



Guia do desenvolvedor

# AWS SDK para SAP ABAP



# AWS SDK para SAP ABAP: Guia do desenvolvedor

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

---

# Table of Contents

O que AWS SDK para SAP ABAPé .....	1
Características do AWS SDK para SAP ABAP .....	1
Manutenção .....	1
Referência de API .....	2
Preços .....	2
Recursos .....	2
Introdução .....	3
Etapa 1: Prepare sua AWS conta .....	3
Perfil do IAM para usuários do SAP .....	3
Autenticação .....	5
Etapa 2: instalar o SDK .....	6
Etapa 3: configurar o SDK .....	6
Etapa 4: configuração funcional .....	8
Etapa 5: Autorizar usuários do SAP .....	10
Etapa 6: escrever o código .....	12
Etapa 7: executar o aplicativo .....	15
Configurar .....	17
Pré-requisitos da SAP .....	17
SDK para SAP ABAP .....	17
SDK para SAP ABAP - edição BTP .....	21
Instalando AWS SDK para SAP ABAP .....	22
Fazer download do SDK .....	22
Verifique o arquivo .....	22
AWS Transportes do SDK .....	23
Instalando o SDK - edição BTP .....	27
Instale o SDK para SAP ABAP - edição BTP .....	27
Modules .....	27
SDK de patches para SAP ABAP - edição BTP .....	28
Configuração .....	29
Configurações globais .....	30
Configurações técnicas .....	31
Configurar cenários .....	31
Configuração do aplicativo .....	32
Perfil do SDK .....	32

Resolvedor lógico de recursos .....	34
Exemplo .....	34
Configurações de runtime .....	35
Registrar e rastrear .....	35
OPT-IN: telemetria aprimorada .....	35
Cenário ativo .....	36
Cenários avançados de conectividade .....	36
Conexão por meio de um servidor proxy .....	36
Conexão por meio de um pacote de inspeção do firewall .....	37
Endpoints de gateway .....	37
Endpoints de interface personalizada .....	38
Roteamento avançado .....	39
Per-service substituição do servidor proxy .....	40
Acesso de endpoints em várias regiões .....	41
Configurações do provedor de serviços .....	42
Atualização, rastreamento e telemetria .....	42
Atualização do sistema SAP .....	43
Rastreamento .....	44
Telemetria .....	45
Uso da SDK .....	46
Representação de dados .....	46
Tipos de dados .....	47
AWS tipos de dados .....	49
Exemplo de programa .....	50
Pré-requisitos .....	50
Código .....	51
Seções de código .....	51
Conceitos .....	54
Classes de API .....	54
Objetos adicionais .....	55
Classes de estrutura .....	55
Matrizes .....	57
Mapas .....	58
Funções de nível superior .....	58
Recursos .....	1
Configuração programática .....	59

Waiters .....	60
Paginadores .....	61
Comportamento de repetição .....	62
Prescritores .....	63
Encadeamento de funções do IAM entre contas .....	64
Criar produtos .....	64
Configurar um ID de produto .....	64
Personalize solicitações HTTP para AWS .....	65
Implemente um aprimoramento .....	65
Filtrar o aprimoramento .....	65
Codifique o aprimoramento .....	65
Limitações .....	66
Exemplos de código .....	68
ACM .....	70
Ações .....	70
API Gateway .....	78
Ações .....	70
Application Recovery Controller .....	87
Ações .....	70
Aurora .....	92
Ações .....	70
ajuste de escala automático .....	98
Ações .....	70
Amazon Bedrock Runtime .....	107
Claude da Anthropic .....	108
Stable Diffusion .....	111
Amazon Bedrock Agents Runtime .....	114
Ações .....	70
CloudFront .....	115
Ações .....	70
CloudWatch .....	118
Ações .....	70
Cenários .....	126
CloudWatch Registros .....	128
Ações .....	70
Provedor de identidade do Amazon Cognito .....	131

Ações .....	70
Amazon Comprehend .....	136
Ações .....	70
AWS Config .....	150
Ações .....	70
AWS Control Tower .....	153
Ações .....	70
Firehose .....	164
Ações .....	70
DynamoDB .....	166
Conceitos básicos .....	167
Ações .....	70
Amazon EC2 .....	181
Ações .....	70
Amazon ECR .....	204
Ações .....	70
Amazon EMR .....	210
Ações .....	70
EventBridge Agendador .....	216
Ações .....	70
AWS Glue .....	221
Ações .....	70
HealthImaging .....	234
Ações .....	70
HealthLake .....	252
Ações .....	70
IAM .....	266
Ações .....	70
AWS IoT data .....	291
Ações .....	70
AWS IoT SiteWise .....	293
Conceitos básicos .....	294
Ações .....	70
Amazon Keyspaces .....	302
Ações .....	70
Kinesis .....	310

Conceitos básicos .....	167
Ações .....	70
AWS KMS .....	320
Conceitos básicos .....	294
Ações .....	70
Lambda .....	341
Conceitos básicos .....	167
Ações .....	70
Organizações .....	355
Ações .....	70
Amazon Pinpoint .....	360
Ações .....	70
API de SMS e voz do Amazon Pinpoint .....	364
Ações .....	70
Amazon Polly .....	374
Ações .....	70
Amazon RDS .....	382
Ações .....	70
banco de dados de origem .....	388
Ações .....	70
Amazon Rekognition .....	395
Ações .....	70
Amazon S3 .....	410
Conceitos básicos .....	167
Ações .....	70
Cenários .....	126
SageMaker IA .....	436
Ações .....	70
Cenários .....	126
Secrets Manager .....	454
Ações .....	70
Amazon SES .....	456
Ações .....	70
API v2 do Amazon SES .....	473
Ações .....	70
Amazon SNS .....	482

Ações .....	70
Cenários .....	126
Amazon SQS .....	493
Ações .....	70
Cenários .....	126
Step Functions .....	503
Ações .....	70
Systems Manager .....	511
Ações .....	70
Amazon Textract .....	521
Ações .....	70
Cenários .....	126
Amazon Transcribe .....	532
Ações .....	70
Amazon Translate .....	540
Ações .....	70
Cenários .....	126
Servidor MCP .....	549
Benefícios .....	549
Configurando o servidor MCP .....	550
O que o servidor MCP pode fazer .....	551
Considerações importantes .....	552
Segurança .....	553
Autenticação do sistema .....	554
Autenticação de metadados .....	554
Autenticação da chave de acesso secreta .....	555
Autenticação baseada em certificado usando o IAM Roles Anywhere .....	555
Autenticação do perfil de origem para acesso entre contas .....	556
Próxima etapa .....	556
Práticas recomendadas de segurança do IAM .....	556
Melhores práticas para perfil de instância do Amazon EC2 .....	557
Perfil do IAM para usuários do SAP .....	558
Considerações sobre a segurança do perfil de origem .....	561
Autorizações da SAP .....	562
Autorizações para configuração .....	562
Autorizações SAP para usuários finais .....	563

Operações seguras .....	565
Criptografia de dados em repouso .....	565
Criptografia de dados em trânsito .....	565
Uso da API .....	2
Chave secreta (SSF) .....	565
Pré-requisitos .....	565
Procedimento .....	566
IAM Roles Anywhere .....	568
Pré-requisitos .....	568
Procedimento .....	569
Perfil de origem .....	64
Pré-requisitos .....	573
Procedimento .....	573
Armazenamento de credenciais .....	21
Etapas de configuração .....	576
Usando o SAP Credential Store com o SDK .....	578
Solução de problemas .....	582
Falha de importação .....	582
Restrição de localização não especificada .....	582
Erro SSL .....	583
Configuração do perfil .....	584
Autorização do IAM .....	585
Autorização para ações .....	585
Cenário ativo .....	36
Caracteres especiais .....	586
Conectividade .....	586
Tópicos adicionais .....	588
Versões .....	588
Estratégia de lançamento .....	588
Práticas recomendadas .....	556
Patches de SDK para SAP ABAP .....	589
Instalação de um módulo adicional .....	590
Desinstalando o SDK para SAP ABAP .....	590
Licenciamento SAP .....	591
Histórico do documento .....	592
.....	dx civ

# O que AWS SDK para SAP ABAP é

AWS SDK para SAP ABAP fornece uma interface para os serviços oferecidos pela AWS linguagem ABAP. Usando o SDK, você pode implementar ABAP BADIs, relatórios, transações, OData serviços e outros artefatos ABAP, Serviços da AWS como Amazon Simple Storage Service (Amazon Amazon DynamoDB S3),,, e muito mais. Amazon Translate Você também pode desenvolver para sistemas baseados em ABAP, a partir do SAP NetWeaver 7.4 e em um ambiente SAP Business Technology Platform. Para obter mais informações, consulte [Instalando o AWS SDK para SAP ABAP - edição BTP](#).

## Tópicos

- [Características do AWS SDK para SAP ABAP](#)
- [Manutenção e suporte para as versões principais do SDK](#)
- [Referência de API](#)
- [Preços](#)
- [Recursos adicionais do](#)

## Características do AWS SDK para SAP ABAP

AWS SDK para SAP ABAP foi projetado para parecer familiar e natural para os desenvolvedores da SAP. Por exemplo, embora todos Serviços da AWS usem as `false` cadeias de caracteres `true` e para representar dados booleanos em estruturas XML e JSON, o SDK para SAP ABAP os converte em valores nativos do ABAP e de um único caractere. 'X' ' ' O SDK para SAP ABAP usa estruturas ABAP nativas o máximo possível, inclusive em tipos de dados e formatos de carimbo de data/hora. Como resultado, o programador ABAP não precisa se preocupar com a serialização subjacente de JSON e XML ou com o formato eletrônico do protocolo da API.

## Manutenção e suporte para as versões principais do SDK

Para obter informações sobre manutenção e suporte para as versões principais do SDK e suas dependências subjacentes, consulte o seguinte no Guia de [referência de ferramentas AWS SDKs e ferramentas](#):

- [AWS SDKs e política de manutenção de ferramentas](#)
- [AWS SDKs e ferramentas e matriz de suporte de versões](#)

## Referência de API

Para ver uma lista completa de AWS SDK para SAP ABAP APIs, consulte [AWS SDK para SAP ABAP - Guia de referência da API](#).

Para ver uma lista completa de módulos AWS SDK para SAP ABAP TLAs, consulte [AWS SDK para SAP ABAP - Lista de módulos](#).

Para ver uma lista completa de módulos do SDK para SAP ABAP - versão prévia para desenvolvedores da edição BTP TLAs, consulte [AWS SDK para SAP ABAP - Edição BTP - Lista de módulos](#).

## Preços

AWS SDK para SAP ABAP está disponível para você sem custo adicional. Você paga somente pelos AWS recursos e serviços que consome com o SDK.

## Recursos adicionais do

Além deste guia, a seguir estão valiosos recursos online para SDK para SAP ABAP.

- [AWS SDK para SAP ABAP Knowledge MCP Server](#) — Conecte seu IDE habilitado para IA para gerar código ABAP preciso para integrações. AWS
- [AWS Biblioteca de exemplos de códigos do SDK](#)
- [SAP na AWS documentação](#)
- [Blog de desenvolvedores da AWS](#)
- [AWS fóruns de desenvolvedores](#)
- [@awsdevelopers](#) (Twitter)

# Começando com AWS SDK para SAP ABAP

Esta seção descreve como começar a usar o SDK. Ele inclui informações sobre como instalar o SDK, realizar a configuração básica e criar um exemplo de código Hello World que traduz uma frase de um idioma para outro. Se você é iniciante no AWS SDK, recomendamos realizar essas etapas em um ambiente sandbox.

## Etapas

- [Etapa 1: Prepare sua AWS conta](#)
- [Etapa 2: instalar o SDK](#)
- [Etapa 3: configurar o SDK](#)
- [Etapa 4: configuração funcional](#)
- [Etapa 5: Autorizar usuários do SAP](#)
- [Etapa 6: escrever o código](#)
- [Etapa 7: executar o aplicativo](#)

## Etapa 1: Prepare sua AWS conta

Para começar a usar o SDK para SAP ABAP, você deve ter uma Conta da AWS ativa. Você precisa de um, Conta da AWS mesmo que seu sistema SAP esteja hospedado localmente, na SAP Business Technology Platform (BTP) ou com outro provedor de nuvem.

Se o seu sistema SAP estiver sendo executado na AWS nuvem, você fará chamadas para AWS serviços em seu Conta da AWS.

## Tópicos

- [Perfil do IAM para usuários do SAP](#)
- [Autenticação](#)

## Perfil do IAM para usuários do SAP

- Crie um perfil do IAM com as instruções fornecidas no Guia do usuário da AWS Identity and Access Management . Para obter mais informações, consulte [Criação de uma função para delegar](#)

[permissões a um AWS serviço](#). Observe o nome do recurso da Amazon (ARN) do perfil do IAM para uso posterior.

- Selecione Amazon EC2 como o caso de uso.
- Use SapDemoTranslate como nome do perfil.
- Anexe o perfil TranslateReadOnly ao perfil.
- O perfil deve ter as seguintes entidades para permitir que o sistema SAP assuma o perfil. Substitua **"111122223333"** pelo número da sua conta da AWS .

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sts:AssumeRole"
      ],
      "Principal": {
        "AWS": "111122223333"
      }
    }
  ]
}
```

Este exemplo mostra que qualquer diretor da Conta da AWS **"111122223333"** pode assumir a função. Essa é uma permissão ampla que é adequada para proof-of-concept. Você pode usar uma entidade principal mais restrito para produção, como nos exemplos a seguir.

- Um usuário específico — quando o sistema SAP está usando um dos seguintes:
  - Credenciais criptografadas por SSF de um sistema SAP local
  - Credenciais do serviço SAP Credential Store no ambiente SAP BTP, ABAP
- Um perfil específico, quando o sistema SAP está no Amazon EC2 e há um perfil de instância.
- Amazon EC2, quando o sistema SAP está no Amazon EC2 e não há perfil de instância.

Para obter mais informações, consulte [Práticas recomendadas de segurança no IAM](#).

## Autenticação

A autenticação depende de onde seu sistema SAP está hospedado.

Locais

- [Na AWS nuvem](#)
- [No local, SAP BTP ou outra nuvem](#)

### Na AWS nuvem

Certifique-se de que a instância EC2 na qual seu sistema SAP está sendo executado tenha um perfil de instância com as seguintes permissões.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::111122223333:role/SapDemoTranslate"
    }
  ]
}
```

Adicione o ARN que você anotou na etapa anterior.

Essa permissão permite que seu sistema SAP assuma o perfil SapDemoTranslate em nome do usuário do ABAP.

### No local, SAP BTP ou outra nuvem

Se seu sistema SAP estiver localizado no local, no SAP BTP ou em outra nuvem, use as etapas a seguir para estabelecer uma conexão para autenticação usando a chave de acesso secreta.

1. Criar um usuário do IAM. Para obter mais informações, consulte [Criar usuários do IAM \(console\)](#).

2. Use SapDemoSID como o nome de usuário do IAM. SID é o ID do sistema do seu sistema SAP.
3. Atribua um perfil SapDemoTranslate a esse usuário.

Guarde a `access_key` e `secret_access_key`. Você deve configurar essas credenciais em seu sistema SAP.

#### Note

Se seu sistema SAP estiver localizado no local, no SAP BTP ou em outra nuvem, você poderá autenticar usando uma das seguintes opções.

- [Autenticação de chave de acesso secreta](#) usando SSF ou SAP Credential Store
- [Usando certificados com o IAM Roles Anywhere](#)

## Etapa 2: instalar o SDK

Consulte as guias a seguir para obter instruções de instalação.

### SDK for SAP ABAP

Importe o SDK para transportes SAP ABAP em seu sistema SAP. Você pode importar os transportes para qualquer cliente. Para obter mais informações, consulte [Instalando o SDK para SAP ABAP](#).

### SDK for SAP ABAP - BTP edition

Instale o SDK para SAP ABAP - edição BTP usando o aplicativo Deploy Product. Para obter mais informações, consulte [Instalando o SDK para SAP ABAP - edição BTP](#).

## Etapa 3: configurar o SDK

Antes de configurar o SDK, verifique se você tem as autorizações necessárias. Para obter mais informações, consulte [Autorizações SAP](#).

Consulte as guias a seguir para obter instruções de configuração.

## SDK for SAP ABAP

Execute a /AWS1/IMG transação para abrir o Guia de implementação do SDK para SAP ABAP. Para executar essa transação, insira /n/AWS1/IMG na barra de comando do seu sistema SAP e escolha Enter.

Conclua as configurações a seguir.

- Acesse Pré-requisitos técnicos.
  - Analise os [parâmetros](#) recomendados e a [conectividade HTTPS](#).
- Vá para Configurações globais → Configurar cenários.
  - Altere as configurações, de acordo com as recomendações em [Configurações globais](#).
- Vá para Configurações globais → Configurações técnicas.
  - Altere as configurações, de acordo com as recomendações em [Configurações globais](#).
- Vá para Configurações de runtime → Registrar e rastrear.
  - Selecione Novas entradas.
    - Nível de rastreamento: sem rastreamento.
    - Linhas de despejo máximas: 100.
    - OPT-IN: enh telemetry: mantenha isso em branco.
  - Selecione Salvar.
- Vá para Configurações de runtime → Cenário ativo.
  - Em Novo cenário, selecione DEFAULT.
  - Selecione Confirmar alteração de cenário.
  - Aceite a solicitação.

### Pré-requisitos para sistemas on-premises

Se o sistema SAP estiver sendo executado on-premises ou em outra nuvem, as credenciais deverão ser armazenadas no banco de dados SAP. As credenciais são criptografadas usando o SAP SSF e exigem uma biblioteca criptográfica configurada, como a da SAP. CommonCryptoLib

As etapas para configurar o SSF para SDK para SAP ABAP estão descritas na transação /AWS1/IMG.

**Note**

O pré-requisito anterior não se aplicará se seu sistema SAP estiver sendo executado no Amazon EC2. Os sistemas SAP executados no Amazon EC2 recuperam credenciais de curta duração e rotativas automáticas dos metadados da instância do Amazon EC2.

## SDK for SAP ABAP - BTP edition

Abra seu ambiente ABAP em um navegador da Web e navegue até o aplicativo Custom Business Configurations.

Conclua as configurações a seguir.

- Vá para Configurar cenários.
  - Altere as configurações, de acordo com as recomendações em [Configurações globais](#).
- Vá para Configurações técnicas.
  - Altere as configurações, de acordo com as recomendações em [Configurações globais](#).

## Etapa 4: configuração funcional

Consulte as guias a seguir para obter instruções de configuração.

## SDK for SAP ABAP

Execute a transação /AWS1/IMG (insira /n/AWS1/IMG na barra de comando e escolha Enter) para abrir o guia de implementação do AWS SDK.

- Vá para Configuração do aplicativo → Perfil do SDK.
  - Selecione Novas entradas.
    - Perfil: DEMO.
    - Descrição: Demo profile.
    - Selecione Salvar.
- Destaque a entrada que você criou e clique na ramificação de árvore Autenticação e Configurações.
  - Selecione Novas entradas.
    - SID: o ID do sistema SAP em que você está atualmente.

- Cliente: o cliente do sistema SAP em que você está atualmente.
- ID do cenário: a lista suspensa onde você encontrará o cenário PADRÃO criado pelo administrador do Basis.
- AWS Região: insira a AWS região para a qual você deseja fazer chamadas. Se o seu sistema SAP estiver sendo executado AWS, insira a AWS região em que ele está sendo executado.
- Método de autenticação:
  - Selecione o Perfil da instância por meio de metadados se seu sistema SAP estiver em execução no Amazon EC2.
  - Selecione Credenciais do armazenamento SSF se seu sistema SAP estiver sendo executado on-premises ou em outra nuvem.
    - Selecione Definir credenciais.
    - Insira a ID da chave de acesso e a chave de acesso secreta que você criou na etapa anterior.
- Mantenha a opção Desativar perfis do IAM.
- Selecione Salvar.
- Clique na ramificação da árvore de Mapeamento de perfis do IAM.
  - Selecione Novas entradas.
    - Insira o número de sequência: 010.
    - Insira o perfil lógico do IAM: TESTUSER.
    - Insira o ARN do perfil do IAM: insira o arn:aws: do perfil do IAM que contém a política TranslateReadOnly criada na etapa anterior.

## SDK for SAP ABAP - BTP edition

Configure a autenticação usando o SAP Credential Store. Para obter mais informações, consulte [Usando o SAP Credential Store](#).

Abra seu ambiente ABAP em um navegador da Web e navegue até o aplicativo Custom Business Configurations.

- Acesse o Perfil do SDK.
  - Selecione Editar para criar um novo perfil.
    - Perfil: DEMO.

- Descrição: Demo profile.
- Selecione a tecla de seta para a direita ao lado da entrada criada para navegar até a guia Autenticação e Configurações.

Selecione Novas entradas.

- SID: o ID do sistema SAP em que você está atualmente.
- Cliente: o cliente do sistema SAP em que você está atualmente.
- ID do cenário: a lista suspensa onde você encontrará o cenário PADRÃO criado pelo administrador do Basis.
- AWS Região: insira a AWS região para a qual você deseja fazer chamadas. Se o seu sistema SAP estiver sendo executado AWS, insira a AWS região em que ele está sendo executado.
- Método de autenticação: selecione as credenciais do SAP Credential Store.
- Insira o namespace e o nome da chave das credenciais armazenadas no SAP Credentials Store.
- Insira o nome do Acordo de Comunicação criado para estabelecer a comunicação entre o SDK para SAP ABAP - edição BTP e o SAP Credential Store.
- Mantenha a opção Desativar perfis do IAM.
- Clique com o botão direito do mouse na tecla de seta direita ao lado da entrada criada para navegar até a guia Mapeamento de funções do IAM.

Selecione Novas entradas.

- Insira o número de sequência: 010.
- Insira o perfil lógico do IAM: TESTUSER.
- Insira o ARN do perfil do IAM: insira o arn:aws: do perfil do IAM que contém a política TranslateReadOnly criada na etapa anterior.

## Etapa 5: Autorizar usuários do SAP

Os usuários do SAP não estão autorizados a usar a AWS funcionalidade por padrão. Os usuários devem ser explicitamente autorizados usando as autorizações do SAP. Consulte as guias a seguir para obter mais detalhes.

## SDK for SAP ABAP

### Crie um perfil PFCG

- Vá para a transação PFCG
- Insira o nome do perfil ZAWS\_SDK\_DEMO\_TESTUSER e selecione Criar perfil único.
  - Descrição: Role for demo AWS SDK functionality
  - Vá até a guia Autorizações.
  - Selecione Alterar dados de autorização e aceite o pop-up informativo.
  - No pop-up Escolher modelo, selecione Não selecionar modelos.
  - Selecione Adicionar manualmente na barra de ferramentas.
  - Adicione os seguintes objetos de autorização:
    - /AWS1/LROL
    - /AWS1/SESS
  - Na árvore de autorização, insira:
    - Perfil para acessar a AWS APIs: DEMO
    - Perfil lógico do IAM: TESTUSER
  - Selecione Salvar.
  - Selecione Gerar.
  - Selecione Voltar.
  - Selecione Salvar para salvar o perfil.

### Atribuir o perfil PFCG aos usuários do SAP

Qualquer usuário que tenha a ZAWS\_SDK\_DEMO\_TESTUSER função atribuída será autorizado a usar as funções do AWS SDK com as configurações definidas no perfil do DEMO SDK. O usuário autorizado também assumirá o perfil do IAM mapeado para o padrão lógico do IAM TESTUSER nesse perfil.

- Execute a transação SU01.
  - Insira o ID de usuário de um usuário do SAP que testará a funcionalidade AWS do SDK.
  - Selecione Alterar.
  - Vá até a guia Perfis e atribua o perfil ZAWS\_SDK\_DEMO\_TESTUSER ao usuário.
  - Selecione Salvar.

## SDK for SAP ABAP - BTP edition

### Crie uma função comercial

- Abra seu ambiente ABAP em um navegador da Web e navegue até o aplicativo Maintain Business Roles.
- Selecione Criar a partir do modelo e insira os detalhes a seguir.
  - Modelo — Escolha **/AWS1/RT\_BTP\_ENDUSER**.
  - Nova ID da função comercial — insira uma ID.
  - Descrição da nova função comercial — insira uma descrição.
- Selecione OK para ver a página da função comercial.
- Na guia Detalhes gerais da função, acesse Categorias de acesso e defina o campo Ajuda para gravação, leitura e valor como Restrito.
- Selecione Manter restrições e expanda Tipos de restrição atribuídos no painel de navegação esquerdo. Atualize o campo a seguir na seção Restrições e valores.
  - Em Escolher sessão do SDK, selecione o ícone de lápis ao lado do Perfil do SDK e navegue até a guia Intervalos. Digite **DEMO** e selecione Adicionar.
  - Em Escolher função lógica do IAM, selecione o ícone de lápis ao lado da função lógica do IAM e navegue até a guia Intervalos. Digite **TESTUSER** e selecione Adicionar.

Selecione o ícone de lápis ao lado do Perfil do SDK e navegue até a guia Intervalos. Digite **DEMO** e selecione Adicionar

- Navegue de volta até o modelo de Função Comercial e abra a guia Usuários Comerciais. Selecione Adicionar para atribuir a função comercial recém-criada a um usuário comercial da SAP que testará a funcionalidade do SDK. Selecione Salvar.

Qualquer usuário comercial atribuído à função comercial criada será autorizado a usar as funções do AWS SDK com as configurações definidas no perfil do DEMO SDK. O usuário autorizado também assumirá o perfil do IAM mapeado para o padrão lógico do IAM TESTUSER nesse perfil.

## Etapa 6: escrever o código

Consulte as guias a seguir para obter mais detalhes.

## SDK for SAP ABAP

## 1. Encerrar transação SE38.

- Insira ZDEMO\_TRANSLATE\_HELLO\_WORLD como nome do programa.
- Selecione Create.
- Insira AWS SDK Hello World In Any Language como título.
- Tipo: escolha Programa executável.
- Status: escolha Programa de teste.
- Selecione Salvar.
- Salve o programa como um objeto local.

Adicione o seguinte código.

```
*&-----*
*& Report  ZAWS1_DEMO_XL8_SIMPLE
*&
*&-----*
*& A simple demo of language translation with AWS Translate
*&
*&-----*
REPORT zaws1_demo_xl8_simple.

START-OF-SELECTION.
  PARAMETERS pv_text TYPE /aws1/xl8boundedlengthstring DEFAULT 'Hello, World'
  OBLIGATORY.

  PARAMETERS pv_lang1 TYPE languageiso DEFAULT 'EN' OBLIGATORY.
  PARAMETERS pv_lang2 TYPE languageiso DEFAULT 'ES' OBLIGATORY.

  TRY.
    DATA(go_session) = /aws1/cl_rt_session_aws=>create( 'DEMO' ).
    DATA(go_xl8)      = /aws1/cl_xl8_factory=>create( go_session ).
    DATA(lo_output) = go_xl8->translatetext(
      iv_text          = pv_text
      iv_sourcelanguagecode = CONV /aws1/xl8languagecodestring( pv_lang1 )
      iv_targetlanguagecode = CONV /aws1/xl8languagecodestring( pv_lang2 )
    ).

    WRITE: / 'Source Phrase: ', pv_text.
```

```

WRITE: / 'Target Phrase: ', lo_output->get_translatedtext( ).
CATCH /aws1/cx_xl8unsuppdedlanguage00 INTO DATA(lo_lang).
WRITE: / 'ERROR' COLOR COL_NEGATIVE,
        'Cannot translate from',
        lo_lang->sourcelanguagecode,
        'to',
        lo_lang->targetlanguagecode.
CATCH cx_root INTO DATA(lo_root).
WRITE: / 'ERROR' COLOR COL_NEGATIVE, lo_root->get_text( ).
ENDTRY.

```

## SDK for SAP ABAP - BTP edition

1. Clique com o botão direito do mouse no pacote em que a classe ABAP precisa ser criada e selecione Novo > Classe ABAP.
2. Insira **ZCL\_DEMO\_XL8\_SIMPLE** o nome da classe e adicione uma descrição da classe. Escolha Próximo.
3. Crie ou escolha uma solicitação de transporte. Selecione Concluir.

Adicione o seguinte código.

```

CLASS zcl_demo_xl8_simple DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

  PUBLIC SECTION.
    INTERFACES if_oo_adt_classrun.
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.

CLASS zcl_demo_xl8_simple IMPLEMENTATION.
  METHOD if_oo_adt_classrun~main.

    TRY.
      " input parameters
      DATA(pv_text) = |Hello, World|.
      DATA(pv_lang1) = |EN|.
      DATA(pv_lang2) = |ES|.

```

```
DATA(go_session) = /aws1/cl_rt_session_aws=>create( 'DEMO' ).
DATA(go_xl8)      = /aws1/cl_xl8_factory=>create( go_session ).
DATA(lo_output) = go_xl8->translatetext(
    iv_text          = pv_text
    iv_sourcelanguagecode = pv_lang1
    iv_targetlanguagecode = pv_lang2
).

out->write( |Source Phrase: { pv_text }| ).
out->write( |Target Phrase: { lo_output->get_translatedtext( ) }| ).
CATCH /aws1/cx_xl8unsuppdedlanguage00 INTO DATA(lo_lang).
out->write( |ERROR - Cannot translate from { lo_lang->sourcelanguagecode }
to { lo_lang->targetlanguagecode }| ).
CATCH cx_root INTO DATA(lo_root).
out->write( |ERROR - { lo_root->get_text( ) }| ).
ENDTRY.
ENDMETHOD.
ENDCLASS.
```

[Para obter detalhes sobre como escrever código ABAP que usa o SDK, consulte Usando. AWS SDK para SAP ABAP](#)

## Etapa 7: executar o aplicativo

Consulte as guias a seguir para obter mais detalhes.

### SDK for SAP ABAP

Execute o aplicativo no SE38. Se for bem-sucedido, o seguinte será o resultado.

```
Source Phrase: Hello, World
Target Phrase: Hola, mundo
```

Se você não tiver autorizações, configurações ou pré-requisitos básicos, poderá receber uma mensagem de erro. Veja o exemplo a seguir.

```
ERROR Could not find configuration under profile DEMO with
scenario DEFAULT for SBX:001
```

Se seu perfil do SAP autorizar você a usar um perfil do SDK e mapeá-lo para um perfil lógico do IAM, enquanto suas permissões do IAM não estiverem configuradas para que o sistema SAP assuma o perfil do IAM, o seguinte será seu resultado.

```
ERROR Could not assume role arn:aws:iam::111122223333:role/SapDemoTranslate
```

Nesse caso, revise suas permissões do IAM e a configuração de confiança nos perfis, nos usuários ou em ambos do IAM definidos em [the section called “Etapa 1: Prepare sua AWS conta”](#).

## SDK for SAP ABAP - BTP edition

Execute o aplicativo no Eclipse > Executar como > Aplicativo ABAP (Console). Se for bem-sucedido, o seguinte será o resultado.

```
Source Phrase: Hello, World  
Target Phrase: Hola, mundo
```

Se você não tiver autorizações, configurações ou pré-requisitos básicos, poderá receber uma mensagem de erro. Veja o exemplo a seguir.

```
ERROR Could not find configuration under profile DEMO with  
scenario DEFAULT for SBX:001
```

Se seu perfil do SAP autorizar você a usar um perfil do SDK e mapeá-lo para um perfil lógico do IAM, enquanto suas permissões do IAM não estiverem configuradas para que o sistema SAP assuma o perfil do IAM, o seguinte será seu resultado.

```
ERROR Could not assume role arn:aws:iam::111122223333:role/SapDemoTranslate
```

Nesse caso, revise suas permissões do IAM e a configuração de confiança nos perfis, nos usuários ou em ambos do IAM definidos em [the section called “Etapa 1: Prepare sua AWS conta”](#).

# Configurar

Esta seção fornece informações sobre como configurar seu ambiente de desenvolvimento para uso do AWS SDK para SAP ABAP.

## Tópicos

- [Pré-requisitos da SAP](#)
- [Instalando AWS SDK para SAP ABAP](#)
- [Instalando o AWS SDK para SAP ABAP - edição BTP](#)

## Pré-requisitos da SAP

Os pré-requisitos a seguir para instalar o SDK são aplicáveis quando seus sistemas SAP estão hospedados no. AWS

## Tópicos

- [Pré-requisitos do AWS SDK for SAP ABAP](#)
- [Pré-requisitos do AWS SDK for SAP ABAP - edição BTP](#)

## Pré-requisitos do AWS SDK for SAP ABAP

A seguir estão os pré-requisitos do AWS SDK for SAP ABAP.

## Tópicos


- [Lançamento básico](#)
- [Lançamento do kernel](#)
- [Parâmetros](#)
- [Observações](#)
- [Conectividade de saída](#)
- [Conectividade HTTPS](#)
- [Acesso aos metadados da EC2 instância Amazon](#)

## Lançamento básico

O SDK para SAP ABAP é compatível com o SAP NetWeaver 7.4 e superior. O SDK para SAP ABAP não toca em nenhuma tabela de aplicativos da SAP. É totalmente independente dos aplicativos, como o SAP Enterprise Resource Planning e o SAP Landscape Transformation Replication Server.

O nível mínimo de SP suportado para SAP\_BASIS 740 é SP 0008. Para obter mais informações, consulte [SAP Note 1856171 - Supporting form fields of the same name in CL\\_HTTP\\_ENTITY](#) (requer acesso ao portal SAP). Com base nas suas necessidades de negócios, você pode escolher um nível de SP mais alto, conforme mostrado na imagem a seguir.

Installed Software Component Versions      Installed Product Versions



Component	Release	SP-Level	Support Package	Short Description of Component
SAP_BASIS	740	0026	SAPKB74026	SAP Basis Component
SAP_ABA	740	0026	SAPKA74026	Cross-Application Component
SAP_GWFND	740	0027	SAPK-74027INSAPGWFND	SAP Gateway Foundation
SAP_UI	754	0008	SAPK-75408INSAPUI	User Interface Technology
PL_BASIS	740	0008	SAPK-74026INSAPBASIS	Basis Plug-In

Não há requisitos mínimos de nível de SP para versões do SAP\_BASIS 750 e superiores.

## Lançamento do kernel

O SDK para SAP ABAP e as ferramentas que usam o Internet Communication Manager (ICM) para conectividade HTTP dependem do kernel SAP para seus recursos criptográficos, HTTP, XML e JSON. Recomendamos usar a versão mais recente do kernel compatível com sua NetWeaver plataforma SAP. O requisito mínimo é a versão 741 do kernel. Para obter mais informações, consulte [SAP Note 2083594 - Versões do SAP Kernel e níveis de patch do SAP Kernel](#) (requer acesso ao portal SAP).

Se você estiver usando a versão 741 ou 742 do kernel, os seguintes níveis de patch são necessários:

- 741, atualização 212
- 742, atualização 111

## Parâmetros

Seu sistema SAP deve oferecer suporte à Indicação de Nome de Servidor (SNI) conforme descrito nas seguintes notas SAP (requer acesso ao portal SAP).

- [SAP Note 2124480 - ICM/Web Dispatcher: Indicação de nome de servidor de extensão TLS \(SNI\) como cliente](#)
- [SAP Note 2582368 - Atualização do SAPSSL para envio da extensão TLS SNI pelo lado do cliente por saphttp, sapkprotp, sldreg](#)

Configure o seguinte parâmetro no arquivo DEFAULT.PFL.

```
icm/HTTPS/client_sni_enabled = TRUE
```

## Observações

Aplique a seguinte nota SAP ao seu sistema.

- <https://launchpad.support.sap.com/#/notas/0001856171>
- <https://launchpad.support.sap.com/#/notas/0002619546>

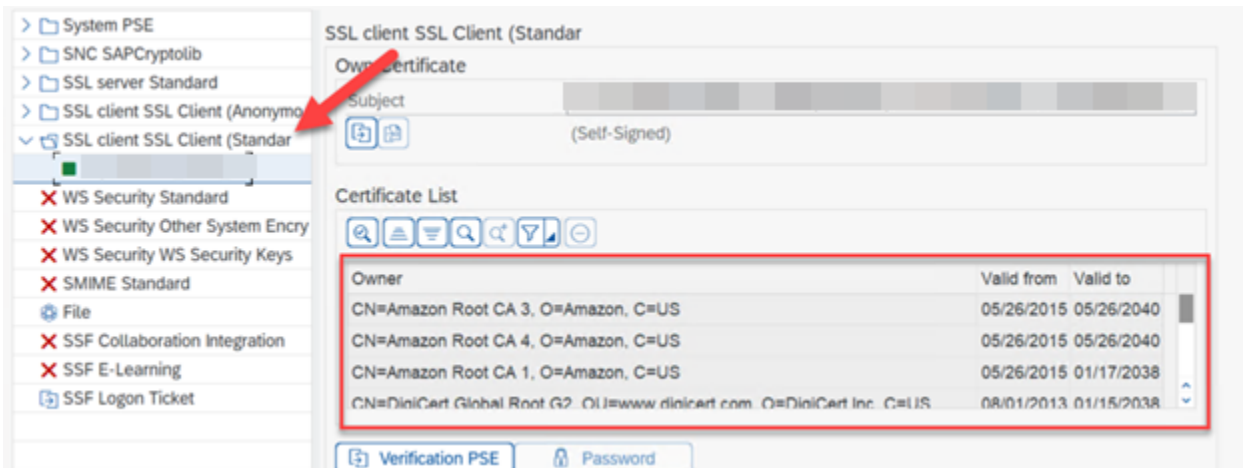
## Conectividade de saída

O SDK para SAP ABAP é um cliente HTTPS. O sistema da SAP envia mensagens HTTPS de saída. Não há exigência de conectividade de entrada.

## Conectividade HTTPS

Todas as chamadas de AWS API são feitas com canais HTTPS criptografados. O sistema SAP deve ser configurado para confiar em AWS certificados para estabelecer uma conexão HTTPS de saída.

1. Acesse <https://www.amazontrust.com/repository/>.
2. Em Raiz CAs, baixe todos os certificados usando o link PEM.
3. Importe esses certificados STRUST de seu SSL Client (Standard) PSE em cada um dos seus sistemas SAP, conforme mostrado na imagem a seguir.



## Acesso aos metadados da EC2 instância Amazon

O sistema ABAP faz conexões HTTP não criptografadas com o localhost (<http://169.254.169.254>) para habilitar os metadados da instância Amazon. EC2 O canal HTTP é usado somente para recuperar AWS credenciais do servidor local. O tráfego HTTP permanece dentro do host.

Os metadados permitem que um sistema SAP AWS se autentique com segurança sem armazenar uma chave secreta no SAP Secure Store. Esse recurso é aplicável somente aos sistemas SAP hospedados na Amazon EC2.

Configure o arquivo DEFAULT.PFL com o parâmetro a seguir para permitir que seu sistema SAP faça uma conexão HTTP de saída não criptografada.

```
icm/server_port_<xx> = PROT=HTTP,PORT=8000,TIMEOUT=60,PROCTIMEOUT=600
```

Use o parâmetro a seguir para habilitar a conexão HTTP de saída sem habilitar a conexão de entrada.

```
icm/server_port_<xx> = PROT=HTTP,PORT=0,TIMEOUT=60,PROCTIMEOUT=600
```

Verifique se seu sistema SAP está configurado para conexões HTTP de saída com as seguintes etapas:

1. Execute a transação SMICM.
2. Vá para Serviços ativos.
3. Verifique se você vê uma marca de seleção verde na linha HTTP, na coluna Ativo, conforme mostrado na imagem a seguir.

Active Services						
No.	Protocol	Service Name/Port	Host Name	Keep Alive	Proc.Timeo	Actv E:
1	HTTPS	50001		60	600	✓
2	HTTP	0		60	600	✓

## Pré-requisitos do AWS SDK for SAP ABAP - edição BTP

A seguir estão os únicos pré-requisitos para o AWS SDK for SAP ABAP - edição BTP.

### Tópicos

- [SAP Landscape Portal — edição BTP](#)
- [SAP Credential Store — edição BTP](#)

### SAP Landscape Portal — edição BTP

Esse pré-requisito é aplicável somente ao AWS SDK for SAP ABAP - edição BTP.

O SAP Landscape Portal é o único mecanismo compatível para instalar complementos em um ambiente SAP BTP. Certifique-se de estar inscrito para usar este serviço. Para obter mais informações, consulte [Landscape Portal](#).

### SAP Credential Store — edição BTP

Esse pré-requisito é aplicável somente ao AWS SDK for SAP ABAP - edição BTP.

Na versão prévia para desenvolvedores, a autenticação por chave de acesso secreta é o único mecanismo compatível para autenticar o AWS SDK para a edição SAP ABAP - BTP. O SDK lê as credenciais do Credential Store e armazena a chave de acesso secreta com segurança.

Você deve atender aos seguintes pré-requisitos.

- Assinatura da Credential Store.
- Armazenamento de credenciais atribuído como um direito à sua subconta BTP. Consulte [Configuração inicial](#) para obter mais detalhes.
- Uma instância de serviço com plano padrão para o Credential Store. Consulte [Criar uma instância de serviço](#) para obter mais detalhes.

Para obter mais informações, consulte [Usando o SAP Credential Store](#).

O serviço SAP Credential Store é executado no SAP BTP fora do sistema ABAP BTP. Consulte o [SAP Credential Store](#) para obter mais detalhes.

## Instalando AWS SDK para SAP ABAP

### Tópicos

- [Fazer download do SDK para SAP ABAP](#)
- [Verifique o SDK para o arquivo SAP ABAP, opcional](#)
- [AWS Transportes do SDK](#)

## Fazer download do SDK para SAP ABAP

Faça o download do SDK em <https://sdk-for-sapabap.aws.amazon.com/awsSdkSapabapV1/release/abapsdk-LATEST.zip> <https://sdk-for-sapabap.amazonaws.cn/> .

```
curl "https://sdk-for-sapabap.aws.amazon.com/awsSdkSapabapV1/release/abapsdk-LATEST.zip" -o "abapsdk-LATEST.zip"
```

Quando o download estiver concluído, recomendamos que você descompacte o arquivo baixado em um diretório, como `/tmp/awssdk`.

## Verifique o SDK para o arquivo SAP ABAP, opcional

Essa etapa opcional de validar a assinatura do seu arquivo SDK ajuda você a confirmar que seu SDK não foi adulterado. Use as etapas a seguir para verificar seu arquivo SDK.

1. Faça download do arquivo SDK SIGNATURE com o seguinte comando:

```
curl "https://sdk-for-sapabap.aws.amazon.com/awsSdkSapabapV1/release/abapsdk-LATEST.sig" -o "abapsdk-LATEST.sig"
```

2. Copie a chave pública a seguir e salve-a em um arquivo chamado `abapsdk-signing-key.pem`.

```
-----BEGIN PUBLIC KEY-----  
MIICIjANBgkqhkiG9w0BAQEFAAOCAG8AMIICGgKCAgEAmS3oN3wKBh4HJ0Ga0tye  
15RR5909nuw0Jx0vEDCT709wUrxS3mjpgEw6b6hvr2dLdoFr+eH4ewT5bV16U3gDv
```

```

051sTdEJJpfLEWJJZZNK3v9fGWKyXgYe+ifmsPmf4lhNd2auzpvIy2Ur1SYijCRB
BWZFW+Ux00kILz+8vCFSXMZ6Z0qtLI1ZFbGrn6A5adbwzsf0qkg9BUEZK0wB6TAi
ZTnkMdBZGCBM9K2MRKKMxtrixUn+TFcAYyh5pM9tUAb2q4XE5m7092UnZG7ur/QY1
1FSZwAhQmk8hUPgUaq00QRC6z3TRzIGK0A/DI0cUPJMzFR4LCxEJkgh4rkRaU9V2
07DthUpj8b7QcQai0pnMpBf3zWlgbjNmX0hB0Eprg8/nVRHspf3zuisJ21MPkz0
cH0R31MNsMLzm+d/gVklT31R/JwAcFCkXTWvR8/V0WNGZZXdVubefrfI/k7fP60B
bzUrI1N4poq16rc4Tk5Derg+wQ7r0WjXkXop2kiCMjbYo0o10kS/At64PLjz8dH
Zg25o79U9EJln+lpqZ297Ks+Hoct0v2GPbeeh0s7+N0fRTy0r81EZIUURLPKLVQUw
otVRzNDgLOA7eA667NrmegZfHCmqEwK9tXakZUHAcMzRPyhALc/HtmovxdStN9h1
JC4ex0GqstAv1fX5QaTbMSECAwEAAQ==
-----END PUBLIC KEY-----

```

3. Verifique o download do ZIP do SDK por meio do comando a seguir. O comando exige `openssl` que faz parte de muitas distribuições Linux.

```

openssl dgst -sha256 -verify abapsdk-signing-key.pem -keyform PEM -signature
abapsdk-LATEST.sig abapsdk-LATEST.zip

```

4. Verifique se a saída do comando anterior é `Verified OK`.
5. Se a saída for `Verification Failure`, repita as etapas anteriores. Se você continuar recebendo uma saída com falha, não instale o SDK e entre em contato com o Suporte.

## AWS Transportes do SDK

### Tópicos

- [Conteúdo](#)
- [Importação](#)
- [Namespace](#)

### Conteúdo

A instalação do SDK para SAP ABAP é concluída por meio do ABAP Transports. Você deve importar esses transportes para seu ambiente de desenvolvimento ou sandbox.

Cada versão do SDK para SAP ABAP substitui completamente a anterior. Não há necessidade de aplicar transportes incrementais. Os transportes são empacotados em um arquivo ZIP. A seguir está a estrutura do arquivo ZIP.

```

transport/

```

```
transports/core/  
transports/core/Knnnnnn.AWS  
transports/core/Rnnnnnn.AWS  
transports/tla1/  
transports/tla1/Knnnnnn.AWS  
transports/tla1/Rnnnnnn.AWS  
transports/tla2/  
transports/tla2/Knnnnnn.AWS  
transports/tla2/Rnnnnnn.AWS  
.  
.  
.
```

A pasta `transports` contém uma subpasta `core`. A subpasta `core` contém os principais transportes de runtime e uma subpasta para cada módulo, nomeada pela abreviatura de três letras do módulo. Para obter uma lista completa de módulos do TLAs, consulte [AWS SDK para SAP ABAP - Lista de módulos](#).

AWS Os transportes do SDK são solicitações de bancada de trabalho. Dependendo da configuração de suas rotas TMS, o SDK pode não ser encaminhado automaticamente para suas filas de garantia de qualidade e produção após a importação para o sistema anterior. Você deve adicioná-los manualmente à fila de cada sistema.

Quando seu projeto estiver pronto para a próxima fase, o AWS SDK poderá ser importado junto com transportes separados contendo seu próprio Z código com funcionalidade comercial. Se você estiver usando um sistema de controle de alterações, como o SAP Change Request Management (ChaRM), consulte o administrador do ChaRM para saber como lidar corretamente com transportes de terceiros.

## Importação

### Tópicos

- [Ponteiros principais](#)
- [Hora da importação](#)

AWS Os transportes do SDK são independentes do cliente. O transporte principal é obrigatório e contém o código de tempo de execução do SDK, a API e a API do Amazon Simple Storage Service. AWS Security Token Service Cada um dos módulos SDK restantes é entregue em um transporte separado. Para manter o tamanho do SDK pequeno em seu sistema, cada módulo do SDK é

opcional. Você pode instalar módulos adicionais posteriormente, se necessário para sua lógica de negócios.

Por exemplo, se você quiser usar os APIs para o Amazon S3 e Amazon Translate, importe os core transportes (contendo o tempo de execução principal, Amazon S3 AWS STS e módulos) e x18 os transportes (contendo o módulo Amazon Translate para).

Para ver uma lista completa do SDK para SAP ABAP APIs, consulte [SDK para SAP ABAP - Guia de referência de API](#).

A seguir estão os principais indicadores ao importar transportes do AWS SDK.

- Cada transporte é entregue como Knnnnnn.AWS e Rnnnnnn.AWS
  - Knnnnnn.AWS deve ser copiado para /usr/sap/trans/cofiles
  - Rnnnnnn.AWS deve ser copiado para /usr/sap/trans/data.
- Ao importar transportes, você deve selecionar a opção Ignorar versão inválida do componente encontrada em Solicitação de transporte de importação > Opções > Opções de importação.
- Todos os transportes desejados podem ser importados simultaneamente.
- Se importar os transportes separadamente, o transporte core deve ser importado primeiro.
- O nível de liberação de todos os transportes deve ser idêntico.

### Hora de importar

AWS Os transportes do SDK podem levar muitos minutos para serem importados. Os transportes são bem-sucedidos se o STMS mostrar uma luz verde (RC=0) ou amarela (RC=4).

- Uma luz vermelha (RC=8) indica que a importação teve um erro de sintaxe.
  - Selecione Solicitar → Exibir → Logs para examinar o erro de importação.
  - Durante a importação, se ocorrer um erro devido à falta de uma interface IF\_SYSTEM\_UUID\_RFC4122\_STATIC, verifique se o SAP Note 2619546 está aplicado ao sistema. Para obter mais informações, consulte [Notas](#).
  - Se a causa do erro for desconhecida, entre em contato Suporte.
- Um raio vermelho (RC=12) indica que os arquivos de transporte não foram carregados corretamente para o /usr/sap/trans ou não têm as permissões necessárias.

## Ponteiros principais

A seguir estão os principais indicadores ao importar transportes do AWS SDK.

- Cada transporte é entregue como Knnnnnn.AWS e Rnnnnnn.AWS
  - Knnnnnn.AWS deve ser copiado para `/usr/sap/trans/cofiles`
  - Rnnnnnn.AWS deve ser copiado para `/usr/sap/trans/data`.
- Ao importar transportes, você deve selecionar a opção Ignorar versão inválida do componente encontrada em Solicitação de transporte de importação > Opções > Opções de importação.
- Todos os transportes desejados podem ser importados simultaneamente.
- Se importar os transportes separadamente, o transporte core deve ser importado primeiro.
- O nível de liberação de todos os transportes deve ser idêntico.

## Hora da importação

AWS Os transportes do SDK podem levar muitos minutos para serem importados. Os transportes são bem-sucedidos se o STMS mostrar uma luz verde (RC=0) ou amarela (RC=4).

- Uma luz vermelha (RC=8) indica que a importação teve um erro de sintaxe.
  - Selecione Solicitar → Exibir → Logs para examinar o erro de importação.
  - Durante a importação, se ocorrer um erro devido à falta de uma interface `IF_SYSTEM_UUID_RFC4122_STATIC`, verifique se o SAP Note 2619546 foi aplicado ao sistema. Para obter mais informações, consulte [Notas](#).
  - Se a causa do erro for desconhecida, entre em contato Suporte.
- Um raio vermelho (RC=12) indica que os arquivos de transporte não foram carregados corretamente `/usr/sap/trans` ou não têm as permissões necessárias.

## Namespace

O SDK para SAP ABAP usa o namespace `/AWS1/` e não modifica objetos SAP ou quaisquer outros objetos em seu sistema, com a seguinte exceção.

- AWS authos objetos estão em uma classe de objeto Auth. As classes de objetos de autenticação são limitadas a quatro caracteres e não oferecem suporte a namespaces. O SDK para SAP ABAP que usa classe de objeto de autenticação é YAW1. Se você já tiver uma classe de objeto de autenticação YAW1 na transaçãoSU21, entre em contato Suporte antes da instalação.

# Instalando o AWS SDK para SAP ABAP - edição BTP

A edição BTP está na versão prévia para desenvolvedores e pode ser instalada juntando-se à versão prévia. Para instalar o SDK, preencha o formulário de participação na versão prévia para desenvolvedores do [AWS SDK for SAP ABAP - BTP edition](#).

Antes de instalar o SDK para a edição SAP ABAP - BTP, certifique-se de atender aos pré-requisitos exigidos. Para obter mais informações, consulte [SAP Landscape Portal](#) e [SAP Credential Store](#).

## Tópicos

- [Instale o SDK para SAP ABAP - edição BTP](#)
- [Modules](#)
- [SDK de patches para SAP ABAP - edição BTP](#)

## Instale o SDK para SAP ABAP - edição BTP

1. Acesse sua instância do SAP Landscape Portal e inicie o aplicativo Deploy Product fiori.
2. Em Produtos, selecione **/AWS1/SDK\_OMNI** em Produtos de parceiros.

Entre em contato Suporte se você não ver **/AWS1/SDK\_OMNI** depois de ser aceito na prévia para desenvolvedores.

3. Em Target Version, escolha a versão do SDK para SAP ABAP - edição BTP que você deseja instalar em seu sistema.
4. Em Sistemas disponíveis, marque as caixas de seleção de todos os SIDs nos quais você deseja instalar o SDK.
5. Selecione Implantar, insira os detalhes do agendamento e selecione Agendar. Você pode monitorar o progresso no Status de implantação da versão do produto.

A instalação pode levar de 30 a 45 minutos e inclui o tempo de inatividade do sistema. Para obter mais detalhes, consulte [Implantar produto](#).

## Modules

Os módulos a seguir estão incluídos na versão prévia para desenvolvedores do AWS SDK for SAP ABAP - edição BTP.

- [Amazon API Gateway \[agw\]](#)
- [Amazon Athena \[ath\]](#)
- [Amazon Bedrock Runtime \[bdr\]](#)
- [Amazon Comprehend \[cpd\]](#)
- [Amazon EventBridge \[evb\]](#)
- [Previsão da Amazon \[fcs\]](#)
- [Amazon Kinesis \[kns\]](#)
- [Amazon Data Firehose \[frh\]](#)
- [SageMaker IA da Amazon \[sgm\]](#)
- [Serviço de notificação simples da Amazon \[...sns\]](#)
- [Amazon Simple Queue Service \[sqs\]](#)
- [Serviço Amazon Simple Storage \[s3\]](#)
- [AWS Systems Manager \[ssm\]](#)
- [Amazon Textract \[tex\]](#)
- [Amazon Transcribe \[tnb\]](#)
- [Amazon Translate \[x18\]](#)
- [AWS CloudTrail \[tr1\]](#)
- [AWS IoT \[iot\]](#)
- [AWS KMS \[kms\]](#)
- [AWS Lambda \[lmd\]](#)
- [AWS Secrets Manager \[smr\]](#)
- [AWS Security Token Service \[sts\]](#)
- [AWS Transfer Family \[trn\]](#)
- [Funções do IAM em qualquer lugar \[...r1a\]](#)
- [API de dados do Amazon Redshift \[rsd\]](#)

## SDK de patches para SAP ABAP - edição BTP

O processo de correção do SDK for SAP ABAP - edição BTP é semelhante ao processo de instalação. Se você instalar o SDK em um sistema que já tenha uma versão antiga instalada, o SDK será corrigido para a nova versão de sua escolha.

# Configurando AWS SDK para SAP ABAP

Antes de usar AWS SDK para SAP ABAP, você deve configurar o SDK com as configurações técnicas e funcionais necessárias para as operações do SDK. Algumas configurações são transportáveis e outras são configurações de runtime. Muitas das configurações são diretamente análogas às configurações definidas nos .INI arquivos para outros SDKs.

As configurações do SDK, exceto as configurações de tempo de execução, devem ser concluídas em seu ambiente de desenvolvimento. Você pode transportar as configurações para o controle de qualidade e produção seguindo as regras usuais de transporte e controle de alterações. A configuração transportável não é recomendada para ambientes de produção.

Se você não tiver permissões para configurar o AWS SDK, consulte [Autorizações SAP](#).

## Configurando AWS SDK para SAP ABAP

Para executar a transação de configuração, insira /n/AWS1/IMG na barra de comando da SAPGUI.

## Configurando o AWS SDK para SAP ABAP - edição BTP

Use as etapas a seguir para configurar o SDK para a edição SAP ABAP - BTP.

1. Abra seu ambiente ABAP em um navegador da web.
2. Navegue até o aplicativo Custom Business Configurations.

Para criar uma solicitação de personalização usando o aplicativo Export Customizing Transports, consulte [Trabalhando no aplicativo Export Customizing Transports](#) - Create Request.

No aplicativo Custom Business Configuration, você pode agrupar configurações com base no tipo de configuração do SDK. Use as etapas a seguir para agrupar as configurações.

1. Abra seu ambiente ABAP em um navegador da Web e navegue até o aplicativo Custom Business Configurations.
2. Selecione Configurações > Grupo e escolha Grupo de configuração na lista suspensa. Selecione OK.
3. As configurações agora estão disponíveis em uma estrutura hierárquica, conforme exibido na imagem. Para salvar a exibição, consulte [Visualizações \(gerenciamento de variantes\) - Componentes](#).

## Custom Business Configurations (4)

Name	Description		
<b>Application Configuration</b>			
SDK Profile	Maintain AWS SDK Profile	>	
Logical Resource Resolver	Maintain Logical Resource Resolution	>	
<b>Global Settings</b>			
Technical Settings	Maintain Technical Settings	>	
Configure Scenarios	Configure Scenarios	>	

Esta seção abrange os seguintes tópicos.

### Tópicos

- [Configurações globais](#)
- [Configuração do aplicativo](#)
- [Configurações de runtime](#)
- [Cenários avançados de conectividade](#)
- [Configurações do provedor de serviços](#)
- [Tópicos de atualização, rastreamento e telemetria para AWS SDK para SAP ABAP](#)

## Configurações globais

Use a transação `/n/AWS1/IMG IMG` para AWS SDK para SAP ABAP e o aplicativo Custom Business Configuration para AWS SDK for SAP ABAP - edição BTP para definir as configurações globais. Este tópico usa IMG e Configuração Empresarial Personalizada de forma intercambiável.

Esta seção abrange os seguintes tópicos.

## Tópicos

- [Configurações técnicas](#)
- [Configurar cenários](#)

## Configurações técnicas

As configurações globais da transação /AWS1/IMG afetam o comportamento de todo o SDK. Essas configurações geralmente são definidas por um administrador do Basis. Você pode definir esses valores para as seguintes configurações recomendadas.

- Selecione Novas entradas.
  - Regionalização S3: [acesse us-east-1 buckets usando s3.amazonaws.com.](#)
  - Regionalização do STS: acesse o STS usando o endpoint global.
  - Desativar EC2 metadados: mantenha esse campo em branco. Esse campo é somente para leitura na edição BTP e está definido como 'Sim' por padrão.
  - Modo de extremidade de metadados: use um endpoint de IPv4 metadados. Esse campo é somente para leitura na edição BTP e é atualizado automaticamente.
  - URL de envio de metadados: mantenha esse campo em branco. Esse campo é somente para leitura na edição BTP.
- Selecione Salvar.

## Configurar cenários

Os cenários permitem que o AWS SDK altere as configurações com mais eficiência durante um cenário de teste de desastres em várias regiões ou de recuperação de desastres. Talvez você não precise desse atributo e, em vez disso, precise apenas configurar o seguinte cenário DEFAULT.

- Selecione Novas entradas.
  - ID do cenário: DEFAULT
  - Descrição do cenário: Cenário padrão
- Selecione Salvar.

Se você tiver uma configuração de recuperação de desastres em várias regiões ou outros casos exclusivos que exijam uma alteração rápida das configurações, poderá configurar vários cenários.

- DEFAULT - Operação padrão.
- DR - Configuração especial se um desastre exigir a transferência de todo o sistema para outra região.
- DR\_TEST - Configuração especial para simular um desastre, por exemplo, em um clone temporário da produção.

## Configuração do aplicativo

A configuração do SDK para SAP ABAP é semelhante à configuração de outros aplicativos baseados em ABAP. Ele é organizado em diferentes perfis para agrupar as configurações de vários cenários. Um perfil ABAP SDK define as configurações necessárias para um cenário de aplicativo específico. Por exemplo, se as transações ZVA01, ZVA02, e ZVA03 forem transações relacionadas a faturas aprimoradas e executadas Serviços da AWS, como Amazon S3, e AWS Lambda Amazon SageMaker AI, então um perfil de SDK chamado pode ser criado. ZINVOICE Esse perfil pode agrupar as configurações técnicas, as autorizações SAP e os mapeamentos de perfis do IAM para a funcionalidade relacionada à fatura.

Use a /n/AWS1/IMG transação para AWS SDK para SAP ABAP e o aplicativo Custom Business Configuration para AWS SDK for SAP ABAP - edição BTP para definir as configurações globais. Este tópico usa IMG e Configuração Empresarial Personalizada de forma intercambiável.

### Tópicos

- [Perfil do SDK](#)
- [Resolvedor lógico de recursos](#)
- [Exemplo](#)

## Perfil do SDK

Um perfil ABAP SDK define o seguinte para cada SID e cliente.

### Note

O cliente está sempre 100% no ambiente SAP BTP, ABAP.

- A AWS região padrão para todas as chamadas de API. Por exemplo, se seus sistemas SAP estão sendo executados na us-east-1 região, é provável que seus outros AWS recursos também estejam na mesma região, e essa deve ser sua região padrão. Seu código ABAP pode substituir a região padrão.
- Método de autenticação
  - Para sistemas SAP executados no Amazon EC2, é altamente recomendável escolher metadados de perfil de instância para se beneficiar das credenciais de curta duração e rotação automática.
  - Para sistemas SAP executados on-premises ou em outra nuvem, você deve escolher as credenciais do armazenamento SSF.
  - Para sistemas ABAP executados no SAP BTP, você deve escolher as credenciais do SAP Credential Store. Para obter mais informações, consulte [Usando o SAP Credential Store para autenticação](#).
  - Para cenários de encadeamento de funções entre contas, escolha Perfil de origem e especifique uma ID do perfil de origem. Isso permite a resolução e a execução automáticas de cadeias de suposições de funções. Para obter mais informações, consulte [Usando o perfil de origem para acesso entre contas](#).
- Um mapeamento dos perfis lógicos do IAM para os perfis do IAM.
  - Esse mapeamento é classificado em ordem decrescente de prioridade.
  - Um perfil do IAM de maior prioridade para a qual um usuário está autorizado em um perfil PFCG será selecionada automaticamente para o usuário.
- Um mapeamento opcional de serviços para endpoints personalizados. Essa configuração é discutida em [Cenários de conectividade avançada](#)

**Note**

As funções do PFCG são chamadas de funções de negócios no ambiente SAP, BTP e ABAP.

Quando um programa ABAP quiser se conectar a um AWS service (Serviço da AWS), ele especificará um perfil ABAP SDK que extrai as configurações necessárias. Uma AUTHORIZATION-CHECK será realizada para confirmar que o usuário tem permissões para acessar o perfil do SDK.

Seu administrador de segurança SAP pode definir um perfil de PFCG concedendo acesso aos usuários apropriados.

## Resolvedor lógico de recursos

O resolvedor lógico de recursos permite que você tenha um local padrão para armazenar nomes de recursos. Ele vem com o SDK para SAP ABAP. Sua ação é semelhante à forma como a transação de FILE mapeia nomes de arquivos lógicos para nomes de arquivos físicos.

Um recurso lógico define o conceito de um AWS recurso, como o bucket Amazon S3 que contém nossas faturas. Esse recurso lógico, por exemplo, pode ser nomeado ZINVOICES\_OUTBOUND e mapeado para um nome de bucket físico diferente, dependendo se o sistema SAP é de desenvolvimento, controle de qualidade ou produção.

O SDK para SAP ABAP é configurado de forma que um sistema de controle de qualidade resolva recursos lógicos para os recursos físicos de controle de qualidade, mesmo após uma atualização do sistema da produção. Os mapeamentos de recursos para TODOS os sistemas são definidos em seu sistema SAP de desenvolvimento e transportados adiante. Essa abordagem é diferente da configuração usual em sistemas SAP, em que o mapeamento é tratado como dados mestre e definido em cada sistema. A vantagem do resolvedor lógico de recursos oferecido pelo SDK para SAP ABAP é que as chances de um transporte incorreto após as atualizações do sistema são quase nulas.

## Exemplo

Há quatro buckets separados do Amazon S3: um para desenvolvimento, produção e controle de qualidade, além de um segundo bucket de controle de qualidade para testes de regressão.

Quando o SDK resolve um recurso lógico, como ZINVOICE\_OUTBOUND para um recurso físico, ele verifica SY-SYSID e SY-MANDT para perguntar em qual SID e cliente estou executando e seleciona automaticamente o recurso físico correto.

Se o mapeamento de um recurso em produção precisar mudar, você deverá alterar o mapeamento no sistema de desenvolvimento IMG e transportá-lo para frente. Isso garante que a reatribuição de AWS recursos a um sistema SAP esteja sujeita ao controle de alterações, como acontece com qualquer outro transporte.

**Note**

Como a configuração do SDK depende do cliente, a reatribuição de recursos é transportada em uma solicitação de personalização e o transporte deve ser importado para cada cliente.

## Configurações de runtime

Esta seção abrange os seguintes tópicos.

**Note**

Essas configurações não são transportáveis e são locais para cada sistema SAP.

### Tópicos

- [Registrar e rastrear](#)
- [OPT-IN: telemetria aprimorada](#)
- [Cenário ativo](#)

## Registrar e rastrear

Você pode ativar um rastreamento para fins de depuração. É recomendável manter o nível de rastreamento em Sem rastreamento, a menos que seja diagnosticado um problema técnico. Para obter mais informações, consulte [operação segura](#).

Essas configurações não são aplicáveis ao SDK para SAP ABAP - edição BTP.

## OPT-IN: telemetria aprimorada

Todos SDKs enviam informações de telemetria AWS para fins de suporte. Você pode optar por aceitar a telemetria aprimorada. Isso é particularmente útil quando você entra em contato Suporte para identificar a origem de uma chamada de API específica. Para obter mais informações, consulte [Rastreamento](#) e [Telemetria](#).

Essas configurações não são aplicáveis ao SDK para SAP ABAP - edição BTP.

## Cenário ativo

Ative seu cenário DEFAULT nesta transação. Essa ativação é necessária apenas uma vez para cada sistema e não deve ser alterada, a menos que o sistema esteja passando por uma recuperação de desastres em várias regiões. Em uma configuração multirregional, você pode usar essa configuração para mudar seu sistema SAP para um ambiente de recuperação de desastres ou cenários de teste de recuperação de desastres.

## Cenários avançados de conectividade

AWS SDK para SAP ABAP consome Serviços da AWS fazendo chamadas HTTPS para AWS endpoints. Em geral, os endpoints da AWS são acessados pela Internet. Um sistema SAP deve ser capaz de acessar a Internet para estabelecer essas conexões de saída. O SDK para SAP ABAP nunca requer uma conexão de entrada da Internet para o sistema SAP.

Os cenários a seguir oferecem maneiras diferentes de estabelecer a conexão de saída.

### Cenários

- [Conexão por meio de um servidor proxy](#)
- [Conexão por meio de um pacote de inspeção do firewall](#)
- [Endpoints de gateway](#)
- [Endpoints de interface personalizada](#)
- [Roteamento avançado](#)
- [Per-service substituição do servidor proxy](#)
- [Acesso de endpoints em várias regiões](#)

## Conexão por meio de um servidor proxy

Para estabelecer uma conexão por meio de um servidor proxy, use as etapas a seguir.

1. No SDK, acesse Transação **SICF**.
2. Clique em Executar.
3. No menu, escolha Cliente > Servidor proxy.
4. Defina a Configuração de proxy como Ativa.

5. No campo Sem proxy para os seguintes endereços, liste todas as exceções separadas por ponto e vírgula.
6. Nos campos Protocolo HTTP e Protocolo HTTPS, especifique os detalhes da conexão do seu servidor proxy.

O SDK não conhece o servidor proxy e não exige nenhuma configuração para usar a configuração do servidor proxy do sistema SAP.

#### Note

Se você usa a [autenticação de metadados de instância do Amazon EC2](#), o sistema SAP não pode usar o servidor proxy para acessar os metadados da instância local em `http://169.254.169.254`. Você deve incluir `169.254.169.254` no campo Sem proxy para os seguintes endereços.

#### Note

Você pode substituir o comportamento do servidor proxy por serviço na seção Roteamento Avançado do. /AWS1/IMG Para obter mais informações, consulte [the section called "Per-service substituição do servidor proxy"](#).

## Conexão por meio de um pacote de inspeção do firewall

Você pode configurar um pacote de inspeção do firewall para conexão de saída. Esses firewalls descriptografam o tráfego SSL e, em seguida, o criptografam novamente antes de passá-lo para o endpoint. Essa configuração geralmente exige que o firewall emita seus próprios certificados para o sistema SAP que está consumindo um AWS service (Serviço da AWS). Você deve instalar o certificado CA do seu firewall em STRUST. Para obter mais informações, consulte [Conectividade HTTPS](#).

## Endpoints de gateway

Alguns Serviços da AWS oferecem endpoints de gateway para fornecer a uma VPC acesso de alto desempenho sem internet. Esses endpoints são transparentes para o SDK para SAP ABAP e não exigem nenhuma configuração.

Para obter mais informações, consulte [Endpoints de gateway](#).

## Endpoints de interface personalizada

Se precisar substituir a resolução padrão do endpoint por um endpoint personalizado, use um endpoint de interface para fornecer à sua VPC acesso de alto desempenho sem Internet. Para obter mais informações, consulte [Configurar um endpoint de interface](#).

Quando não estão usando DNS privado, esses endpoints têm seus próprios endereços DNS, e um programa ABAP deve substituir explicitamente a lógica usual de resolução de endpoints. Para obter mais informações, consulte AWS re:Post — [Por que não consigo resolver nomes de domínio de serviço para um endpoint VPC de interface?](#)

No exemplo a seguir, um endpoint de interface é criado para AWS STS e Amazon Translate. O sistema SAP não está usando DNS privado e chama os serviços com um endpoint personalizado. Os recursos lógicos definidos em `/AWS1/IMG` representam os endereços dos endpoints da interface física, como `vpce-0123456789abcdef-hd52vxz.translate.us-west-2.vpce.amazonaws.com`. Isso evita a codificação rígida do DNS no código.

No código a seguir, os recursos lógicos no `/AWS1/IMG` são primeiro resolvidos em nomes de endpoints físicos. Em seguida, eles são fornecidos aos métodos de fábrica da classe de AWS sessão (que usa AWS STS para assumir uma função do IAM) e da classe de API de tradução.

```
" This example assumes we have defined our logical endpoints in /AWS1/IMG
" as logical resources so that we don't hardcode our endpoints in code.
" The endpoints may be different in Dev, QA and Prod environments.
DATA(lo_config) = /aws1/cl_rt_config=>create( 'DEMO' ).
DATA(lo_resolver) = /aws1/cl_rt_lresource_resolver=>create( lo_config ).

" logical resource STS_ENDPOINT should resolve to the interface endpoint
" for example vpce-0123456789-abcdefg.sts.us-west-2.vpce.amazonaws.com
DATA(lv_sts_endpoint) = lo_resolver->resolve_lresource( 'STS_ENDPOINT' ).

" logical resource XL8_ENDPOINT should resolve to the interface endpoint
" e.g. vpce-0123456789abcdefg-12345567.translate.us-west-2.vpce.amazonaws.com
DATA(lv_xl8_endpoint) = lo_resolver->resolve_lresource( 'XL8_ENDPOINT' ).

" the session itself uses the sts service to assume a role, so the
" session creation process requires a custom endpoint, specified here
DATA(lo_session) = /aws1/cl_rt_session_aws=>create(
    iv_profile_id = 'DEMO'
```

```

iv_custom_sts_endpoint = |https://{ lv_sts_endpoint }|
).

" now we create an API object, and override the default endpoint with
" the custom endpoint
DATA(lo_xl8)      = /aws1/cl_xl8_factory=>create(
  io_session = lo_session
  iv_custom_endpoint = |https://{ lv_xl8_endpoint }| " provide custom endpoint
).
" now calls to lo_xl8 go to custom endpoint...

```

Conforme mostrado no exemplo, qualquer chamada de método no `go_xl8` vai para o endpoint `https://vpce-0123456789abcdefg-12345567.translate.us-west-2.vpce.amazonaws.com`. Também é possível definir o endpoint personalizado de roteamento na configuração IMG em vez de no código, conforme mostrado na próxima seção.

## Roteamento avançado

Na seção anterior, mostramos como um endpoint personalizado pode ser especificado no `iv_custom_endpoint` argumento dos métodos de fábrica para os módulos do SDK. À medida que o número de programas ABAP usando o SDK aumenta, isso pode se tornar difícil de gerenciar. É possível configurar um mapeamento de um AWS service (Serviço da AWS) para um endpoint personalizado no perfil do SDK. Para cada SID, cliente e cenário, a abreviação de três letras do serviço (TLA) pode ser mapeada para uma URL de endpoint:

TLA	URL de endpoint personalizado
BDR	<code>https://vpce-23456789abcdef012-3c4d5e6f.bedrock-runtime.us-east-1.vpce.amazonaws.com</code>
LMD	<code>https://vpce-123456789abcdef01-2b3c4d5e.lambda.us-east-1.vpce.amazonaws.com</code>
S3	<code>https://vpce-0123456789abcdef0-1a2b3c4d.s3.us-east-1.vpce.amazonaws.com</code>

Com essa configuração, você não precisa especificar `iv_custom_endpoint` nas chamadas do método de fábrica. O endpoint personalizado é selecionado automaticamente na tabela de

configuração. A configuração é específica para o perfil do SDK para que você possa criar vários perfis com roteamento diferente para atender às suas necessidades. Assim como em outras configurações do SDK Profile, o roteamento é específico do SID e do cliente, portanto, um roteamento separado pode ser definido para sistemas diferentes.

## Per-service substituição do servidor proxy

Por padrão, o SDK usa as configurações do servidor proxy definidas em Transação SICF (consulte [the section called “Conexão por meio de um servidor proxy”](#)). As configurações de proxy em SICF se aplicam globalmente a todas as conexões HTTP de saída do sistema SAP. Em alguns ambientes, você pode precisar de um controle mais preciso sobre quais Serviços da AWS usam o servidor proxy e quais se conectam diretamente.

Na seção Roteamento Avançado de /AWS1/IMG, você pode definir uma configuração Usar Servidor Proxy para cada serviço. Essa configuração controla se o SDK encaminha as solicitações desse serviço por meio do servidor proxy definido em SICF, independentemente das configurações globais de ativação ou filtro do proxy.

Os seguintes valores estão disponíveis:

- Padrão — Use qualquer comportamento de proxy configurado em SICF. Se o proxy estiver ativo e o endpoint não for excluído pelo filtro, o proxy será usado. Esse é o comportamento padrão.
- Sempre — Sempre encaminhe as solicitações desse serviço por meio do servidor proxy definido em SICF, independentemente das configurações globais de ativação ou filtro.
- Nunca — Nunca encaminhe solicitações desse serviço por meio do servidor proxy, independentemente das configurações globais de ativação ou filtro. Use isso quando um endpoint de serviço puder ser acessado diretamente sem um proxy, por exemplo, ao usar um endpoint VPC.

Por exemplo, você pode configurar o Amazon S3 e nunca AWS STS usar o proxy (porque eles são acessados por meio de endpoints de gateway VPC), enquanto o Amazon SNS sempre usa o proxy porque ele só pode ser acessado pela Internet.

TLA	Use o servidor proxy
EC2	Padrão
S3	Nunca

TLA	Use o servidor proxy
SNS	Sempre
STS	Nunca

Essa configuração é definida por perfil, SID, cliente e cenário do SDK. Você pode criar vários perfis com roteamento de proxy diferente para se adequar a diferentes ambientes ou casos de uso.

### Note

O host e a porta do servidor proxy são sempre definidos na guia Protocolo HTTPS das configurações de proxy em TransaçãoSICF. A substituição por serviço controla apenas se o proxy é usado, não qual servidor proxy é usado.

## Acesso de endpoints em várias regiões

AWS O endpoint é determinado automaticamente a partir do padrão Região da AWS definido no perfil do SDK. Você também pode especificar uma região programaticamente, substituindo a região padrão. Isso pode ser substituído no método CREATE() de fábrica ou posteriormente com o objeto de configuração do SDK. Para obter mais informações, consulte [Configuração programática](#).

No exemplo a seguir, o método de fábrica CREATE() é usado para definir a região e listar as filas do Amazon SQS nas regiões us-east-1 e us-west-2.

```
REPORT zdemo_sqs_queue_list.  
parameters: profile type /AWS1/RT_PROFILE_ID OBLIGATORY.  
  
START-OF-SELECTION.  
DATA(go_session) = /aws1/cl_rt_session_aws=>create( profile ).  
data(lt_region) = VALUE stringtab(  
  ( |us-east-1| )  
  ( |us-west-2| )  
).  
  
LOOP AT lt_region INTO DATA(lv_region).  
  DATA(go_sqs) = /aws1/cl_sqs_factory=>create(  
    io_session = go_session  
    iv_region = conv /AWS1/RT_REGION_ID( lv_region )
```

```
) .  
WRITE: / lv_region COLOR COL_HEADING.  
LOOP AT go_sqs->listqueues( )->get_queueurls( ) INTO DATA(lo_url).  
    WRITE: / lo_url->get_value( ).  
ENDLOOP.  
ENDLOOP.
```

## Configurações do provedor de serviços

Às vezes, os administradores do Basis precisam controlar certos recursos do SDK em todo o sistema, a partir do cliente. 000 Esse é um cenário comum para provedores de hospedagem e serviços que operam sistemas em suas próprias AWS contas em nome de seus clientes. AWS O SDK para SAP ABAP oferece suporte às configurações do provedor de serviços. Essas configurações são definidas no cliente 000 e afetam o SDK em todos os clientes. As configurações do provedor de serviços não são suportadas no SDK para SAP ABAP - edição BTP.

As configurações do provedor de serviços são definidas na transação /AWS1/IMG e devem ser definidas no cliente000. As configurações do provedor de serviços em outros clientes são ignoradas. As configurações no cliente entram em 000 vigor em todos os clientes e substituem outras IMG configurações em caso de conflito.

Use as etapas a seguir para definir as configurações do provedor de serviços no cliente000.

1. Expanda a ramificação Configurações do provedor de serviços na transação/AWS1/IMG.
2. Escolha as grades de proteção do provedor de serviços
3. Selecione Novas entradas e ajuste as configurações com base nos requisitos da sua empresa.
  - Desativar EC2 metadados — impede que o SDK acesse os metadados da EC2 instância em todos os clientes, mesmo que um perfil do SDK esteja configurado para autenticação usando metadados da instância. EC2 O SDK gera uma exceção se um programa ABAP tentar acessar os metadados da instância usando o SDK.
4. Selecione Salvar.

## Tópicos de atualização, rastreamento e telemetria para AWS SDK para SAP ABAP

Esta seção abrange os seguintes tópicos.

## Tópicos

- [Atualização do sistema SAP](#)
- [Rastreamento](#)
- [Telemetria](#)

## Atualização do sistema SAP

Depois de uma atualização do sistema, o principal desafio para um administrador do Basis é garantir que os sistemas separados não estejam acessando os recursos uns dos outros. Por exemplo, talvez você queira garantir que seu sistema SAP de controle de qualidade não esteja acessando os recursos, como um bucket S3, do seu cenário de produção.

O SDK para SAP ABAP fornece uma abordagem segura dos recursos lógicos para esse desafio. Um analista de negócios pode realizar as seguintes etapas:

1. Definir um recurso lógico, como ZINVOICE\_OUTBOUND.
2. Mapear todos os sistemas e clientes no sistema de desenvolvimento.
3. Transportar a configuração de TODOS os sistemas até o cenário de produção.

### Etapas básicas após uma atualização

1. Verifique a autenticação
  - Se o sistema estiver usando a autenticação de chave de acesso secreta, as credenciais criptografadas por SSF serão inválidas porque estão armazenadas nos dados principais. As credenciais devem ser inseridas novamente, o que pode exigir a regeneração de uma nova chave de acesso secreta. <https://console.aws.amazon.com/iam/>
  - Se o sistema estiver se autenticando com metadados da EC2 instância, nenhuma etapa será necessária.

### Verifique as configurações de rastreamento

- No /AWS1/IMG, verifique se as configurações de rastreamento são as desejadas. Essas configurações não são transportáveis.

## Rastreamento

A saída do rastreamento é controlada nas configurações de runtime do IMG.

Os níveis de rastreamento que você pode usar são:

- Sem rastreamento
- Chamadas de API de rastreamento
- Chamadas de API de rastreamento e carga útil

Essa opção contém informações de carga não criptografadas.

- Chamadas de API de rastreamento, carga útil e transformação interna de XML

Essa opção contém informações de carga não criptografadas.

Se o rastreamento da API for ativado, os rastreamentos serão gravados no DIR\_WORK no arquivo `aws1_trace-YYYY-MM-DD.log`.

Se o rastreamento de carga útil também for ativado, arquivos adicionais com o título `aws1_payload_*` serão criados para cada chamada e componente de carga útil. O comprimento do rastreamento da carga útil pode ser limitado pelo comprimento aplicado a cada falha individual do rastreamento da carga útil.

Os rastreamentos de carga útil têm como objetivo principal coletar informações a serem fornecidas Suporte no caso de um erro de serialização. Recomendamos que você escolha Sem rastreamento, a menos que esteja tentando diagnosticar um erro do SDK.

### Note

Os rastreamentos de carga podem conter informações comerciais não criptografadas. Recomendamos ativar esses rastreamentos somente para uma solicitação do AWS Support para ajudá-lo a solucionar problemas. Você pode desativar esses rastreamentos após a resolução. Os rastreamentos não são excluídos automaticamente e precisam ser removidos pelo administrador do sistema quando não são mais necessários.

Essas configurações não são aplicáveis ao SDK para SAP ABAP - edição BTP.

## Telemetria

SDKs enviar informações de telemetria para. Suporte O SDK para SAP ABAP coleta as seguintes informações:

- Versão do sistema operacional e nível de patch
- Nível de lançamento e patch do SAP\_BASIS
- Versão do SAP Kernel e nível de patch

Você pode optar por enviar as seguintes informações para Suporte.

- SAP SID e nome da instância (host\_sid\_nn)
- Cliente SAP (SY-MANDT)
- Código da transação (SY-TCODE) e relatório (SY-REPID)

As informações adicionais Suporte permitem ajudá-lo melhor. Suporte pode detectar por que uma determinada chamada de API foi feita e pode encontrar ainda mais a transação relevante em um sistema SAP.

A telemetria é limitada às versões SDK e API do SDK for SAP ABAP - edição BTP.

# Usando AWS SDK para SAP ABAP

O SDK para SAP ABAP tem dois componentes principais.

- SDK Runtime (pacote /AWS1/RT): um conjunto de objetos que sustentam a segurança, a autenticação, o rastreamento, a configuração, a conversão de dados e outras funções entre APIs. Os módulos de API para Amazon S3 AWS STS, IAM Roles Anywhere e Secrets Manager são obrigatórios.
- APIs (pacote /AWS1/API e seus subpacotes) — um subpacote para cada API em que os objetos de cada API são completamente independentes uns dos outros, garantindo que uma alteração em uma API não interrompa outra API. Para ver uma lista completa de AWS SDK para SAP ABAP APIs, consulte [AWS SDK para SAP ABAP - Guia de referência da API](#).

Esta seção abrange os seguintes tópicos.

## Tópicos

- [Representação de dados no ABAP](#)
- [Exemplo de programa do Amazon S3](#)
- [Conceitos do SDK para SAP ABAP](#)
- [AWS SDK para SAP ABAP features](#)
- [Criar produtos com SDK](#)
- [Personalize solicitações HTTP para AWS](#)
- [Limitações](#)

## Representação de dados no ABAP

Esta seção abrange os seguintes tópicos.

## Tópicos

- [Tipos de dados](#)
- [AWS tipos de dados](#)

## Tipos de dados

Serviços da AWS têm um conjunto padrão de tipos de dados que devem ser mapeados para tipos de dados ABAP. Para obter mais detalhes, consulte a tabela a seguir.

AWS tipo de dados	Tipo de dados do ABAP	Comentários
boolean	C	Caractere único "X" e " "
String	STRING	
Byte	INT2	INT2 tem um intervalo maior que 0-255. A maioria Serviços da AWS truncará os estouros, mas esse comportamento não está definido formalmente.
Short	INT2	
Inteiro	INT4	
Longo	DEC19	INT8 não está disponível até o ABAP 750. DEC19 é usado para fins de compatibilidade e consistência em todas as plataformas ABAP suportadas.
Blob	XSTRING	Representa dados binários
Float	STRING	Embora o ABAP ofereça suporte DECFLOATs, ele não pode representar valores como NaN, Infinity e -Infinity . AWS O SDK os representa internamente como STRINGs e os converte em em tempo de execução. DECFLOAT1 6 Se NaN, Infinity ou +Infinity estiverem representados,
Duplo	STRING	

AWS tipo de dados	Tipo de dados do ABAP	Comentários
		o desenvolvedor poderá processá-los em resposta a um conjunto especial de exceções ou mapeamentos.
bigInteger	STRING	Esses valores representam números de comprimento infinito que não podem ser representados em ABAP e STRINGS são usados em vez de BigInteger.
bigDecimal	STRING	
Timestamp	TZNTSTMPS	TZNTSTMPS permite o processamento com funções nativas de carimbo de data/hora ABAP.

Serviços da AWS também retorne os seguintes tipos de dados agregados.

AWS tipo de dados	Tipo de dados do ABAP	Comentários
Estrutura	Classe	
Union	Classe	Uma união é o mesmo que uma estrutura, exceto que uma união nunca terá mais de um conjunto de campos. Todos os outros campos serão definidos como Sem valor.
Array	TABELA PADRÃO	
Hash	TABELA COM HASH	A tabela com hash terá apenas duas colunas: uma

AWS tipo de dados	Tipo de dados do ABAP	Comentários
		KEY (string) e uma VALUE (classe).

## AWS tipos de dados

As seguintes abordagens foram integradas para dar suporte Serviços da AWS ao ABAP.

- Certos tipos de AWS dados não podem ser representados no ABAP. Por exemplo, o tipo de dados float em ABAP não suporta os valores NaN, Infinity ou -Infinity. Portanto, o tipo de dados float é representado como STRING e traduzido para DECFLOAT16 em runtime.
- AWS os dados são representados no fio como JSON ou XML, e os valores são opcionais. Por exemplo, veja os exemplos a seguir retornados por um AWS service (Serviço da AWS) em JSON.

```
Fullname: {  
  Firstname: "Ana",  
  Middlename: "Carolina",  
  Lastname: "Silva"  
}
```

Se Ana não tiver um segundo nome, o serviço retornará a seguinte saída.

```
Fullname: {  
  Firstname: "Ana",  
  Lastname: "Silva"  
}
```

O ABAP não distingue entre uma string de comprimento 0 e uma string que não tem valor. Outras linguagens podem atribuir um valor NULL à string ou envolvê-la em uma estrutura (como o wrapper `Optional` do Java). Eles não são suportados no ABAP. Portanto, o SDK for SAP ABAP facilita a distinção de valores ao fornecer variantes do método getter.

# Exemplo de programa do Amazon S3

Esta seção mostra um exemplo de programa simples para listar o conteúdo de um bucket do Amazon S3 por meio de uma chamada de `ListObjectsV2`.

## Tópicos

- [Pré-requisitos](#)
- [Código](#)
- [Seções de código](#)

## Pré-requisitos

Você deve cumprir os seguintes pré-requisitos para executar este programa de exemplo.

- Você tem um bucket do Amazon S3. Neste tutorial, o bucket tem um nome `demo-invoices.customer.com`.
- Transação `/AWS1/IMG`:
  - Tem um perfil SDK definido chamado `DEMO_S3`.
    - No perfil do SDK, o perfil lógico do IAM TESTUSER deve ser mapeado para um perfil do IAM, como a `arn:aws:iam::111122223333:role/SapDemoFinance` que concede a permissão `s3:ListBucket` para listar o conteúdo do seu bucket do Amazon S3.
  - Tem um recurso lógico chamado `DEMO_BUCKET` que é mapeado para seu bucket do Amazon S3 com o SID e o cliente do seu sistema SAP.
- Seu usuário tem uma função PFCG que:
  - Autoriza o usuário a acessar o perfil do SDK `DEMO_S3` por meio do objeto de autenticação `- /AWS1/SESS`.
  - Autoriza o usuário a acessar o perfil lógico do IAM TESTUSER por meio do objeto de autenticação `- /AWS1/LROL`.
- Seu sistema SAP pode se autenticar AWS usando o método definido no perfil do SDK.
- Seu perfil de EC2 instância da Amazon concede ao seu sistema SAP os direitos sobre a `sts:assumeRole` função IAM `arn:aws:iam::111122223333:role/SapDemoFinance` mapeada no perfil do SDK.

## Código

O bloco de código a seguir mostra como deve ser a aparência do seu código.

```
REPORT zdemo_s3_listbuckets.

START-OF-SELECTION.
  PARAMETERS pv_lres TYPE /aws1/rt_resource_logical
                DEFAULT 'DEMO_BUCKET' OBLIGATORY.

  DATA(go_session) = /aws1/cl_rt_session_aws=>create( 'DEMO_S3' ).
  DATA(gv_bucket)  = go_session->resolve_lresource( pv_lres ).

  DATA(go_s3)      = /aws1/cl_s3_factory=>create( go_session ).

  TRY.
    DATA(lo_output) = go_s3->listobjectsv2(
      iv_bucket = CONV string( gv_bucket )
      iv_maxkeys = 100
    ).
    LOOP AT lo_output->get_contents( ) INTO DATA(lo_object).
      DATA lv_mdate TYPE datum.
      CONVERT TIME STAMP lo_object->get_lastmodified( )
        TIME ZONE 'UTC'
        INTO DATE lv_mdate.
      WRITE: / CONV text30( lo_object->get_key( ) ),
              lv_mdate, lo_object->get_size( ).
    ENDLOOP.
  CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
    DATA(lv_msg) = lo_ex->if_message~get_text( ).
    MESSAGE lv_msg TYPE 'I'.
  ENDTRY.
```

## Seções de código

A seguir, é apresentada uma análise do código nas seções.

```
PARAMETERS pv_lres TYPE /aws1/rt_resource_logical
                DEFAULT 'DEMO_BUCKET' OBLIGATORY.
```

O usuário não pode especificar um nome de bucket físico. Eles especificam um bucket lógico e os administradores do sistema (especificamente o analista de negócios), em coordenação com o administrador da AWS, mapeiam os buckets lógicos para os buckets físicos no /AWS1/IMG. Na maioria dos cenários de negócios, o usuário não tem a chance de escolher o bucket lógico: o ID do recurso lógico tem codificação rígida ou configurado em uma tabela de configuração personalizada.

```
DATA(go_session) = /aws1/cl_rt_session_aws=>create( 'DEMO_S3' ).
```

Essa linha estabelece uma sessão de segurança e declara que esse programa ABAP espera usar o perfil do SDK DEMO\_S3. Essa chamada é a conexão com a configuração do SDK e extrai a região padrão, as configurações de autenticação e o perfil do IAM desejado. Uma chamada para AUTHORIZATION-CHECK é feita automaticamente para garantir que o objeto de autorização /AWS1/SESS seja satisfeito. Além disso, serão feitas chamadas AUTHORIZATION-CHECK para determinar o perfil lógico do IAM mais poderoso (número de sequência inferior) para o qual o usuário está autorizado, com base no objeto de autorização /AWS1/LROL. O SDK assumirá que o perfil do IAM está mapeado para o perfil lógico do IAM para o SID e o cliente. Em seguida, o objeto da sessão ativa o rastreamento com base nas configurações de rastreamento no IMG.

Se o usuário não estiver autorizado para o perfil do SDK solicitado ou para qualquer perfil lógico do IAM disponível, uma exceção será gerada.

```
DATA(gv_bucket) = go_session->resolve_lresource( pv_lres ).
```

Essa linha resolve o recurso lógico para um nome de bucket físico. Se o recurso lógico não puder ser resolvido porque a configuração não tem mapeamento para essa combinação de SID/cliente, uma exceção será gerada.

```
DATA(go_s3) = /aws1/cl_s3_factory=>create( go_session ).
```

Essa linha cria um objeto de API para o Amazon S3 usando o método create() de /aws1/cl\_s3\_factory. O objeto retornado é do tipo /aws1/if\_s3 que é a interface de uma API do Amazon S3. Um objeto de API separado deve ser criado para cada serviço. Por exemplo, se um programa ABAP estiver consumindo Amazon S3 e DynamoDB AWS Lambda, ele criará objetos de API a partir de, e. /aws1/cl\_s3\_factory /aws1/cl\_lmd\_factory /aws1/cl\_dyn\_factory

Existem alguns parâmetros opcionais para a estrutura que permitem especificar a região se você quiser substituir a região padrão configurada IMG. Dessa forma, pode haver duas instâncias de /aws1/uf\_s3, uma para us-east-1 e outra para us-west-2, se você quiser copiar objetos de um bucket em uma região para um bucket em outra região. Da mesma forma, você pode criar dois objetos de sessão de segurança diferentes e usá-los para criar duas instâncias separadas de /aws1/c1\_s3, se precisar de um relatório para ler de um bucket relacionado a finanças e gravar objetos em um bucket relacionado à logística.

```
DATA(lo_output) = go_s3->listobjectsv2(  
    iv_bucket = CONV string( gv_bucket )  
    iv_maxkeys = 100  
).
```

Esta linha é uma chamada para ListObjectsV2. Ela exige argumentos de entrada simples e retorna um único objeto. Esses objetos podem representar dados JSON e XML profundos, desserializados em uma estrutura orientada a objetos ABAP. Isso pode ser bastante complicado em alguns casos. Agora você só precisa processar a saída para listar o conteúdo do bucket.

```
LOOP AT lo_output->get_contents( ) INTO DATA(lo_object).  
    DATA lv_mdate TYPE datum.  
    CONVERT TIME STAMP lo_object->get_lastmodified( )  
        TIME ZONE 'UTC'  
        INTO DATE lv_mdate.  
    WRITE: / CONV text30( lo_object->get_key( ) ),  
        lv_mdate, lo_object->get_size( ).  
ENDLOOP.
```

Os dados são acessados usando um método de estilo GET...() que oculta a representação interna dos dados. GET\_CONTENTS() retorna uma tabela ABAP e cada linha contém um objeto representando uma única entrada do Amazon S3. Na maioria dos casos, o AWS SDK adota essa abordagem orientada a objetos e todos os dados são representados como objetos e tabelas. O LastModified campo é representado como um timestamp que pode ser convertido em uma data com o CONVERT TIME STAMP comando ABAP-native. Ele GET\_SIZE() retorna um INT4 para facilitar as operações matemáticas e de formatação.

```
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
```

```
DATA(lv_msg) = lo_ex->if_message~get_text( ).  
MESSAGE lv_msg TYPE 'I'.
```

Todos os erros, conexão, cliente 4xx, servidor 5xx ou qualquer erro ABAP, como erros de autorização ou configuração, são representados como exceções. Você pode lidar com cada exceção separadamente. Você pode escolher se uma exceção deve ser tratada como um erro informativo, uma nova tentativa, um aviso, um atalho ou qualquer outro tipo de tratamento.

## Conceitos do SDK para SAP ABAP

Esta seção aborda os conceitos básicos de AWS SDK para SAP ABAP.

### Tópicos

- [Classes de API](#)
- [Objetos adicionais](#)
- [Classes de estrutura](#)
- [Matrizes](#)
- [Mapas](#)
- [Funções de nível superior](#)

## Classes de API

Cada um AWS service (Serviço da AWS) recebe um acrônimo de três letras ou. TLA O serviço é representado por uma interface no formato /AWS1/IF\_<TLA>. Chamaremos isso de interface de serviço. A classe de API está no pacote /AWS1/API\_<TLA>. A interface de serviço consiste em um método para cada AWS operação (chamaremos esses métodos de Métodos de Operação). Para ver uma lista completa de módulos AWS SDK para SAP ABAP TLAs, consulte [AWS SDK para SAP ABAP - Lista de módulos](#).

Cada método de operação tem alguns argumentos IMPORTING e, no máximo, um argumento RETURNING. Normalmente, esses argumentos serão objetos com construtores complicados e um longo conjunto de métodos GET...(). Em muitos casos, os objetos conterão objetos aninhados, referências recursivas, tabelas de objetos, tabelas de tabelas e assim por diante. Isso ocorre porque Serviços da AWS estão passando estruturas XML e JSON profundas, que não podem ser representadas por um conjunto simples de argumentos.

O RETURNING argumento é sempre uma classe, mesmo que a classe contenha somente um único atributo.

## Objetos adicionais

Além de conter a classe de API primária, cada pacote de API contém vários objetos relacionados de repositório e dicionário de dados.

- Uma classe para cada objeto do tipo estrutura.
- Uma classe para qualquer tipo de dados primitivo que aparece em uma tabela. Por exemplo, se um serviço retornar uma tabela de strings, a API ABAP a representará como uma tabela de objetos, em que cada objeto é uma classe wrapper que encapsula uma string. Isso é para que a classe wrapper possa ocultar os detalhes da representação de uma string nula que não pode ser representada nativamente no ABAP.
- Uma classe de exceção para quaisquer erros específicos definidos pelo serviço.
- Elementos de dados para cada tipo de dados primitivo. Cada tipo de dados tem seu próprio elemento de dados para ser autodocumentado.
- Objetos adicionais para processamento interno, como transformações XSLT para serializar e desserializar cargas XML e JSON.

## Classes de estrutura

A maioria dos AWS dados enviados e recebidos pelo serviço é representada pelo AWS SDK como classes. Essas classes representam estruturas de dados e ocultam os detalhes internos do armazenamento. Em particular, as classes ocultam que a forma como o SDK representa esse campo não tem valor.

Para cada campo em uma classe de estrutura, há três métodos.

### **GET\_field( )**

O método GET\_field( )

- Retorna o valor do campo ou
- Se o campo não tiver valor, ele retornará um valor padrão, que você pode definir como um parâmetro opcional.

Por exemplo, considere o código a seguir que imprime a restrição de localização de um bucket.

```
DATA(lo_location) = go_s3->getbucketlocation( iv_bucket = CONV string( gv_bucket ) ).  
WRITE: / 'Bucket Location: ',  
       lo_location->get_locationconstraint( ).
```

Se o bucket não tiver nenhuma restrição de localização (como no caso de `us-east-1`), ele `GET_LOCATIONCONSTRAINT( )` retornará a string vazia. Você pode substituir esse comportamento e especificar o valor desejado se o campo não tiver nenhum valor.

```
DATA(lo_location) = go_s3->getbucketlocation( iv_bucket = CONV string( gv_bucket ) ).  
WRITE: / 'Bucket Location: ',  
       lo_location->get_locationconstraint( iv_value_if_missing = 'assuming us-east-1' ).
```

Agora o programa gravará `Bucket Location: assuming us-east-1` se o resultado de `getbucketlocation( )` não retornar um local.

É possível solicitar que o método `GET( )` retorne um resultado específico se o valor solicitado estiver completamente ausente, veja o exemplo de código a seguir.

```
data(lo_location) = go_s3->GETBUCKETLOCATION(  
    new /AWS1/CL_S3_GET_BUCKET_LOC_REQ( iv_bucket = gv_bucket )  
).  
write: / 'Location constraint: ',  
       lo_location->GET_LOCATIONCONSTRAINT( 'NopeNopeNope' ).
```

Nesse caso, se não houver restrição de localização, `GET_LOCATIONCONSTRAINT( )` retornará `NopeNopeNope`.

## **HAS\_field( )**

O método `HAS_field( )` é uma forma de descobrir se o campo tem um valor ou não. Veja o exemplo a seguir.

```
if NOT lo_location->HAS_LOCATIONCONSTRAINT( ).  
    write: / 'There is no location constraint'.  
endif.
```

Se um determinado campo sempre tiver um valor, não haverá método `HAS_field( )`.

## **ASK\_field( )**

O método `ASK_field( )` retorna o valor do campo ou gera uma exceção se não tiver valor. Essa é uma maneira conveniente de processar vários campos, sair da lógica e adotar uma abordagem diferente se algum dos campos não tiver valor.

```
TRY.  
    WRITE: / 'Location constraint: ', lo_location->ask_locationconstraint( ).  
CATCH /aws1/cx_rt_value_missing.  
    WRITE: / 'Never mind, there is no location constraint'.  
ENDTRY.
```

Observe que `/AWS1/CX_RT_VALUE_MISSING` é uma exceção estática e você receberá um aviso se optar por não pegá-la.

### Práticas recomendadas

Em geral, você pode usar o método `GET_field( )`, pois ele trata uma string nula como uma string vazia e é a mais parecida com ABAP das três opções. No entanto, ele não permite distinguir facilmente entre situações em que o campo tem um valor em branco e em que o campo não tem valor. Se sua lógica de negócios depende da distinção entre dados ausentes e dados em branco, os métodos `ASK` ou `HAS` permitem lidar com esses casos.

## Matrizes

As matrizes são representadas como tabelas de objetos padrão ABAP.

Uma matriz JSON pode conter valores nulos, como a seguinte matriz: `['cat', 'dog', null, 'horse']`. Isso é chamado de matriz esparsa. Ele é representado no ABAP como uma tabela interna de referências de objetos e o valor `null` é representado na tabela como um valor ABAP `null` verdadeiro. Ao iterar por meio de uma tabela esparsa, você deve verificar os valores `null` para evitar acessar um objeto `null` e obter uma exceção `CX_SY_REF_IS_INITIAL`. Na prática, matrizes esparsas são raras em serviços AWS.

Para inicializar uma matriz de objetos, é conveniente usar as novas estruturas do ABAP 7.40. Considere o lançamento de uma EC2 instância da Amazon com vários grupos de segurança atribuídos:

```
ao_ec2->runinstances(  
    iv_imageid                = lo_latest_ami->get_imageid( )  
    iv_instancetype           = 't2.micro'  
    iv_maxcount               = 1  
    iv_mincount               = 1  
    it_securitygroupids       = VALUE /aws1/  
cl_ec2secgrpiddstringlist_w=>tt_securitygroupidstringlist(  
    ( NEW /aws1/  
cl_ec2secgrpiddstringlist_w( 'sg-12345678' ) )  
    ( NEW /aws1/  
cl_ec2secgrpiddstringlist_w( 'sg-55555555' ) )  
    ( NEW /aws1/  
cl_ec2secgrpiddstringlist_w( 'sg-99999999' ) )  
    )  
    iv_subnetid              = ao_snet->get_subnetid( )  
    it_tagsspecifications     = make_tag_spec( 'instance' )  
)
```

## Mapas

Os mapas JSON são representados em ABAP como Hashed Tables onde cada linha da tabela tem apenas dois componentes.

- KEY: uma string que é a UNIQUE KEY da tabela.
- VALUE: um objeto contendo o valor.

Um mapa é um dos poucos casos em que o AWS SDK usa uma estrutura verdadeira, em vez de uma classe. Isso é necessário porque as tabelas com hash ABAP não podem ter uma referência de objeto como o campo chave, e as chaves de AWS mapa são sempre cadeias de caracteres não nulas.

## Funções de nível superior

O [Classes de API](#) descrito na seção anterior reflete com precisão o AWS serviço APIs e os representa APIs como classes ABAP familiares. Em alguns casos, o SDK também inclui funções de nível superior que se baseiam nas classes da API para simplificar determinadas operações. As funções de nível superior são incluídas para conveniência do programador e não substituem as classes de API de nível inferior.

Se o SDK incluir funções de nível superior para um módulo, elas serão incluídas no mesmo transporte e poderão ser acessadas por meio de uma classe de fábrica chamada `/AWS1/CL_TLA_L2_FACTORY`. A classe de fábrica inclui métodos para criar vários clientes de nível superior para o módulo que são documentados junto com o restante da API com a [documentação da API](#).

## AWS SDK para SAP ABAP features

AWS SDK para SAP ABAP fornece os seguintes recursos.

### Tópicos

- [Configuração programática](#)
- [Waiters](#)
- [Paginadores](#)
- [Comportamento de repetição](#)
- [Prescritores](#)
- [Encadeamento de funções do IAM entre contas](#)

## Configuração programática

Use a transação `/n/AWS1/IMG IMG` para AWS SDK para SAP ABAP e o aplicativo Custom Business Configuration para AWS SDK for SAP ABAP - edição BTP para configuração programática.

Para iniciar a configuração programática, comece recuperando um objeto de configuração com o comando `get_config( )`.

```
data(lo_config) = lo_s3->get_config( ).
```

Cada objeto de configuração implementa a interface `/AWS1/IF_RT_CONFIG` que inclui GETters e SETters correspondentes ao IMG. Por exemplo, a região padrão pode ser substituída. No exemplo de comando a seguir:

```
lo_s3->get_config( )->aws1/if_rt_config~set_region( 'us-east-1' ).
```

Alguns objetos de configuração não têm representação IMG e só podem ser definidos programaticamente, como o máximo de tentativas. No exemplo de comando a seguir:

```
lo_s3->get_config( )->/aws1/if_rt_config~set_max_attempts( 10 ).
```

O objeto de configuração do também Serviços da AWS pode incluir métodos específicos de serviço que não estão representados em `/aws1/if_rt_config`. Por exemplo, o Amazon S3 pode endereçar um bucket chamado foobucket usando um endpoint virtual `foobucket.s3.region.amazonaws.com` virtual ou um estilo de caminho `s3.region.amazonaws.com/foobucket`. Você pode impor o uso do estilo de caminho com o comando de exemplo a seguir.

```
lo_s3->get_config( )->set_forcepathstyle( abap_true ).
```

Para obter mais informações sobre configurações de serviço, consulte [AWS SDK para SAP ABAP — Guia de referência da API](#).

## Waiters

Ao trabalhar com o modo assíncrono AWS APIs, você precisa esperar que um determinado recurso fique disponível antes de realizar outras ações. Por exemplo, a `CREATETABLE()` API do Amazon DynamoDB responde imediatamente com o status `CREATING` da tabela. Você pode iniciar operações de leitura ou gravação somente depois que o status da tabela for alterado para `ACTIVE`. Os garçons permitem que você confirme se AWS os recursos estão em um determinado estado antes de realizar ações neles.

Os garçons usam operações de serviço para pesquisar o status dos AWS recursos até que o recurso atinja o estado pretendido ou até que seja determinado que o recurso não atinge o estado desejado. Escrever o código para pesquisar recursos da AWS continuamente pode ser demorado e propenso a erros. Os waiters ajudam a simplificar essa complexidade, assumindo a responsabilidade de realizar pesquisas em seu nome.

Veja o exemplo a seguir do Amazon S3 usando o waiter.

```
DATA(lo_session) = /aws1/cl_rt_session_aws=>create( cv_pfl ).
DATA(lo_s3) = /aws1/cl_s3_factory=>create( lo_session ).

" Create a bucket - initiates the process of creating an S3 bucket and might return
  before the bucket exists
lo_s3#createbucket( iv_bucket = |amzn-s3-demo-bucket| ).
```

```
" Wait until the newly created bucket becomes available
lo_s3->get_waiter( )->bucketexists(
    iv_max_wait_time = 200
    iv_bucket = |amzn-s3-demo-bucket|
).
```

- Neste exemplo, o cliente Amazon S3 é usado para criar um bucket. O comando `get_waiter()` é implementado para especificar quando o `bucketexists`.
- Você deve especificar o parâmetro `iv_max_wait_time` para cada `waiter`. Representa a quantidade total de tempo que um `waiter` deve esperar antes de terminar. No exemplo anterior, um `waiter` pode correr por 200 segundos.
- Talvez seja necessário fornecer entradas adicionais para os parâmetros necessários. No exemplo anterior, o nome do bucket do Amazon S3 é obrigatório para o parâmetro `iv_bucket`.
- A exceção `/AWS1/CX_RT_WAITER_FAILURE` indica que o `waiter` excedeu o tempo máximo especificado no parâmetro `iv_max_wait_time`.
- A exceção `/AWS1/CX_RT_WAITER_TIMEOUT` indica que o `waiter` parou por não atingir o estado desejado.

## Paginadores

Algumas AWS service (Serviço da AWS) operações oferecem respostas paginadas. Elas são paginadas para retornar uma quantidade fixa de dados com cada resposta. Você precisa fazer solicitações subsequentes com um token ou marcador para recuperar todo o conjunto de resultados. Por exemplo, a operação `ListObjectsV2` do Amazon S3 retorna só até 1.000 objetos por vez. Você deve fazer solicitações subsequentes com o token apropriado para obter a próxima página de resultados.

Paginação é o processo de enviar solicitações sucessivas para continuar de onde uma solicitação anterior parou. Paginadores são iteradores de resultados fornecidos pelo SDK para SAP ABAP. Você pode usar paginado APIs com facilidade e sem entender o mecanismo subjacente da API usando tokens de paginação.

### Trabalhando com paginadores

Você pode criar paginadores com o método `get_paginator()` que retorna um objeto paginador. O objeto paginador chama a operação que está sendo paginada. O objeto paginador aceita

os parâmetros obrigatórios a serem fornecidos à API subjacente. Esse processo retorna um objeto iterador que pode ser usado para iterar sobre resultados paginados, usando os métodos `has_next()` e `get_next()`.

- `has_next()` retorna um valor booleano indicando se há mais respostas ou páginas disponíveis para a operação chamada.
- `get_next()` retorna a resposta da operação.

O exemplo a seguir lista todos os objetos em um bucket do S3 recuperados usando o paginador.

```
DATA(lo_session) = /aws1/cl_rt_session_aws=>create( 'DEMO' ).
DATA(lo_s3) = /aws1/cl_s3_factory=>create( lo_session ).

TRY.
  DATA(lo_paginador) = lo_s3->get_paginador( ).
  DATA(lo_iterator) = lo_paginador->listobjectsv2(
    iv_bucket = 'example_bucket'
  ).
  WHILE lo_iterator->has_next( ).
    DATA(lo_output) = lo_iterator->get_next( ).
    LOOP AT lo_output->get_contents( ) INTO DATA(lo_object).
      WRITE: / lo_object->get_key( ), lo_object->get_size( ).
    ENDLLOOP.
  ENDWHILE.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
  MESSAGE lo_ex->if_message~get_text( ) TYPE 'I'.
ENDTRY.
```

## Comportamento de repetição

O SDK para SAP ABAP permite que você configure o número máximo de novas tentativas de solicitações de Serviços da AWS que falham devido a erros transitórios ou de controle de utilização. O número de novas tentativas permitidas no nível do cliente de serviço, ou seja, o número de vezes que o SDK repete a operação antes de falhar e gerar uma exceção, é especificado pelo atributo `AV_MAX_ATTEMPTS` no objeto de configuração do serviço. Quando um objeto de cliente de serviço é criado, o SDK configura o atributo `AV_MAX_ATTEMPTS` com um valor padrão de 3. O objeto de configuração do serviço pode ser usado para definir programaticamente o número máximo de novas tentativas para um valor desejado. Consulte mais detalhes no exemplo a seguir.

```
" Retrieve configuration object using Amazon S3 service's get_config( ) method
```

```
DATA(lo_config) = lo_s3->get_config( ).

" Set the maximum number of retries to 5
lo_config->/aws1/if_rt_config~set_max_attempts( 5 ).

" Get the value of the maximum retry attempt.
DATA(lv_max_retry_attempts) = lo_config->/aws1/if_rt_config~get_max_attempts( ).
```

### Note

Embora o objeto de configuração do ABAP SDK permita que o modo de nova tentativa seja definido com o método `/AWS1/IF_RT_CONFIG~SET_RETRY_MODE()`, o SDK só permite o modo de nova tentativa `standard`. Para obter mais informações, consulte o [comportamento de repetição](#) AWS SDKs e o Guia de referência de ferramentas.

## Prescritores

Você pode usar presigned URLs para conceder acesso por tempo limitado a alguns. Serviços da AWS Um URL pré-assinado pode ser inserido em um navegador ou usado por um programa para realizar a operação do serviço. É possível usar o pre-signed URL várias vezes, até a data e a hora de expiração. Para obter mais informações, consulte [Trabalhar com URLs SDK pré-assinado](#) para clientes SAP ABAP para serviços que oferecem suporte a presignatários terá um método especial chamado criar um presignatário `GET_PRESIGNER()` para esse serviço. Em seguida, chame um método do presigner, que corresponde aos métodos do cliente da API, exceto que eles retornam uma URL pré-assinada em vez de realmente realizar a operação.

```
" Retrieve a presigner for Amazon S3
DATA(lo_presigner) = lo_s3->get_presigner( iv_expires_sec = 600 ).

" the presigner getObject() method has the same signature as
" lo_s3->getObject(), but it doesn't actually make the call.
" to the service. It just prepares a presigned URL for a future call
DATA(lo_presigned_req) = lo_presigner->getObject( iv_bucket = iv_bucket_name iv_key =
  iv_key ).

" You can provide this URL to a web page, user, email etc so they
" can retrieve the file. The URL will expire in 10 minutes.
ov_url = lo_presigned_req->get_url( ).
```

## Encadeamento de funções do IAM entre contas

O suporte ao encadeamento de funções do IAM entre contas permite acesso contínuo aos recursos em várias AWS contas por meio da configuração do perfil de origem. Esse recurso permite que você configure várias suposições de função, em que um perfil assume uma função e, em seguida, assume outra função, permitindo padrões complexos de acesso entre contas.

Para obter mais informações, consulte [Usando o perfil de origem para acesso entre contas](#).

## Criar produtos com SDK

Um produto ou complemento ABAP que consome Serviços da AWS pode aprimorar e ampliar os recursos do SDK. É possível criar esses produtos para usar com o SDK.

Tópicos

- [Configurar um ID de produto](#)

## Configurar um ID de produto

É recomendável que você defina um ID de produto ao estabelecer uma sessão dentro de um produto ou complemento. Consulte mais detalhes no exemplo a seguir.

```
DATA(lo_session) = /aws1/cl_rt_session_aws=>create( 'DEMO' ).  
lo_session->set_product_id( 'INVOICE_ANALYZER' ).
```

O ID do produto deve conter somente letras, números e sublinhados sem espaços ou caracteres especiais. Você pode associá-lo ao nome técnico do produto ou a qualquer outro identificador. Se você desenvolver vários produtos ou complementos, o ID de produto deve ser exclusivo para cada produto. Por exemplo, o produto IDs para os produtos Invoice Analyzer, Tax Calculator e Pricing Engine pode ser INVOICE\_ANALYZERTAX\_CALCULATOR, e. PRICING\_ENGINE

Adicionar uma ID do produto à sessão aprimora a telemetria enviada AWS com cada chamada de serviço. O ID de produto e o namespace do objeto que está fazendo a chamada estão incluídos na telemetria. Com essa telemetria, Suporte você pode identificar o produto que está fazendo a chamada caso seu cliente enfrente problemas com o SDK. Isso pode ajudar a esclarecer que a chamada está realmente sendo feita pelo produto, e não pelo código do cliente.

# Personalize solicitações HTTP para AWS

O AWS SDK para SAP ABAP manipula o processo de criar uma solicitação HTTP, enviar uma carga e receber uma resposta. Você pode personalizar o comportamento ou o conteúdo da solicitação HTTP para atender aos seus próprios requisitos de TI. O SDK define o ponto de aprimoramento / AWS1/RT\_EHN\_HTTP\_CLIENT como um local central para aprimorar a comunicação HTTP. O ponto de aprimoramento suporta a adição de cabeçalhos HTTP à solicitação feita para AWS.

## Implemente um aprimoramento

A SAP fornece as seguintes instruções para implementar um ponto de aprimoramento:

- [ABAP clássico](#)
- [BTP ABAP](#)

## Filtrar o aprimoramento

O ponto de aprimoramento oferece suporte a várias implementações que podem ser ativas simultaneamente. Você pode filtrar a execução do BAdi com base nos seguintes atributos, se precisar garantir que seu aprimoramento seja executado somente em chamadas para um AWS serviço específico ou ação de API:

- TLA- A abreviatura de três letras do serviço, em letras maiúsculas.
- OPERATION- O nome da ação da API. Por exemplo, a operação para obter um objeto de um bucket do S3 é [GetObject](#). O nome da ação diferencia maiúsculas de minúsculas e pode não corresponder exatamente ao nome do método ABAP.

## Codifique o aprimoramento

O aprimoramento fornece o método a seguir.

### MODIFY\_REQ\_HEADERS

```
CHANGING CT_HEADERS TYPE /AWS1/RT_STRINGMAP_TT
```

Você pode acrescentar e modificar cabeçalhos na tabela interna. CT\_HEADERS Não recomendamos modificar os cabeçalhos, pois isso altera os dados que o AWS serviço usa. Todos os cabeçalhos

adicionados são ignorados pelo AWS serviço, mas podem ser processados pela sua infraestrutura de TI, como servidores proxy ou outro middleware.

O ponto de aprimoramento é chamado antes do cálculo dos cabeçalhos de autenticação e telemetria, portanto, eles não podem ser modificados pelo aprimoramento.

Veja a seguir um exemplo de implementação.

```
METHOD /aws1/if_rt_badi_http_client~modify_req_headers.  
  APPEND VALUE /aws1/rt_stringpair_ts( name = 'x-test-example' value = 'value' )  
  TO ct_headers.  
ENDMETHOD.
```

## Limitações

AWS SDK para SAP ABAP inclui módulos SDK para todos Serviços da AWS. Alguns desses módulos podem ter limitações, conforme descrito aqui.

- Módulos que dependem de vinculações de MQTT protocolo, como `iotevents`, não funcionarão. MQTT não é um protocolo baseado em HTTP e atualmente não é suportado pelo AWS SDK para SAP ABAP
- As operações que retornam fluxos de eventos são suportadas, mas esperarão até que todo o fluxo seja recebido antes de retornar o controle ao ABAP. Exemplos são `Amazon Bedrock Agents Runtime InvokeAgent` e `AWS Lambda InvokeWithResponseStream`
- As operações que recebem fluxos de eventos não são suportadas devido a limitações na plataforma ABAP subjacente. Exemplos são `Amazon Q Business Chat` e `Amazon Lex StartConversation`

AWS SDK para SAP ABAP tem as seguintes limitações de recursos.

- Os seguintes atributos do Amazon S3 não são suportados.
  - Pontos de acesso multirregionais
  - Criptografia do lado do cliente do Amazon S3

AWS O SDK para SAP ABAP - A edição BTP tem as seguintes limitações durante a versão prévia para desenvolvedores.

- Alguns módulos podem não estar disponíveis.
- Ele não pode ser desinstalado.
- Ele é atualizado com menos frequência.

# Exemplos de código SDK para SAP ABAP

Os exemplos de código neste tópico mostram como usar o AWS SDK para SAP ABAP com. AWS

As noções básicas são exemplos de código que mostram como realizar as operações essenciais em um serviço.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Alguns serviços contêm categorias de exemplo adicionais que mostram como utilizar bibliotecas ou funções específicas do serviço.

## Services

- [Exemplos de ACM usando SDK para SAP ABAP](#)
- [Exemplos de API Gateway usando SDK para SAP ABAP](#)
- [Exemplos de controladores de recuperação de aplicativos usando SDK para SAP ABAP](#)
- [Exemplos do Aurora usando o SDK para SAP ABAP](#)
- [Exemplos de Auto Scaling usando SDK para SAP ABAP](#)
- [Exemplos de Amazon Bedrock Runtime usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon Bedrock Agents Runtime usando o SDK para SAP ABAP](#)
- [CloudFront exemplos usando SDK para SAP ABAP](#)
- [CloudWatch exemplos usando SDK para SAP ABAP](#)
- [CloudWatch Exemplos de registros usando o SDK para SAP ABAP](#)
- [Exemplos de provedores de identidade do Amazon Cognito usando SDK para SAP ABAP](#)
- [Exemplos do Amazon Comprehend usando o SDK para SAP ABAP](#)
- [AWS Config exemplos usando SDK para SAP ABAP](#)
- [AWS Control Tower exemplos usando SDK para SAP ABAP](#)
- [Exemplos de Firehose usando o SDK para SAP ABAP](#)

- [Exemplos do DynamoDB usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon EC2 usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon ECR usando SDK para SAP ABAP](#)
- [Exemplos do Amazon EMR usando SDK para SAP ABAP](#)
- [EventBridge Exemplos de agendador usando o SDK para SAP ABAP](#)
- [AWS Glue exemplos usando SDK para SAP ABAP](#)
- [HealthImaging exemplos usando o SDK para SAP ABAP](#)
- [HealthLake exemplos usando o SDK para SAP ABAP](#)
- [Exemplos de IAM usando SDK para SAP ABAP](#)
- [AWS IoT data exemplos usando o SDK para SAP ABAP](#)
- [AWS IoT SiteWise exemplos usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon Keyspaces usando SDK para SAP ABAP](#)
- [Exemplos do Kinesis usando o SDK para SAP ABAP](#)
- [AWS KMS exemplos usando o SDK para SAP ABAP](#)
- [Exemplos do Lambda usando o SDK para SAP ABAP](#)
- [Exemplos de organizações usando o SDK for SAP ABAP](#)
- [Exemplos do Amazon Pinpoint usando SDK para SAP ABAP](#)
- [Exemplos de API de voz e SMS do Amazon Pinpoint usando SDK para SAP ABAP](#)
- [Exemplos do Amazon Polly usando SDK para SAP ABAP](#)
- [Exemplos do Amazon RDS usando SDK para SAP ABAP](#)
- [Exemplos do Amazon Redshift usando SDK para SAP ABAP](#)
- [Exemplos do Amazon Rekognition usando SDK para SAP ABAP](#)
- [Exemplos do Amazon S3 usando o SDK para SAP ABAP](#)
- [SageMaker Exemplos de IA usando o SDK para SAP ABAP](#)
- [Exemplos de Secrets Manager usando SDK para SAP ABAP](#)
- [Exemplos do Amazon SES usando SDK para SAP ABAP](#)
- [Exemplos da API v2 do Amazon SES usando SDK para SAP ABAP](#)
- [Exemplos do Amazon SNS usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon SQS usando o SDK para SAP ABAP](#)

- [Exemplos de Step Functions usando o SDK para SAP ABAP](#)
- [Exemplos do Systems Manager usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon Textract usando o SDK para SAP ABAP](#)
- [Exemplos do Amazon Transcribe usando SDK para SAP ABAP](#)
- [Exemplos do Amazon Translate usando o SDK para SAP ABAP](#)

## Exemplos de ACM usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com ACM.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### AddTagsToCertificate

O código de exemplo a seguir mostra como usar AddTagsToCertificate.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

TRY.

```

    " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/
certificate-id'
    lo_acm->addtagstocertificate(
        iv_certificatearn = iv_certificate_arn
        it_tags = it_tags
    ).
    MESSAGE 'Tags added to certificate successfully.' TYPE 'I'.
CATCH /aws1/cx_acminvalidarnex.
    MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.
CATCH /aws1/cx_acmresourcenotfoundex.
    MESSAGE 'Certificate not found.' TYPE 'I'.
CATCH /aws1/cx_acminvalidtagex.
    MESSAGE 'Invalid tag provided.' TYPE 'I'.
CATCH /aws1/cx_acmtoomanytagsex.
    MESSAGE 'Too many tags for certificate.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [AddTagsToCertificate](#) referência da API AWS SDK for SAP ABAP.

## DeleteCertificate

O código de exemplo a seguir mostra como usar DeleteCertificate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/
certificate-id'
    lo_acm->deletecertificate( iv_certificatearn = iv_certificate_arn ).
    MESSAGE 'Certificate deleted successfully.' TYPE 'I'.
CATCH /aws1/cx_acminvalidarnex.
    MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.
CATCH /aws1/cx_acmresourcenotfoundex.

```

```
MESSAGE 'Certificate not found.' TYPE 'I'.
CATCH /aws1/cx_acmresourceinuseex.
MESSAGE 'Certificate is in use and cannot be deleted.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteCertificate](#) referência da API AWS SDK for SAP ABAP.

## DescribeCertificate

O código de exemplo a seguir mostra como usar DescribeCertificate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/
certificate-id'
    oo_result = lo_acm->describecertificate( iv_certificatearn =
iv_certificate_arn ).
    MESSAGE 'Certificate details retrieved.' TYPE 'I'.
CATCH /aws1/cx_acminvalidarnex.
    MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.
CATCH /aws1/cx_acmresourcefoundex.
    MESSAGE 'Certificate not found.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeCertificate](#) referência da API AWS SDK for SAP ABAP.

## GetCertificate

O código de exemplo a seguir mostra como usar GetCertificate.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/  
certificate-id'  
    oo_result = lo_acm->getcertificate( iv_certificatearn =  
iv_certificate_arn ).  
    MESSAGE 'Certificate body and chain retrieved.' TYPE 'I'.  
    CATCH /aws1/cx_acminvalidarnex.  
    MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.  
    CATCH /aws1/cx_acmresourcenotfoundex.  
    MESSAGE 'Certificate not found.' TYPE 'I'.  
    CATCH /aws1/cx_acmrequestinprgssex.  
    MESSAGE 'Certificate request is in progress.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetCertificate](#) referência da API AWS SDK for SAP ABAP.

**ImportCertificate**

O código de exemplo a seguir mostra como usar ImportCertificate.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " Only pass certificate chain if it's provided (it's optional)
```

```
IF iv_certificate_chain IS NOT INITIAL.  
  DATA(lo_result) = lo_acm->importcertificate(  
    iv_certificate = iv_certificate  
    iv_privatekey = iv_private_key  
    iv_certificatechain = iv_certificate_chain  
  ).  
ELSE.  
  lo_result = lo_acm->importcertificate(  
    iv_certificate = iv_certificate  
    iv_privatekey = iv_private_key  
  ).  
ENDIF.  
ov_certificate_arn = lo_result->get_certificatearn( ).  
MESSAGE 'Certificate imported successfully.' TYPE 'I'.  
CATCH /aws1/cx_acminvalidparameterex.  
  MESSAGE 'Invalid parameter provided.' TYPE 'I'.  
CATCH /aws1/cx_acmlimitexceededex.  
  MESSAGE 'Certificate limit exceeded.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ImportCertificate](#) referência da API AWS SDK for SAP ABAP.

## ListCertificates

O código de exemplo a seguir mostra como usar ListCertificates.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_acm->listcertificates(  
    iv_maxitems = iv_max_items  
    it_certificatestatuses = it_statuses  
    io_includes = io_includes
```

```

    ).
    MESSAGE 'Certificates listed successfully.' TYPE 'I'.
  CATCH /aws1/cx_acminvalidargsex.
    MESSAGE 'Invalid arguments provided.' TYPE 'I'.
  CATCH /aws1/cx_acmvalidationex.
    MESSAGE 'Validation error occurred.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [ListCertificates](#) referência da API AWS SDK for SAP ABAP.

## ListTagsForCertificate

O código de exemplo a seguir mostra como usar ListTagsForCertificate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

  TRY.
    " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/
certificate-id'
    DATA(lo_result) = lo_acm->listtagsforcertificate(
      iv_certificatearn = iv_certificate_arn
    ).
    ot_tags = lo_result->get_tags( ).
    MESSAGE 'Certificate tags retrieved successfully.' TYPE 'I'.
  CATCH /aws1/cx_acminvalidarnex.
    MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.
  CATCH /aws1/cx_acmresourcenotfoundex.
    MESSAGE 'Certificate not found.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [ListTagsForCertificate](#) referência da API AWS SDK for SAP ABAP.

## RemoveTagsFromCertificate

O código de exemplo a seguir mostra como usar `RemoveTagsFromCertificate`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/  
certificate-id'  
    lo_acm->removetagsfromcertificate(  
        iv_certificatearn = iv_certificate_arn  
        it_tags = it_tags  
    ).  
    MESSAGE 'Tags removed from certificate successfully.' TYPE 'I'.  
CATCH /aws1/cx_acminvalidarnex.  
    MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.  
CATCH /aws1/cx_acmresourcenotfoundex.  
    MESSAGE 'Certificate not found.' TYPE 'I'.  
CATCH /aws1/cx_acminvalidtagex.  
    MESSAGE 'Invalid tag provided.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [RemoveTagsFromCertificate](#) referência da API AWS SDK for SAP ABAP.

## RequestCertificate

O código de exemplo a seguir mostra como usar `RequestCertificate`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  " iv_domain_name = 'example.com'
  " iv_validation_method = 'DNS' or 'EMAIL'
  DATA(lo_result) = lo_acm->requestcertificate(
    iv_domainname = iv_domain_name
    it_subjectalternativenames = COND #( WHEN it_alterate_domains IS NOT
INITIAL
                                     THEN it_alterate_domains )
    iv_validationmethod = iv_validation_method
  ).
  ov_certificate_arn = lo_result->get_certificatearn( ).
  MESSAGE 'Certificate requested successfully.' TYPE 'I'.
CATCH /aws1/cx_acminvalidparameterex.
  MESSAGE 'Invalid parameter provided.' TYPE 'I'.
CATCH /aws1/cx_acmlimitexceeddex.
  MESSAGE 'Certificate limit exceeded.' TYPE 'I'.
CATCH /aws1/cx_acminvdomvationoptsex.
  MESSAGE 'Invalid domain validation options.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [RequestCertificate](#) referência da API AWS SDK for SAP ABAP.

## ResendValidationEmail

O código de exemplo a seguir mostra como usar ResendValidationEmail.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_certificate_arn = 'arn:aws:acm:region:123456789012:certificate/
certificate-id'
  " iv_domain = 'example.com'
  " iv_validation_domain = 'example.com'
  lo_acm->resendvalidationemail(
    iv_certificatearn = iv_certificate_arn
    iv_domain = iv_domain
    iv_validationdomain = iv_validation_domain
  ).
  MESSAGE 'Validation email resent successfully.' TYPE 'I'.
CATCH /aws1/cx_acminvalidarnex.
  MESSAGE 'The certificate ARN is not valid.' TYPE 'I'.
CATCH /aws1/cx_acmresourcenotfoundex.
  MESSAGE 'Certificate not found.' TYPE 'I'.
CATCH /aws1/cx_acminvalidstateex.
  MESSAGE 'Certificate is not in a valid state.' TYPE 'I'.
CATCH /aws1/cx_acminvdomvationoptsex.
  MESSAGE 'Invalid domain validation options.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ResendValidationEmail](#) referência da API AWS SDK for SAP ABAP.

## Exemplos de API Gateway usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com API Gateway.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateDeployment

O código de exemplo a seguir mostra como usar CreateDeployment.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_agw->createdeployment(  
        iv_restapiid = iv_rest_api_id  
        iv_stagename = iv_stage_name  
        iv_description = 'Deployment created by ABAP SDK' ).  
    DATA(lv_deployment_id) = oo_result->get_id( ).  
    MESSAGE 'Deployment created with ID: ' && lv_deployment_id TYPE 'I'.  
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
    MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
    MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_not_found.  
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
    MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_too_many.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateDeployment](#) referência da API AWS SDK for SAP ABAP.

## CreateResource

O código de exemplo a seguir mostra como usar CreateResource.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_agw->createresource(  
        iv_restapiid = iv_rest_api_id  
        iv_parentid = iv_parent_id  
        iv_pathpart = iv_resource_path ).  
    DATA(lv_resource_id) = oo_result->get_id( ).  
    MESSAGE 'Resource created with ID: ' && lv_resource_id TYPE 'I'.  
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
    MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
    MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_not_found.  
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
    MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateResource](#) referência da API AWS SDK for SAP ABAP.

## CreateRestApi

O código de exemplo a seguir mostra como usar CreateRestApi.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_agw->createrestapi(  
        iv_name = iv_api_name  
        iv_description = 'Sample REST API created by ABAP SDK' ).  
    DATA(lv_api_id) = oo_result->get_id( ).  
    MESSAGE 'REST API created with ID: ' && lv_api_id TYPE 'I'.  
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
    MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
    MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_too_many.  
CATCH /aws1/cx_agwunauthorizedex INTO DATA(lo_unauthorized).  
    MESSAGE lo_unauthorized->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_unauthorized.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateRestApi](#) referência da API AWS SDK for SAP ABAP.

## DeleteRestApi

O código de exemplo a seguir mostra como usar DeleteRestApi.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_agw->deleterestapi(  
    iv_restapiid = iv_rest_api_id ).  
  MESSAGE 'REST API deleted successfully' TYPE 'I'.  
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
  MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
  RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
  MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
  RAISE EXCEPTION lo_not_found.  
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
  MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
  RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteRestApi](#) referência da API AWS SDK for SAP ABAP.

## GetResources

O código de exemplo a seguir mostra como usar GetResources.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

oo_result = lo_agw->getresources(
  iv_restapiid = iv_rest_api_id ).
DATA(lt_resources) = oo_result->get_items( ).
DATA(lv_count) = lines( lt_resources ).
MESSAGE 'Found ' && lv_count && ' resources' TYPE 'I'.
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).
  MESSAGE lo_bad_request->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_bad_request.
CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).
  MESSAGE lo_not_found->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_not_found.
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).
  MESSAGE lo_too_many->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_too_many.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetResources](#) referência da API AWS SDK for SAP ABAP.

## GetRestApis

O código de exemplo a seguir mostra como usar GetRestApis.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  oo_result = lo_agw->getrestapis( ).
  DATA(lt_apis) = oo_result->get_items( ).
  DATA(lv_count) = lines( lt_apis ).
  MESSAGE 'Found ' && lv_count && ' REST APIs' TYPE 'I'.
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).
  MESSAGE lo_bad_request->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_bad_request.
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).

```

```
MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetRestApis](#) referência da API AWS SDK for SAP ABAP.

## PutIntegration

O código de exemplo a seguir mostra como usar PutIntegration.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_agw->putintegration(  
        iv_restapiid = iv_rest_api_id  
        iv_resourceid = iv_resource_id  
        iv_httpmethod = iv_http_method  
        iv_type = 'AWS_PROXY'  
        iv_integrationhttpmethod = 'POST'  
        iv_uri = iv_integration_uri ).  
    MESSAGE 'Integration configured for method' TYPE 'I'.  
    CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
        MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
        RAISE EXCEPTION lo_bad_request.  
    CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
        MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
        RAISE EXCEPTION lo_not_found.  
    CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
        MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
        RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutIntegration](#) referência da API AWS SDK for SAP ABAP.

## PutIntegrationResponse

O código de exemplo a seguir mostra como usar PutIntegrationResponse.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_agw->putintegrationresponse(  
        iv_restapiid = iv_rest_api_id  
        iv_resourceid = iv_resource_id  
        iv_httpmethod = iv_http_method  
        iv_statuscode = '200' ).  
    MESSAGE 'Integration response configured for status 200' TYPE 'I'.  
    CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
    MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request.  
    CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
    MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_not_found.  
    CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
    MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutIntegrationResponse](#) referência da API AWS SDK for SAP ABAP.

## PutMethod

O código de exemplo a seguir mostra como usar PutMethod.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_agw->putmethod(  
        iv_restapiid = iv_rest_api_id  
        iv_resourceid = iv_resource_id  
        iv_httpmethod = iv_http_method  
        iv_authorizationtype = 'NONE' ).  
    MESSAGE 'Method ' && iv_http_method && ' added to resource' TYPE 'I'.  
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
    MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
    MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_not_found.  
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
    MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutMethod](#) referência da API AWS SDK for SAP ABAP.

## PutMethodResponse

O código de exemplo a seguir mostra como usar PutMethodResponse.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_agw->putmethodresponse(  
        iv_restapiid = iv_rest_api_id  
        iv_resourceid = iv_resource_id  
        iv_httpmethod = iv_http_method  
        iv_statuscode = '200' ).  
    MESSAGE 'Method response configured for status 200' TYPE 'I'.  
CATCH /aws1/cx_agwbadrequestex INTO DATA(lo_bad_request).  
    MESSAGE lo_bad_request->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_agwnotfoundexception INTO DATA(lo_not_found).  
    MESSAGE lo_not_found->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_not_found.  
CATCH /aws1/cx_agwtoomanyrequestsex INTO DATA(lo_too_many).  
    MESSAGE lo_too_many->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_too_many.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutMethodResponse](#) referência da API AWS SDK for SAP ABAP.

## Exemplos de controladores de recuperação de aplicativos usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Application Recovery Controller.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### GetRoutingControlState

O código de exemplo a seguir mostra como usar `GetRoutingControlState`.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

CONSTANTS cv_pfl TYPE /aws1/rt_profile_id VALUE 'ZCODE_DEMO'.
DATA lo_exception TYPE REF TO /aws1/cx_rt_generic.
DATA lo_session TYPE REF TO /aws1/cl_rt_session_base.
DATA lo_client TYPE REF TO /aws1/if_r5v.
DATA lt_endpoints TYPE TABLE OF string.
DATA lv_endpoint TYPE string.
DATA lv_region TYPE /aws1/rt_region_id.

" Parse the comma-separated cluster endpoints
" Expected format: "https://endpoint1.com|us-west-2,https://endpoint2.com|us-
east-1"
SPLIT iv_cluster_endpoints AT ',' INTO TABLE lt_endpoints.

" As a best practice, shuffle cluster endpoints to distribute load
" For more information, see https://docs.aws.amazon.com/r53recovery/latest/dg/
route53-arc-best-practices.html#route53-arc-best-practices.regional
" For simplicity, we'll try them in order (shuffling can be added if needed)

" Try each endpoint in order
LOOP AT lt_endpoints INTO lv_endpoint.
  TRY.
    " Parse endpoint and region from the format "url|region"
    DATA(lv_pos) = find( val = lv_endpoint sub = '|' ).
    IF lv_pos > 0.
      DATA(lv_url) = substring( val = lv_endpoint len = lv_pos ).
      lv_region = substring( val = lv_endpoint off = lv_pos + 1 ).
    ELSE.
      " If no region specified, use default

```

```
        lv_url = lv_endpoint.  
        lv_region = 'us-east-1'.  
    ENDIF.  
  
    " Create session for this region  
    lo_session = /aws1/cl_rt_session_aws=>create( cv_pfl ).  
  
    " Create client with the specific endpoint  
    lo_client = create_recovery_client(  
        iv_endpoint = lv_url  
        iv_region    = lv_region  
        io_session   = lo_session ).  
  
    " Try to get the routing control state  
    oo_result = lo_client->getroutingcontrolstate(  
        iv_routingcontrolarn = iv_routing_control_arn ).  
  
    " If successful, return the result  
    RETURN.  
  
CATCH /aws1/cx_r5vendpttmpyunavailex INTO DATA(lo_endpoint_ex).  
    " This endpoint is temporarily unavailable, try the next one  
    lo_exception = lo_endpoint_ex.  
    CONTINUE.  
  
CATCH /aws1/cx_r5vaccessdeniedex  
        /aws1/cx_r5vinternalserverex  
        /aws1/cx_r5vresourcenotfoundex  
        /aws1/cx_r5vthrottlingex  
        /aws1/cx_r5vvalidationex  
        /aws1/cx_rt_generic INTO lo_exception.  
    " For other errors, re-raise immediately  
    RAISE EXCEPTION lo_exception.  
ENDTRY.  
ENDLOOP.  
  
" If we get here, all endpoints failed - re-raise the last exception  
IF lo_exception IS BOUND.  
    RAISE EXCEPTION lo_exception.  
ENDIF.
```

- Para obter detalhes da API, consulte a [GetRoutingControlState](#) referência da API AWS SDK for SAP ABAP.

## UpdateRoutingControlState

O código de exemplo a seguir mostra como usar UpdateRoutingControlState.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

CONSTANTS cv_pfl TYPE /aws1/rt_profile_id VALUE 'ZCODE_DEMO'.
DATA lo_exception TYPE REF TO /aws1/cx_rt_generic.
DATA lo_session TYPE REF TO /aws1/cl_rt_session_base.
DATA lo_client TYPE REF TO /aws1/if_r5v.
DATA lt_endpoints TYPE TABLE OF string.
DATA lv_endpoint TYPE string.
DATA lv_region TYPE /aws1/rt_region_id.

" Parse the comma-separated cluster endpoints
" Expected format: "https://endpoint1.com|us-west-2,https://endpoint2.com|us-
east-1"
SPLIT iv_cluster_endpoints AT ',' INTO TABLE lt_endpoints.

" As a best practice, shuffle cluster endpoints to distribute load
" For more information, see https://docs.aws.amazon.com/r53recovery/latest/dg/
route53-arc-best-practices.html#route53-arc-best-practices.regional
" For simplicity, we'll try them in order (shuffling can be added if needed)

" Try each endpoint in order
LOOP AT lt_endpoints INTO lv_endpoint.
  TRY.
    " Parse endpoint and region from the format "url|region"
    DATA(lv_pos) = find( val = lv_endpoint sub = '|' ).
    IF lv_pos > 0.
      DATA(lv_url) = substring( val = lv_endpoint len = lv_pos ).
      lv_region = substring( val = lv_endpoint off = lv_pos + 1 ).
    ELSE.

```

```
" If no region specified, use default
lv_url = lv_endpoint.
lv_region = 'us-east-1'.
ENDIF.

" Create session for this region
lo_session = /aws1/cl_rt_session_aws=>create( cv_pfl ).

" Create client with the specific endpoint
lo_client = create_recovery_client(
  iv_endpoint = lv_url
  iv_region   = lv_region
  io_session  = lo_session ).

" Try to update the routing control state
oo_result = lo_client->updateroutingcontrolstate(
  iv_routingcontrolarn      = iv_routing_control_arn
  iv_routingcontrolstate    = iv_routing_control_state
  it_safetyrulestooverride = it_safety_rules_override ).

" If successful, return the result
RETURN.

CATCH /aws1/cx_r5vendpttmpyunavailex INTO DATA(lo_endpoint_ex).
" This endpoint is temporarily unavailable, try the next one
lo_exception = lo_endpoint_ex.
CONTINUE.

CATCH /aws1/cx_r5vaccessdeniedex
      /aws1/cx_r5vconflictexception
      /aws1/cx_r5vinternalserverex
      /aws1/cx_r5vresourcenotfoundex
      /aws1/cx_r5vthrottlingex
      /aws1/cx_r5vvalidationex
      /aws1/cx_rt_generic INTO lo_exception.
" For other errors, re-raise immediately
RAISE EXCEPTION lo_exception.
ENDTRY.
ENDLOOP.

" If we get here, all endpoints failed - re-raise the last exception
IF lo_exception IS BOUND.
  RAISE EXCEPTION lo_exception.
ENDIF.
```

- Para obter detalhes da API, consulte a [UpdateRoutingControlState](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Aurora usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Aurora.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateDBClusterParameterGroup

O código de exemplo a seguir mostra como usar CreateDBClusterParameterGroup.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
  DATA(lo_output) = lo_rds->createdbclusterparamgroup(  
    iv_dbclusterparamgroupname = iv_param_group_name  
    iv_dbparametergroupfamily = iv_param_group_family  
    iv_description = iv_description
```

```

    ).
    oo_result = lo_output->get_dbclusterparametergroup( ).
  CATCH /aws1/cx_rdsdbparmgralrexfault.
    " Re-raise exception - parameter group already exists
    RAISE EXCEPTION TYPE /aws1/cx_rdsdbparmgralrexfault.
  CATCH /aws1/cx_rdsdbprimgrquotaexcd00.
    " Re-raise exception - quota exceeded
    RAISE EXCEPTION TYPE /aws1/cx_rdsdbprimgrquotaexcd00.
  ENDTRY.

```

- Para obter detalhes da API, consulte a referência da API [Create DBCluster ParameterGroup](#) in AWS SDK for SAP ABAP.

## DeleteDBClusterParameterGroup

O código de exemplo a seguir mostra como usar DeleteDBClusterParameterGroup.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

  TRY.
    lo_rds->deletedbclusterparamgroup(
      iv_dbclusterparamgroupname = iv_param_group_name
    ).
  CATCH /aws1/cx_rdsdbprimgrnotfndfault.
    " Re-raise exception - parameter group not found
    RAISE EXCEPTION TYPE /aws1/cx_rdsdbprimgrnotfndfault.
  CATCH /aws1/cx_rdsinvdbprimgrstatef00.
    " Re-raise exception - invalid state
    RAISE EXCEPTION TYPE /aws1/cx_rdsinvdbprimgrstatef00.
  ENDTRY.

```

- Para obter detalhes da API, consulte [Excluir DBCluster ParameterGroup](#) no AWS SDK para referência da API SAP ABAP.

## DescribeDBClusterParameterGroups

O código de exemplo a seguir mostra como usar DescribeDBClusterParameterGroups.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  DATA(lo_output) = lo_rds->describedbclusterparamgroups(  
    iv_dbclusterparamgroupname = iv_param_group_name  
  ).  
  DATA(lt_param_groups) = lo_output->get_dbclusterparametergroups( ).  
  IF lines( lt_param_groups ) > 0.  
    oo_result = lt_param_groups[ 1 ].  
  ENDIF.  
  CATCH /aws1/cx_rdsdbprimgrnotfndfault.  
ENDTRY.
```

- Para obter detalhes da API, consulte [Descrição DBCluster ParameterGroups](#) na referência da API ABAP do AWS SDK for SAP.

## DescribeDBClusterParameters

O código de exemplo a seguir mostra como usar DescribeDBClusterParameters.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

DATA lv_marker TYPE /aws1/rdsstring VALUE ''.
DATA lt_all_parameters TYPE /aws1/cl_rdsparameter=>tt_parameterslist.

DO.
  DATA(lo_output) = lo_rds->describedbclusterparameters(
    iv_dbclusterparamgroupname = iv_param_group_name
    iv_source = iv_source
    iv_marker = lv_marker
  ).

  LOOP AT lo_output->get_parameters( ) INTO DATA(lo_param).
    IF iv_name_prefix IS INITIAL OR
       lo_param->get_parametername( ) CP |{ iv_name_prefix }*|.
      APPEND lo_param TO lt_all_parameters.
    ENDIF.
  ENDLLOOP.

  lv_marker = lo_output->get_marker( ).
  IF lv_marker IS INITIAL.
    EXIT.
  ENDIF.
ENDDO.

ot_parameters = lt_all_parameters.
CATCH /aws1/cx_rdsdbprimgrnotfndfault.
  " Re-raise exception - parameter group not found
  RAISE EXCEPTION TYPE /aws1/cx_rdsdbprimgrnotfndfault.
ENDTRY.


```

- Para obter detalhes da API, consulte [Descrição DBCluster dos parâmetros](#) no AWS SDK para referência da API SAP ABAP.

## DescribeDBEngineVersions

O código de exemplo a seguir mostra como usar DescribeDBEngineVersions.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" iv_engine          = 'mysql'
" iv_dbparametergroupfamily = 'mysql8.0' (optional - filters by parameter group
family)
TRY.
  oo_result = lo_rds->describedbengineversions(
    iv_engine          = iv_engine
    iv_dbparametergroupfamily = iv_dbparametergroupfamily ).
  DATA(lv_version_count) = lines( oo_result->get_dbengineversions( ) ).
  MESSAGE |Retrieved { lv_version_count } engine versions.| TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte [Descrição DBEngine das versões](#) no AWS SDK para referência da API SAP ABAP.

**DescribeOrderableDBInstanceOptions**

O código de exemplo a seguir mostra como usar DescribeOrderableDBInstanceOptions.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" iv_engine          = 'mysql'
" iv_engineversion   = '8.0.35'
TRY.
  oo_result = lo_rds->descorderabledbinstoptions(
```

```

        iv_engine          = iv_engine
        iv_engineversion = iv_engineversion ).
DATA(lv_option_count) = lines( oo_result->get_orderabledbinoptions( ) ).
MESSAGE |Retrieved { lv_option_count } orderable DB instance options.| TYPE
'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte [DescribeOrderableDBInstanceOpções](#) no AWS SDK para referência da API SAP ABAP.

## ModifyDBClusterParameterGroup

O código de exemplo a seguir mostra como usar `ModifyDBClusterParameterGroup`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_rds->modifydbclusterparamgroup(
        iv_dbclusterparamgroupname = iv_param_group_name
        it_parameters = it_update_parameters
    ).
CATCH /aws1/cx_rdsdbprmgrnotfndfault.
    " Re-raise exception - parameter group not found
    RAISE EXCEPTION TYPE /aws1/cx_rdsdbprmgrnotfndfault.
CATCH /aws1/cx_rdsinldbprmgrstatef00.
    " Re-raise exception - invalid state
    RAISE EXCEPTION TYPE /aws1/cx_rdsinldbprmgrstatef00.
ENDTRY.

```

- Para obter detalhes da API, consulte [Modificar DBCluster ParameterGroup](#) no AWS SDK para referência da API SAP ABAP.

# Exemplos de Auto Scaling usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Auto Scaling.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateAutoScalingGroup

O código de exemplo a seguir mostra como usar CreateAutoScalingGroup.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lo_launch_template TYPE REF TO /aws1/cl_aslaunchtemplatespec.

" Example: iv_group_name = 'my-auto-scaling-group'
" Example: iv_launch_template_name = 'my-launch-template'
" Example: iv_min_size = 1
" Example: iv_max_size = 3
" Example: iv_vpc_zone_identifier = 'subnet-12345,subnet-67890' (for VPC)

TRY.
    " Create launch template specification
```

```

lo_launch_template = NEW /aws1/cl_asclaunchtemplatespec(
  iv_launchtemplatename = iv_launch_template_name
  iv_version = '$Default' ).

" Create the Auto Scaling group
" Use VPCZoneIdentifier for VPC subnets, or AvailabilityZones for EC2-
Classic
IF iv_vpc_zone_identifier IS NOT INITIAL.
  " VPC-based deployment - use subnet IDs
  ao_asc->createautoscalinggroup(
    iv_autoscalinggroupname = iv_group_name
    iv_vpczoneidentifier = iv_vpc_zone_identifier
    io_launchtemplate = lo_launch_template
    iv_minsize = iv_min_size
    iv_maxsize = iv_max_size ).
ELSE.
  " EC2-Classic or default VPC - use availability zones
  ao_asc->createautoscalinggroup(
    iv_autoscalinggroupname = iv_group_name
    it_availabilityzones = it_group_zones
    io_launchtemplate = lo_launch_template
    iv_minsize = iv_min_size
    iv_maxsize = iv_max_size ).
ENDIF.

" Wait for the group to be created (simplified - in production use proper
polling)
WAIT UP TO 10 SECONDS.

MESSAGE 'Auto Scaling group created successfully' TYPE 'I'.

CATCH /aws1/cx_ascalreadyexistsfault INTO DATA(lo_already_exists).
  RAISE EXCEPTION lo_already_exists.
CATCH /aws1/cx_asclimitexceededfault INTO DATA(lo_limit_exceeded).
  RAISE EXCEPTION lo_limit_exceeded.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateAutoScalingGroup](#) preferência da API AWS SDK for SAP ABAP.

## DeleteAutoScalingGroup

O código de exemplo a seguir mostra como usar DeleteAutoScalingGroup.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_group_name = 'my-auto-scaling-group'

TRY.
  ao_asc->deleteautoscalinggroup(
    iv_autoscalinggroupname = iv_group_name ).

    " Wait for the group to be deleted (simplified - in production use proper
polling)
    WAIT UP TO 10 SECONDS.

    MESSAGE 'Auto Scaling group deleted successfully' TYPE 'I'.


CATCH /aws1/cx_ascscaactivityinprg00 INTO DATA(lo_activity_in_progress).
  RAISE EXCEPTION lo_activity_in_progress.
CATCH /aws1/cx_ascresourceinusefault INTO DATA(lo_resource_in_use).
  RAISE EXCEPTION lo_resource_in_use.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAutoScalingGroup](#) referência da API AWS SDK for SAP ABAP.

## DescribeAutoScalingGroups

O código de exemplo a seguir mostra como usar DescribeAutoScalingGroups.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_group_names TYPE /aws1/
cl_ascautoscgroupnames_w=>tt_autoscalinggroupnames.
DATA lo_group_name TYPE REF TO /aws1/cl_ascautoscgroupnames_w.

" Example: iv_group_name = 'my-auto-scaling-group'

TRY.
  " Build group names parameter
  CREATE OBJECT lo_group_name
  EXPORTING
    iv_value = iv_group_name.
  APPEND lo_group_name TO lt_group_names.

  " Describe the Auto Scaling group
  DATA(lo_output) = ao_asc->describeautoscalinggroups(
    it_autoscalinggroupnames = lt_group_names ).

  " Return the first (and only) group in the result
  DATA(lt_groups) = lo_output->get_autoscalinggroups( ).
  IF lines( lt_groups ) > 0.
    READ TABLE lt_groups INDEX 1 INTO DATA(lo_group).
    oo_output = lo_group.
  ENDIF.

  MESSAGE 'Auto Scaling group information retrieved successfully' TYPE 'I'.

  CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
    RAISE EXCEPTION lo_contention.
  CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
    RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeAutoScalingGroups](#) referência da API AWS SDK for SAP ABAP.

## DescribeAutoScalingInstances

O código de exemplo a seguir mostra como usar `DescribeAutoScalingInstances`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: it_instance_ids contains a list of instance IDs

TRY.
  DATA(lo_output) = ao_asc->describeautoscalinginstances(
    it_instanceids = it_instance_ids ).

  ot_output = lo_output->get_autoscalinginstances( ).

  MESSAGE 'Auto Scaling instances information retrieved successfully' TYPE
'I'.


CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
  RAISE EXCEPTION lo_contention.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeAutoScalingInstances](#) referência da API AWS SDK for SAP ABAP.

## DescribeScalingActivities

O código de exemplo a seguir mostra como usar `DescribeScalingActivities`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_group_name = 'my-auto-scaling-group'

TRY.
  DATA(lo_output) = ao_asc->describescalingactivities(
    iv_autoscalinggroupname = iv_group_name ).

  ot_output = lo_output->get_activities( ).

  MESSAGE 'Scaling activities retrieved successfully' TYPE 'I'.


CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
  RAISE EXCEPTION lo_contention.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeScalingActivities](#) referência da API AWS SDK for SAP ABAP.

## DisableMetricsCollection

O código de exemplo a seguir mostra como usar `DisableMetricsCollection`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_group_name = 'my-auto-scaling-group'

TRY.
  ao_asc->disablemetricscollection(
    iv_autoscalinggroupname = iv_group_name ).

  MESSAGE 'Metrics collection disabled successfully' TYPE 'I'.

CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
  RAISE EXCEPTION lo_contention.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DisableMetricsCollection](#) referência da API AWS SDK for SAP ABAP.

## EnableMetricsCollection

O código de exemplo a seguir mostra como usar EnableMetricsCollection.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_group_name = 'my-auto-scaling-group'
" Example: it_metrics contains list of metrics like 'GroupMinSize',
'GroupMaxSize', etc.

TRY.
  ao_asc->enablemetricscollection(
    iv_autoscalinggroupname = iv_group_name
    it_metrics = it_metrics
    iv_granularity = '1Minute' ).

  MESSAGE 'Metrics collection enabled successfully' TYPE 'I'.
```

```
CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
  RAISE EXCEPTION lo_contention.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [EnableMetricsCollection](#) referência da API AWS SDK for SAP ABAP.

## SetDesiredCapacity

O código de exemplo a seguir mostra como usar SetDesiredCapacity.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_group_name = 'my-auto-scaling-group'
" Example: iv_capacity = 2

TRY.
  ao_asc->setdesiredcapacity(
    iv_autoscalinggroupname = iv_group_name
    iv_desiredcapacity = iv_capacity
    iv_honorcooldown = abap_false ).

  MESSAGE 'Desired capacity set successfully' TYPE 'I'.

CATCH /aws1/cx_ascscaactivityinprg00 INTO DATA(lo_activity_in_progress).
  RAISE EXCEPTION lo_activity_in_progress.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [SetDesiredCapacity](#) referência da API AWS SDK for SAP ABAP.

## TerminateInstanceInAutoScalingGroup

O código de exemplo a seguir mostra como usar `TerminateInstanceInAutoScalingGroup`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_instance_id = 'i-1234567890abcdef0'
" Example: iv_decrease_capacity = abap_true

TRY.
  DATA(lo_output) = ao_asc->terminateinstinautosgroup(
    iv_instanceid = iv_instance_id
    iv_shoulddecrementdesiredcap = iv_decrease_capacity ).

  oo_output = lo_output->get_activity( ).

  MESSAGE 'Instance terminated successfully' TYPE 'I'.

CATCH /aws1/cx_ascscaactivityinprg00 INTO DATA(lo_activity_in_progress).
  RAISE EXCEPTION lo_activity_in_progress.
CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
  RAISE EXCEPTION lo_contention.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [TerminateInstanceInAutoScalingGroup](#) referência da API AWS SDK for SAP ABAP.

## UpdateAutoScalingGroup

O código de exemplo a seguir mostra como usar UpdateAutoScalingGroup.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Example: iv_group_name = 'my-auto-scaling-group'
" Example: iv_max_size = 5

TRY.
  ao_asc->updateautoscalinggroup(
    iv_autoscalinggroupname = iv_group_name
    iv_maxsize = iv_max_size
    iv_minsize = iv_min_size ).

  MESSAGE 'Auto Scaling group updated successfully' TYPE 'I'.

CATCH /aws1/cx_ascresrccontionfault INTO DATA(lo_contention).
  RAISE EXCEPTION lo_contention.
CATCH /aws1/cx_ascscaactivityinprg00 INTO DATA(lo_activity_in_progress).
  RAISE EXCEPTION lo_activity_in_progress.
CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_exception).
  RAISE EXCEPTION lo_generic_exception.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateAutoScalingGroup](#) referência da API AWS SDK for SAP ABAP.

## Exemplos de Amazon Bedrock Runtime usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Bedrock Runtime.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Claude da Anthropic](#)
- [Stable Diffusion](#)

## Claude da Anthropic

### InvokeModel

O exemplo de código a seguir mostra como enviar uma mensagem de texto ao Claude da Anthropic usando a API Invoke Model.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Invoque o modelo de base Claude 2 da Anthropic para gerar texto. Este exemplo usa recursos de/US2/CL\_JSON que podem não estar disponíveis em algumas versões. NetWeaver

```
"Claude V2 Input Parameters should be in a format like this:
* {
*   "prompt":"\n\nHuman:\nTell me a joke\n\nAssistant:\n",
*   "max_tokens_to_sample":2048,
*   "temperature":0.5,
*   "top_k":250,
*   "top_p":1.0,
*   "stop_sequences":[]
* }

DATA: BEGIN OF ls_input,
      prompt                TYPE string,
      max_tokens_to_sample TYPE /aws1/rt_shape_integer,
      temperature           TYPE /aws1/rt_shape_float,
      top_k                 TYPE /aws1/rt_shape_integer,
```

```

        top_p                TYPE /aws1/rt_shape_float,
        stop_sequences       TYPE /aws1/rt_stringtab,
    END OF ls_input.

    "Leave ls_input-stop_sequences empty.
    ls_input-prompt = |\n\nHuman:\n{ iv_prompt }\n\nAssistant:\n|.
    ls_input-max_tokens_to_sample = 2048.
    ls_input-temperature = '0.5'.
    ls_input-top_k = 250.
    ls_input-top_p = 1.

    "Serialize into JSON with /ui2/cl_json -- this assumes SAP_UI is installed.
    DATA(lv_json) = /ui2/cl_json=>serialize(
        data = ls_input
        pretty_name = /ui2/cl_json=>pretty_mode-low_case ).

    TRY.
        DATA(lo_response) = lo_bdr->invokemodel(
            iv_body = /aws1/cl_rt_util=>string_to_xstring( lv_json )
            iv_modelid = 'anthropic.claude-v2'
            iv_accept = 'application/json'
            iv_contenttype = 'application/json' ).

        "Claude V2 Response format will be:
        * {
        *   "completion": "Knock Knock...",
        *   "stop_reason": "stop_sequence"
        * }
        DATA: BEGIN OF ls_response,
            completion TYPE string,
            stop_reason TYPE string,
        END OF ls_response.

        /ui2/cl_json=>deserialize(
            EXPORTING jsonx = lo_response->get_body( )
            pretty_name = /ui2/cl_json=>pretty_mode-camel_case
            CHANGING data = ls_response ).

        DATA(lv_answer) = ls_response-completion.
        CATCH /aws1/cx_bdraccessdeniedex INTO DATA(lo_ex).
        WRITE / lo_ex->get_text( ).
        WRITE / |Don't forget to enable model access at https://
console.aws.amazon.com/bedrock/home?#/modelaccess|.

```

```
ENDTRY.
```

Invoque o modelo de base Claude da Anthropic 2 para gerar texto usando um cliente de nível alto de L2.

```
TRY.
    DATA(lo_bdr_l2_claude) = /aws1/cl_bdr_l2_factory=>create_claude_2( lo_bdr ).
    " iv_prompt can contain a prompt like 'tell me a joke about Java
    programmers'.
    DATA(lv_answer) = lo_bdr_l2_claude->prompt_for_text( iv_prompt ).
    CATCH /aws1/cx_bdraccessdeniedex INTO DATA(lo_ex).
    WRITE / lo_ex->get_text( ).
    WRITE / |Don't forget to enable model access at https://
    console.aws.amazon.com/bedrock/home?#/modelaccess|.

ENDTRY.
```

Invoque o modelo de base Claude da Anthropic 3 para gerar texto usando um cliente de nível alto de L2.

```
TRY.
    " Choose a model ID from Anthropic that supports the Messages API -
    currently this is
    " Claude v2, Claude v3 and v3.5. For the list of model ID, see:
    " https://docs.aws.amazon.com/bedrock/latest/userguide/model-ids.html

    " for the list of models that support the Messages API see:
    " https://docs.aws.amazon.com/bedrock/latest/userguide/model-parameters-
    anthropic-claude-messages.html
    DATA(lo_bdr_l2_claude) = /aws1/cl_bdr_l2_factory=>create_anthropic_msg_api(
        io_bdr = lo_bdr
        iv_model_id = 'anthropic.claude-3-sonnet-20240229-v1:0' ). " choosing
    Claude v3 Sonnet
    " iv_prompt can contain a prompt like 'tell me a joke about Java
    programmers'.
    DATA(lv_answer) = lo_bdr_l2_claude->prompt_for_text( iv_prompt = iv_prompt
        iv_max_tokens = 100 ).

    CATCH /aws1/cx_bdraccessdeniedex INTO DATA(lo_ex).
    WRITE / lo_ex->get_text( ).
    WRITE / |Don't forget to enable model access at https://
    console.aws.amazon.com/bedrock/home?#/modelaccess|.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [InvokeModel](#) referência da API AWS SDK for SAP ABAP.

## Stable Diffusion

### InvokeModel

O exemplo de código a seguir mostra como invocar o Stable Diffusion XL da Stability.ai no Amazon Bedrock para gerar uma imagem.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie uma imagem com o Stable Diffusion.

```
"Stable Diffusion Input Parameters should be in a format like this:
*  {
*    "text_prompts": [
*      {"text":"Draw a dolphin with a mustache"},
*      {"text":"Make it photorealistic"}
*    ],
*    "cfg_scale":10,
*    "seed":0,
*    "steps":50
*  }
TYPES: BEGIN OF prompt_ts,
        text TYPE /aws1/rt_shape_string,
      END OF prompt_ts.

DATA: BEGIN OF ls_input,
        text_prompts TYPE STANDARD TABLE OF prompt_ts,
        cfg_scale   TYPE /aws1/rt_shape_integer,
```

```

        seed          TYPE /aws1/rt_shape_integer,
        steps        TYPE /aws1/rt_shape_integer,
    END OF ls_input.

APPEND VALUE prompt_ts( text = iv_prompt ) TO ls_input-text_prompts.
ls_input-cfg_scale = 10.
ls_input-seed = 0. "or better, choose a random integer.
ls_input-steps = 50.

DATA(lv_json) = /ui2/cl_json=>serialize(
    data = ls_input
        pretty_name = /ui2/cl_json=>pretty_mode-low_case ).

TRY.
    DATA(lo_response) = lo_bdr->invokemodel(
        iv_body = /aws1/cl_rt_util=>string_to_xstring( lv_json )
        iv_modelid = 'stability.stable-diffusion-xl-v1'
        iv_accept = 'application/json'
        iv_contenttype = 'application/json' ).

    "Stable Diffusion Result Format:
    *
    * {
    *   "result": "success",
    *   "artifacts": [
    *     {
    *       "seed": 0,
    *       "base64": "iVBORw0KGgoAAAANSUhEUgAAAgAAA...
    *       "finishReason": "SUCCESS"
    *     }
    *   ]
    * }
    TYPES: BEGIN OF artifact_ts,
            seed          TYPE /aws1/rt_shape_integer,
            base64        TYPE /aws1/rt_shape_string,
            finishreason TYPE /aws1/rt_shape_string,
    END OF artifact_ts.

    DATA: BEGIN OF ls_response,
            result TYPE /aws1/rt_shape_string,
            artifacts TYPE STANDARD TABLE OF artifact_ts,
    END OF ls_response.

    /ui2/cl_json=>deserialize(
        EXPORTING jsonx = lo_response->get_body( )

```

```

                pretty_name = /ui2/cl_json=>pretty_mode-camel_case
    CHANGING data = ls_response ).
    IF ls_response-artifacts IS NOT INITIAL.
        DATA(lv_image) =
    cl_http_utility=>if_http_utility~decode_x_base64( ls_response-artifacts[ 1 ]-
base64 ).
        ENDIF.
    CATCH /aws1/cx_bdraccessdeniedex INTO DATA(lo_ex).
        WRITE / lo_ex->get_text( ).
        WRITE / |Don't forget to enable model access at https://
console.aws.amazon.com/bedrock/home?#/modelaccess|.

    ENDTRY.

```

Invoque o modelo de base Stable Diffusion XL da Stability AI para gerar imagens usando um cliente L2 de alto nível.

```

    TRY.
        DATA(lo_bdr_l2_sd) = /aws1/
    cl_bdr_l2_factory=>create_stable_diffusion_xl_1( lo_bdr ).
        " iv_prompt contains a prompt like 'Show me a picture of a unicorn reading
an enterprise financial report'.
        DATA(lv_image) = lo_bdr_l2_sd->text_to_image( iv_prompt ).
        CATCH /aws1/cx_bdraccessdeniedex INTO DATA(lo_ex).
        WRITE / lo_ex->get_text( ).
        WRITE / |Don't forget to enable model access at https://
console.aws.amazon.com/bedrock/home?#/modelaccess|.

    ENDTRY.

```

- Para obter detalhes da API, consulte a [InvokeModel](#) referência da API AWS SDK for SAP ABAP.

# Exemplos do Amazon Bedrock Agents Runtime usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Bedrock Agents Runtime.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### InvokeAgent

O código de exemplo a seguir mostra como usar InvokeAgent.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA(lo_result) = lo_bdz->invokeagent(  
  iv_agentid      = iv_agentid  
  iv_agentaliasid = iv_agentaliasid  
  iv_enabletrace  = abap_true  
  iv_sessionid    = CONV #( cl_system_uuid=>create_uuid_c26_static( ) )  
  iv_inputtext    = |Let's play "rock, paper, scissors". I choose rock.| ).  
DATA(lo_stream) = lo_result->get_completion( ).
```

```

TRY.
  " loop while there are still events in the stream
  WHILE lo_stream->/aws1/if_rt_stream_reader~data_available( ) = abap_true.
    DATA(lo_evt) = lo_stream->read( ).
    " each /AWS1/CL_BDZRESPONSESTREAM_EV event contains exactly one member
    " all others are INITIAL. For each event, process the non-initial
    " member if desired
    IF lo_evt->get_chunk( ) IS NOT INITIAL.
      " Process a Chunk event
      DATA(lv_xstr) = lo_evt->get_chunk( )->get_bytes( ).
      DATA(lv_answer) = /aws1/cl_rt_util=>xstring_to_string( lv_xstr ).
      " the answer says something like "I chose paper, so you lost"
    ELSEIF lo_evt->get_files( ) IS NOT INITIAL.
      " process a Files event if desired
    ELSEIF lo_evt->get_returncontrol( ) IS NOT INITIAL.
      " process a ReturnControl event if desired
    ELSEIF lo_evt->get_trace( ) IS NOT INITIAL.
      " process a Trace event if desired
    ENDIF.
  ENDWHILE.
  " the stream of events can possibly contain an exception
  " which will be raised to break the loop
  " catch /AWS1/CX_BDZACCESSDENIEDEX.
  " catch /AWS1/CX_BDZINTERNALSERVEREX.
  " catch /AWS1/CX_BDZMODELNOTREADYEX.
  " catch /AWS1/CX_BDZVALIDATIONEX.
  " catch /AWS1/CX_BDZTHROTTLINGEX.
  " catch /AWS1/CX_BDZDEPENDENCYFAILEDEX.
  " catch /AWS1/CX_BDZBADGATEWAYEX.
  " catch /AWS1/CX_BDZRESOURCENOTFOUNDEX.
  " catch /AWS1/CX_BDZSERVICEQUOTAEXCDEX.
  " catch /AWS1/CX_BDZCONFLICTEXCEPTION.
ENDTRY.

```

- Para obter detalhes da API, consulte a [InvokeAgent](#) referência da API AWS SDK for SAP ABAP.

## CloudFront exemplos usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. CloudFront

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### ListDistributions

O código de exemplo a seguir mostra como usar ListDistributions.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_fnt->listdistributions( ). " oo_result is returned for  
testing purposes. "  
    MESSAGE 'Retrieved list of CloudFront distributions.' TYPE 'I'.  
    CATCH /aws1/cx_fntinvalidargument.  
        MESSAGE 'Invalid argument provided.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListDistributions](#) referência da API AWS SDK for SAP ABAP.

### UpdateDistribution

O código de exemplo a seguir mostra como usar UpdateDistribution.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Get the current distribution configuration and ETag "
  DATA(lo_distribution_config_result) = lo_fnt->getdistributionconfig( iv_id =
iv_distribution_id ).
  DATA(lo_old_config) = lo_distribution_config_result-
>get_distributionconfig( ).
  DATA(lv_etag) = lo_distribution_config_result->get_etag( ).

  " Create a new distribution config with the updated comment "
  " Since the config object is immutable, we need to create a new one with all
existing values "
  DATA(lo_new_config) = NEW /aws1/cl_fntdistributionconfig(
  iv_callerreference = lo_old_config->get_callerreference( )
  io_aliases = lo_old_config->get_aliases( )
  iv_defaultrootobject = lo_old_config->get_defaultrootobject( )
  io_origins = lo_old_config->get_origins( )
  io_oringroups = lo_old_config->get_oringroups( )
  io_defaultcachebehavior = lo_old_config->get_defaultcachebehavior( )
  io_cachebehaviors = lo_old_config->get_cachebehaviors( )
  io_customerrorresponses = lo_old_config->get_customerrorresponses( )
  iv_comment = iv_comment
  io_logging = lo_old_config->get_logging( )
  iv_priceclass = lo_old_config->get_priceclass( )
  iv_enabled = lo_old_config->get_enabled( )
  io_viewercertificate = lo_old_config->get_viewercertificate( )
  io_restrictions = lo_old_config->get_restrictions( )
  iv_webaclid = lo_old_config->get_webaclid( )
  iv_httpversion = lo_old_config->get_httpversion( )
  iv_isipv6enabled = lo_old_config->get_isipv6enabled( ) ).

  " Update the distribution with the modified configuration "
  lo_fnt->updatedistribution(
  io_distributionconfig = lo_new_config
  iv_id = iv_distribution_id
```

```
        iv_ifmatch = lv_etag ).
    MESSAGE 'CloudFront distribution updated successfully.' TYPE 'I'.
    CATCH /aws1/cx_fntnosuchdistribution.
    MESSAGE 'Distribution does not exist.' TYPE 'E'.
    CATCH /aws1/cx_fntpreconditionfailed.
    MESSAGE 'Precondition failed - ETag mismatch.' TYPE 'E'.
    CATCH /aws1/cx_fntinvalidifmatchvrs.
    MESSAGE 'Invalid If-Match version.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateDistribution](#) referência da API AWS SDK for SAP ABAP.

## CloudWatch exemplos usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. CloudWatch

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)
- [Cenários](#)

## Ações

### DeleteAlarms

O código de exemplo a seguir mostra como usar DeleteAlarms.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_cwt->deletealarms(  
    it_alarmnames = it_alarm_names ).  
  MESSAGE 'Alarms deleted.' TYPE 'I'.  
CATCH /aws1/cx_cwtresourcenotfound.  
  MESSAGE 'Resource being accessed is not found.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAlarms](#) referência da API AWS SDK for SAP ABAP.

**DescribeAlarms**

O código de exemplo a seguir mostra como usar DescribeAlarms.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_cwt->describealarms(                                " oo_result is returned  
for testing purposes. "  
    it_alarmnames = it_alarm_names ).  
  MESSAGE 'Alarms retrieved.' TYPE 'I'.  
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-  
>av_err_msg }|. 
```

```
MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeAlarms](#) referência da API AWS SDK for SAP ABAP.

## DescribeAlarmsForMetric

O código de exemplo a seguir mostra como usar `DescribeAlarmsForMetric`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_cwt->describealarmsformetric(           " oo_result is
returned for testing purposes. "
    iv_namespace   = iv_namespace
    iv_metricname  = iv_metric_name
    it_dimensions  = it_dimensions
    iv_statistic   = iv_statistic
    iv_period      = iv_period
    iv_unit        = iv_unit ).
    MESSAGE 'Alarms for metric retrieved.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeAlarmsForMetric](#) referência da API AWS SDK for SAP ABAP.

## DisableAlarmActions

O código de exemplo a seguir mostra como usar `DisableAlarmActions`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Disables actions on the specified alarm. "  
TRY.  
    lo_cwt->disablealarmactions(  
        it_alarmnames = it_alarm_names ).  
    MESSAGE 'Alarm actions disabled.' TYPE 'I'.  
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DisableAlarmActions](#) referência da API AWS SDK for SAP ABAP.

## EnableAlarmActions

O código de exemplo a seguir mostra como usar `EnableAlarmActions`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

"Enable actions on the specified alarm."
TRY.
  lo_cwt->enablealarmactions(
    it_alarmnames = it_alarm_names ).
  MESSAGE 'Alarm actions enabled.' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [EnableAlarmActions](#) referência da API AWS SDK for SAP ABAP.

## GetMetricStatistics

O código de exemplo a seguir mostra como usar GetMetricStatistics.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  oo_result = lo_cwt->getmetricstatistics(           " oo_result is
returned for testing purposes. "
  iv_namespace   = iv_namespace
  iv_metricname  = iv_metric_name
  iv_starttime   = iv_start_time
  iv_endtime     = iv_end_time
  iv_period      = iv_period
  it_statistics  = it_statistics ).
  MESSAGE 'Metric statistics retrieved.' TYPE 'I'.
CATCH /aws1/cx_cwtinparamvalueex.
  MESSAGE 'The specified argument was not valid.' TYPE 'E'.

```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetMetricStatistics](#) referência da API AWS SDK for SAP ABAP.

## ListMetrics

O código de exemplo a seguir mostra como usar ListMetrics.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
"The following list-metrics example displays the metrics for Amazon CloudWatch."
TRY.
    oo_result = lo_cwt->listmetrics(           " oo_result is returned for
testing purposes. "
    iv_namespace = iv_namespace ).
    DATA(lt_metrics) = oo_result->get_metrics( ).
    MESSAGE 'Metrics retrieved.' TYPE 'I'.
CATCH /aws1/cx_cwtinvparamvalueex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListMetrics](#) referência da API AWS SDK for SAP ABAP.

## PutMetricAlarm

O código de exemplo a seguir mostra como usar PutMetricAlarm.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  lo_cwt->putmetricalarm(
    iv_alarmname           = iv_alarm_name
    iv_comparisonoperator  = iv_comparison_operator
    iv_evaluationperiods   = iv_evaluation_periods
    iv_metricname          = iv_metric_name
    iv_namespace           = iv_namespace
    iv_statistic           = iv_statistic
    iv_threshold           = iv_threshold
    iv_actionsenabled      = iv_actions_enabled
    iv_alarmdescription    = iv_alarm_description
    iv_unit                = iv_unit
    iv_period              = iv_period
    it_dimensions          = it_dimensions ).
  MESSAGE 'Alarm created.' TYPE 'I'.
CATCH /aws1/cx_cwtlimitexceededfault.
  MESSAGE 'The request processing has exceeded the limit' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutMetricAlarm](#) referência da API AWS SDK for SAP ABAP.

## PutMetricData

O código de exemplo a seguir mostra como usar PutMetricData.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_metricdata TYPE /aws1/cl_cwtmetricdatum=>tt_metricdata.
```

```
"Create metric data object.
```

```
DATA(lo_metricdatum) = NEW /aws1/cl_cwtmetricdatum(  
  iv_metricname = iv_metric_name  
  iv_value      = iv_value  
  iv_unit       = iv_unit ).
```

```
INSERT lo_metricdatum INTO TABLE lt_metricdata.
```

```
TRY.
```

```
  lo_cwt->putmetricdata(  
    iv_namespace = iv_namespace  
    it_metricdata = lt_metricdata ).  
  MESSAGE 'Metric data added.' TYPE 'I'.
```

```
CATCH /aws1/cx_cwtinvalueparamvalueex.
```

```
  MESSAGE 'The specified argument was not valid.' TYPE 'E'.
```

```
ENDTRY.
```

Coloque um conjunto de dados em uma CloudWatch métrica.

```
DATA lt_metricdata TYPE /aws1/cl_cwtmetricdatum=>tt_metricdata.
```

```
"Create metric data object with values and counts.
```

```
DATA(lo_metricdatum) = NEW /aws1/cl_cwtmetricdatum(  
  iv_metricname = iv_metric_name  
  iv_timestamp  = iv_timestamp  
  iv_unit       = iv_unit  
  it_values     = it_values  
  it_counts     = it_counts ).
```

```
INSERT lo_metricdatum INTO TABLE lt_metricdata.

TRY.
  lo_cwt->putmetricdata(
    iv_namespace = iv_namespace
    it_metricdata = lt_metricdata ).
  MESSAGE 'Metric data set added.' TYPE 'I'.
CATCH /aws1/cx_cwtinvsparamvalueex.
  MESSAGE 'The specified argument was not valid.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutMetricData](#) referência da API AWS SDK for SAP ABAP.

## Cenários

### Começar a usar alarmes

O exemplo de código a seguir mostra como:

- Criar um alarme.
- Desabilitar ações de alarme.
- Descrever um alarme.
- Excluir um alarme.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_alarmnames TYPE /aws1/cl_cwtalarmnames_w=>tt_alarmnames.
DATA lo_alarmname TYPE REF TO /aws1/cl_cwtalarmnames_w.

"Create an alarm"
```

```

TRY.
  lo_cwt->putmetricalarm(
    iv_alarmname           = iv_alarm_name
    iv_comparisonoperator  = iv_comparison_operator
    iv_evaluationperiods   = iv_evaluation_periods
    iv_metricname          = iv_metric_name
    iv_namespace           = iv_namespace
    iv_statistic           = iv_statistic
    iv_threshold           = iv_threshold
    iv_actionsenabled      = iv_actions_enabled
    iv_alarmdescription    = iv_alarm_description
    iv_unit                = iv_unit
    iv_period              = iv_period
    it_dimensions          = it_dimensions ).
  MESSAGE 'Alarm created' TYPE 'I'.
CATCH /aws1/cx_cwtlimitexceededfault.
  MESSAGE 'The request processing has exceeded the limit' TYPE 'E'.
ENDTRY.

"Create an ABAP internal table for the created alarm."
lo_alarmname = NEW #( iv_value = iv_alarm_name ).
INSERT lo_alarmname INTO TABLE lt_alarmnames.

"Disable alarm actions."
TRY.
  lo_cwt->disablealarmactions(
    it_alarmnames          = lt_alarmnames ).
  MESSAGE 'Alarm actions disabled' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_disablealarm_exception).
  DATA(lv_disablealarm_error) = |"{ lo_disablealarm_exception->av_err_code }"
- { lo_disablealarm_exception->av_err_msg }|.
  MESSAGE lv_disablealarm_error TYPE 'E'.
ENDTRY.

"Describe alarm using the same ABAP internal table."
TRY.
  oo_result = lo_cwt->describealarms(
returned for testing purpose "                                " oo_result is
    it_alarmnames          = lt_alarmnames ).
  MESSAGE 'Alarms retrieved' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_describealarms_exception).
  DATA(lv_describealarms_error) = |"{ lo_describealarms_exception-
>av_err_code }" - { lo_describealarms_exception->av_err_msg }|.
  MESSAGE lv_describealarms_error TYPE 'E'.

```

```
ENDTRY.  
  
"Delete alarm."  
TRY.  
    lo_cwt->deletealarms(  
        it_alarmnames = lt_alarmnames ).  
    MESSAGE 'Alarms deleted' TYPE 'I'.  
CATCH /aws1/cx_cwtresourcenotfound.  
    MESSAGE 'Resource being access is not found.' TYPE 'E'.  
ENDTRY.
```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [DeleteAlarms](#)
  - [DescribeAlarms](#)
  - [DisableAlarmActions](#)
  - [PutMetricAlarm](#)

## CloudWatch Exemplos de registros usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP with CloudWatch Logs.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos


- [Ações](#)

### Ações

#### **GetQueryResults**

O código de exemplo a seguir mostra como usar `GetQueryResults`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_cwl->getqueryresults(
    iv_queryid = iv_query_id ).

  " Display query status and result count
  DATA(lv_status) = oo_result->get_status( ).
  DATA(lt_results) = oo_result->get_results( ).
  DATA(lv_result_count) = lines( lt_results ).


  MESSAGE |Query status: { lv_status }. Retrieved { lv_result_count } log
event(s).| TYPE 'I'.
  CATCH /aws1/cx_cwlinvalidparameterex.
    MESSAGE 'Invalid parameter.' TYPE 'E'.
  CATCH /aws1/cx_cwlresourcenotfoundex.
    MESSAGE 'Resource not found.' TYPE 'E'.
  CATCH /aws1/cx_cwlserviceunavailex.
    MESSAGE 'Service unavailable.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetQueryResults](#) referência da API AWS SDK for SAP ABAP.

## StartQuery

O código de exemplo a seguir mostra como usar StartQuery.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_log_group_name = '/aws/lambda/my-function'
  " iv_query_string = 'fields @timestamp, @message | sort @timestamp desc |
limit 20'
  " iv_start_time and iv_end_time must be in Unix epoch milliseconds (ms since
Jan 1, 1970 00:00:00 UTC)
  oo_result = lo_cwl->startquery(
    iv_loggroupname = iv_log_group_name
    iv_starttime    = iv_start_time
    iv_endtime      = iv_end_time
    iv_querystring  = iv_query_string
    iv_limit        = iv_limit ).

  " Display the query ID for tracking
  DATA(lv_query_id) = oo_result->get_queryid( ).
  MESSAGE |Query started successfully with ID: { lv_query_id }| TYPE 'I'.
  CATCH /aws1/cx_cwlinvalidparameterex.
    MESSAGE 'Invalid parameter.' TYPE 'E'.
  CATCH /aws1/cx_cwllimitexceededx.
    MESSAGE 'Limit exceeded.' TYPE 'E'.
  CATCH /aws1/cx_cwlmalformedqueryex.
    MESSAGE 'Malformed query.' TYPE 'E'.
  CATCH /aws1/cx_cwlresourcenotfoundex.
    MESSAGE 'Resource not found.' TYPE 'E'.
  CATCH /aws1/cx_cwlserviceunavailex.
    MESSAGE 'Service unavailable.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartQuery](#) referência da API AWS SDK for SAP ABAP.

# Exemplos de provedores de identidade do Amazon Cognito usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Cognito Identity Provider.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### AdminInitiateAuth

O código de exemplo a seguir mostra como usar AdminInitiateAuth.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " Set up authentication parameters  
    DATA(lt_auth_params) = VALUE /aws1/  
cl_cgpathparamstype_w=>tt_authparameterstype(  
    ( VALUE /aws1/cl_cgpathparamstype_w=>ts_authparameterstype_maprow(  
        key = 'USERNAME'  
        value = NEW /aws1/cl_cgpathparamstype_w( iv_user_name ) ) )  
    ( VALUE /aws1/cl_cgpathparamstype_w=>ts_authparameterstype_maprow(  
        key = 'PASSWORD'  
        value = NEW /aws1/cl_cgpathparamstype_w( iv_password ) ) ) ) )
```

```

        key = 'PASSWORD'
        value = NEW /aws1/cl_cgpaauthparamstype_w( iv_password ) ) )
    ).

" Add SECRET_HASH if provided
IF iv_secret_hash IS NOT INITIAL.
    INSERT VALUE #(
        key = 'SECRET_HASH'
        value = NEW /aws1/cl_cgpaauthparamstype_w( iv_secret_hash )
    ) INTO TABLE lt_auth_params.
ENDIF.

oo_result = lo_cgp->admininitiateauth(
    iv_userpoolid = iv_user_pool_id
    iv_clientid = iv_client_id
    iv_authflow = 'ADMIN_USER_PASSWORD_AUTH'
    it_authparameters = lt_auth_params
).

DATA(lv_challenge) = oo_result->get_challengename( ).

IF lv_challenge IS INITIAL.
    MESSAGE 'User successfully signed in.' TYPE 'I'.
ELSE.
    MESSAGE |Authentication challenge required: { lv_challenge }.| TYPE 'I'.
ENDIF.

CATCH /aws1/cx_cgpusernotfoundex INTO DATA(lo_user_ex).
    MESSAGE |User { iv_user_name } not found.| TYPE 'E'.

CATCH /aws1/cx_cgpnnotauthorizedex INTO DATA(lo_auth_ex).
    MESSAGE 'Not authorized. Check credentials.' TYPE 'E'.
ENDTRY.


```

- Para obter detalhes da API, consulte a [AdminInitiateAuth](#) referência da API AWS SDK for SAP ABAP.

## AdminRespondToAuthChallenge

O código de exemplo a seguir mostra como usar AdminRespondToAuthChallenge.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Build challenge responses
  DATA(lt_challenge_responses) = VALUE /aws1/
cl_cgpchallengerspstyp00=>tt_challengeresponsestype(
    ( VALUE /aws1/cl_cgpchallengerspstyp00=>ts_challengerspstype_maprow(
      key = 'USERNAME'
      value = NEW /aws1/cl_cgpchallengerspstyp00( iv_user_name ) ) )
    ( VALUE /aws1/cl_cgpchallengerspstyp00=>ts_challengerspstype_maprow(
      key = 'SOFTWARE_TOKEN_MFA_CODE'
      value = NEW /aws1/cl_cgpchallengerspstyp00( iv_mfa_code ) ) )
  ).

  " Add SECRET_HASH if provided
  IF iv_secret_hash IS NOT INITIAL.
    INSERT VALUE #(
      key = 'SECRET_HASH'
      value = NEW /aws1/cl_cgpchallengerspstyp00( iv_secret_hash )
    ) INTO TABLE lt_challenge_responses.
  ENDIF.

  DATA(lo_result) = lo_cgp->adminrespondtoauthchallenge(
    iv_userpoolid = iv_user_pool_id
    iv_clientid = iv_client_id
    iv_challenge_name = 'SOFTWARE_TOKEN_MFA'
    it_challengeresponses = lt_challenge_responses
    iv_session = iv_session
  ).

  oo_auth_result = lo_result->get_authenticationresult( ).

  IF oo_auth_result IS BOUND.
    MESSAGE 'MFA challenge completed successfully.' TYPE 'I'.
  ELSE.
    " Another challenge might be required
```

```
DATA(lv_next_challenge) = lo_result->get_challenge_name( ).
MESSAGE |Additional challenge required: { lv_next_challenge }.| TYPE 'I'.
ENDIF.

CATCH /aws1/cx_cgpcodemismatchex INTO DATA(lo_code_ex).
MESSAGE 'Invalid MFA code provided.' TYPE 'E'.

CATCH /aws1/cx_cgpxpiredcodeex INTO DATA(lo_expired_ex).
MESSAGE 'MFA code has expired.' TYPE 'E'.

CATCH /aws1/cx_cgpnauthorizedex INTO DATA(lo_auth_ex).
MESSAGE 'Not authorized. Check MFA configuration.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [AdminRespondToAuthChallenge](#) referência da API AWS SDK for SAP ABAP.

## AssociateSoftwareToken

O código de exemplo a seguir mostra como usar AssociateSoftwareToken.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
DATA(lo_result) = lo_cgp->associatesoftwaretoken(
    iv_session = iv_session
).

ov_secret_code = lo_result->get_secretcode( ).

MESSAGE 'MFA secret code generated successfully.' TYPE 'I'.

CATCH /aws1/cx_cgpxresourcenotfoundex INTO DATA(lo_ex).
MESSAGE 'Session not found or expired.' TYPE 'E'.
```

```
CATCH /aws1/cx_cgpnnotauthorizedex INTO DATA(lo_auth_ex).
  MESSAGE 'Not authorized to associate software token.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [AssociateSoftwareToken](#) referência da API AWS SDK for SAP ABAP.

## ListUsers

O código de exemplo a seguir mostra como usar ListUsers.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  DATA(lo_result) = lo_cgp->listusers(
    iv_userpoolid = iv_user_pool_id
  ).

  ot_users = lo_result->get_users( ).

  MESSAGE |Found { lines( ot_users ) } users in the pool.| TYPE 'I'.

CATCH /aws1/cx_cgpresourcenotfoundex INTO DATA(lo_ex).
  MESSAGE |User pool { iv_user_pool_id } not found.| TYPE 'E'.

CATCH /aws1/cx_cgpnnotauthorizedex INTO DATA(lo_auth_ex).
  MESSAGE 'Not authorized to list users.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListUsers](#) referência da API AWS SDK for SAP ABAP.

## VerifySoftwareToken

O código de exemplo a seguir mostra como usar VerifySoftwareToken.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  DATA(lo_result) = lo_cgp->verifysoftwaretoken(
    iv_session = iv_session
    iv_usercode = iv_user_code
  ).

  ov_status = lo_result->get_status( ).

  IF ov_status = 'SUCCESS'.
    MESSAGE 'MFA token verified successfully.' TYPE 'I'.
  ELSE.
    MESSAGE |MFA verification status: { ov_status }.| TYPE 'I'.
  ENDIF.

  CATCH /aws1/cx_cgpcodemismatchex INTO DATA(lo_code_ex).
    MESSAGE 'Invalid MFA code provided.' TYPE 'E'.

  CATCH /aws1/cx_cgpenbsoftwaretokmf00 INTO DATA(lo_enabled_ex).
    MESSAGE 'Software token MFA is already enabled.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [VerifySoftwareToken](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon Comprehend usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Comprehend.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateDocumentClassifier

O código de exemplo a seguir mostra como usar CreateDocumentClassifier.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_cpd->createdocumentclassifier(
    iv_documentclassifiername = iv_classifier_name
    iv_languagecode = iv_language_code
    io_inputdataconfig = NEW /aws1/cl_cpddocclifierinpdat00(
      iv_s3uri = iv_training_s3_uri
    )
    iv_dataaccessrolelearn = iv_data_access_role_arn
    iv_mode = iv_mode
  ).
  MESSAGE 'Document classifier creation started.' TYPE 'I'.
CATCH /aws1/cx_cpdivalidrequestex.
  MESSAGE 'Invalid request.' TYPE 'E'.
CATCH /aws1/cx_cpdrresrclimitexcdex.
  MESSAGE 'Resource limit exceeded.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanyrequestsex.
```

```
MESSAGE 'Too many requests.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanytagsex.
MESSAGE 'Too many tags.' TYPE 'E'.
CATCH /aws1/cx_cpdingernalserverex.
MESSAGE 'Internal server error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateDocumentClassifier](#) referência da API AWS SDK for SAP ABAP.

## DeleteDocumentClassifier

O código de exemplo a seguir mostra como usar DeleteDocumentClassifier.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_cpd->deletedocumentclassifier(
    iv_documentclassifierarn = iv_classifier_arn
  ).
  MESSAGE 'Document classifier deleted.' TYPE 'I'.
CATCH /aws1/cx_cpdingalidrequestex.
  MESSAGE 'Invalid request.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanyrequestsex.
  MESSAGE 'Too many requests.' TYPE 'E'.
CATCH /aws1/cx_cpdingourcenotfoundex.
  MESSAGE 'Resource not found.' TYPE 'E'.
CATCH /aws1/cx_cpdingourceninuseex.
  MESSAGE 'Resource in use.' TYPE 'E'.
CATCH /aws1/cx_cpdingernalserverex.
  MESSAGE 'Internal server error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteDocumentClassifier](#) referência da API AWS SDK for SAP ABAP.

## DescribeDocumentClassificationJob

O código de exemplo a seguir mostra como usar DescribeDocumentClassificationJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->describedocclassificationjob(  
        iv_jobid = iv_job_id  
    ).  
    MESSAGE 'Document classification job described.' TYPE 'I'.  
CATCH /aws1/cx_cpdivalidrequestex.  
    MESSAGE 'Invalid request.' TYPE 'E'.  
CATCH /aws1/cx_cpdjobnotfoundex.  
    MESSAGE 'Job not found.' TYPE 'E'.  
CATCH /aws1/cx_cpdtomanyrequestsex.  
    MESSAGE 'Too many requests.' TYPE 'E'.  
CATCH /aws1/cx_cpdinternalserverex.  
    MESSAGE 'Internal server error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeDocumentClassificationJob](#) referência da API AWS SDK for SAP ABAP.

## DescribeDocumentClassifier

O código de exemplo a seguir mostra como usar DescribeDocumentClassifier.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->describedocumentclassifier(  
        iv_documentclassifierarn = iv_classifier_arn  
    ).  
    MESSAGE 'Document classifier described.' TYPE 'I'.  
CATCH /aws1/cx_cpinvalidrequestex.  
    MESSAGE 'Invalid request.' TYPE 'E'.  
CATCH /aws1/cx_cpdtoomanyrequestsex.  
    MESSAGE 'Too many requests.' TYPE 'E'.  
CATCH /aws1/cx_cpresourcefoundex.  
    MESSAGE 'Resource not found.' TYPE 'E'.  
CATCH /aws1/cx_cpinternalserverex.  
    MESSAGE 'Internal server error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeDocumentClassifier](#) referência da API AWS SDK for SAP ABAP.

**DescribeTopicsDetectionJob**

O código de exemplo a seguir mostra como usar DescribeTopicsDetectionJob.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
oo_result = lo_cpd->describetopicdetectionjob(
  iv_jobid = iv_job_id
).
MESSAGE 'Topics detection job described.' TYPE 'I'.
CATCH /aws1/cx_cpinvalidrequestex.
  MESSAGE 'Invalid request.' TYPE 'E'.
CATCH /aws1/cx_cpjobnotfoundex.
  MESSAGE 'Job not found.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanyrequestsex.
  MESSAGE 'Too many requests.' TYPE 'E'.
CATCH /aws1/cx_cpinternalserverex.
  MESSAGE 'Internal server error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeTopicsDetectionJob](#) referência da API AWS SDK for SAP ABAP.

## DetectDominantLanguage

O código de exemplo a seguir mostra como usar DetectDominantLanguage.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_cpd->detectdominantlanguage( iv_text = iv_text ).
  MESSAGE 'Languages detected.' TYPE 'I'.
CATCH /aws1/cx_cpdtextrsizeexceedex.
  MESSAGE 'Text size exceeds limit.' TYPE 'E'.
CATCH /aws1/cx_cpinternalserverex.
  MESSAGE 'Internal server error occurred.' TYPE 'E'.
CATCH /aws1/cx_cpinvalidrequestex.
  MESSAGE 'Invalid request.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectDominantLanguage](#) referência da API AWS SDK for SAP ABAP.

## DetectEntities

O código de exemplo a seguir mostra como usar DetectEntities.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_cpd->detectentities(  
        iv_text = iv_text  
        iv_languagecode = iv_language_code  
    ).  
    MESSAGE 'Entities detected.' TYPE 'I'.  
CATCH /aws1/cx_cpdtextrisizelmtexcdex.  
    MESSAGE 'Text size exceeds limit.' TYPE 'E'.  
CATCH /aws1/cx_cpdundsuppedlanguageex.  
    MESSAGE 'Unsupported language.' TYPE 'E'.  
CATCH /aws1/cx_cpdiinternalserverex.  
    MESSAGE 'Internal server error occurred.' TYPE 'E'.  
CATCH /aws1/cx_cpdiinvalidrequestex.  
    MESSAGE 'Invalid request.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectEntities](#) referência da API AWS SDK for SAP ABAP.

## DetectKeyPhrases

O código de exemplo a seguir mostra como usar DetectKeyPhrases.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_cpd->detectkeyphrases(  
        iv_text = iv_text  
        iv_languagecode = iv_language_code  
    ).  
    MESSAGE 'Key phrases detected.' TYPE 'I'.  
CATCH /aws1/cx_cpdtextrsizeexceedex.  
    MESSAGE 'Text size exceeds limit.' TYPE 'E'.  
CATCH /aws1/cx_cpdundisupportedlanguageex.  
    MESSAGE 'Unsupported language.' TYPE 'E'.  
CATCH /aws1/cx_cpdiinternalserverex.  
    MESSAGE 'Internal server error occurred.' TYPE 'E'.  
CATCH /aws1/cx_cpdiinvalidrequestex.  
    MESSAGE 'Invalid request.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectKeyPhrases](#) referência da API AWS SDK for SAP ABAP.

## DetectPiiEntities

O código de exemplo a seguir mostra como usar DetectPiiEntities.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->detectpiientities(  
        iv_text = iv_text  
        iv_languagecode = iv_language_code  
    ).  
    MESSAGE 'PII entities detected.' TYPE 'I'.  
CATCH /aws1/cx_cpdtextrsizeexceedex.  
    MESSAGE 'Text size exceeds limit.' TYPE 'E'.  
CATCH /aws1/cx_cpdundisupportedlanguageex.  
    MESSAGE 'Unsupported language.' TYPE 'E'.  
CATCH /aws1/cx_cpdiinternalserverex.  
    MESSAGE 'Internal server error occurred.' TYPE 'E'.  
CATCH /aws1/cx_cpdiinvalidrequestex.  
    MESSAGE 'Invalid request.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectPiiEntities](#) referência da API AWS SDK for SAP ABAP.

## DetectSentiment

O código de exemplo a seguir mostra como usar DetectSentiment.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->detectsentiment(  
        iv_text = iv_text  
        iv_languagecode = iv_language_code  
    ).  
    MESSAGE 'Sentiment detected.' TYPE 'I'.  
CATCH /aws1/cx_cpdtextrsizeexceedex.  
    MESSAGE 'Text size exceeds limit.' TYPE 'E'.  
CATCH /aws1/cx_cpdundisupportedlanguageex.
```

```
MESSAGE 'Unsupported language.' TYPE 'E'.
CATCH /aws1/cx_cpdinternalserverex.
MESSAGE 'Internal server error occurred.' TYPE 'E'.
CATCH /aws1/cx_cpdivalidrequestex.
MESSAGE 'Invalid request.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectSentiment](#) referência da API AWS SDK for SAP ABAP.

## DetectSyntax

O código de exemplo a seguir mostra como usar DetectSyntax.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_cpd->detectsyntax(
    iv_text = iv_text
    iv_languagecode = iv_language_code
  ).
  MESSAGE 'Syntax tokens detected.' TYPE 'I'.
  CATCH /aws1/cx_cpdtextrsizefmtexcdex.
  MESSAGE 'Text size exceeds limit.' TYPE 'E'.
  CATCH /aws1/cx_cpdunsuppdedlanguageex.
  MESSAGE 'Unsupported language.' TYPE 'E'.
  CATCH /aws1/cx_cpdinternalserverex.
  MESSAGE 'Internal server error occurred.' TYPE 'E'.
  CATCH /aws1/cx_cpdivalidrequestex.
  MESSAGE 'Invalid request.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectSyntax](#) referência da API AWS SDK for SAP ABAP.

## ListDocumentClassificationJobs

O código de exemplo a seguir mostra como usar ListDocumentClassificationJobs.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->listdocclassificationjobs( ).  
    MESSAGE 'Document classification jobs listed.' TYPE 'I'.  
    CATCH /aws1/cx_cpdivalidrequestex.  
        MESSAGE 'Invalid request.' TYPE 'E'.  
    CATCH /aws1/cx_cpdtoomanyrequestsex.  
        MESSAGE 'Too many requests.' TYPE 'E'.  
    CATCH /aws1/cx_cpdivalidfilterex.  
        MESSAGE 'Invalid filter.' TYPE 'E'.  
    CATCH /aws1/cx_cpdivernalserverex.  
        MESSAGE 'Internal server error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListDocumentClassificationJobs](#) referência da API AWS SDK for SAP ABAP.

## ListDocumentClassifiers

O código de exemplo a seguir mostra como usar ListDocumentClassifiers.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->listdocumentclassifiers( ).  
    MESSAGE 'Document classifiers listed.' TYPE 'I'.  
CATCH /aws1/cx_cpinvalidrequestex.  
    MESSAGE 'Invalid request.' TYPE 'E'.  
CATCH /aws1/cx_cpdtoomanyrequestsex.  
    MESSAGE 'Too many requests.' TYPE 'E'.  
CATCH /aws1/cx_cpinvalidfilterex.  
    MESSAGE 'Invalid filter.' TYPE 'E'.  
CATCH /aws1/cx_cpinternalserverex.  
    MESSAGE 'Internal server error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListDocumentClassifiers](#) referência da API AWS SDK for SAP ABAP.

**ListTopicsDetectionJobs**

O código de exemplo a seguir mostra como usar ListTopicsDetectionJobs.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_cpd->listtopicsdetectionjobs( ).  
    MESSAGE 'Topics detection jobs listed.' TYPE 'I'.
```

```
CATCH /aws1/cx_cpinvalidrequestex.  
  MESSAGE 'Invalid request.' TYPE 'E'.  
CATCH /aws1/cx_cpdtoomanyrequestsex.  
  MESSAGE 'Too many requests.' TYPE 'E'.  
CATCH /aws1/cx_cpinvalidfilterex.  
  MESSAGE 'Invalid filter.' TYPE 'E'.  
CATCH /aws1/cx_cpinternalserverex.  
  MESSAGE 'Internal server error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListTopicsDetectionJobs](#) referência da API AWS SDK for SAP ABAP.

## StartDocumentClassificationJob

O código de exemplo a seguir mostra como usar StartDocumentClassificationJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_cpd->startdocclassificationjob(  
    iv_jobname = iv_job_name  
    iv_documentclassifierarn = iv_classifier_arn  
    io_inputdataconfig = NEW /aws1/cl_cpinputdataconfig(  
      iv_s3uri = iv_input_s3_uri  
      iv_inputformat = iv_input_format  
    )  
    io_outputdataconfig = NEW /aws1/cl_cpoutputdataconfig(  
      iv_s3uri = iv_output_s3_uri  
    )  
    iv_dataaccessrolearn = iv_data_access_role_arn  
  ).  
  MESSAGE 'Document classification job started.' TYPE 'I'.  
CATCH /aws1/cx_cpinvalidrequestex.
```

```
MESSAGE 'Invalid request.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanyrequestsex.
MESSAGE 'Too many requests.' TYPE 'E'.
CATCH /aws1/cx_cpdtresourcenotfoundex.
MESSAGE 'Resource not found.' TYPE 'E'.
CATCH /aws1/cx_cpdtresourceunavailex.
MESSAGE 'Resource unavailable.' TYPE 'E'.
CATCH /aws1/cx_cpdkmskeyvalidationex.
MESSAGE 'KMS key validation error.' TYPE 'E'.
CATCH /aws1/cx_cpdttoomanytagsex.
MESSAGE 'Too many tags.' TYPE 'E'.
CATCH /aws1/cx_cpdtresrclimitexcdex.
MESSAGE 'Resource limit exceeded.' TYPE 'E'.
CATCH /aws1/cx_cpdtinternalserverex.
MESSAGE 'Internal server error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartDocumentClassificationJob](#) referência da API AWS SDK for SAP ABAP.

## StartTopicsDetectionJob

O código de exemplo a seguir mostra como usar StartTopicsDetectionJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_cpd->starttopicsdetectionjob(
    iv_jobname = iv_job_name
    io_inputdataconfig = NEW /aws1/cl_cpdinputdataconfig(
      iv_s3uri = iv_input_s3_uri
      iv_inputformat = iv_input_format
    )
    io_outputdataconfig = NEW /aws1/cl_cpdoutputdataconfig(
```

```
        iv_s3uri = iv_output_s3_uri
    )
    iv_dataaccessrolearn = iv_data_access_role_arn
).
MESSAGE 'Topics detection job started.' TYPE 'I'.
CATCH /aws1/cx_cpinvalidrequestex.
    MESSAGE 'Invalid request.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanyrequestsex.
    MESSAGE 'Too many requests.' TYPE 'E'.
CATCH /aws1/cx_cpdkmskeyvalidationex.
    MESSAGE 'KMS key validation error.' TYPE 'E'.
CATCH /aws1/cx_cpdtoomanytagsex.
    MESSAGE 'Too many tags.' TYPE 'E'.
CATCH /aws1/cx_cpdrresrclimitexcdex.
    MESSAGE 'Resource limit exceeded.' TYPE 'E'.
CATCH /aws1/cx_cpinternalserverex.
    MESSAGE 'Internal server error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartTopicsDetectionJob](#) referência da API AWS SDK for SAP ABAP.

## AWS Config exemplos usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. AWS Config

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### DeleteConfigRule

O código de exemplo a seguir mostra como usar DeleteConfigRule.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
lo_cfs->deleteconfigrule( iv_rule_name ).  
MESSAGE 'Deleted AWS Config rule.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [DeleteConfigRule](#) referência da API AWS SDK for SAP ABAP.

### DescribeConfigRules

O código de exemplo a seguir mostra como usar DescribeConfigRules.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA(lo_result) = lo_cfs->describeconfigrules(  
  it_configrulenames = VALUE /aws1/cl_cfsconfigrulenames_w=>tt_configrulenames(  
    ( NEW /aws1/cl_cfsconfigrulenames_w( iv_rule_name ) )  
  )  
).  
ot_cfg_rules = lo_result->get_configrules( ).  
MESSAGE 'Retrieved AWS Config rule data.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [DescribeConfigRules](#) referência da API AWS SDK for SAP ABAP.

## PutConfigRule

O código de exemplo a seguir mostra como usar PutConfigRule.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Create a config rule for S3 bucket public read prohibition
lo_cfs->putconfigrule(
  io_configrule = NEW /aws1/cl_cfssconfigrule(
    iv_configrulename = iv_rule_name
    iv_description = |S3 Public Read Prohibited Bucket Rule|
    io_scope = NEW /aws1/cl_cfssscope(
      it_complianceresourcetypes = VALUE /aws1/
cl_cfsclnresrctypes_w=>tt_complianceresourcetypes(
        ( NEW /aws1/cl_cfsclnresrctypes_w( |AWS::S3::Bucket| ) )
      )
    )
    io_source = NEW /aws1/cl_cfsssource(
      iv_owner = |AWS|
      iv_sourceidentifier = |S3_BUCKET_PUBLIC_READ_PROHIBITED|
    )
    iv_inputparameters = '{} '
    iv_configrulestate = |ACTIVE|
  )
).
MESSAGE 'Created AWS Config rule.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [PutConfigRule](#) referência da API AWS SDK for SAP ABAP.

# AWS Control Tower exemplos usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. AWS Control Tower

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### **DisableBaseline**

O código de exemplo a seguir mostra como usar `DisableBaseline`.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Disable the baseline  
  DATA(lo_output) = io_ctt->disablebaseline(  
    iv_enabledbaselineidentifier = iv_enabled_baseline_identifier  
  ).  
  
  DATA(lv_operation_id) = lo_output->get_operationidentifier( ).  
  
  " Wait for operation to complete  
  DATA lv_status TYPE /aws1/cttbaselineopstatus.
```

```
DO 100 TIMES.
  lv_status = get_baseline_operation(
    io_ctt = io_ctt
    iv_operation_id = lv_operation_id
  ).

  DATA(lv_msg) = |Baseline operation status: { lv_status }|.
  MESSAGE lv_msg TYPE 'I'.

  IF lv_status = 'SUCCEEDED' OR lv_status = 'FAILED'.
    EXIT.
  ENDIF.

  " Wait 30 seconds
  WAIT UP TO 30 SECONDS.
ENDDO.

ov_operation_id = lv_operation_id.
MESSAGE 'Baseline disabled successfully.' TYPE 'I'.
CATCH /aws1/cx_cttconflictexception INTO DATA(lo_conflict).
  " Log conflict but don't fail - return empty operation ID
  DATA(lv_msg2) = |Conflict disabling baseline: { lo_conflict->get_text( ) }.
  Skipping disable step.|.
  MESSAGE lv_msg2 TYPE 'I'.
  CLEAR ov_operation_id.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DisableBaseline](#) referência da API AWS SDK for SAP ABAP.

## DisableControl

O código de exemplo a seguir mostra como usar `DisableControl`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Disable the control
DATA(lo_output) = io_ctt->disablecontrol(
  iv_controlidentifier = iv_control_arn
  iv_targetidentifier  = iv_target_identifier
).

DATA(lv_operation_id) = lo_output->get_operationidentifier( ).

" Wait for operation to complete
DATA lv_status TYPE /aws1/cttcontrolopstatus.
DO 100 TIMES.
  lv_status = get_control_operation(
    io_ctt = io_ctt
    iv_operation_id = lv_operation_id
  ).

DATA(lv_msg) = |Control operation status: { lv_status }|.
MESSAGE lv_msg TYPE 'I'.

IF lv_status = 'SUCCEEDED' OR lv_status = 'FAILED'.
  EXIT.
ENDIF.

" Wait 30 seconds
WAIT UP TO 30 SECONDS.
ENDDO.


ov_operation_id = lv_operation_id.
MESSAGE 'Control disabled successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [DisableControl](#) referência da API AWS SDK for SAP ABAP.

## EnableBaseline

O código de exemplo a seguir mostra como usar EnableBaseline.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" Prepare parameters for enabling baseline
DATA lt_parameters TYPE /aws1/
cl_cttenbdbaselineparam=>tt_enabledbaselineparameters.

" Add Identity Center baseline parameter if provided
IF iv_identity_center_baseline IS NOT INITIAL.
  " Create a JSON document with the baseline ARN value
  DATA(lv_json) = |\{ "IdentityCenterEnabledBaselineArn":
"{ iv_identity_center_baseline }" \}|.
  DATA(lo_param) = NEW /aws1/cl_cttenbdbaselineparam(
    iv_key = 'IdentityCenterEnabledBaselineArn'
    io_value = /aws1/cl_rt_document=>from_json_str( lv_json )
  ).
  APPEND lo_param TO lt_parameters.
ENDIF.

" Enable the baseline
DATA(lo_output) = io_ctt->enablebaseline(
  iv_baselineidentifier = iv_baseline_identifier
  iv_baselineversion    = iv_baseline_version
  iv_targetidentifier   = iv_target_identifier
  it_parameters         = lt_parameters
).

DATA(lv_operation_id) = lo_output->get_operationidentifier( ).

" Wait for operation to complete
DATA lv_status TYPE /aws1/cttbaselineopstatus.
DO 100 TIMES.
  lv_status = get_baseline_operation(
    io_ctt = io_ctt
    iv_operation_id = lv_operation_id
  ).

```

```
DATA(lv_msg) = |Baseline operation status: { lv_status }|.
MESSAGE lv_msg TYPE 'I'.

IF lv_status = 'SUCCEEDED' OR lv_status = 'FAILED'.
  EXIT.
ENDIF.

" Wait 30 seconds
WAIT UP TO 30 SECONDS.
ENDDO.

ov_enabled_baseline_arn = lo_output->get_arn( ).
MESSAGE 'Baseline enabled successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [EnableBaseline](#) referência da API AWS SDK for SAP ABAP.

## EnableControl

O código de exemplo a seguir mostra como usar EnableControl.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Enable the control
DATA(lo_output) = io_ctt->enablecontrol(
  iv_controlidentifier = iv_control_arn
  iv_targetidentifier  = iv_target_identifier
).

DATA(lv_operation_id) = lo_output->get_operationidentifier( ).

" Wait for operation to complete
DATA lv_status TYPE /aws1/cttcontrolopstatus.
DO 100 TIMES.
```

```
lv_status = get_control_operation(  
    io_ctt = io_ctt  
    iv_operation_id = lv_operation_id  
).  
  
DATA(lv_msg) = |Control operation status: { lv_status }|.  
MESSAGE lv_msg TYPE 'I'.  
  
IF lv_status = 'SUCCEEDED' OR lv_status = 'FAILED'.  
    EXIT.  
ENDIF.  
  
" Wait 30 seconds  
WAIT UP TO 30 SECONDS.  
ENDDO.  
  
ov_operation_id = lv_operation_id.  
MESSAGE 'Control enabled successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [EnableControl](#) referência da API AWS SDK for SAP ABAP.

## GetBaselineOperation

O código de exemplo a seguir mostra como usar GetBaselineOperation.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA(lo_output) = io_ctt->getbaselineoperation(  
    iv_operationidentifier = iv_operation_id  
).  
  
ov_status = lo_output->get_baselineoperation( )->get_status( ).
```

- Para obter detalhes da API, consulte a [GetBaselineOperation](#) referência da API AWS SDK for SAP ABAP.

## GetControlOperation

O código de exemplo a seguir mostra como usar `GetControlOperation`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA(lo_output) = io_ctt->getcontroloperation(  
    iv_operationidentifier = iv_operation_id  
    ).  
  
ov_status = lo_output->get_controloperation( )->get_status( ).
```

- Para obter detalhes da API, consulte a [GetControlOperation](#) referência da API AWS SDK for SAP ABAP.

## ListBaselines

O código de exemplo a seguir mostra como usar `ListBaselines`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_baselines TYPE /aws1/cl_cttbaselinessummary=>tt_baselines.  
DATA lv_nexttoken TYPE /aws1/cttstring.
```

```

" List all baselines using pagination
DO.
  DATA(lo_output) = io_ctt->listbaselines(
    iv_nexttoken = lv_nexttoken
  ).

  APPEND LINES OF lo_output->get_baselines( ) TO lt_baselines.

  lv_nexttoken = lo_output->get_nexttoken( ).
  IF lv_nexttoken IS INITIAL.
    EXIT.
  ENDIF.
ENDDO.

ot_baselines = lt_baselines.
MESSAGE 'Listed baselines successfully.' TYPE 'I'.

```

- Para obter detalhes da API, consulte a [ListBaselines](#) referência da API AWS SDK for SAP ABAP.

## ListEnabledBaselines

O código de exemplo a seguir mostra como usar ListEnabledBaselines.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA lt_enabled_baselines TYPE /aws1/
cl_cttenbdbaselinesumm=>tt_enabledbaselines.
DATA lv_nexttoken TYPE /aws1/cttlstenbdbaselinesnex00.

" List all enabled baselines using pagination
DO.
  DATA(lo_output) = io_ctt->listenabledbaselines(

```

```

        lv_nexttoken = lv_nexttoken
    ).

    APPEND LINES OF lo_output->get_enabledbaselines( ) TO lt_enabled_baselines.

    lv_nexttoken = lo_output->get_nexttoken( ).
    IF lv_nexttoken IS INITIAL.
        EXIT.
    ENDIF.
ENDDO.

ot_enabled_baselines = lt_enabled_baselines.
MESSAGE 'Listed enabled baselines successfully.' TYPE 'I'.

```

- Para obter detalhes da API, consulte a [ListEnabledBaselines](#) referência da API AWS SDK for SAP ABAP.

## ListEnabledControls

O código de exemplo a seguir mostra como usar ListEnabledControls.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA lt_enabled_controls TYPE /aws1/
cl_cttenabledcontrolsumm=>tt_enabledcontrols.
DATA lv_nexttoken TYPE /aws1/cttstring.

" List all enabled controls using pagination
DO.
    DATA(lo_output) = io_ctt->listenabledcontrols(
        iv_targetidentifier = iv_target_identifier
        iv_nexttoken       = lv_nexttoken
    ).

```

```
APPEND LINES OF lo_output->get_enabledcontrols( ) TO lt_enabled_controls.

lv_nexttoken = lo_output->get_nexttoken( ).
IF lv_nexttoken IS INITIAL.
    EXIT.
ENDIF.
ENDDO.

ot_enabled_controls = lt_enabled_controls.
MESSAGE 'Listed enabled controls successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [ListEnabledControls](#) referência da API AWS SDK for SAP ABAP.

## ListLandingZones

O código de exemplo a seguir mostra como usar ListLandingZones.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_landing_zones TYPE /aws1/
cl_cttlandingzonessummary=>tt_landingzonessummaries.
DATA lv_nexttoken TYPE /aws1/cttstring.

" List all landing zones using pagination
DO.
    DATA(lo_output) = io_ctt->listlandingzones(
        iv_nexttoken = lv_nexttoken
    ).

    APPEND LINES OF lo_output->get_landingzones( ) TO lt_landing_zones.

    lv_nexttoken = lo_output->get_nexttoken( ).
    IF lv_nexttoken IS INITIAL.
```

```
EXIT.  
ENDIF.  
ENDDO.  
  
ot_landing_zones = lt_landing_zones.  
MESSAGE 'Listed landing zones successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [ListLandingZones](#) referência da API AWS SDK for SAP ABAP.

## ResetEnabledBaseline

O código de exemplo a seguir mostra como usar `ResetEnabledBaseline`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Reset the enabled baseline  
DATA(lo_output) = io_ctt->resetenabledbaseline(  
    iv_enabledbaselineidentifier = iv_enabled_baseline_identifier  
).  
  
DATA(lv_operation_id) = lo_output->get_operationidentifier( ).  
  
" Wait for operation to complete  
DATA lv_status TYPE /aws1/cttbaselineopstatus.  
DO 100 TIMES.  
    lv_status = get_baseline_operation(  
        io_ctt = io_ctt  
        iv_operation_id = lv_operation_id  
    ).  
  
DATA(lv_msg) = |Baseline operation status: { lv_status }|.  
MESSAGE lv_msg TYPE 'I'.
```

```
IF lv_status = 'SUCCEEDED' OR lv_status = 'FAILED'.  
  EXIT.  
ENDIF.  
  
" Wait 30 seconds  
WAIT UP TO 30 SECONDS.  
ENDDO.  
  
ov_operation_id = lv_operation_id.  
MESSAGE 'Baseline reset successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [ResetEnabledBaseline](#) referência da API AWS SDK for SAP ABAP.

## Exemplos de Firehose usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Firehose.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### PutRecord

O código de exemplo a seguir mostra como usar PutRecord.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  DATA(lo_record) = NEW /aws1/cl_frhrecord( iv_data = iv_data ).  
  
  DATA(lo_result) = lo_frh->putrecord(  
    iv_deliverystreamname = iv_deliv_stream_name  
    io_record              = lo_record ).  
  
  MESSAGE 'Record sent to Firehose delivery stream.' TYPE 'I'.  
  CATCH /aws1/cx_frhresourcenotfoundex.  
    MESSAGE 'Delivery stream not found.' TYPE 'E'.  
  CATCH /aws1/cx_frhinvalidargumentex.  
    MESSAGE 'Invalid argument provided.' TYPE 'E'.  
  CATCH /aws1/cx_frhserviceunavailex.  
    MESSAGE 'Service temporarily unavailable.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutRecord](#) referência da API AWS SDK for SAP ABAP.

**PutRecordBatch**

O código de exemplo a seguir mostra como usar PutRecordBatch.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
DATA(lo_result) = lo_frh->putrecordbatch(
  iv_deliverystreamname = iv_deliv_stream_name
  it_records             = it_records ).

DATA(lv_failed_count) = lo_result->get_failedputcount( ).

IF lv_failed_count > 0.
  MESSAGE |{ lv_failed_count } records failed to send.| TYPE 'I'.
ELSE.
  MESSAGE 'All records sent successfully to Firehose delivery stream.' TYPE
'I'.
ENDIF.
CATCH /aws1/cx_frhresourcefoundex.
  MESSAGE 'Delivery stream not found.' TYPE 'E'.
CATCH /aws1/cx_frhinvalidargumentex.
  MESSAGE 'Invalid argument provided.' TYPE 'E'.
CATCH /aws1/cx_frhserviceunavailex.
  MESSAGE 'Service temporarily unavailable.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutRecordBatch](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do DynamoDB usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o DynamoDB.

As noções básicas são exemplos de código que mostram como realizar as operações essenciais em um serviço.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Conceitos básicos](#)

- [Ações](#)

## Conceitos básicos

Conheça os conceitos básicos

O exemplo de código a seguir mostra como:

- Criar uma tabela que possa conter dados de filmes.
- Colocar, obter e atualizar um único filme na tabela.
- Gravar dados de filmes na tabela usando um arquivo JSON de exemplo.
- Consultar filmes que foram lançados em determinado ano.
- Verificar filmes que foram lançados em um intervalo de anos.
- Excluir um filme da tabela e, depois, excluir a tabela.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Create an Amazon Dynamo DB table.

TRY.
  DATA(lo_session) = /aws1/cl_rt_session_aws=>create( cv_pfl ).
  DATA(lo_dyn) = /aws1/cl_dyn_factory=>create( lo_session ).
  DATA(lt_keyschema) = VALUE /aws1/cl_dynkeyschemaelement=>tt_keyschema(
    ( NEW /aws1/cl_dynkeyschemaelement( iv_attributename = 'year'
                                          iv_keytype = 'HASH' ) )
    ( NEW /aws1/cl_dynkeyschemaelement( iv_attributename = 'title'
                                          iv_keytype = 'RANGE' ) ) ).
  DATA(lt_attributedefinitions) = VALUE /aws1/
cl_dynattributedefn=>tt_attributedefinitions(
    ( NEW /aws1/cl_dynattributedefn( iv_attributename = 'year'
                                     iv_attributetype = 'N' ) )
    ( NEW /aws1/cl_dynattributedefn( iv_attributename = 'title'
                                     iv_attributetype = 'S' ) ) ).
```

```

" Adjust read/write capacities as desired.
DATA(lo_dynprovthroughput) = NEW /aws1/cl_dynprovthroughput(
  iv_readcapacityunits = 5
  iv_writecapacityunits = 5 ).
DATA(oo_result) = lo_dyn->createtable(
  it_keyschema = lt_keyschema
  iv_tablename = iv_table_name
  it_attributedefinitions = lt_attributedefinitions
  io_provisionedthroughput = lo_dynprovthroughput ).
" Table creation can take some time. Wait till table exists before
returning.
lo_dyn->get_waiter( )->tableexists(
  iv_max_wait_time = 200
  iv_tablename      = iv_table_name ).
MESSAGE 'DynamoDB Table' && iv_table_name && 'created.' TYPE 'I'.
" It throws exception if the table already exists.
CATCH /aws1/cx_dynresourceinuseex INTO DATA(lo_resourceinuseex).
  DATA(lv_error) = |"{ lo_resourceinuseex->av_err_code }" -
{ lo_resourceinuseex->av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.

" Describe table
TRY.
  DATA(lo_table) = lo_dyn->describetable( iv_tablename = iv_table_name ).
  DATA(lv_tablename) = lo_table->get_table( )->ask_tablename( ).
  MESSAGE 'The table name is ' && lv_tablename TYPE 'I'.
CATCH /aws1/cx_dynresourcenotfoundex.
  MESSAGE 'The table does not exist' TYPE 'E'.
ENDTRY.

" Put items into the table.
TRY.
  DATA(lo_resp_putitem) = lo_dyn->putitem(
    iv_tablename = iv_table_name
    it_item       = VALUE /aws1/
cl_dynattributevalue=>tt_putiteminputattributemap(
  ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
    key = 'title' value = NEW /aws1/cl_dynattributevalue( iv_s =
'Jaws' ) ) )
  ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
    key = 'year' value = NEW /aws1/cl_dynattributevalue( iv_n = |
{ '1975' }| ) ) ) )

```

```

        ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
          key = 'rating' value = NEW /aws1/cl_dynattributevalue( iv_n = |
{ '7.5' }| ) ) ) )
      ) ).
    lo_resp_putitem = lo_dyn->putitem(
      iv_tablename = iv_table_name
      it_item       = VALUE /aws1/
cl_dynattributevalue=>tt_putiteminputattributemap(
      ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
        key = 'title' value = NEW /aws1/cl_dynattributevalue( iv_s = 'Star
Wars' ) ) ) )
      ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
        key = 'year' value = NEW /aws1/cl_dynattributevalue( iv_n = |
{ '1978' }| ) ) ) )
      ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
        key = 'rating' value = NEW /aws1/cl_dynattributevalue( iv_n = |
{ '8.1' }| ) ) ) )
      ) ).
    lo_resp_putitem = lo_dyn->putitem(
      iv_tablename = iv_table_name
      it_item       = VALUE /aws1/
cl_dynattributevalue=>tt_putiteminputattributemap(
      ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
        key = 'title' value = NEW /aws1/cl_dynattributevalue( iv_s =
'Speed' ) ) ) )
      ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
        key = 'year' value = NEW /aws1/cl_dynattributevalue( iv_n = |
{ '1994' }| ) ) ) )
      ( VALUE /aws1/cl_dynattributevalue=>ts_putiteminputattrmap_maprow(
        key = 'rating' value = NEW /aws1/cl_dynattributevalue( iv_n = |
{ '7.9' }| ) ) ) )
      ) ).
    " TYPE REF TO /AWSEX/CL_AWS1_dyn_PUT_ITEM_OUTPUT
    MESSAGE '3 rows inserted into DynamoDB Table' && iv_table_name TYPE 'I'.
    CATCH /aws1/cx_dyncondalcheckfaile00.
    MESSAGE 'A condition specified in the operation could not be evaluated.'
TYPE 'E'.
    CATCH /aws1/cx_dynresourcenotfoundex.
    MESSAGE 'The table or index does not exist' TYPE 'E'.
    CATCH /aws1/cx_dyntransactconflictex.
    MESSAGE 'Another transaction is using the item' TYPE 'E'.
    ENDTRY.

    " Get item from table.

```

```

TRY.
  DATA(lo_resp_getitem) = lo_dyn->getitem(
    iv_tablename          = iv_table_name
    it_key                = VALUE /aws1/cl_dynattributevalue=>tt_key(
      ( VALUE /aws1/cl_dynattributevalue=>ts_key_maprow(
        key = 'title' value = NEW /aws1/cl_dynattributevalue( iv_s =
'Jaws' ) ) )
      ( VALUE /aws1/cl_dynattributevalue=>ts_key_maprow(
        key = 'year' value = NEW /aws1/cl_dynattributevalue( iv_n =
'1975' ) ) )
    ) ).
  DATA(lt_attr) = lo_resp_getitem->get_item( ).
  DATA(lo_title) = lt_attr[ key = 'title' ]-value.
  DATA(lo_year) = lt_attr[ key = 'year' ]-value.
  DATA(lo_rating) = lt_attr[ key = 'year' ]-value.
  MESSAGE 'Movie name is: ' && lo_title->get_s( ) TYPE 'I'.
  MESSAGE 'Movie year is: ' && lo_year->get_n( ) TYPE 'I'.
  MESSAGE 'Movie rating is: ' && lo_rating->get_n( ) TYPE 'I'.
  CATCH /aws1/cx_dynresourcenotfoundex.
    MESSAGE 'The table or index does not exist' TYPE 'E'.
  ENDTRY.

" Query item from table.
TRY.
  DATA(lt_attributelist) = VALUE /aws1/
cl_dynattributevalue=>tt_attributelist(
    ( NEW /aws1/cl_dynattributevalue( iv_n = '1975' ) ) ).
  DATA(lt_keyconditions) = VALUE /aws1/cl_dyncondition=>tt_keyconditions(
    ( VALUE /aws1/cl_dyncondition=>ts_keyconditions_maprow(
      key = 'year'
      value = NEW /aws1/cl_dyncondition(
        it_attributelist = lt_attributelist
        iv_comparisonoperator = |EQ|
      ) ) ) ).
  DATA(lo_query_result) = lo_dyn->query(
    iv_tablename = iv_table_name
    it_keyconditions = lt_keyconditions ).
  DATA(lt_items) = lo_query_result->get_items( ).
  READ TABLE lo_query_result->get_items( ) INTO DATA(lt_item) INDEX 1.
  lo_title = lt_item[ key = 'title' ]-value.
  lo_year = lt_item[ key = 'year' ]-value.
  lo_rating = lt_item[ key = 'rating' ]-value.
  MESSAGE 'Movie name is: ' && lo_title->get_s( ) TYPE 'I'.
  MESSAGE 'Movie year is: ' && lo_year->get_n( ) TYPE 'I'.

```

```

    MESSAGE 'Movie rating is: ' && lo_rating->get_n( ) TYPE 'I'.
  CATCH /aws1/cx_dynresourcenotfoundex.
    MESSAGE 'The table or index does not exist' TYPE 'E'.
ENDTRY.

" Scan items from table.
TRY.
  DATA(lo_scan_result) = lo_dyn->scan( iv_tablename = iv_table_name ).
  lt_items = lo_scan_result->get_items( ).
  " Read the first item and display the attributes.
  READ TABLE lo_query_result->get_items( ) INTO lt_item INDEX 1.
  lo_title = lt_item[ key = 'title' ]-value.
  lo_year = lt_item[ key = 'year' ]-value.
  lo_rating = lt_item[ key = 'rating' ]-value.
  MESSAGE 'Movie name is: ' && lo_title->get_s( ) TYPE 'I'.
  MESSAGE 'Movie year is: ' && lo_year->get_n( ) TYPE 'I'.
  MESSAGE 'Movie rating is: ' && lo_rating->get_n( ) TYPE 'I'.
  CATCH /aws1/cx_dynresourcenotfoundex.
    MESSAGE 'The table or index does not exist' TYPE 'E'.
ENDTRY.

" Update items from table.
TRY.
  DATA(lt_attributeupdates) = VALUE /aws1/
cl_dynattrvalueupdate=>tt_attributeupdates(
    ( VALUE /aws1/cl_dynattrvalueupdate=>ts_attributeupdates_maprow(
      key = 'rating' value = NEW /aws1/cl_dynattrvalueupdate(
        io_value = NEW /aws1/cl_dynattributevalue( iv_n = '7.6' )
        iv_action = |PUT| ) ) ) ).
  DATA(lt_key) = VALUE /aws1/cl_dynattributevalue=>tt_key(
    ( VALUE /aws1/cl_dynattributevalue=>ts_key_maprow(
      key = 'year' value = NEW /aws1/cl_dynattributevalue( iv_n = '1975' ) ) )
    ( VALUE /aws1/cl_dynattributevalue=>ts_key_maprow(
      key = 'title' value = NEW /aws1/cl_dynattributevalue( iv_s =
'1980' ) ) ) ).
  DATA(lo_resp) = lo_dyn->updateitem(
    iv_tablename      = iv_table_name
    it_key            = lt_key
    it_attributeupdates = lt_attributeupdates ).
  MESSAGE '1 item updated in DynamoDB Table' && iv_table_name TYPE 'I'.
  CATCH /aws1/cx_dyncondalcheckfaile00.
    MESSAGE 'A condition specified in the operation could not be evaluated.'
TYPE 'E'.
  CATCH /aws1/cx_dynresourcenotfoundex.

```

```
MESSAGE 'The table or index does not exist' TYPE 'E'.
CATCH /aws1/cx_dyntransactconflictex.
MESSAGE 'Another transaction is using the item' TYPE 'E'.
ENDTRY.

" Delete table.
TRY.
  lo_dyn->deletetable( iv_tablename = iv_table_name ).
  lo_dyn->get_waiter( )->tablenotexists(
    iv_max_wait_time = 200
    iv_tablename      = iv_table_name ).
  MESSAGE 'DynamoDB Table deleted.' TYPE 'I'.
CATCH /aws1/cx_dynresourcefoundex.
MESSAGE 'The table or index does not exist' TYPE 'E'.
CATCH /aws1/cx_dynresourceinuseex.
MESSAGE 'The table cannot be deleted as it is in use' TYPE 'E'.
ENDTRY.
```


- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [BatchWriteItem](#)
  - [CreateTable](#)
  - [DeleteItem](#)
  - [DeleteTable](#)
  - [DescribeTable](#)
  - [GetItem](#)
  - [PutItem](#)
  - [Query](#)
  - [Scan](#)
  - [UpdateItem](#)

## Ações

### CreateTable

O código de exemplo a seguir mostra como usar CreateTable.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  DATA(lt_keyschema) = VALUE /aws1/cl_dynkeyschemaelement=>tt_keyschema(
    ( NEW /aws1/cl_dynkeyschemaelement( iv_attributename = 'year'
                                          iv_keytype = 'HASH' ) )
    ( NEW /aws1/cl_dynkeyschemaelement( iv_attributename = 'title'
                                          iv_keytype = 'RANGE' ) ) ).
  DATA(lt_attributedefinitions) = VALUE /aws1/
cl_dynattributedefn=>tt_attributedefinitions(
    ( NEW /aws1/cl_dynattributedefn( iv_attributename = 'year'
                                     iv_attributetype = 'N' ) )
    ( NEW /aws1/cl_dynattributedefn( iv_attributename = 'title'
                                     iv_attributetype = 'S' ) ) ).

  " Adjust read/write capacities as desired.
  DATA(lo_dynprovthroughput) = NEW /aws1/cl_dynprovthroughput(
    iv_readcapacityunits = 5
    iv_writecapacityunits = 5 ).
  oo_result = lo_dyn->createtable(
    it_keyschema = lt_keyschema
    iv_tablename = iv_table_name
    it_attributedefinitions = lt_attributedefinitions
    io_provisionedthroughput = lo_dynprovthroughput ).
  " Table creation can take some time. Wait till table exists before
  returning.
  lo_dyn->get_waiter( )->tableexists(
    iv_max_wait_time = 200
    iv_tablename      = iv_table_name ).
  MESSAGE 'DynamoDB Table' && iv_table_name && 'created.' TYPE 'I'.
  " This exception can happen if the table already exists.
  CATCH /aws1/cx_dynresourceinuseex INTO DATA(lo_resourceinuseex).
    DATA(lv_error) = |"{ lo_resourceinuseex->av_err_code }" -
{ lo_resourceinuseex->av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateTable](#) referência da API AWS SDK for SAP ABAP.

## DeleteItem

O código de exemplo a seguir mostra como usar DeleteItem.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  DATA(lo_resp) = lo_dyn->deleteitem(  
    iv_tablename          = iv_table_name  
    it_key                = it_key_input ).  
  MESSAGE 'Deleted one item.' TYPE 'I'.  
CATCH /aws1/cx_dyncondalcheckfaile00.  
  MESSAGE 'A condition specified in the operation could not be evaluated.'  
TYPE 'E'.  
CATCH /aws1/cx_dynresourcenotfoundex.  
  MESSAGE 'The table or index does not exist' TYPE 'E'.  
CATCH /aws1/cx_dyntransactconflictex.  
  MESSAGE 'Another transaction is using the item' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteItem](#) referência da API AWS SDK for SAP ABAP.

## DeleteTable

O código de exemplo a seguir mostra como usar DeleteTable.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_dyn->deletetable( iv_tablename = iv_table_name ).  
  " Wait till the table is actually deleted.  
  lo_dyn->get_waiter( )->tablenotexists(  
    iv_max_wait_time = 200  
    iv_tablename     = iv_table_name ).  
  MESSAGE 'Table ' && iv_table_name && ' deleted.' TYPE 'I'.  
CATCH /aws1/cx_dynresourcenotfoundex.  
  MESSAGE 'The table ' && iv_table_name && ' does not exist' TYPE 'E'.  
CATCH /aws1/cx_dynresourceinuseex.  
  MESSAGE 'The table cannot be deleted since it is in use' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteTable](#) referência da API AWS SDK for SAP ABAP.

## DescribeTable

O código de exemplo a seguir mostra como usar DescribeTable.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_dyn->describetable( iv_tablename = iv_table_name ).  
  DATA(lv_tablename) = oo_result->get_table( )->ask_tablename( ).
```

```

DATA(lv_tablearn) = oo_result->get_table( )->ask_tablearn( ).
DATA(lv_tablestatus) = oo_result->get_table( )->ask_tablestatus( ).
DATA(lv_itemcount) = oo_result->get_table( )->ask_itemcount( ).
MESSAGE 'The table name is ' && lv_tablename
        && '. The table ARN is ' && lv_tablearn
        && '. The tablestatus is ' && lv_tablestatus
        && '. Item count is ' && lv_itemcount TYPE 'I'.
CATCH /aws1/cx_dynresourcenotfoundex.
    MESSAGE 'The table ' && lv_tablename && ' does not exist' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeTable](#) referência da API AWS SDK for SAP ABAP.

## GetItem

O código de exemplo a seguir mostra como usar GetItem.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_item = lo_dyn->getitem(
        iv_tablename      = iv_table_name
        it_key             = it_key ).
    DATA(lt_attr) = oo_item->get_item( ).
    DATA(lo_title) = lt_attr[ key = 'title' ]-value.
    DATA(lo_year) = lt_attr[ key = 'year' ]-value.
    DATA(lo_rating) = lt_attr[ key = 'rating' ]-value.
    MESSAGE 'Movie name is: ' && lo_title->get_s( )
            && 'Movie year is: ' && lo_year->get_n( )
            && 'Moving rating is: ' && lo_rating->get_n( ) TYPE 'I'.
    CATCH /aws1/cx_dynresourcenotfoundex.
        MESSAGE 'The table or index does not exist' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetItem](#) referência da API AWS SDK for SAP ABAP.

## ListTables

O código de exemplo a seguir mostra como usar ListTables.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_dyn->listtables( ).  
    " You can loop over the oo_result to get table properties like this.  
    LOOP AT oo_result->get_tablenames( ) INTO DATA(lo_table_name).  
        DATA(lv_tablename) = lo_table_name->get_value( ).  
    ENDLOOP.  
    DATA(lv_tablecount) = lines( oo_result->get_tablenames( ) ).  
    MESSAGE 'Found ' && lv_tablecount && ' tables' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->av_err_msg }|.  
        MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListTables](#) referência da API AWS SDK for SAP ABAP.

## PutItem

O código de exemplo a seguir mostra como usar PutItem.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  DATA(lo_resp) = lo_dyn->putitem(  
    iv_tablename = iv_table_name  
    it_item      = it_item ).  
  MESSAGE '1 row inserted into DynamoDB Table' && iv_table_name TYPE 'I'.  
CATCH /aws1/cx_dyncondalcheckfaile00.  
  MESSAGE 'A condition specified in the operation could not be evaluated.'  
TYPE 'E'.  
CATCH /aws1/cx_dynresourcenotfoundex.  
  MESSAGE 'The table or index does not exist' TYPE 'E'.  
CATCH /aws1/cx_dyntransactconflictex.  
  MESSAGE 'Another transaction is using the item' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutItem](#) referência da API AWS SDK for SAP ABAP.

## Query

O código de exemplo a seguir mostra como usar Query.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

" Query movies for a given year .
DATA(lt_attributelist) = VALUE /aws1/
cl_dynattributevalue=>tt_attributevaluelist(
  ( NEW /aws1/cl_dynattributevalue( iv_n = |{ iv_year }| ) ) ).
DATA(lt_key_conditions) = VALUE /aws1/cl_dyncondition=>tt_keyconditions(
  ( VALUE /aws1/cl_dyncondition=>ts_keyconditions_maprow(
    key = 'year'
    value = NEW /aws1/cl_dyncondition(
      it_attributevaluelist = lt_attributelist
      iv_comparisonoperator = |EQ|
    ) ) ) ).
oo_result = lo_dyn->query(
  iv_tablename = iv_table_name
  it_keyconditions = lt_key_conditions ).
DATA(lt_items) = oo_result->get_items( ).
"You can loop over the results to get item attributes.
LOOP AT lt_items INTO DATA(lt_item).
  DATA(lo_title) = lt_item[ key = 'title' ]-value.
  DATA(lo_year) = lt_item[ key = 'year' ]-value.
ENDLOOP.
DATA(lv_count) = oo_result->get_count( ).
MESSAGE 'Item count is: ' && lv_count TYPE 'I'.
CATCH /aws1/cx_dynresourcenotfoundex.
MESSAGE 'The table or index does not exist' TYPE 'E'.
ENDTRY.

```

- Para obter os detalhes da API, consulte [Query](#) na Referência da API AWS SDK para SAP ABAP.

## Scan

O código de exemplo a seguir mostra como usar Scan.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Scan movies for rating greater than or equal to the rating specified
  DATA(lt_attributelist) = VALUE /aws1/
cl_dynattributevalue=>tt_attributevaluelist(
  ( NEW /aws1/cl_dynattributevalue( iv_n = |{ iv_rating }| ) ) ).
  DATA(lt_filter_conditions) = VALUE /aws1/
cl_dyncondition=>tt_filterconditionmap(
  ( VALUE /aws1/cl_dyncondition=>ts_filterconditionmap_maprow(
    key = 'rating'
    value = NEW /aws1/cl_dyncondition(
      it_attributelist = lt_attributelist
      iv_comparisonoperator = |GE|
    ) ) ) ).
  oo_scan_result = lo_dyn->scan( iv_tablename = iv_table_name
    it_scanfilter = lt_filter_conditions ).
  DATA(lt_items) = oo_scan_result->get_items( ).
  LOOP AT lt_items INTO DATA(lo_item).
    " You can loop over to get individual attributes.
    DATA(lo_title) = lo_item[ key = 'title' ]-value.
    DATA(lo_year) = lo_item[ key = 'year' ]-value.
  ENDLLOOP.
  DATA(lv_count) = oo_scan_result->get_count( ).
  MESSAGE 'Found ' && lv_count && ' items' TYPE 'I'.
  CATCH /aws1/cx_dynresourcenotfoundex.
  MESSAGE 'The table or index does not exist' TYPE 'E'.
ENDTRY.

```

- Para obter os detalhes da API, consulte [Scan](#) na Referência da API AWS SDK para SAP ABAP.

## UpdateItem

O código de exemplo a seguir mostra como usar UpdateItem.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_output = lo_dyn->updateitem(  
        iv_tablename      = iv_table_name  
        it_key             = it_item_key  
        it_attributeupdates = it_attribute_updates ).  
    MESSAGE '1 item updated in DynamoDB Table' && iv_table_name TYPE 'I'.  
CATCH /aws1/cx_dyncondalcheckfaile00.  
    MESSAGE 'A condition specified in the operation could not be evaluated.'  
TYPE 'E'.  
CATCH /aws1/cx_dynresourcenotfoundex.  
    MESSAGE 'The table or index does not exist' TYPE 'E'.  
CATCH /aws1/cx_dyntransactconflictex.  
    MESSAGE 'Another transaction is using the item' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateItem](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon EC2 usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon EC2.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos


- [Ações](#)

### Ações

#### **AllocateAddress**

O código de exemplo a seguir mostra como usar `AllocateAddress`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_ec2->allocateaddress( iv_domain = 'vpc' ).    " oo_result is  
returned for testing purposes. "  
    MESSAGE 'Allocated an Elastic IP address.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [AllocateAddress](#) referência da API AWS SDK for SAP ABAP.

## AssociateAddress

O código de exemplo a seguir mostra como usar AssociateAddress.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ec2->associateaddress(                               " oo_result is  
returned for testing purposes. "  
        iv_allocationid = iv_allocation_id  
        iv_instanceid = iv_instance_id ).
```

```

    MESSAGE 'Associated an Elastic IP address with an EC2 instance.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
    ENTRY.

```

- Para obter detalhes da API, consulte a [AssociateAddress](#) referência da API AWS SDK for SAP ABAP.

## AuthorizeSecurityGroupIngress

O código de exemplo a seguir mostra como usar AuthorizeSecurityGroupIngress.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" Create IP permissions for SSH access (port 22)
" iv_cidr_ip = '192.0.2.0/24'
DATA lt_ip_permissions TYPE /aws1/cl_ec2ippermission=>tt_ippermissionlist.
DATA(lo_ip_permission) = NEW /aws1/cl_ec2ippermission(
  iv_ipprotocol = 'tcp'
  iv_fromport = 22
  iv_toport = 22
  it_ipranges = VALUE /aws1/cl_ec2iprange=>tt_iprangelist(
    ( NEW /aws1/cl_ec2iprange( iv_cidrip = iv_cidr_ip ) )
  )
).
APPEND lo_ip_permission TO lt_ip_permissions.

TRY.
  oo_result = lo_ec2->authsecuritygroupingress(           " oo_result is
returned for testing purposes. "
  iv_groupid = iv_group_id
  it_ippermissions = lt_ip_permissions ).

```

```

    MESSAGE 'Authorized ingress rule for security group.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
    ENTRY.

```

- Para obter detalhes da API, consulte a [AuthorizeSecurityGroupIngress](#) referência da API AWS SDK for SAP ABAP.

## CreateKeyPair

O código de exemplo a seguir mostra como usar CreateKeyPair.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

    TRY.
        oo_result = lo_ec2->createkeypair( iv_keyname = iv_key_name ).
        " oo_result is returned for testing purposes. "
        MESSAGE 'Amazon EC2 key pair created.' TYPE 'I'.
        CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
        ENTRY.


```

- Para obter detalhes da API, consulte a [CreateKeyPair](#) referência da API AWS SDK for SAP ABAP.

## CreateSecurityGroup

O código de exemplo a seguir mostra como usar CreateSecurityGroup.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
    oo_result = lo_ec2->createsecuritygroup(
        iv_description = 'Security group example'
        iv_groupname = iv_security_group_name
        iv_vpcid = iv_vpc_id ).
    MESSAGE 'Security group created.' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
    >av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateSecurityGroup](#) referência da API AWS SDK for SAP ABAP.

## CreateVpc

O código de exemplo a seguir mostra como usar CreateVpc.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" iv_cidr_block = '10.0.0.0/16'
TRY.
```

```

        oo_result = lo_ec2->createvpc( iv_cidrblock = iv_cidr_block ).
oo_result is returned for testing purposes. "
        DATA(lv_vpc_id) = oo_result->get_vpc( )->get_vpcid( ).
        MESSAGE 'Created VPC.' TYPE 'I'.
        CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateVpc](#) referência da API AWS SDK for SAP ABAP.

## CreateVpcEndpoint

O código de exemplo a seguir mostra como usar CreateVpcEndpoint.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" iv_vpc_id = 'vpc-abc123'
" iv_service_name = 'com.amazonaws.region.service'
TRY.
    oo_result = lo_ec2->createvpcendpoint(
for testing purposes. "
        iv_vpcid = iv_vpc_id
        iv_servicename = iv_service_name
        it_routetableids = it_route_table_ids ).
    DATA(lv_vpc_endpoint_id) = oo_result->get_vpcendpoint( )-
>get_vpcendpointid( ).
    MESSAGE 'Created VPC endpoint.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateVpcEndpoint](#) referência da API AWS SDK for SAP ABAP.

## DeleteKeyPair

O código de exemplo a seguir mostra como usar DeleteKeyPair.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    lo_ec2->deletekeypair( iv_keyname = iv_key_name ).  
    MESSAGE 'Amazon EC2 key pair deleted.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteKeyPair](#) referência da API AWS SDK for SAP ABAP.

## DeleteSecurityGroup

O código de exemplo a seguir mostra como usar DeleteSecurityGroup.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    lo_ec2->deletesecuritygroup( iv_groupid = iv_security_group_id ).  
    MESSAGE 'Security group deleted.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteSecurityGroup](#) preferência da API AWS SDK for SAP ABAP.

## DeleteVpc

O código de exemplo a seguir mostra como usar DeleteVpc.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ec2->deletevpc( iv_vpcid = iv_vpc_id ).  
    MESSAGE 'Deleted VPC.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|. 
```

```
MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteVpc](#) referência da API AWS SDK for SAP ABAP.

## DeleteVpcEndpoints

O código de exemplo a seguir mostra como usar DeleteVpcEndpoints.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    lo_ec2->deletevpcendpoints( it_vpcendpointids = it_vpc_endpoint_ids ).  
    MESSAGE 'Deleted VPC endpoint(s).' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteVpcEndpoints](#) referência da API AWS SDK for SAP ABAP.

## DescribeAddresses

O código de exemplo a seguir mostra como usar DescribeAddresses.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_ec2->describeaddresses( ). " oo_result  
is returned for testing purposes. "  
    DATA(lt_addresses) = oo_result->get_addresses( ).  
    MESSAGE 'Retrieved information about Elastic IP addresses.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeAddresses](#) referência da API AWS SDK for SAP ABAP.

## DescribeAvailabilityZones

O código de exemplo a seguir mostra como usar DescribeAvailabilityZones.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ec2->describeavailabilityzones( ). "  
oo_result is returned for testing purposes. "  
    DATA(lt_zones) = oo_result->get_availabilityzones( ).
```

```

    MESSAGE 'Retrieved information about Availability Zones.' TYPE 'I'.

    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
      DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
      MESSAGE lv_error TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeAvailabilityZones](#) referência da API AWS SDK for SAP ABAP.

## DescribeImages

O código de exemplo a seguir mostra como usar DescribeImages.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

    TRY.
      oo_result = lo_ec2->describeimages( it_imageids = it_image_ids ).
      " oo_result is returned for testing purposes. "
      DATA(lt_images) = oo_result->get_images( ).
      MESSAGE 'Retrieved information about Amazon Machine Images (AMIs).' TYPE
'I'.
      CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
      ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeImages](#) referência da API AWS SDK for SAP ABAP.

## DescribeInstanceTypes

O código de exemplo a seguir mostra como usar DescribeInstanceTypes.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" Create filters for architecture and instance type patterns
" iv_architecture = 'x86_64'
DATA lt_filters TYPE /aws1/cl_ec2filter=>tt_filterlist.
APPEND NEW /aws1/cl_ec2filter(
  iv_name = 'processor-info.supported-architecture'
  it_values = VALUE /aws1/cl_ec2valuestringlist_w=>tt_valuestringlist(
    ( NEW /aws1/cl_ec2valuestringlist_w( iv_architecture ) )
  )
) TO lt_filters.
" Filter for instance type patterns like '*.micro', '*.small'
APPEND NEW /aws1/cl_ec2filter(
  iv_name = 'instance-type'
  it_values = VALUE /aws1/cl_ec2valuestringlist_w=>tt_valuestringlist(
    ( NEW /aws1/cl_ec2valuestringlist_w( '*.micro' ) )
    ( NEW /aws1/cl_ec2valuestringlist_w( '*.small' ) )
  )
) TO lt_filters.

TRY.
  oo_result = lo_ec2->describeinstancetype( it_filters = lt_filters ).
  " oo_result is returned for testing purposes. "
  DATA(lt_instance_types) = oo_result->get_instancetypes( ).
  MESSAGE 'Retrieved information about EC2 instance types.' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeInstanceTypes](#) referência da API AWS SDK for SAP ABAP.

## DescribeInstances

O código de exemplo a seguir mostra como usar DescribeInstances.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_ec2->describeinstances( ).
    is returned for testing purposes. " oo_result

    " Retrieving details of EC2 instances. "
    DATA: lv_instance_id    TYPE /aws1/ec2string,
           lv_status         TYPE /aws1/ec2instancename,
           lv_instance_type  TYPE /aws1/ec2instancetype,
           lv_image_id       TYPE /aws1/ec2string.
    LOOP AT oo_result->get_reservations( ) INTO DATA(lo_reservation).
        LOOP AT lo_reservation->get_instances( ) INTO DATA(lo_instance).
            lv_instance_id = lo_instance->get_instanceid( ).
            lv_status = lo_instance->get_state( )->get_name( ).
            lv_instance_type = lo_instance->get_instancetype( ).
            lv_image_id = lo_instance->get_imageid( ).
        ENDLLOOP.
    ENDLLOOP.
    MESSAGE 'Retrieved information about EC2 instances.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeInstances](#) referência da API AWS SDK for SAP ABAP.

## DescribeKeyPairs

O código de exemplo a seguir mostra como usar DescribeKeyPairs.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ec2->describekeypairs( ).                " oo_result  
is returned for testing purposes. "  
    DATA(lt_key_pairs) = oo_result->get_keypairs( ).  
    MESSAGE 'Retrieved information about key pairs.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.   
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeKeyPairs](#) referência da API AWS SDK for SAP ABAP.

## DescribeRegions

O código de exemplo a seguir mostra como usar DescribeRegions.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_ec2->describeregions( ).
    " oo_result
is returned for testing purposes. "
    DATA(lt_regions) = oo_result->get_regions( ).
    MESSAGE 'Retrieved information about Regions.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeRegions](#) referência da API AWS SDK for SAP ABAP.

**DescribeRouteTables**

O código de exemplo a seguir mostra como usar DescribeRouteTables.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" Create filter for VPC ID
" iv_vpc_id = 'vpc-abc123'
DATA lt_filters TYPE /aws1/cl_ec2filter=>tt_filterlist.
APPEND NEW /aws1/cl_ec2filter(

```

```

    iv_name = 'vpc-id'
    it_values = VALUE /aws1/cl_ec2valuestringlist_w=>tt_valuestringlist(
      ( NEW /aws1/cl_ec2valuestringlist_w( iv_vpc_id ) )
    )
  ) TO lt_filters.

  TRY.
    oo_result = lo_ec2->describeroutetables( it_filters = lt_filters ).
    " oo_result is returned for testing purposes. "
    DATA(lt_route_tables) = oo_result->get_routetables( ).
    MESSAGE 'Retrieved information about route tables.' TYPE 'I'.
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeRouteTables](#) referência da API AWS SDK for SAP ABAP.

## DescribeSecurityGroups

O código de exemplo a seguir mostra como usar DescribeSecurityGroups.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

  TRY.
    DATA lt_group_ids TYPE /aws1/cl_ec2groupidstrlist_w=>tt_groupidstringlist.
    APPEND NEW /aws1/cl_ec2groupidstrlist_w( iv_value = iv_group_id ) TO
lt_group_ids.
    oo_result = lo_ec2->describesecuritygroups( it_groupids = lt_group_ids ).
    " oo_result is returned for testing purposes. "
    DATA(lt_security_groups) = oo_result->get_securitygroups( ).
    MESSAGE 'Retrieved information about security groups.' TYPE 'I'.
  TRY.

```

```

CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeSecurityGroups](#) referência da API AWS SDK for SAP ABAP.

## MonitorInstances

O código de exemplo a seguir mostra como usar MonitorInstances.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA lt_instance_ids TYPE /aws1/
cl_ec2instidstringlist_w=>tt_instanceidstringlist.
  APPEND NEW /aws1/cl_ec2instidstringlist_w( iv_value = iv_instance_id ) TO
lt_instance_ids.

"Perform dry run"
TRY.
  " DryRun is set to true. This checks for the required permissions to monitor
the instance without actually making the request. "
  lo_ec2->monitorinstances(
    it_instanceids = lt_instance_ids
    iv_dryrun = abap_true ).
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  " If the error code returned is `DryRunOperation`, then you have the
required permissions to monitor this instance. "
  IF lo_exception->av_err_code = 'DryRunOperation'.
    MESSAGE 'Dry run to enable detailed monitoring completed.' TYPE 'I'.
  " DryRun is set to false to enable detailed monitoring. "

```

```

    lo_ec2->monitorinstances(
        it_instanceids = lt_instance_ids
        iv_dryrun = abap_false ).
    MESSAGE 'Detailed monitoring enabled.' TYPE 'I'.
    " If the error code returned is `UnauthorizedOperation`, then you don't
    have the required permissions to monitor this instance. "
    ELSEIF lo_exception->av_err_code = 'UnauthorizedOperation'.
        MESSAGE 'Dry run to enable detailed monitoring failed. User does not have
        the permissions to monitor the instance.' TYPE 'E'.
    ELSE.
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
        >av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
    ENDIF.
ENDTRY.
ENDTRY.

```

- Para obter detalhes da API, consulte a [MonitorInstances](#) referência da API AWS SDK for SAP ABAP.

## RebootInstances

O código de exemplo a seguir mostra como usar RebootInstances.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

    DATA lt_instance_ids TYPE /aws1/
    cl_ec2instidstringlist_w=>tt_instanceidstringlist.
    APPEND NEW /aws1/cl_ec2instidstringlist_w( iv_value = iv_instance_id ) TO
    lt_instance_ids.

    "Perform dry run"
    TRY.
        " DryRun is set to true. This checks for the required permissions to reboot
        the instance without actually making the request. "

```

```

lo_ec2->rebootinstances(
  it_instanceids = lt_instance_ids
  iv_dryrun = abap_true ).
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
" If the error code returned is `DryRunOperation`, then you have the
required permissions to reboot this instance. "
IF lo_exception->av_err_code = 'DryRunOperation'.
  MESSAGE 'Dry run to reboot instance completed.' TYPE 'I'.
  " DryRun is set to false to make a reboot request. "
  lo_ec2->rebootinstances(
    it_instanceids = lt_instance_ids
    iv_dryrun = abap_false ).
  MESSAGE 'Instance rebooted.' TYPE 'I'.
  " If the error code returned is `UnauthorizedOperation`, then you don't
have the required permissions to reboot this instance. "
  ELSEIF lo_exception->av_err_code = 'UnauthorizedOperation'.
    MESSAGE 'Dry run to reboot instance failed. User does not have permissions
to reboot the instance.' TYPE 'E'.
  ELSE.
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
  ENDIF.
ENDTRY.

```

- Para obter detalhes da API, consulte a [RebootInstances](#) referência da API AWS SDK for SAP ABAP.

## ReleaseAddress

O código de exemplo a seguir mostra como usar ReleaseAddress.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  lo_ec2->releaseaddress( iv_allocationid = iv_allocation_id ).
  MESSAGE 'Elastic IP address released.' TYPE 'I'.
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ReleaseAddress](#) referência da API AWS SDK for SAP ABAP.

## RunInstances

O código de exemplo a seguir mostra como usar RunInstances.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" Create tags for resource created during instance launch. "
DATA lt_tagsspecifications TYPE /aws1/
cl_ec2tagsspecification=>tt_tagsspecificationlist.
DATA ls_tagsspecifications LIKE LINE OF lt_tagsspecifications.
ls_tagsspecifications = NEW /aws1/cl_ec2tagsspecification(
  iv_resourcetype = 'instance'
  it_tags = VALUE /aws1/cl_ec2tag=>tt_taglist(
    ( NEW /aws1/cl_ec2tag( iv_key = 'Name' iv_value = iv_tag_value ) )
  ) ).
APPEND ls_tagsspecifications TO lt_tagsspecifications.

TRY.
  " Create/launch Amazon Elastic Compute Cloud (Amazon EC2) instance. "
  oo_result = lo_ec2->runinstances( " oo_result is
returned for testing purposes. "

```

```

        iv_imageid = iv_ami_id
        iv_instancetype = 't3.micro'
        iv_maxcount = 1
        iv_mincount = 1
        it_tagspecifications = lt_tagspecifications
        iv_subnetid = iv_subnet_id ).
    MESSAGE 'EC2 instance created.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [RunInstances](#) referência da API AWS SDK for SAP ABAP.

## StartInstances

O código de exemplo a seguir mostra como usar StartInstances.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

    DATA lt_instance_ids TYPE /aws1/
cl_ec2instidstringlist_w=>tt_instanceidstringlist.
    APPEND NEW /aws1/cl_ec2instidstringlist_w( iv_value = iv_instance_id ) TO
lt_instance_ids.

    "Perform dry run"
    TRY.
        " DryRun is set to true. This checks for the required permissions to start
the instance without actually making the request. "
        lo_ec2->startinstances(
            it_instanceids = lt_instance_ids

```

```

        iv_dryrun = abap_true ).
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    " If the error code returned is `DryRunOperation`, then you have the
    required permissions to start this instance. "
    IF lo_exception->av_err_code = 'DryRunOperation'.
        MESSAGE 'Dry run to start instance completed.' TYPE 'I'.
        " DryRun is set to false to start instance. "
        oo_result = lo_ec2->startinstances(          " oo_result is returned for
testing purposes. "
        it_instanceids = lt_instance_ids
        iv_dryrun = abap_false ).
        MESSAGE 'Successfully started the EC2 instance.' TYPE 'I'.
        " If the error code returned is `UnauthorizedOperation`, then you don't
        have the required permissions to start this instance. "
        ELSEIF lo_exception->av_err_code = 'UnauthorizedOperation'.
            MESSAGE 'Dry run to start instance failed. User does not have permissions
to start the instance.' TYPE 'E'.
        ELSE.
            DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
            MESSAGE lv_error TYPE 'E'.
        ENDIF.
    ENDTRY.
ENDTRY.

```

- Para obter detalhes da API, consulte a [StartInstances](#) referência da API AWS SDK for SAP ABAP.

## StopInstances

O código de exemplo a seguir mostra como usar StopInstances.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA lt_instance_ids TYPE /aws1/
cl_ec2instidstringlist_w=>tt_instanceidstringlist.
  APPEND NEW /aws1/cl_ec2instidstringlist_w( iv_value = iv_instance_id ) TO
  lt_instance_ids.

  "Perform dry run"
  TRY.
    " DryRun is set to true. This checks for the required permissions to stop
    the instance without actually making the request. "
    lo_ec2->stopinstances(
      it_instanceids = lt_instance_ids
      iv_dryrun = abap_true ).
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    " If the error code returned is `DryRunOperation`, then you have the
    required permissions to stop this instance. "
    IF lo_exception->av_err_code = 'DryRunOperation'.
      MESSAGE 'Dry run to stop instance completed.' TYPE 'I'.
      " DryRun is set to false to stop instance. "
      oo_result = lo_ec2->stopinstances( " oo_result is returned for
testing purposes. "
      it_instanceids = lt_instance_ids
      iv_dryrun = abap_false ).
      MESSAGE 'Successfully stopped the EC2 instance.' TYPE 'I'.
      " If the error code returned is `UnauthorizedOperation`, then you don't
      have the required permissions to stop this instance. "
      ELSEIF lo_exception->av_err_code = 'UnauthorizedOperation'.
        MESSAGE 'Dry run to stop instance failed. User does not have permissions
to stop the instance.' TYPE 'E'.
      ELSE.
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
        MESSAGE lv_error TYPE 'E'.
      ENDIF.
    ENDTRY.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [StopInstances](#) referência da API AWS SDK for SAP ABAP.

# Exemplos do Amazon ECR usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Amazon ECR.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateRepository

O código de exemplo a seguir mostra como usar CreateRepository.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_repository_name = 'my-repository'
  oo_result = lo_ecr->createrepository(
    iv_repositoryname = iv_repository_name ).
  DATA(lv_repository_uri) = oo_result->get_repository( )-
>get_repositoryuri( ).
  MESSAGE |Repository created with URI: { lv_repository_uri }| TYPE 'I'.
CATCH /aws1/cx_ecrrepositoryalrexex.
  " If repository already exists, retrieve it
  DATA lt_repo_names TYPE /aws1/
cl_ecrrepositorynames00=>tt_repositorynamelist.
```

```

        APPEND NEW /aws1/cl_ecrrepositorynames00( iv_value = iv_repository_name )
    TO lt_repo_names.
    DATA(lo_describe_result) = lo_ecr->describerepositories( it_repositorynames
= lt_repo_names ).
    DATA(lt_repos) = lo_describe_result->get_repositories( ).
    IF lines( lt_repos ) > 0.
        READ TABLE lt_repos INDEX 1 INTO DATA(lo_repo).
        oo_result = NEW /aws1/cl_ecrcrerepositoryrsp( io_repository = lo_repo ).
        MESSAGE |Repository { iv_repository_name } already exists.| TYPE 'I'.
    ENDIF.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateRepository](#) referência da API AWS SDK for SAP ABAP.

## DeleteRepository

O código de exemplo a seguir mostra como usar DeleteRepository.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    " iv_repository_name = 'my-repository'
    lo_ecr->deleterepository(
        iv_repositoryname = iv_repository_name
        iv_force = abap_true ).
    MESSAGE |Repository { iv_repository_name } deleted.| TYPE 'I'.
CATCH /aws1/cx_ecrrepositorynotfound.
    MESSAGE 'Repository not found.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteRepository](#) referência da API AWS SDK for SAP ABAP.

## DescribeImages

O código de exemplo a seguir mostra como usar DescribeImages.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_repository_name = 'my-repository'
  " it_image_ids = VALUE #( ( NEW /aws1/cl_ecrimageidentifier( iv_imagetag =
'latest' ) ) )
  IF it_image_ids IS NOT INITIAL.
    oo_result = lo_ecr->describeimages(
      iv_repositoryname = iv_repository_name
      it_imageids = it_image_ids ).
  ELSE.
    oo_result = lo_ecr->describeimages(
      iv_repositoryname = iv_repository_name ).
  ENDIF.
  DATA(lt_image_details) = oo_result->get_imagedetails( ).
  MESSAGE |Found { lines( lt_image_details ) } images in repository.| TYPE
'I'.
  CATCH /aws1/cx_ecrrepositorynotfoundex.
    MESSAGE 'Repository not found.' TYPE 'I'.
  CATCH /aws1/cx_ecrimagenotfoundex.
    MESSAGE 'Image not found.' TYPE 'I'.
  CATCH /aws1/cx_ecrinvalidparameterex.
    MESSAGE 'Invalid parameter provided.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeImages](#) referência da API AWS SDK for SAP ABAP.

## DescribeRepositories

O código de exemplo a seguir mostra como usar DescribeRepositories.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " it_repository_names = VALUE #( ( NEW /aws1/  
cl_ocrrepositorynames00( iv_value = 'my-repository' ) ) )  
    oo_result = lo_ocr->describerepositories(  
        it_repository_names = it_repository_names ).  
    DATA(lt_repositories) = oo_result->get_repositories( ).  
    MESSAGE |Found { lines( lt_repositories ) } repositories.| TYPE 'I'.  
CATCH /aws1/cx_ocrrepositorynotfound.  
    MESSAGE 'Repository not found.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeRepositories](#) referência da API AWS SDK for SAP ABAP.

## GetAuthorizationToken

O código de exemplo a seguir mostra como usar GetAuthorizationToken.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

oo_result = lo_ecr->getauthorizationtoken( ).
DATA(lt_auth_data) = oo_result->get_authorizationdata( ).
IF lines( lt_auth_data ) > 0.
  READ TABLE lt_auth_data INDEX 1 INTO DATA(lo_auth_data).
  DATA(lv_token) = lo_auth_data->get_authorizationtoken( ).
  MESSAGE 'Authorization token retrieved.' TYPE 'I'.
ENDIF.
CATCH /aws1/cx_ecrserverexception.
  MESSAGE 'Server exception occurred.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetAuthorizationToken](#) referência da API AWS SDK for SAP ABAP.

## GetRepositoryPolicy

O código de exemplo a seguir mostra como usar GetRepositoryPolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_repository_name = 'my-repository'
  oo_result = lo_ecr->getrepositorypolicy(
    iv_repositoryname = iv_repository_name ).
  DATA(lv_policy_text) = oo_result->get_policytext( ).
  MESSAGE 'Repository policy retrieved.' TYPE 'I'.
CATCH /aws1/cx_ecrrepositorynotfound.
  MESSAGE 'Repository not found.' TYPE 'I'.
CATCH /aws1/cx_ecrrepositoryplynot00.
  MESSAGE 'Repository policy not found.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetRepositoryPolicy](#) referência da API AWS SDK for SAP ABAP.

## PutLifecyclePolicy

O código de exemplo a seguir mostra como usar PutLifecyclePolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_repository_name = 'my-repository'
  " iv_lifecycle_policy_text = '{"rules":
[{"rulePriority":1,"description":"Expire images older than 14 days",...}]}'
  lo_ecr->putlifecyclepolicy(
    iv_repositoryname = iv_repository_name
    iv_lifecyclepolicytext = iv_lifecycle_policy_text ).
  MESSAGE |Lifecycle policy set for repository { iv_repository_name }.| TYPE
'I'.
  CATCH /aws1/cx_ecrrepositorynotfoundex.
    MESSAGE 'Repository not found.' TYPE 'I'.
  CATCH /aws1/cx_ecrvalidationex.
    MESSAGE 'Invalid lifecycle policy format.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutLifecyclePolicy](#) referência da API AWS SDK for SAP ABAP.

## SetRepositoryPolicy

O código de exemplo a seguir mostra como usar SetRepositoryPolicy.

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_repository_name = 'my-repository'
  " iv_policy_text = '{"Version":"2012-10-17",          "Statement":[...]}'
  lo_ecr->setrepositorypolicy(
    iv_repositoryname = iv_repository_name
    iv_policytext = iv_policy_text ).
  MESSAGE |Policy set for repository { iv_repository_name }.| TYPE 'I'.
CATCH /aws1/cx_ecrrepositorynotfound.
  MESSAGE 'Repository not found.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [SetRepositoryPolicy](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon EMR usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon EMR.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

# Ações

## AddJobFlowSteps

O código de exemplo a seguir mostra como usar AddJobFlowSteps.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Build args list for Spark submit
  DATA lt_args TYPE /aws1/cl_emrxmlstringlist_w=>tt_xmlstringlist.
  APPEND NEW /aws1/cl_emrxmlstringlist_w( 'spark-submit' ) TO lt_args.
  APPEND NEW /aws1/cl_emrxmlstringlist_w( '--deploy-mode' ) TO lt_args.
  APPEND NEW /aws1/cl_emrxmlstringlist_w( 'cluster' ) TO lt_args.
  APPEND NEW /aws1/cl_emrxmlstringlist_w( iv_script_uri ) TO lt_args.
  APPEND LINES OF it_script_args TO lt_args.

  " Create step configuration
  DATA(lo_hadoop_jar_step) = NEW /aws1/cl_emrhadoopjarstepcfg(
    iv_jar = 'command-runner.jar'
    it_args = lt_args
  ).

  DATA(lo_step_config) = NEW /aws1/cl_emrstepconfig(
    iv_name = iv_name
    iv_actiononfailure = 'CONTINUE'
    io_hadoopjarstep = lo_hadoop_jar_step
  ).

  DATA lt_steps TYPE /aws1/cl_emrstepconfig=>tt_stepconfiglist.
  APPEND lo_step_config TO lt_steps.

  DATA(lo_result) = lo_emr->addjobflowsteps(
    iv_jobflowid = iv_cluster_id
    it_steps = lt_steps
  ).
```

```

" Get first step ID
DATA(lt_step_ids) = lo_result->get_stepids( ).
READ TABLE lt_step_ids INDEX 1 INTO DATA(lo_step_id_obj).
IF sy-subrc = 0.
    ov_step_id = lo_step_id_obj->get_value( ).
    MESSAGE |Step added with ID { ov_step_id }| TYPE 'I'.
ENDIF.
CATCH /aws1/cx_emrinternalservererr INTO DATA(lo_internal_error).
DATA(lv_error) = lo_internal_error->if_message~get_text( ).
MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [AddJobFlowSteps](#) referência da API AWS SDK for SAP ABAP.

## DescribeCluster

O código de exemplo a seguir mostra como usar DescribeCluster.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_emr->describecluster(
        iv_clusterid = iv_cluster_id
    ).
    DATA(lo_cluster) = oo_result->get_cluster( ).
    DATA(lv_cluster_name) = lo_cluster->get_name( ).
    MESSAGE |Retrieved cluster information for { lv_cluster_name }| TYPE 'I'.
CATCH /aws1/cx_emrinternalserverex INTO DATA(lo_internal_error).
    DATA(lv_error) = lo_internal_error->if_message~get_text( ).
    MESSAGE lv_error TYPE 'E'.
CATCH /aws1/cx_emrinvalidrequestex INTO DATA(lo_invalid_error).
    lv_error = lo_invalid_error->if_message~get_text( ).

```

```
MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeCluster](#) referência da API AWS SDK for SAP ABAP.

## DescribeStep

O código de exemplo a seguir mostra como usar DescribeStep.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_emr->describestep(  
        iv_clusterid = iv_cluster_id  
        iv_stepid = iv_step_id  
    ).  
    DATA(lo_step) = oo_result->get_step( ).  
    DATA(lv_step_name) = lo_step->get_name( ).  
    MESSAGE |Retrieved step information for { lv_step_name }| TYPE 'I'.  
CATCH /aws1/cx_emrinternalserverex INTO DATA(lo_internal_error).  
    DATA(lv_error) = lo_internal_error->if_message~get_text( ).  
    MESSAGE lv_error TYPE 'E'.  
CATCH /aws1/cx_emrinvalidrequestex INTO DATA(lo_invalid_error).  
    lv_error = lo_invalid_error->if_message~get_text( ).  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeStep](#) referência da API AWS SDK for SAP ABAP.

## ListSteps

O código de exemplo a seguir mostra como usar ListSteps.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_emr->liststeps(  
        iv_clusterid = iv_cluster_id  
    ).  
    DATA(lt_steps) = oo_result->get_steps( ).  
    DATA(lv_step_count) = lines( lt_steps ).  
    MESSAGE |Retrieved { lv_step_count } steps for cluster| TYPE 'I'.  
    CATCH /aws1/cx_emrinternalserverex INTO DATA(lo_internal_error).  
        DATA(lv_error) = lo_internal_error->if_message~get_text( ).  
        MESSAGE lv_error TYPE 'E'.  
    CATCH /aws1/cx_emrinvalidrequestex INTO DATA(lo_invalid_error).  
        lv_error = lo_invalid_error->if_message~get_text( ).  
        MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListSteps](#) referência da API AWS SDK for SAP ABAP.

## RunJobFlow

O código de exemplo a seguir mostra como usar RunJobFlow.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create instances configuration
  DATA(lo_instances) = NEW /aws1/cl_emrjobflowinstsconfig(
    iv_masterinstancetype = 'm5.xlarge'
    iv_slaveinstancetype = 'm5.xlarge'
    iv_instancecount = 3
    iv_keepjobflowalivewhennos00 = iv_keep_alive
    iv_emrmanagedmastersecgroup = iv_primary_sec_grp
    iv_emrmanagedslavesecgroup = iv_secondary_sec_grp
  ).

  DATA(lo_result) = lo_emr->runjobflow(
    iv_name = iv_name
    iv_loguri = iv_log_uri
    iv_releaselabel = 'emr-5.30.1'
    io_instances = lo_instances
    it_steps = it_steps
    it_applications = it_applications
    iv_jobflowrole = iv_job_flow_role
    iv_servicerole = iv_service_role
    iv_ebsrootvolumesize = 10
    iv_visibletoallusers = abap_true
  ).


  ov_cluster_id = lo_result->get_jobflowid( ).
  MESSAGE 'EMR cluster created successfully.' TYPE 'I'.
  CATCH /aws1/cx_emrinternalservererr INTO DATA(lo_internal_error).
  DATA(lv_error) = lo_internal_error->if_message~get_text( ).
  MESSAGE lv_error TYPE 'E'.
  CATCH /aws1/cx_emrclientexc INTO DATA(lo_client_error).
  lv_error = lo_client_error->if_message~get_text( ).
  MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [RunJobFlow](#) referência da API AWS SDK for SAP ABAP.

## TerminateJobFlows

O código de exemplo a seguir mostra como usar TerminateJobFlows.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  DATA lt_cluster_ids TYPE /aws1/cl_emrxmlstringlist_w=>tt_xmlstringlist.  
  APPEND NEW /aws1/cl_emrxmlstringlist_w( iv_cluster_id ) TO lt_cluster_ids.  
  
  lo_emr->terminatejobflows(  
    it_jobflowids = lt_cluster_ids  
  ).  
  MESSAGE 'EMR cluster terminated successfully.' TYPE 'I'.  
CATCH /aws1/cx_emrinternalservererr INTO DATA(lo_internal_error).  
  DATA(lv_error) = lo_internal_error->if_message~get_text( ).  
  MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [TerminateJobFlows](#) referência da API AWS SDK for SAP ABAP.

## EventBridge Exemplos de agendador usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com EventBridge Scheduler.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### CreateSchedule

O código de exemplo a seguir mostra como usar CreateSchedule.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Constants for time calculations
  DATA lv_start_date TYPE /aws1/scdstartdate.
  DATA lv_end_date TYPE /aws1/scdenndate.
  DATA lv_start_timestamp TYPE timestamp.
  DATA lv_end_timestamp TYPE timestamp.
  DATA lv_hours_to_run TYPE i VALUE 1.

  " Get current timestamp
  GET TIME STAMP FIELD lv_start_timestamp.

  " Add 1 hour to the current timestamp using CL_ABAP_TSTMP
  lv_end_timestamp = cl_abap_tstmp=>add(
    tstmp = lv_start_timestamp
    secs = lv_hours_to_run * 3600 ).

  " Convert timestamps to decimal format for AWS API
  lv_start_date = lv_start_timestamp.
  lv_end_date = lv_end_timestamp.

  " Prepare flexible time window configuration
  DATA lo_flexible_time_window TYPE REF TO /aws1/cl_scdflexiblewindow.
  IF iv_use_flexible_time_win = abap_true.
    " iv_use_flexible_time_win = ABAP_TRUE
    " Example: Set MaximumWindowInMinutes to 10 for flexible window
    lo_flexible_time_window = NEW /aws1/cl_scdflexiblewindow(
```

```

        iv_mode = 'FLEXIBLE'
        iv_maximumwindowinminutes = 10 ).
ELSE.
    lo_flexible_time_window = NEW /aws1/cl_scdflexibletimewindow(
        iv_mode = 'OFF' ).
ENDIF.

" Prepare target configuration
" Example iv_target_arn = 'arn:aws:sqs:us-east-1:123456789012:my-queue'
" Example iv_role_arn = 'arn:aws:iam::123456789012:role/SchedulerRole'
" Example iv_input = '{"message": "Hello from EventBridge Scheduler"}'
DATA(lo_target) = NEW /aws1/cl_scdtarget(
    iv_arn = iv_target_arn
    iv_rolearn = iv_role_arn
    iv_input = iv_input ).

" Set action after completion if needed
DATA lv_action_after_completion TYPE /aws1/scdactionaftercompletion.
IF iv_delete_after_completion = abap_true.
    " iv_delete_after_completion = ABAP_TRUE
    lv_action_after_completion = 'DELETE'.
ELSE.
    lv_action_after_completion = 'NONE'.
ENDIF.

" Create the schedule
" Example iv_name = 'my-schedule'
" Example iv_schedule_expression = 'rate(15 minutes)'
" Example iv_schedule_group_name = 'my-schedule-group'
DATA(lo_result) = lo_scd->createschedule(
    iv_name = iv_name
    iv_scheduleexpression = iv_schedule_expression
    iv_groupname = iv_schedule_group_name
    io_target = lo_target
    io_flexibletimewindow = lo_flexible_time_window
    iv_startdate = lv_start_date
    iv_enddate = lv_end_date
    iv_actionaftercompletion = lv_action_after_completion ).

ov_schedule_arn = lo_result->get_scheduledarn( ).
MESSAGE 'Schedule created successfully.' TYPE 'I'.

CATCH /aws1/cx_scdconflictexception INTO DATA(lo_conflict_ex).

```

```

        DATA(lv_error) = |Conflict creating schedule: { lo_conflict_ex-
>if_message~get_text( ) }|.
        MESSAGE lv_error TYPE 'I'.
        CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_ex).
        DATA(lv_generic_error) = |Error creating schedule: { lo_generic_ex-
>if_message~get_text( ) }|.
        MESSAGE lv_generic_error TYPE 'I'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateSchedule](#) referência da API AWS SDK for SAP ABAP.

## CreateScheduleGroup

O código de exemplo a seguir mostra como usar CreateScheduleGroup.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

    TRY.
        " Example iv_name = 'my-schedule-group'
        DATA(lo_result) = lo_scd->createschedulegroup(
            iv_name = iv_name ).

        ov_schedule_group_arn = lo_result->get_schedulegrouparn( ).
        MESSAGE 'Schedule group created successfully.' TYPE 'I'.

        CATCH /aws1/cx_scdconflictexception INTO DATA(lo_conflict_ex).
        DATA(lv_error) = |Conflict creating schedule group: { lo_conflict_ex-
>if_message~get_text( ) }|.
        MESSAGE lv_error TYPE 'I'.
        CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_ex).
        DATA(lv_generic_error) = |Error creating schedule group: { lo_generic_ex-
>if_message~get_text( ) }|.
        MESSAGE lv_generic_error TYPE 'I'.
    
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateScheduleGroup](#) referência da API AWS SDK for SAP ABAP.

## DeleteSchedule

O código de exemplo a seguir mostra como usar DeleteSchedule.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Example iv_name = 'my-schedule'  
  " Example iv_schedule_group_name = 'my-schedule-group'  
  lo_scd->deleteschedule(  
    iv_name = iv_name  
    iv_groupname = iv_schedule_group_name ).  
  MESSAGE 'Schedule deleted successfully.' TYPE 'I'.  
  
  CATCH /aws1/cx_scdresourcenotfoundex INTO DATA(lo_not_found_ex).  
    DATA(lv_error) = |Schedule not found: { lo_not_found_ex-  
>if_message~get_text( ) }|.  
    MESSAGE lv_error TYPE 'I'.  
  CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_ex).  
    DATA(lv_generic_error) = |Error deleting schedule: { lo_generic_ex-  
>if_message~get_text( ) }|.  
    MESSAGE lv_generic_error TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteSchedule](#) referência da API AWS SDK for SAP ABAP.

## DeleteScheduleGroup

O código de exemplo a seguir mostra como usar DeleteScheduleGroup.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Example iv_name = 'my-schedule-group'
  lo_scd->deleteschedulegroup(
    iv_name = iv_name ).
  MESSAGE 'Schedule group deleted successfully.' TYPE 'I'.

  CATCH /aws1/cx_scdresourcenotfoundex INTO DATA(lo_not_found_ex).
  DATA(lv_error) = |Schedule group not found: { lo_not_found_ex-
>if_message~get_text( ) }|.
  MESSAGE lv_error TYPE 'I'.
  CATCH /aws1/cx_rt_generic INTO DATA(lo_generic_ex).
  DATA(lv_generic_error) = |Error deleting schedule group: { lo_generic_ex-
>if_message~get_text( ) }|.
  MESSAGE lv_generic_error TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteScheduleGroup](#) referência da API AWS SDK for SAP ABAP.

## AWS Glue exemplos usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. AWS Glue

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Ações](#)

## Ações

### CreateCrawler

O código de exemplo a seguir mostra como usar CreateCrawler.

#### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_crawler_name = 'my-crawler'
  " iv_role_arn = 'arn:aws:iam::123456789012:role/AWSGlueServiceRole-Test'
  " iv_database_name = 'my-database'
  " iv_table_prefix = 'test_'
  " iv_s3_target = 's3://example-bucket/data/'

  DATA(lt_s3_targets) = VALUE /aws1/cl_glus3target=>tt_s3targetlist(
    ( NEW /aws1/cl_glus3target( iv_path = iv_s3_target ) ) ).

  DATA(lo_targets) = NEW /aws1/cl_gluclawlertargets(
    it_s3targets = lt_s3_targets ).

  lo_glu->createcrawler(
    iv_name = iv_crawler_name
    iv_role = iv_role_arn
    iv_databasename = iv_database_name
    iv_tableprefix = iv_table_prefix
    io_targets = lo_targets ).
  MESSAGE 'Crawler created successfully.' TYPE 'I'.
```

```

CATCH /aws1/cx_glualreadyexistsex.
  MESSAGE 'Crawler already exists.' TYPE 'E'.
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
  MESSAGE lv_invalid_error TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
  MESSAGE lv_timeout_error TYPE 'E'.
CATCH /aws1/cx_gluresrcnumlmtexcdex INTO DATA(lo_limit_ex).
  DATA(lv_limit_error) = lo_limit_ex->if_message~get_longtext( ).
  MESSAGE lv_limit_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateCrawler](#) referência da API AWS SDK for SAP ABAP.

## CreateJob

O código de exemplo a seguir mostra como usar CreateJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_job_name = 'my-etl-job'
  " iv_description = 'ETL job for data transformation'
  " iv_role_arn = 'arn:aws:iam::123456789012:role/AWSGlueServiceRole-Test'
  " iv_script_location = 's3://example-bucket/scripts/my-script.py'

  DATA(lo_command) = NEW /aws1/cl_glujobcommand(
    iv_name = 'glueetl'
    iv_scriptlocation = iv_script_location
    iv_pythonversion = '3' ).

  lo_glu->createjob(

```

```

        iv_name = iv_job_name
        iv_description = iv_description
        iv_role = iv_role_arn
        io_command = lo_command
        iv_glueversion = '3.0' ).
    MESSAGE 'Job created successfully.' TYPE 'I'.
CATCH /aws1/cx_glualreadyexistsex.
    MESSAGE 'Job already exists.' TYPE 'E'.
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).
    DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
    MESSAGE lv_invalid_error TYPE 'E'.
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).
    DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).
    MESSAGE lv_internal_error TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
    DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
    MESSAGE lv_timeout_error TYPE 'E'.
CATCH /aws1/cx_gluressrcnumlmtexcdex INTO DATA(lo_limit_ex).
    DATA(lv_limit_error) = lo_limit_ex->if_message~get_longtext( ).
    MESSAGE lv_limit_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateJob](#) referência da API AWS SDK for SAP ABAP.

## DeleteCrawler

O código de exemplo a seguir mostra como usar DeleteCrawler.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    " iv_crawler_name = 'my-crawler'
    lo_glu->deletecrawler( iv_name = iv_crawler_name ).
    MESSAGE 'Crawler deleted successfully.' TYPE 'I'.

```

```

CATCH /aws1/cx_gluerrorrunningex.
  MESSAGE 'Crawler is currently running.' TYPE 'E'.
CATCH /aws1/cx_gluentitynotfoundex.
  MESSAGE 'Crawler does not exist.' TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
  MESSAGE lv_timeout_error TYPE 'E'.
CATCH /aws1/cx_gluschedrtrtransingex.
  MESSAGE 'Scheduler is transitioning.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteCrawler](#) referência da API AWS SDK for SAP ABAP.

## DeleteDatabase

O código de exemplo a seguir mostra como usar DeleteDatabase.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_database_name = 'my-database'
  lo_glu->deletedatabase( iv_name = iv_database_name ).
  MESSAGE 'Database deleted successfully.' TYPE 'I'.
CATCH /aws1/cx_gluentitynotfoundex.
  MESSAGE 'Database does not exist.' TYPE 'E'.
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
  MESSAGE lv_invalid_error TYPE 'E'.
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).
  DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).
  MESSAGE lv_internal_error TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).

```

```
MESSAGE lv_timeout_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteDatabase](#) referência da API AWS SDK for SAP ABAP.

## DeleteJob

O código de exemplo a seguir mostra como usar DeleteJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " iv_job_name = 'my-etl-job'  
  lo_glu->deletejob( iv_jobname = iv_job_name ).  
  MESSAGE 'Job deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).  
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).  
  MESSAGE lv_invalid_error TYPE 'E'.  
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).  
  DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).  
  MESSAGE lv_internal_error TYPE 'E'.  
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).  
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).  
  MESSAGE lv_timeout_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteJob](#) referência da API AWS SDK for SAP ABAP.

## DeleteTable

O código de exemplo a seguir mostra como usar DeleteTable.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " iv_database_name = 'my-database'  
  " iv_table_name = 'my-table'  
  lo_glu->deletetable(  
    iv_databasename = iv_database_name  
    iv_name = iv_table_name ).  
  MESSAGE 'Table deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_gluentitynotfoundex.  
  MESSAGE 'Table or database does not exist.' TYPE 'E'.  
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).  
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).  
  MESSAGE lv_invalid_error TYPE 'E'.  
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).  
  DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).  
  MESSAGE lv_internal_error TYPE 'E'.  
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).  
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).  
  MESSAGE lv_timeout_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteTable](#) referência da API AWS SDK for SAP ABAP.

## GetCrawler

O código de exemplo a seguir mostra como usar GetCrawler.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " iv_crawler_name = 'my-crawler'  
  oo_result = lo_glu->getcrawler( iv_name = iv_crawler_name ).  
  DATA(lo_crawler) = oo_result->get_crawler( ).  
  MESSAGE 'Crawler information retrieved.' TYPE 'I'.  
CATCH /aws1/cx_gluentitynotfoundex.  
  MESSAGE 'Crawler does not exist.' TYPE 'I'.  
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).  
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).  
  MESSAGE lv_timeout_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetCrawler](#) referência da API AWS SDK for SAP ABAP.

## GetDatabase

O código de exemplo a seguir mostra como usar GetDatabase.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " iv_database_name = 'my-database'  
  oo_result = lo_glu->getdatabase( iv_name = iv_database_name ).  
  DATA(lo_database) = oo_result->get_database( ).
```

```

    MESSAGE 'Database information retrieved.' TYPE 'I'.
  CATCH /aws1/cx_gluentitynotfoundex.
    MESSAGE 'Database does not exist.' TYPE 'E'.
  CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).
    DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
    MESSAGE lv_invalid_error TYPE 'E'.
  CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).
    DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).
    MESSAGE lv_internal_error TYPE 'E'.
  CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
    DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
    MESSAGE lv_timeout_error TYPE 'E'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [GetDatabase](#) referência da API AWS SDK for SAP ABAP.

## GetJobRun

O código de exemplo a seguir mostra como usar GetJobRun.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_job_name = 'my-etl-job'
  " iv_run_id = 'jr_abcd1234567890abcdef1234567890abcdef12345678'
  oo_result = lo_glu->getjobrun(
    iv_jobname = iv_job_name
    iv_runid = iv_run_id ).
  DATA(lo_job_run) = oo_result->get_jobrun( ).
  MESSAGE 'Job run information retrieved.' TYPE 'I'.
  CATCH /aws1/cx_gluentitynotfoundex.
    MESSAGE 'Job or job run does not exist.' TYPE 'E'.
  CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).

```

```

DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
MESSAGE lv_invalid_error TYPE 'E'.
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).
DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).
MESSAGE lv_internal_error TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
MESSAGE lv_timeout_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetJobRun](#) referência da API AWS SDK for SAP ABAP.

## GetJobRuns

O código de exemplo a seguir mostra como usar GetJobRuns.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_job_name = 'my-etl-job'
  oo_result = lo_glu->getjobruns( iv_jobname = iv_job_name ).
  DATA(lt_job_runs) = oo_result->get_jobruns( ).
  MESSAGE 'Job runs retrieved successfully.' TYPE 'I'.
CATCH /aws1/cx_gluentitynotfoundex.
  MESSAGE 'Job does not exist.' TYPE 'E'.
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
  MESSAGE lv_invalid_error TYPE 'E'.
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).
  DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).
  MESSAGE lv_internal_error TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
  MESSAGE lv_timeout_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetJobRuns](#) referência da API AWS SDK for SAP ABAP.

## GetTables

O código de exemplo a seguir mostra como usar GetTables.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " iv_database_name = 'my-database'  
  oo_result = lo_glu->gettables( iv_databasename = iv_database_name ).  
  DATA(lt_tables) = oo_result->get_tablelist( ).  
  MESSAGE 'Tables retrieved successfully.' TYPE 'I'.  
CATCH /aws1/cx_gluentitynotfoundex.  
  MESSAGE 'Database does not exist.' TYPE 'E'.  
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).  
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).  
  MESSAGE lv_invalid_error TYPE 'E'.  
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).  
  DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).  
  MESSAGE lv_internal_error TYPE 'E'.  
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).  
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).  
  MESSAGE lv_timeout_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetTables](#) referência da API AWS SDK for SAP ABAP.

## ListJobs

O código de exemplo a seguir mostra como usar ListJobs.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_glu->listjobs( ).  
    DATA(lt_job_names) = oo_result->get_jobnames( ).  
    MESSAGE 'Job list retrieved successfully.' TYPE 'I'.  
CATCH /aws1/cx_gluentitynotfoundex.  
    MESSAGE 'No jobs found.' TYPE 'I'.  
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).  
    DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).  
    MESSAGE lv_invalid_error TYPE 'E'.  
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).  
    DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).  
    MESSAGE lv_internal_error TYPE 'E'.  
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).  
    DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).  
    MESSAGE lv_timeout_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListJobs](#) referência da API AWS SDK for SAP ABAP.

## StartCrawler

O código de exemplo a seguir mostra como usar StartCrawler.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_crawler_name = 'my-crawler'
  lo_glu->startcrawler( iv_name = iv_crawler_name ).
  MESSAGE 'Crawler started successfully.' TYPE 'I'.
CATCH /aws1/cx_glucrawlerrunningex.
  MESSAGE 'Crawler is already running.' TYPE 'I'.
CATCH /aws1/cx_gluentitynotfoundex.
  MESSAGE 'Crawler does not exist.' TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
  MESSAGE lv_timeout_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [StartCrawler](#) referência da API AWS SDK for SAP ABAP.

## StartJobRun

O código de exemplo a seguir mostra como usar StartJobRun.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_job_name = 'my-etl-job'
  " iv_input_database = 'my-database'
  " iv_input_table = 'my-table'
  " iv_output_bucket_url = 's3://example-output-bucket/'

  DATA lt_arguments TYPE /aws1/cl_glugenericmap_w=>tt_genericmap.
  lt_arguments = VALUE #(
    ( VALUE /aws1/cl_glugenericmap_w=>ts_genericmap_maprow(
      key = '--input_database'
      value = NEW /aws1/cl_glugenericmap_w( iv_value = iv_input_database ) ) )
    ( VALUE /aws1/cl_glugenericmap_w=>ts_genericmap_maprow(
      key = '--input_table'

```

```

        value = NEW /aws1/cl_glugenericmap_w( iv_value = iv_input_table ) ) )
    ( VALUE /aws1/cl_glugenericmap_w=>ts_genericmap_maprow(
      key = '--output_bucket_url'
      value = NEW /aws1/cl_glugenericmap_w( iv_value =
iv_output_bucket_url ) ) ) ).

DATA(oo_result) = lo_glu->startjobrun(
  iv_jobname = iv_job_name
  it_arguments = lt_arguments ).
ov_job_run_id = oo_result->get_jobrunid( ).
MESSAGE 'Job run started successfully.' TYPE 'I'.
CATCH /aws1/cx_gluconcurrentrunsex00.
  MESSAGE 'Maximum concurrent runs exceeded.' TYPE 'E'.
CATCH /aws1/cx_gluentitynotfoundex.
  MESSAGE 'Job does not exist.' TYPE 'E'.
CATCH /aws1/cx_gluinvalidinputex INTO DATA(lo_invalid_ex).
  DATA(lv_invalid_error) = lo_invalid_ex->if_message~get_longtext( ).
  MESSAGE lv_invalid_error TYPE 'E'.
CATCH /aws1/cx_gluinternalserviceex INTO DATA(lo_internal_ex).
  DATA(lv_internal_error) = lo_internal_ex->if_message~get_longtext( ).
  MESSAGE lv_internal_error TYPE 'E'.
CATCH /aws1/cx_gluoperationtimeoutex INTO DATA(lo_timeout_ex).
  DATA(lv_timeout_error) = lo_timeout_ex->if_message~get_longtext( ).
  MESSAGE lv_timeout_error TYPE 'E'.
CATCH /aws1/cx_gluresrcnumlmtexcdex INTO DATA(lo_limit_ex).
  DATA(lv_limit_error) = lo_limit_ex->if_message~get_longtext( ).
  MESSAGE lv_limit_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [StartJobRun](#) referência da API AWS SDK for SAP ABAP.

## HealthImaging exemplos usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. HealthImaging

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Ações](#)

## Ações

### CopyImageSet

O código de exemplo a seguir mostra como usar CopyImageSet.

#### SDK para SAP ABAP

```
TRY.
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'
  " iv_source_image_set_id = '1234567890123456789012345678901234567890'
  " iv_source_version_id = '1'
  " iv_destination_image_set_id =
'12345678901234567890123456789012345678901234567890' (optional)
  " iv_destination_version_id = '1' (optional)
  " iv_force = abap_false
  DATA(lo_source_info) = NEW /aws1/cl_migcpsrcimagesetinf00(
    iv_latestversionid = iv_source_version_id ).
  DATA(lo_copy_info) = NEW /aws1/cl_migcpimagesetinfmtion(
    io_sourceimageset = lo_source_info ).
  IF iv_destination_image_set_id IS NOT INITIAL AND
    iv_destination_version_id IS NOT INITIAL.
    DATA(lo_dest_info) = NEW /aws1/cl_migcopydstimageset(
      iv_imagesetid = iv_destination_image_set_id
      iv_latestversionid = iv_destination_version_id ).
    lo_copy_info = NEW /aws1/cl_migcpimagesetinfmtion(
      io_sourceimageset = lo_source_info
      io_destinationimageset = lo_dest_info ).
  ENDIF.
  oo_result = lo_mig->copyimageset(
    iv_datastoreid = iv_datastore_id
    iv_sourceimagesetid = iv_source_image_set_id
    io_copyimagesetinformation = lo_copy_info
    iv_force = iv_force ).
  DATA(lo_dest_props) = oo_result->get_dstimagesetproperties( ).
  DATA(lv_new_id) = lo_dest_props->get_imagesetid( ).
```

```

    MESSAGE |Image set copied with new ID: { lv_new_id }.| TYPE 'I'.
  CATCH /aws1/cx_migaccessdeniedex.
    MESSAGE 'Access denied.' TYPE 'I'.
  CATCH /aws1/cx_migconflictexception.
    MESSAGE 'Conflict error.' TYPE 'I'.
  CATCH /aws1/cx_miginternalserverex.
    MESSAGE 'Internal server error.' TYPE 'I'.
  CATCH /aws1/cx_migresourcenotfoundex.
    MESSAGE 'Image set not found.' TYPE 'I'.
  CATCH /aws1/cx_migservicequotaexcdex.
    MESSAGE 'Service quota exceeded.' TYPE 'I'.
  CATCH /aws1/cx_migthrottlingex.
    MESSAGE 'Request throttled.' TYPE 'I'.
  CATCH /aws1/cx_migvalidationex.
    MESSAGE 'Validation error.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [CopyImageSet](#) referência da API AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## CreateDatastore

O código de exemplo a seguir mostra como usar CreateDatastore.

SDK para SAP ABAP

```

TRY.
  " iv_datastore_name = 'my-datastore-name'
  oo_result = lo_mig->createdatastore( iv_datastorename = iv_datastore_name ).
  DATA(lv_datastore_id) = oo_result->get_datastoreid( ).
  MESSAGE 'Data store created.' TYPE 'I'.
  CATCH /aws1/cx_migaccessdeniedex.
    MESSAGE 'Access denied.' TYPE 'I'.
  CATCH /aws1/cx_migconflictexception.
    MESSAGE 'Conflict. Data store may already exist.' TYPE 'I'.

```

```

CATCH /aws1/cx_miginternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
CATCH /aws1/cx_migservicequotaexcdex.
  MESSAGE 'Service quota exceeded.' TYPE 'I'.
CATCH /aws1/cx_migthrottlingex.
  MESSAGE 'Request throttled.' TYPE 'I'.
CATCH /aws1/cx_migvalidationex.
  MESSAGE 'Validation error.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateDatastore](#) referência da API AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## DeleteDatastore

O código de exemplo a seguir mostra como usar DeleteDatastore.

### SDK para SAP ABAP

```

TRY.
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'
  oo_result = lo_mig->deletedatastore( iv_datastoreid = iv_datastore_id ).
  MESSAGE 'Data store deleted.' TYPE 'I'.
CATCH /aws1/cx_migaccessdeniedex.
  MESSAGE 'Access denied.' TYPE 'I'.
CATCH /aws1/cx_migconflictexception.
  MESSAGE 'Conflict. Data store may contain resources.' TYPE 'I'.
CATCH /aws1/cx_miginternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
CATCH /aws1/cx_migresourcenotfoundex.
  MESSAGE 'Data store not found.' TYPE 'I'.
CATCH /aws1/cx_migthrottlingex.
  MESSAGE 'Request throttled.' TYPE 'I'.
CATCH /aws1/cx_migvalidationex.
  MESSAGE 'Validation error.' TYPE 'I'.

```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteDatastore](#) referência da API AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## DeleteImageSet

O código de exemplo a seguir mostra como usar DeleteImageSet.

SDK para SAP ABAP

```
TRY.  
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'  
  " iv_image_set_id = '12345678901234567890123456789012345678901234567890'  
  oo_result = lo_mig->deleteimageset(  
    iv_datastoreid = iv_datastore_id  
    iv_imagesetid = iv_image_set_id ).  
  MESSAGE 'Image set deleted.' TYPE 'I'.  
CATCH /aws1/cx_migaccessdeniedex.  
  MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_migconflictexception.  
  MESSAGE 'Conflict error.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
  MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migresourcenotfoundex.  
  MESSAGE 'Image set not found.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
  MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
  MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteImageSet](#) referência da API AWS SDK for SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

**GetDICOMImportJob**

O código de exemplo a seguir mostra como usar GetDICOMImportJob.

## SDK para SAP ABAP

```
TRY.
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'
  " iv_job_id = '12345678901234567890123456789012'
  oo_result = lo_mig->getdicomimportjob(
    iv_datastoreid = iv_datastore_id
    iv_jobid = iv_job_id ).
  DATA(lo_job_props) = oo_result->get_jobproperties( ).
  DATA(lv_job_status) = lo_job_props->get_jobstatus( ).
  MESSAGE |Job status: { lv_job_status }.| TYPE 'I'.
CATCH /aws1/cx_migaccessdeniedex.
  MESSAGE 'Access denied.' TYPE 'I'.
CATCH /aws1/cx_migconflictexception.
  MESSAGE 'Conflict error.' TYPE 'I'.
CATCH /aws1/cx_miginternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
CATCH /aws1/cx_migresourcenotfoundex.
  MESSAGE 'Job not found.' TYPE 'I'.
CATCH /aws1/cx_migthrottlingex.
  MESSAGE 'Request throttled.' TYPE 'I'.
CATCH /aws1/cx_migvalidationex.
  MESSAGE 'Validation error.' TYPE 'I'.
ENDTRY.
```

- Para [obter detalhes da API, consulte a referência da API Get DICOMImport Job](#) in AWS SDK for SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

**GetDatastore**

O código de exemplo a seguir mostra como usar GetDatastore.

**SDK para SAP ABAP**

```
TRY.  
    " iv_datastore_id = '12345678901234567890123456789012345678901234567890'  
    oo_result = lo_mig->getdatastore( iv_datastoreid = iv_datastore_id ).  
    DATA(lo_properties) = oo_result->get_datastoreproperties( ).  
    DATA(lv_name) = lo_properties->get_datastorename( ).  
    DATA(lv_status) = lo_properties->get_datastorestatus( ).  
    MESSAGE 'Data store properties retrieved.' TYPE 'I'.  
CATCH /aws1/cx_migaccessdeniedex.  
    MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
    MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migresourcenotfoundex.  
    MESSAGE 'Data store not found.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
    MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
    MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetDatastore](#) referência da API AWS SDK for SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## GetImageFrame

O código de exemplo a seguir mostra como usar GetImageFrame.

### SDK para SAP ABAP

```
TRY.  
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'  
  " iv_image_set_id = '12345678901234567890123456789012345678901234567890'  
  " iv_image_frame_id = '12345678901234567890123456789012345678901234567890'  
  oo_result = lo_mig->getimageframe(  
    iv_datastoreid = iv_datastore_id  
    iv_imagesetid = iv_image_set_id  
    io_imageframeinformation = NEW /aws1/cl_migimageframeinfmtion(  
      iv_imageframeid = iv_image_frame_id ) ).  
  DATA(lv_frame_blob) = oo_result->get_imageframeblob( ).  
  MESSAGE 'Image frame retrieved.' TYPE 'I'.  
CATCH /aws1/cx_migaccessdeniedex.  
  MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_migconflictexception.  
  MESSAGE 'Conflict error.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
  MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migresourcenotfoundex.  
  MESSAGE 'Image frame not found.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
  MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
  MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetImageFrame](#) referência da API AWS SDK for SAP ABAP.

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## GetImageSet

O código de exemplo a seguir mostra como usar GetImageSet.

### SDK para SAP ABAP

```
TRY.
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'
  " iv_image_set_id = '12345678901234567890123456789012345678901234567890'
  " iv_version_id = '1' (optional)
  IF iv_version_id IS NOT INITIAL.
    oo_result = lo_mig->getimageset(
      iv_datastoreid = iv_datastore_id
      iv_imagesetid = iv_image_set_id
      iv_versionid = iv_version_id ).
  ELSE.
    oo_result = lo_mig->getimageset(
      iv_datastoreid = iv_datastore_id
      iv_imagesetid = iv_image_set_id ).
  ENDIF.
  DATA(lv_state) = oo_result->get_imagesetstate( ).
  MESSAGE |Image set retrieved with state: { lv_state }.| TYPE 'I'.
CATCH /aws1/cx_migaccessdeniedex.
  MESSAGE 'Access denied.' TYPE 'I'.
CATCH /aws1/cx_migconflictexception.
  MESSAGE 'Conflict error.' TYPE 'I'.
CATCH /aws1/cx_miginternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
CATCH /aws1/cx_migresourcenotfoundex.
  MESSAGE 'Image set not found.' TYPE 'I'.
CATCH /aws1/cx_migthrottlingex.
  MESSAGE 'Request throttled.' TYPE 'I'.
CATCH /aws1/cx_migvalidationex.
  MESSAGE 'Validation error.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetImageSet](#) referência da API AWS SDK for SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


## GetImageSetMetadata

O código de exemplo a seguir mostra como usar GetImageSetMetadata.

### SDK para SAP ABAP

```
TRY.
  " iv_datastore_id = '1234567890123456789012345678901234567890'
  " iv_image_set_id = '1234567890123456789012345678901234567890'
  " iv_version_id = '1' (optional)
  IF iv_version_id IS NOT INITIAL.
    oo_result = lo_mig->getimagesetmetadata(
      iv_datastoreid = iv_datastore_id
      iv_imagesetid = iv_image_set_id
      iv_versionid = iv_version_id ).
  ELSE.
    oo_result = lo_mig->getimagesetmetadata(
      iv_datastoreid = iv_datastore_id
      iv_imagesetid = iv_image_set_id ).
  ENDIF.
  DATA(lv_metadata_blob) = oo_result->get_imagesetmetadatablob( ).
  MESSAGE 'Image set metadata retrieved.' TYPE 'I'.
CATCH /aws1/cx_migaccessdeniedex.
  MESSAGE 'Access denied.' TYPE 'I'.
CATCH /aws1/cx_migconflictexception.
  MESSAGE 'Conflict error.' TYPE 'I'.
CATCH /aws1/cx_miginternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
CATCH /aws1/cx_migresourcenotfoundex.
  MESSAGE 'Image set not found.' TYPE 'I'.
CATCH /aws1/cx_migthrottlingex.
  MESSAGE 'Request throttled.' TYPE 'I'.
CATCH /aws1/cx_migvalidationex.
  MESSAGE 'Validation error.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetImageSetMetadatarreferência](#) da API AWS SDK for SAP ABAP.

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## ListDICOMImportJobs

O código de exemplo a seguir mostra como usar ListDICOMImportJobs.

SDK para SAP ABAP

```
TRY.  
  " iv_datastore_id = '12345678901234567890123456789012345678901234567890'  
  oo_result = lo_mig->listdicomimportjobs( iv_datastoreid = iv_datastore_id ).  
  DATA(lt_jobs) = oo_result->get_jobsummaries( ).  
  DATA(lv_count) = lines( lt_jobs ).  
  MESSAGE |Found { lv_count } DICOM import jobs.| TYPE 'I'.  
CATCH /aws1/cx_migaccessdeniedex.  
  MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_migconflictexception.  
  MESSAGE 'Conflict error.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
  MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migresourcenotfoundex.  
  MESSAGE 'Resource not found.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
  MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
  MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte [Listar DICOMImport trabalhos](#) no AWS SDK para referência da API SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## ListDatastores

O código de exemplo a seguir mostra como usar ListDatastores.

### SDK para SAP ABAP

```
TRY.  
    oo_result = lo_mig->listdatastores( ).  
    DATA(lt_datastores) = oo_result->get_datastoresummaries( ).  
    DATA(lv_count) = lines( lt_datastores ).  
    MESSAGE |Found { lv_count } data stores.| TYPE 'I'.  
CATCH /aws1/cx_migaccessdeniedex.  
    MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
    MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
    MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
    MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListDatastores](#) referência da API AWS SDK for SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## ListImageSetVersions

O código de exemplo a seguir mostra como usar ListImageSetVersions.

## SDK para SAP ABAP

```
TRY.  
  " iv_datastore_id = '1234567890123456789012345678901234567890'  
  " iv_image_set_id = '1234567890123456789012345678901234567890'  
  oo_result = lo_mig->listimagesetversions(  
    iv_datastoreid = iv_datastore_id  
    iv_imagesetid = iv_image_set_id ).  
  DATA(lt_versions) = oo_result->get_imagesetpropertieslist( ).  
  DATA(lv_count) = lines( lt_versions ).  
  MESSAGE |Found { lv_count } image set versions.| TYPE 'I'.  
CATCH /aws1/cx_migaccessdeniedex.  
  MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_migconflictexception.  
  MESSAGE 'Conflict error.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
  MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migresourcenotfoundex.  
  MESSAGE 'Image set not found.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
  MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
  MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListImageSetVersions](#) referência da API AWS SDK for SAP ABAP.

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

**ListTagsForResource**

O código de exemplo a seguir mostra como usar ListTagsForResource.

## SDK para SAP ABAP

```
TRY.
```

```

    " iv_resource_arn = 'arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012'
    oo_result = lo_mig->listtagsforresource( iv_resourcearn = iv_resource_arn ).
    DATA(lt_tags) = oo_result->get_tags( ).
    DATA(lv_count) = lines( lt_tags ).
    MESSAGE |Found { lv_count } tags for resource.| TYPE 'I'.
  CATCH /aws1/cx_migaccessdeniedex.
    MESSAGE 'Access denied.' TYPE 'I'.
  CATCH /aws1/cx_miginternalserverex.
    MESSAGE 'Internal server error.' TYPE 'I'.
  CATCH /aws1/cx_migresourcenotfoundex.
    MESSAGE 'Resource not found.' TYPE 'I'.
  CATCH /aws1/cx_migthrottlingex.
    MESSAGE 'Request throttled.' TYPE 'I'.
  CATCH /aws1/cx_migvalidationex.
    MESSAGE 'Validation error.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [ListTagsForResource](#) referência da API AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## SearchImageSets

O código de exemplo a seguir mostra como usar SearchImageSets.

### SDK para SAP ABAP

```

  TRY.
    " iv_datastore_id = '1234567890123456789012345678901234567890'
    oo_result = lo_mig->searchimagesets(
      iv_datastoreid = iv_datastore_id
      io_searchcriteria = io_search_criteria ).
    DATA(lt_imagesets) = oo_result->get_imagesetsmetadatasums( ).
    DATA(lv_count) = lines( lt_imagesets ).
    MESSAGE |Found { lv_count } image sets.| TYPE 'I'.
  TRY.

```

```
CATCH /aws1/cx_migaccessdeniedex.  
    MESSAGE 'Access denied.' TYPE 'I'.  
CATCH /aws1/cx_migconflictexception.  
    MESSAGE 'Conflict error.' TYPE 'I'.  
CATCH /aws1/cx_miginternalserverex.  
    MESSAGE 'Internal server error.' TYPE 'I'.  
CATCH /aws1/cx_migresourcenotfoundex.  
    MESSAGE 'Resource not found.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
    MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
    MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [SearchImageSets](#) referência da API AWS SDK for SAP ABAP.

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## StartDICOMImportJob

O código de exemplo a seguir mostra como usar StartDICOMImportJob.

### SDK para SAP ABAP

```
TRY.  
    " iv_job_name = 'import-job-1'  
    " iv_datastore_id = '12345678901234567890123456789012345678901234567890'  
    " iv_role_arn = 'arn:aws:iam::123456789012:role/ImportJobRole'  
    " iv_input_s3_uri = 's3://my-bucket/input/'  
    " iv_output_s3_uri = 's3://my-bucket/output/'  
    oo_result = lo_mig->startdicomimportjob(  
        iv_jobname = iv_job_name  
        iv_datastoreid = iv_datastore_id  
        iv_dataaccessrolearn = iv_role_arn  
        iv_inputs3uri = iv_input_s3_uri  
        iv_outputs3uri = iv_output_s3_uri ).
```

```

DATA(lv_job_id) = oo_result->get_jobid( ).
MESSAGE |DICOM import job started with ID: { lv_job_id }.| TYPE 'I'.
CATCH /aws1/cx_migaccessdeniedex.
MESSAGE 'Access denied.' TYPE 'I'.
CATCH /aws1/cx_migconflictexception.
MESSAGE 'Conflict error.' TYPE 'I'.
CATCH /aws1/cx_miginternalserverex.
MESSAGE 'Internal server error.' TYPE 'I'.
CATCH /aws1/cx_migresourcenotfoundex.
MESSAGE 'Resource not found.' TYPE 'I'.
CATCH /aws1/cx_migservicequotaexcdex.
MESSAGE 'Service quota exceeded.' TYPE 'I'.
CATCH /aws1/cx_migthrottlingex.
MESSAGE 'Request throttled.' TYPE 'I'.
CATCH /aws1/cx_migvalidationex.
MESSAGE 'Validation error.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte [a referência da API Start DICOMImport Job](#) in AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## TagResource

O código de exemplo a seguir mostra como usar TagResource.

### SDK para SAP ABAP

```

TRY.
  " iv_resource_arn = 'arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012'
  lo_mig->tagresource(
    iv_resourcearn = iv_resource_arn
    it_tags = it_tags ).
  MESSAGE 'Resource tagged successfully.' TYPE 'I'.
  CATCH /aws1/cx_migaccessdeniedex.

```

```

    MESSAGE 'Access denied.' TYPE 'I'.
    CATCH /aws1/cx_miginternalserverex.
    MESSAGE 'Internal server error.' TYPE 'I'.
    CATCH /aws1/cx_migresourcenotfoundex.
    MESSAGE 'Resource not found.' TYPE 'I'.
    CATCH /aws1/cx_migthrottlingex.
    MESSAGE 'Request throttled.' TYPE 'I'.
    CATCH /aws1/cx_migvalidationex.
    MESSAGE 'Validation error.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [TagResource](#) referência da API AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## UntagResource

O código de exemplo a seguir mostra como usar UntagResource.

### SDK para SAP ABAP

```

TRY.
  " iv_resource_arn = 'arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012'
  lo_mig->untagresource(
    iv_resourcearn = iv_resource_arn
    it_tagkeys = it_tag_keys ).
  MESSAGE 'Resource untagged successfully.' TYPE 'I'.
  CATCH /aws1/cx_migaccessdeniedex.
  MESSAGE 'Access denied.' TYPE 'I'.
  CATCH /aws1/cx_miginternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
  CATCH /aws1/cx_migresourcenotfoundex.
  MESSAGE 'Resource not found.' TYPE 'I'.
  CATCH /aws1/cx_migthrottlingex.
  MESSAGE 'Request throttled.' TYPE 'I'.

```

```
CATCH /aws1/cx_migvalidationex.
  MESSAGE 'Validation error.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UntagResource](#) referência da API AWS SDK for SAP ABAP.

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## UpdateImageSetMetadata

O código de exemplo a seguir mostra como usar UpdateImageSetMetadata.

### SDK para SAP ABAP

```
TRY.
  " iv_datastore_id = '1234567890123456789012345678901234567890'
  " iv_image_set_id = '1234567890123456789012345678901234567890'
  " iv_latest_version_id = '1'
  " iv_force = abap_false
  oo_result = lo_mig->updateimagesetmetadata(
    iv_datastoreid = iv_datastore_id
    iv_imagesetid = iv_image_set_id
    iv_latestversionid = iv_latest_version_id
    io_updateimagesetmetupdates = io_metadata_updates
    iv_force = iv_force ).
  DATA(lv_new_version) = oo_result->get_latestversionid( ).
  MESSAGE |Image set metadata updated to version: { lv_new_version }.| TYPE
'I'.
  CATCH /aws1/cx_migaccessdeniedex.
    MESSAGE 'Access denied.' TYPE 'I'.
  CATCH /aws1/cx_migconflictexception.
    MESSAGE 'Conflict error.' TYPE 'I'.
  CATCH /aws1/cx_miginternalserverex.
    MESSAGE 'Internal server error.' TYPE 'I'.
  CATCH /aws1/cx_migresourcenotfoundex.
    MESSAGE 'Image set not found.' TYPE 'I'.
```

```
CATCH /aws1/cx_migservicequotaexcdex.  
  MESSAGE 'Service quota exceeded.' TYPE 'I'.  
CATCH /aws1/cx_migthrottlingex.  
  MESSAGE 'Request throttled.' TYPE 'I'.  
CATCH /aws1/cx_migvalidationex.  
  MESSAGE 'Validation error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateImageSetMetadata](#) referência da API AWS SDK for SAP ABAP.

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## HealthLake exemplos usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com HealthLake

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos


- [Ações](#)

## Ações

### CreateFHIRDatastore

O código de exemplo a seguir mostra como usar CreateFHIRDatastore.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_datastore_name = 'MyHealthLakeDataStore'
  oo_result = lo_hll->createfhirdatastore(
    iv_datastorename = iv_datastore_name
    iv_datastoretypeversion = 'R4'
  ).
  MESSAGE 'Data store created successfully.' TYPE 'I'.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
    DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_validation_ex.
  CATCH /aws1/cx_hllinternalserverex INTO DATA(lo_internal_ex).
    lv_error = |Internal server error: { lo_internal_ex->av_err_code }-
{ lo_internal_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_internal_ex.
  CATCH /aws1/cx_hllthrottlingex INTO DATA(lo_throttling_ex).
    lv_error = |Throttling error: { lo_throttling_ex->av_err_code }-
{ lo_throttling_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_throttling_ex.
ENDTRY.


```

- Para obter detalhes da API, consulte a referência da API [Create FHIRDatastore](#) in AWS SDK for SAP ABAP.

**DeleteFHIRDatastore**

O código de exemplo a seguir mostra como usar DeleteFHIRDatastore.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " iv_datastore_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'  
  oo_result = lo_hll->deletefhirdatastore(  
    iv_datastoreid = iv_datastore_id  
  ).  
  MESSAGE 'Data store deleted successfully.' TYPE 'I'.  
  CATCH /aws1/cx_hllaccessdeniedex INTO DATA(lo_access_ex).  
    DATA(lv_error) = |Access denied: { lo_access_ex->av_err_code }-  
{ lo_access_ex->av_err_msg }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_access_ex.  
  CATCH /aws1/cx_hllconflictexception INTO DATA(lo_conflict_ex).  
    lv_error = |Conflict error: { lo_conflict_ex->av_err_code }-  
{ lo_conflict_ex->av_err_msg }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_conflict_ex.  
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).  
    lv_error = |Resource not found: { lo_notfound_ex->av_err_code }-  
{ lo_notfound_ex->av_err_msg }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_notfound_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte [Excluir FHIRDatastore](#) no AWS SDK para referência da API SAP ABAP.

## DescribeFHIRDatastore

O código de exemplo a seguir mostra como usar DescribeFHIRDatastore.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_datastore_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  oo_result = lo_hll->describefhirdatastore(
    iv_datastoreid = iv_datastore_id
  ).
  DATA(lo_datastore_properties) = oo_result->get_datastoreproperties( ).
  IF lo_datastore_properties IS BOUND.
    DATA(lv_datastore_name) = lo_datastore_properties->get_datastorename( ).
    DATA(lv_datastore_status) = lo_datastore_properties-
>get_datastorestatus( ).
    MESSAGE 'Data store described successfully.' TYPE 'I'.
  ENDIF.
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
    DATA(lv_error) = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_notfound_ex.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
    lv_error = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_validation_ex.
ENDTRY.


```

- Para obter detalhes da API, consulte [Descrição FHIRDatastore](#) na referência da API ABAP do AWS SDK for SAP.

**DescribeFHIRExportJob**

O código de exemplo a seguir mostra como usar DescribeFHIRExportJob.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  " iv_datastore_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  " iv_job_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  oo_result = lo_hll->describefhirexportjob(
    iv_datastoreid = iv_datastore_id
    iv_jobid = iv_job_id
  ).
  DATA(lo_export_job_properties) = oo_result->get_exportjobproperties( ).
  IF lo_export_job_properties IS BOUND.
    DATA(lv_job_status) = lo_export_job_properties->get_jobstatus( ).
    MESSAGE |Export job status: { lv_job_status }.| TYPE 'I'.
  ENDIF.
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
  DATA(lv_error) = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_notfound_ex.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
  lv_error = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_validation_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a referência da API [Describe FHIRExport Job](#) in AWS SDK for SAP ABAP.

## DescribeFHIRImportJob

O código de exemplo a seguir mostra como usar DescribeFHIRImportJob.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_datastore_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  " iv_job_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  oo_result = lo_hll->describefhirimportjob(
    iv_datastoreid = iv_datastore_id
    iv_jobid = iv_job_id
  ).
  DATA(lo_import_job_properties) = oo_result->get_importjobproperties( ).
  IF lo_import_job_properties IS BOUND.
    DATA(lv_job_status) = lo_import_job_properties->get_jobstatus( ).
    MESSAGE |Import job status: { lv_job_status }.| TYPE 'I'.
  ENDIF.
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
  DATA(lv_error) = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_notfound_ex.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
  lv_error = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_validation_ex.
ENDTRY.


```

- Para obter detalhes da API, consulte a referência da API [Describe FHIRImport Job](#) in AWS SDK for SAP ABAP.

## ListFHIRDatastores

O código de exemplo a seguir mostra como usar ListFHIRDatastores.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_hll->listfhirdatastores( ).  
    DATA(lt_datastores) = oo_result->get_datastorepropertieslist( ).  
    DATA(lv_datastore_count) = lines( lt_datastores ).  
    MESSAGE |Found { lv_datastore_count } data store(s).| TYPE 'I'.  
    CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).  
        DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-  
{ lo_validation_ex->av_err_msg }|.  
        MESSAGE lv_error TYPE 'I'.  
        RAISE EXCEPTION lo_validation_ex.  
    CATCH /aws1/cx_hllthrottlingex INTO DATA(lo_throttling_ex).  
        lv_error = |Throttling error: { lo_throttling_ex->av_err_code }-  
{ lo_throttling_ex->av_err_msg }|.  
        MESSAGE lv_error TYPE 'I'.  
        RAISE EXCEPTION lo_throttling_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte [Lista FHIRDatastores](#) no AWS SDK para referência da API SAP ABAP.

## ListFHIRExportJobs

O código de exemplo a seguir mostra como usar ListFHIRExportJobs.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_datastore_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  IF iv_submitted_after IS NOT INITIAL.
    oo_result = lo_hll->listfhirexportjobs(
      iv_datastoreid = iv_datastore_id
      iv_submittedafter = iv_submitted_after
    ).
  ELSE.
    oo_result = lo_hll->listfhirexportjobs(
      iv_datastoreid = iv_datastore_id
    ).
  ENDIF.
  DATA(lt_export_jobs) = oo_result->get_exportjobpropertieslist( ).
  DATA(lv_job_count) = lines( lt_export_jobs ).
  MESSAGE |Found { lv_job_count } export job(s).| TYPE 'I'.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
    DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_validation_ex.
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
    lv_error = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_notfound_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte [Listar FHIRExport trabalhos](#) no AWS SDK para referência da API SAP ABAP.

## ListFHIRImportJobs

O código de exemplo a seguir mostra como usar ListFHIRImportJobs.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_datastore_id = 'a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  IF iv_submitted_after IS NOT INITIAL.
    oo_result = lo_hll->listfhirimportjobs(
      iv_datastoreid = iv_datastore_id
      iv_submittedafter = iv_submitted_after
    ).
  ELSE.
    oo_result = lo_hll->listfhirimportjobs(
      iv_datastoreid = iv_datastore_id
    ).
  ENDIF.
  DATA(lt_import_jobs) = oo_result->get_importjobpropertieslist( ).
  DATA(lv_job_count) = lines( lt_import_jobs ).
  MESSAGE |Found { lv_job_count } import job(s).| TYPE 'I'.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
    DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_validation_ex.
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
    lv_error = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_notfound_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte [Listar FHIRImport trabalhos](#) no AWS SDK para referência da API SAP ABAP.

## ListTagsForResource

O código de exemplo a seguir mostra como usar ListTagsForResource.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_resource_arn = 'arn:aws:healthlake:us-east-1:123456789012:datastore/
fhir/a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  DATA(lo_result) = lo_hll->listtagsforresource(
    iv_resourcearn = iv_resource_arn
  ).
  ot_tags = lo_result->get_tags( ).
  DATA(lv_tag_count) = lines( ot_tags ).
  MESSAGE |Found { lv_tag_count } tag(s).| TYPE 'I'.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
  DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_validation_ex.
  CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
  lv_error = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_notfound_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListTagsForResource](#) referência da API AWS SDK for SAP ABAP.

## StartFHIRExportJob

O código de exemplo a seguir mostra como usar StartFHIRExportJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_job_name = 'MyExportJob'
  " iv_output_s3_uri = 's3://my-bucket/export/output/'

```

```

    " iv_kms_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012'
    " iv_data_access_role_arn = 'arn:aws:iam::123456789012:role/
HealthLakeExportRole'
    oo_result = lo_hll->startfhirexportjob(
      iv_jobname = iv_job_name
      io_outputdataconfig = NEW /aws1/cl_hlloutputdataconfig(
        io_s3configuration = NEW /aws1/cl_hlls3configuration(
          iv_s3uri = iv_output_s3_uri
          iv_kmskeyid = iv_kms_key_id
        )
      )
      iv_dataaccessrolearn = iv_data_access_role_arn
      iv_datastoreid = iv_datastore_id
    ).
    DATA(lv_job_id) = oo_result->get_jobid( ).
    MESSAGE |Export job started with ID { lv_job_id }.| TYPE 'I'.
    CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
    DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_validation_ex.
    CATCH /aws1/cx_hllthrottlingex INTO DATA(lo_throttling_ex).
    lv_error = |Throttling error: { lo_throttling_ex->av_err_code }-
{ lo_throttling_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_throttling_ex.
    CATCH /aws1/cx_hllaccessdeniedex INTO DATA(lo_access_ex).
    lv_error = |Access denied: { lo_access_ex->av_err_code }-{ lo_access_ex-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_access_ex.
  ENTRY.


```

- Para obter detalhes da API, consulte [a referência da API Start FHIRExport Job](#) in AWS SDK for SAP ABAP.

## StartFHIRImportJob

O código de exemplo a seguir mostra como usar StartFHIRImportJob.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_job_name = 'MyImportJob'
  " iv_input_s3_uri = 's3://my-bucket/import/data.ndjson'
  " iv_job_output_s3_uri = 's3://my-bucket/import/output/'
  " iv_kms_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012'
  " iv_data_access_role_arn = 'arn:aws:iam::123456789012:role/
HealthLakeImportRole'
  oo_result = lo_hll->startfhirimportjob(
    iv_jobname = iv_job_name
    io_inputdataconfig = NEW /aws1/cl_hllinputdataconfig( iv_s3uri =
iv_input_s3_uri )
    io_joboutputdataconfig = NEW /aws1/cl_hlloutputdataconfig(
      io_s3configuration = NEW /aws1/cl_hlls3configuration(
        iv_s3uri = iv_job_output_s3_uri
        iv_kmskeyid = iv_kms_key_id
      )
    )
    iv_dataaccessrolearn = iv_data_access_role_arn
    iv_datastoreid = iv_datastore_id
  ).
  DATA(lv_job_id) = oo_result->get_jobid( ).
  MESSAGE |Import job started with ID { lv_job_id }.| TYPE 'I'.
  CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
  DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_validation_ex.
  CATCH /aws1/cx_hllthrottlingex INTO DATA(lo_throttling_ex).
  lv_error = |Throttling error: { lo_throttling_ex->av_err_code }-
{ lo_throttling_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_throttling_ex.
  CATCH /aws1/cx_hllaccessdeniedex INTO DATA(lo_access_ex).

```

```

lv_error = |Access denied: { lo_access_ex->av_err_code }-{ lo_access_ex->av_err_msg }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_access_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte [a referência da API Start FHIRImport Job](#) in AWS SDK for SAP ABAP.

## TagResource

O código de exemplo a seguir mostra como usar TagResource.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_resource_arn = 'arn:aws:healthlake:us-east-1:123456789012:datastore/
fhir/a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
  lo_hll->tagresource(
    iv_resourcearn = iv_resource_arn
    it_tags = it_tags
  ).
  MESSAGE 'Resource tagged successfully.' TYPE 'I'.
CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
  DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_validation_ex.
CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
  lv_error = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_notfound_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [TagResource](#) referência da API AWS SDK for SAP ABAP.

## UntagResource

O código de exemplo a seguir mostra como usar UntagResource.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    " iv_resource_arn = 'arn:aws:healthlake:us-east-1:123456789012:datastore/
fhir/a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6'
    lo_hll->untagresource(
        iv_resourcearn = iv_resource_arn
        it_tagkeys = it_tag_keys
    ).
    MESSAGE 'Resource untagged successfully.' TYPE 'I'.
CATCH /aws1/cx_hllvalidationex INTO DATA(lo_validation_ex).
    DATA(lv_error) = |Validation error: { lo_validation_ex->av_err_code }-
{ lo_validation_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_validation_ex.
CATCH /aws1/cx_hllresourcenotfoundex INTO DATA(lo_notfound_ex).
    lv_error = |Resource not found: { lo_notfound_ex->av_err_code }-
{ lo_notfound_ex->av_err_msg }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_notfound_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UntagResource](#) referência da API AWS SDK for SAP ABAP.

# Exemplos de IAM usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com IAM.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### **AttachRolePolicy**

O código de exemplo a seguir mostra como usar `AttachRolePolicy`.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_iam->attachrolepolicy(  
    iv_rolename = iv_role_name  
    iv_policyarn = iv_policy_arn ).  
  MESSAGE 'Policy attached to role successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'Role or policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceededex.  
  MESSAGE 'Policy attachment limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [AttachRolePolicy](#) referência da API AWS SDK for SAP ABAP.

## AttachUserPolicy

O código de exemplo a seguir mostra como usar `AttachUserPolicy`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_iam->attachuserpolicy(  
    iv_username = iv_user_name  
    iv_policyarn = iv_policy_arn ).  
  MESSAGE 'Policy attached to user successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'User or policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceededex.  
  MESSAGE 'Policy attachment limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [AttachUserPolicy](#) referência da API AWS SDK for SAP ABAP.

## CreateAccessKey

O código de exemplo a seguir mostra como usar `CreateAccessKey`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_iam->createaccesskey(  
        iv_username = iv_user_name ).  
    MESSAGE 'Access key created successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'User does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceedex.  
    MESSAGE 'Maximum number of access keys reached.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateAccessKey](#) referência da API AWS SDK for SAP ABAP.

## CreateAccountAlias

O código de exemplo a seguir mostra como usar CreateAccountAlias.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_iam->createaccountalias(  
        iv_accountalias = iv_account_alias ).  
    MESSAGE 'Account alias created successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamentityalrddyexex.
```

```
MESSAGE 'Account alias already exists.' TYPE 'E'.
CATCH /aws1/cx_iamlimitexceeddex.
MESSAGE 'Account alias limit exceeded.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateAccountAlias](#) referência da API AWS SDK for SAP ABAP.

## CreatePolicy

O código de exemplo a seguir mostra como usar CreatePolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_iam->createpolicy(
    iv_policyname = iv_policy_name
    iv_policydocument = iv_policy_document
    iv_description = iv_description ).
  MESSAGE 'Policy created successfully.' TYPE 'I'.
CATCH /aws1/cx_iamentityalrddyexex.
  MESSAGE 'Policy already exists.' TYPE 'E'.
CATCH /aws1/cx_iammalformedplydocex.
  MESSAGE 'Policy document is malformed.' TYPE 'E'.
CATCH /aws1/cx_iamlimitexceeddex.
  MESSAGE 'Policy limit exceeded.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreatePolicy](#) referência da API AWS SDK for SAP ABAP.

## CreatePolicyVersion

O código de exemplo a seguir mostra como usar CreatePolicyVersion.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->createpolicyversion(  
        iv_policyarn = iv_policy_arn  
        iv_policydocument = iv_policy_document  
        iv_setasdefault = iv_set_as_default ).  
    MESSAGE 'Policy version created successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iammalformedplydocex.  
    MESSAGE 'Policy document is malformed.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceeddex.  
    MESSAGE 'Policy version limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreatePolicyVersion](#) referência da API AWS SDK for SAP ABAP.

## CreateRole

O código de exemplo a seguir mostra como usar CreateRole.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_iam->createrole(  
    iv_rolename = iv_role_name  
    iv_assumerolepolicydocument = iv_assume_role_policy_document ).  
  MESSAGE 'Role created successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamentityalrddyexex.  
  MESSAGE 'Role already exists.' TYPE 'E'.  
CATCH /aws1/cx_iammalformedplydocex.  
  MESSAGE 'Assume role policy document is malformed.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceeddex.  
  MESSAGE 'Role limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateRole](#) referência da API AWS SDK for SAP ABAP.

## CreateServiceLinkedRole

O código de exemplo a seguir mostra como usar CreateServiceLinkedRole.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_iam->listpolicyversions(  
    iv_policyarn = iv_policy_arn ).  
  MESSAGE 'Retrieved policy versions list.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'Policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamservicefailureex.  
  MESSAGE 'Service failure when listing policy versions.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateServiceLinkedRole](#) referência da API AWS SDK for SAP ABAP.

## CreateUser

O código de exemplo a seguir mostra como usar CreateUser.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->createuser(  
        iv_username = iv_user_name ).  
    MESSAGE 'User created successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamentityalrddyexex.  
    MESSAGE 'User already exists.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceeddex.  
    MESSAGE 'Limit exceeded for IAM users.' TYPE 'E'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Entity does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateUser](#) referência da API AWS SDK for SAP ABAP.

## DeleteAccessKey

O código de exemplo a seguir mostra como usar DeleteAccessKey.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
lo_iam->deleteaccesskey(  
    iv_accesskeyid = iv_access_key_id  
    iv_username = iv_user_name ).  
MESSAGE 'Access key deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
MESSAGE 'Access key or user does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAccessKey](#) referência da API AWS SDK for SAP ABAP.

## DeleteAccountAlias

O código de exemplo a seguir mostra como usar DeleteAccountAlias.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
lo_iam->deleteaccountalias(  
    iv_accountalias = iv_account_alias ).  
MESSAGE 'Account alias deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
MESSAGE 'Account alias does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAccountAlias](#) referência da API AWS SDK for SAP ABAP.

## DeletePolicy

O código de exemplo a seguir mostra como usar DeletePolicy.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_iam->deletepolicy( iv_policyarn = iv_policy_arn ).  
  MESSAGE 'Policy deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'Policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamdeleteconflictex.  
  MESSAGE 'Policy cannot be deleted due to attachments.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeletePolicy](#) referência da API AWS SDK for SAP ABAP.

## DeletePolicyVersion

O código de exemplo a seguir mostra como usar DeletePolicyVersion.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_iam->deletepolicyversion(  
    iv_policyarn = iv_policy_arn  
    iv_versionid = iv_version_id ).  
  MESSAGE 'Policy version deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'Policy or version does not exist.' TYPE 'E'.
```

```
CATCH /aws1/cx_iamdeleteconflictex.  
    MESSAGE 'Cannot delete default policy version.' TYPE 'E'.  
CATCH /aws1/cx_iamlimitexceedex.  
    MESSAGE 'Limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeletePolicyVersion](#) referência da API AWS SDK for SAP ABAP.

## DeleteRole

O código de exemplo a seguir mostra como usar DeleteRole.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    lo_iam->deleterole( iv_rolename = iv_role_name ).  
    MESSAGE 'Role deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Role does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamdeleteconflictex.  
    MESSAGE 'Role cannot be deleted due to attached resources.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteRole](#) referência da API AWS SDK for SAP ABAP.

## DeleteUser

O código de exemplo a seguir mostra como usar DeleteUser.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_iam->deleteuser( iv_username = iv_user_name ).  
  MESSAGE 'User deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'User does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamdeleteconflictex.  
  MESSAGE 'User cannot be deleted due to attached resources.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteUser](#) referência da API AWS SDK for SAP ABAP.

**DetachRolePolicy**

O código de exemplo a seguir mostra como usar DetachRolePolicy.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_iam->detachrolepolicy(  
    iv_rolename = iv_role_name  
    iv_policyarn = iv_policy_arn ).  
  MESSAGE 'Policy detached from role successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'Role or policy does not exist.' TYPE 'E'.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetachRolePolicy](#) referência da API AWS SDK for SAP ABAP.

## DetachUserPolicy

O código de exemplo a seguir mostra como usar DetachUserPolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_iam->detachuserpolicy(  
    iv_username = iv_user_name  
    iv_policyarn = iv_policy_arn ).  
  MESSAGE 'Policy detached from user successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'User or policy does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetachUserPolicy](#) referência da API AWS SDK for SAP ABAP.

## GenerateCredentialReport

O código de exemplo a seguir mostra como usar GenerateCredentialReport.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->generatecredentialreport( ).  
    MESSAGE 'Credential report generation started.' TYPE 'I'.  
    CATCH /aws1/cx_iamlimitexceedex.  
        MESSAGE 'Report generation limit exceeded.' TYPE 'E'.  
    CATCH /aws1/cx_iamservicefailureex.  
        MESSAGE 'Service failure when generating credential report.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GenerateCredentialReport](#) referência da API AWS SDK for SAP ABAP.

**GetAccessKeyLastUsed**

O código de exemplo a seguir mostra como usar GetAccessKeyLastUsed.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->getaccesskeylastused(  
        iv_accesskeyid = iv_access_key_id ).  
    MESSAGE 'Retrieved access key last used information.' TYPE 'I'.  
    CATCH /aws1/cx_iamnosuchentityex.  
        MESSAGE 'Access key does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetAccessKeyLastUsed](#) referência da API AWS SDK for SAP ABAP.

## GetAccountAuthorizationDetails

O código de exemplo a seguir mostra como usar `GetAccountAuthorizationDetails`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->getaccountauthdetails( ).  
    MESSAGE 'Retrieved account authorization details.' TYPE 'I'.  
    CATCH /aws1/cx_iamservicefailureex.  
        MESSAGE 'Service failure when getting account authorization details.' TYPE  
'E'.  
    ENDTRY.
```

- Para obter detalhes da API, consulte a [GetAccountAuthorizationDetails](#) referência da API AWS SDK for SAP ABAP.

## GetAccountPasswordPolicy

O código de exemplo a seguir mostra como usar `GetAccountPasswordPolicy`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->getaccountpasswordpolicy( ).  
    MESSAGE 'Retrieved account password policy.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'No password policy exists.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when getting password policy.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetAccountPasswordPolicy](#) referência da API AWS SDK for SAP ABAP.

## GetAccountSummary

O código de exemplo a seguir mostra como usar GetAccountSummary.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_iam->getaccountsummary( ).  
    MESSAGE 'Retrieved account summary.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when getting account summary.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetAccountSummary](#) referência da API AWS SDK for SAP ABAP.

## GetCredentialReport

O código de exemplo a seguir mostra como usar GetCredentialReport.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_iam->getcredentialreport( ).  
    MESSAGE 'Retrieved credential report.' TYPE 'I'.  
CATCH /aws1/cx_iamcredrptnotpresen00.  
    MESSAGE 'Credential report not present.' TYPE 'E'.  
CATCH /aws1/cx_iamcredrptexpiredex.  
    MESSAGE 'Credential report expired.' TYPE 'E'.  
CATCH /aws1/cx_iamcredrptnotreadyex.  
    MESSAGE 'Credential report not ready.' TYPE 'E'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when getting credential report.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetCredentialReport](#) referência da API AWS SDK for SAP ABAP.

## GetPolicy

O código de exemplo a seguir mostra como usar GetPolicy.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->getpolicy( iv_policyarn = iv_policy_arn ).
```

```
MESSAGE 'Retrieved policy information.' TYPE 'I'.
CATCH /aws1/cx_iamnosuchentityex.
MESSAGE 'Policy does not exist.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetPolicy](#) referência da API AWS SDK for SAP ABAP.

## GetPolicyVersion

O código de exemplo a seguir mostra como usar `GetPolicyVersion`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  oo_result = lo_iam->getpolicyversion(
    iv_policyarn = iv_policy_arn
    iv_versionid = iv_version_id ).
  MESSAGE 'Retrieved policy version information.' TYPE 'I'.
CATCH /aws1/cx_iamnosuchentityex.
  MESSAGE 'Policy or version does not exist.' TYPE 'E'.
CATCH /aws1/cx_iaminvalidinputex.
  MESSAGE 'Invalid input provided.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetPolicyVersion](#) referência da API AWS SDK for SAP ABAP.

## GetRole

O código de exemplo a seguir mostra como usar `GetRole`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_iam->getrole( iv_rolename = iv_role_name ).  
    MESSAGE 'Retrieved role information.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Role does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetRole](#) referência da API AWS SDK for SAP ABAP.

## ListAccessKeys

O código de exemplo a seguir mostra como usar ListAccessKeys.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listaccesskeys(  
        iv_username = iv_user_name ).  
    MESSAGE 'Retrieved access key list.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'User does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListAccessKeys](#) referência da API AWS SDK for SAP ABAP.

## ListAccountAliases

O código de exemplo a seguir mostra como usar ListAccountAliases.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listaccountaliases( ).  
    MESSAGE 'Retrieved account alias list.' TYPE 'I'.  
    CATCH /aws1/cx_iamservicefailureex.  
        MESSAGE 'Service failure when listing account aliases.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListAccountAliases](#) referência da API AWS SDK for SAP ABAP.

## ListAttachedRolePolicies

O código de exemplo a seguir mostra como usar ListAttachedRolePolicies.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
oo_result = lo_iam->listattachedrolepolicies(  
    iv_rolename = iv_role_name ).  
MESSAGE 'Retrieved attached policy list for role.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
MESSAGE 'Role does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListAttachedRolePolicies](#) referência da API AWS SDK for SAP ABAP.

## ListGroups

O código de exemplo a seguir mostra como usar ListGroups.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listgroups( ).  
MESSAGE 'Retrieved group list.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
MESSAGE 'Service failure when listing groups.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListGroups](#) referência da API AWS SDK for SAP ABAP.

## ListPolicies

O código de exemplo a seguir mostra como usar ListPolicies.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listpolicies( iv_scope = iv_scope ).  
    MESSAGE 'Retrieved policy list.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when listing policies.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListPolicies](#) referência da API AWS SDK for SAP ABAP.

**ListPolicyVersions**

O código de exemplo a seguir mostra como usar ListPolicyVersions.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listpolicyversions(  
        iv_policyarn = iv_policy_arn ).  
    MESSAGE 'Retrieved policy versions list.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when listing policy versions.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListPolicyVersions](#) referência da API AWS SDK for SAP ABAP.

## ListRolePolicies

O código de exemplo a seguir mostra como usar ListRolePolicies.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listrolepolicies(  
        iv_rolename = iv_role_name ).  
    MESSAGE 'Retrieved inline policy list for role.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Role does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListRolePolicies](#) referência da API AWS SDK for SAP ABAP.

## ListRoles

O código de exemplo a seguir mostra como usar ListRoles.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_iam->listroles( ).  
    MESSAGE 'Retrieved role list.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when listing roles.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListRoles](#) referência da API AWS SDK for SAP ABAP.

## ListSAMLProviders

O código de exemplo a seguir mostra como usar ListSAMLProviders.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_iam->listsamlproviders( ).  
    MESSAGE 'Retrieved SAML provider list.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when listing SAML providers.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte [Lista SAMLProviders](#) no AWS SDK para referência da API SAP ABAP.

## ListUsers

O código de exemplo a seguir mostra como usar ListUsers.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_iam->listusers( ).  
    MESSAGE 'Retrieved user list.' TYPE 'I'.  
CATCH /aws1/cx_iamservicefailureex.  
    MESSAGE 'Service failure when listing users.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListUsers](#) referência da API AWS SDK for SAP ABAP.

**SetDefaultPolicyVersion**

O código de exemplo a seguir mostra como usar SetDefaultPolicyVersion.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_iam->setdefaultpolicyversion(  
        iv_policyarn = iv_policy_arn  
        iv_versionid = iv_version_id ).  
    MESSAGE 'Default policy version set successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'Policy or version does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iaminvalidinputex.  
    MESSAGE 'Invalid input provided.' TYPE 'E'.
```

```
CATCH /aws1/cx_iamlimitexceedex.  
  MESSAGE 'Limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [SetDefaultPolicyVersion](#) referência da API AWS SDK for SAP ABAP.

## UpdateAccessKey

O código de exemplo a seguir mostra como usar UpdateAccessKey.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_iam->updateaccesskey(  
    iv_accesskeyid = iv_access_key_id  
    iv_status = iv_status  
    iv_username = iv_user_name ).  
  MESSAGE 'Access key updated successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'Access key or user does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateAccessKey](#) referência da API AWS SDK for SAP ABAP.

## UpdateUser

O código de exemplo a seguir mostra como usar UpdateUser.

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_iam->updateuser(  
    iv_username = iv_user_name  
    iv_newusername = iv_new_user_name ).  
  MESSAGE 'User updated successfully.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
  MESSAGE 'User does not exist.' TYPE 'E'.  
CATCH /aws1/cx_iamentityalrдыexex.  
  MESSAGE 'New user name already exists.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateUser](#) referência da API AWS SDK for SAP ABAP.

## AWS IoT data exemplos usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. AWS IoT data

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### GetThingShadow

O código de exemplo a seguir mostra como usar GetThingShadow.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    DATA(lo_result) = lo_iop->getthingshadow( iv_thingname = iv_thing_name ).  
  
    " Convert xstring payload to JSON string  
    DATA(lv_payload) = lo_result->get_payload( ).  
    ov_shadow = /aws1/cl_rt_util=>xstring_to_string( lv_payload ).  
    MESSAGE 'Thing shadow retrieved successfully.' TYPE 'I'.  
CATCH /aws1/cx_iopresourcenotfoundex.  
    MESSAGE 'Thing shadow not found.' TYPE 'E'.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |{ lo_exception->get_text( ) }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetThingShadow](#) referência da API AWS SDK for SAP ABAP.

### UpdateThingShadow

O código de exemplo a seguir mostra como usar UpdateThingShadow.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Convert JSON string to xstring for payload  
  DATA(lv_payload) = /aws1/cl_rt_util=>string_to_xstring( iv_shadow_state ).  
  
  lo_iop->updatethingsshadow(  
    iv_thingname = iv_thing_name  
    iv_payload = lv_payload ).  
  MESSAGE 'Thing shadow updated successfully.' TYPE 'I'.  
CATCH /aws1/cx_iopresourcenotfoundex.  
  MESSAGE 'Thing not found.' TYPE 'E'.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_exception).  
  DATA(lv_error) = |{ lo_exception->get_text( ) }|.  
  MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateThingShadow](#) referência da API AWS SDK for SAP ABAP.

## AWS IoT SiteWise exemplos usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. AWS IoT SiteWise

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Conceitos básicos](#)
- [Ações](#)

## Conceitos básicos

Olá AWS IoT SiteWise

O exemplo de código a seguir mostra como começar a usar o AWS IoT SiteWise.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ios->listassetmodels( ). " oo_result is returned for testing  
purposes. "  
    DATA(lt_asset_models) = oo_result->get_assetmodelsummaries( ).  
    MESSAGE 'Retrieved list of asset models.' TYPE 'I'.  
    CATCH /aws1/cx_rt_generic.  
        MESSAGE 'Unable to list asset models.' TYPE 'E'.  
ENDTRY.
```


- Para obter detalhes da API, consulte a [ListAssetModels](#) referência da API AWS SDK for SAP ABAP.

## Ações

### BatchPutAssetPropertyValue

O código de exemplo a seguir mostra como usar BatchPutAssetPropertyValue.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_ios->batchputassetpropertyvalue(  
    it_entries = it_entries  
  ).  
  MESSAGE 'Data sent to IoT SiteWise asset successfully.' TYPE 'I'.  
CATCH /aws1/cx_iosresourcenotfoundex.  
  MESSAGE 'Asset does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [BatchPutAssetPropertyValue](#) referência da API AWS SDK for SAP ABAP.

## CreateAsset

O código de exemplo a seguir mostra como usar CreateAsset.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_ios->createasset(  
    iv_assetname = iv_asset_name  
    iv_assetmodelid = iv_asset_model_id  
  ). " oo_result is returned for testing purposes. "  
  MESSAGE 'IoT SiteWise asset created' TYPE 'I'.
```

```
CATCH /aws1/cx_iosresourcenotfoundex.  
  MESSAGE 'Asset model does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateAsset](#) referência da API AWS SDK for SAP ABAP.

## CreateAssetModel

O código de exemplo a seguir mostra como usar CreateAssetModel.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  oo_result = lo_ios->createassetmodel(  
    iv_assetmodelname = iv_asset_model_name  
    iv_assetmodeldescription = 'This is a sample asset model description.'  
    it_assetmodelproperties = it_properties  
  ). " oo_result is returned for testing purposes. "  
  MESSAGE 'IoT SiteWise asset model created' TYPE 'I'.  
CATCH /aws1/cx_iosresrcalrddyexistsex.  
  MESSAGE 'Asset model already exists.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateAssetModel](#) referência da API AWS SDK for SAP ABAP.

## CreateGateway

O código de exemplo a seguir mostra como usar CreateGateway.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  oo_result = lo_ios->creategateway(
    iv_gatewayname = iv_gateway_name
    io_gatewayplatform = NEW /aws1/cl_iosgatewayplatform(
      io_greengrassv2 = NEW /aws1/cl_iosgreengrassv2(
        iv_coredevicethingname = iv_core_device_thing_name
      )
    )
    it_tags = VALUE /aws1/cl_iostagmap_w=>tt_tagmap(
      (
        VALUE /aws1/cl_iostagmap_w=>ts_tagmap_maprow(
          key = 'Environment'
          value = NEW /aws1/cl_iostagmap_w( 'Production' )
        )
      )
    )
  ). " oo_result is returned for testing purposes. "
  MESSAGE 'IoT SiteWise gateway created' TYPE 'I'.
CATCH /aws1/cx_iosresrcalrddyexistsex.
  MESSAGE 'Gateway already exists.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateGateway](#) referência da API AWS SDK for SAP ABAP.

## DeleteAsset

O código de exemplo a seguir mostra como usar DeleteAsset.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_ios->deleteasset(  
    iv_assetid = iv_asset_id  
  ).  
  MESSAGE 'IoT SiteWise asset deleted.' TYPE 'I'.  
CATCH /aws1/cx_rt_generic.  
  MESSAGE 'Unable to delete asset.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAsset](#) referência da API AWS SDK for SAP ABAP.

**DeleteAssetModel**

O código de exemplo a seguir mostra como usar DeleteAssetModel.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_ios->deleteassetmodel(  
    iv_assetmodelid = iv_asset_model_id  
  ).  
  MESSAGE 'IoT SiteWise asset model deleted.' TYPE 'I'.  
CATCH /aws1/cx_rt_generic.  
  MESSAGE 'Unable to delete asset model.' TYPE 'E'.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAssetModel](#) referência da API AWS SDK for SAP ABAP.

## DeleteGateway

O código de exemplo a seguir mostra como usar DeleteGateway.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_ios->deletegateway(  
    iv_gatewayid = iv_gateway_id  
  ).  
  MESSAGE 'IoT SiteWise gateway deleted.' TYPE 'I'.  
CATCH /aws1/cx_iosresourcenotfoundex.  
  MESSAGE 'Gateway does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteGateway](#) referência da API AWS SDK for SAP ABAP.

## DescribeGateway

O código de exemplo a seguir mostra como usar DescribeGateway.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_ios->describegateway(  
        iv_gatewayid = iv_gateway_id  
    ). " oo_result is returned for testing purposes. "  
    MESSAGE 'Retrieved gateway description.' TYPE 'I'.  
CATCH /aws1/cx_iosresourcenotfoundex.  
    MESSAGE 'Gateway does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeGateway](#) referência da API AWS SDK for SAP ABAP.

## GetAssetPropertyValue

O código de exemplo a seguir mostra como usar GetAssetPropertyValue.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ios->getassetpropertyvalue(  
        iv_assetid = iv_asset_id  
        iv_propertyid = iv_property_id  
    ). " oo_result is returned for testing purposes. "  
    MESSAGE 'Retrieved asset property value.' TYPE 'I'.
```

```
CATCH /aws1/cx_iosresourcenotfoundex.
  MESSAGE 'Asset or property does not exist.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetAssetPropertyValue](#) referência da API AWS SDK for SAP ABAP.

## ListAssetModelProperties

O código de exemplo a seguir mostra como usar `ListAssetModelProperties`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_ios->listassetmodelproperties(
    iv_assetmodelid = iv_asset_model_id
  ). " oo_result is returned for testing purposes. "
  DATA(lt_properties) = oo_result->get_assetmodelpropertysums( ).
  MESSAGE 'Retrieved list of asset model properties.' TYPE 'I'.
CATCH /aws1/cx_rt_generic.
  MESSAGE 'Unable to list asset model properties.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListAssetModelProperties](#) referência da API AWS SDK for SAP ABAP.

## ListAssetModels

O código de exemplo a seguir mostra como usar `ListAssetModels`.

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ios->listassetmodels( ). " oo_result is returned for testing  
purposes. "  
    DATA(lt_asset_models) = oo_result->get_assetmodelsummaries( ).  
    MESSAGE 'Retrieved list of asset models.' TYPE 'I'.  
    CATCH /aws1/cx_rt_generic.  
        MESSAGE 'Unable to list asset models.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListAssetModels](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon Keyspaces usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Amazon Keyspaces.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### CreateKeyspace

O código de exemplo a seguir mostra como usar CreateKeyspace.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kys->createkeyspace(  
        iv_keyspacename = iv_keyspace_name ).  
    MESSAGE 'Keyspace created successfully.' TYPE 'I'.  
    CATCH /aws1/cx_kysconflictexception.  
        MESSAGE 'Keyspace already exists.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->av_err_msg }|.  
        MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateKeyspace](#) referência da API AWS SDK for SAP ABAP.

### CreateTable

O código de exemplo a seguir mostra como usar CreateTable.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Define schema with columns
  DATA(lt_columns) = VALUE /aws1/
cl_kyscolumndefinition=>tt_columndefinitionlist(
  ( NEW /aws1/cl_kyscolumndefinition( iv_name = 'title' iv_type = 'text' ) )
  ( NEW /aws1/cl_kyscolumndefinition( iv_name = 'year' iv_type = 'int' ) )
  ( NEW /aws1/cl_kyscolumndefinition( iv_name = 'release_date' iv_type =
'timestamp' ) )
  ( NEW /aws1/cl_kyscolumndefinition( iv_name = 'plot' iv_type = 'text' ) )
  ).

  " Define partition keys
  DATA(lt_partition_keys) = VALUE /aws1/
cl_kyspartitionkey=>tt_partitionkeylist(
  ( NEW /aws1/cl_kyspartitionkey( iv_name = 'year' ) )
  ( NEW /aws1/cl_kyspartitionkey( iv_name = 'title' ) )
  ).

  " Create schema definition
  DATA(lo_schema) = NEW /aws1/cl_kysschemadefinition(
    it_allcolumns = lt_columns
    it_partitionkeys = lt_partition_keys ).

  " Enable point-in-time recovery
  DATA(lo_pitr) = NEW /aws1/cl_kyspointintimerec(
    iv_status = 'ENABLED' ).

  oo_result = lo_kys->createtable(
    iv_keyspacename = iv_keyspace_name
    iv_tablename = iv_table_name
    io_schemadefinition = lo_schema
    io_pointintimerecovery = lo_pitr ).
  MESSAGE 'Table created successfully.' TYPE 'I'.
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateTable](#) referência da API AWS SDK for SAP ABAP.

## DeleteKeyspace

O código de exemplo a seguir mostra como usar DeleteKeyspace.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_kys->deletekeyspace(  
    iv_keyspacename = iv_keyspace_name ).  
  MESSAGE 'Keyspace deleted successfully.' TYPE 'I'.  
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
  ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteKeyspace](#) referência da API AWS SDK for SAP ABAP.

## DeleteTable

O código de exemplo a seguir mostra como usar DeleteTable.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

lo_kys->deletetable(
  iv_keyspacename = iv_keyspace_name
  iv_tablename = iv_table_name ).
MESSAGE 'Table deleted successfully.' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteTable](#) referência da API AWS SDK for SAP ABAP.

## GetKeyspace

O código de exemplo a seguir mostra como usar GetKeyspace.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  oo_result = lo_kys->getkeyspace(
    iv_keyspacename = iv_keyspace_name ).
  MESSAGE 'Keyspace retrieved successfully.' TYPE 'I'.
  CATCH /aws1/cx_kysresourcenotfoundex.
    MESSAGE 'Keyspace does not exist.' TYPE 'I'.
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetKeyspace](#) referência da API AWS SDK for SAP ABAP.

## GetTable

O código de exemplo a seguir mostra como usar GetTable.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kys->gettable(  
        iv_keyspacename = iv_keyspace_name  
        iv_tablename = iv_table_name ).  
    MESSAGE 'Table information retrieved successfully.' TYPE 'I'.  
CATCH /aws1/cx_kysresourcenotfoundex.  
    MESSAGE 'Table does not exist.' TYPE 'I'.  
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetTable](#) referência da API AWS SDK for SAP ABAP.

## ListKeyspaces

O código de exemplo a seguir mostra como usar ListKeyspaces.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kys->listkeyspaces(  
        iv_maxresults = iv_max_results ).  
    MESSAGE 'Keyspaces listed successfully.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListKeyspaces](#) referência da API AWS SDK for SAP ABAP.

## ListTables

O código de exemplo a seguir mostra como usar ListTables.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kys->listtables(  
        iv_keyspacename = iv_keyspace_name ).  
    MESSAGE 'Tables listed successfully.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListTables](#) referência da API AWS SDK for SAP ABAP.

## RestoreTable

O código de exemplo a seguir mostra como usar RestoreTable.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kys->restoretale(  
        iv_sourcekeyspacename = iv_source_keyspace_name  
        iv_sourcetablename = iv_source_table_name  
        iv_targetkeyspacename = iv_target_keyspace_name  
        iv_targettablename = iv_target_table_name  
        iv_restoretimestamp = iv_restore_timestamp ).  
    MESSAGE 'Table restore initiated successfully.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
        DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
        MESSAGE lv_error TYPE 'E'.  
    ENDTRY.
```

- Para obter detalhes da API, consulte a [RestoreTable](#) referência da API AWS SDK for SAP ABAP.

## UpdateTable

O código de exemplo a seguir mostra como usar UpdateTable.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " Add a new column to track watched movies  
    DATA(lt_add_columns) = VALUE /aws1/  
cl_kyscolumndefinition=>tt_columndefinitionlist(  
    ( NEW /aws1/cl_kyscolumndefinition( iv_name = 'watched' iv_type =  
'boolean' ) )  
    ).  
  
    oo_result = lo_kys->updatetable(  
        iv_keyspacename = iv_keyspace_name  
        iv_tablename = iv_table_name  
        it_addcolumns = lt_add_columns ).  
    MESSAGE 'Table updated successfully.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateTable](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Kinesis usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Kinesis.

As noções básicas são exemplos de código que mostram como realizar as operações essenciais em um serviço.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Conceitos básicos](#)

- [Ações](#)

## Conceitos básicos

Conheça os conceitos básicos

O exemplo de código a seguir mostra como:

- Criar um fluxo e inserir um registro nele.
- Criar um iterador de fragmento.
- Ler o registro e, em seguida, limpar os recursos.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lo_stream_describe_result TYPE REF TO /aws1/cl_knsdescrstreamoutput.
DATA lo_stream_description TYPE REF TO /aws1/cl_knsstreamdescription.
DATA lo_sharditerator TYPE REF TO /aws1/cl_knsgetsharditerator01.
DATA lo_record_result TYPE REF TO /aws1/cl_knsputrecordoutput.

"Create stream."
TRY.
    lo_kns->createstream(
        iv_streamname = iv_stream_name
        iv_shardcount = iv_shard_count ).
    MESSAGE 'Stream created.' TYPE 'I'.
CATCH /aws1/cx_knsinvalidargumentex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_knslimitexceededex.
    MESSAGE 'The request processing has failed because of a limit exceeded
exception.' TYPE 'E'.
CATCH /aws1/cx_knsresourceinuseex.
    MESSAGE 'The request processing has failed because the resource is in use.'
TYPE 'E'.
```

```

ENDTRY.

"Wait for stream to becomes active."
lo_stream_describe_result = lo_kns->describestream( iv_streamname =
iv_stream_name ).
lo_stream_description = lo_stream_describe_result->get_streamdescription( ).
WHILE lo_stream_description->get_streamstatus( ) <> 'ACTIVE'.
  IF sy-index = 30.
    EXIT.          "maximum 5 minutes"
  ENDIF.
  WAIT UP TO 10 SECONDS.
  lo_stream_describe_result = lo_kns->describestream( iv_streamname =
iv_stream_name ).
  lo_stream_description = lo_stream_describe_result->get_streamdescription( ).
ENDWHILE.

"Create record."
TRY.
  lo_record_result = lo_kns->putrecord(
    iv_streamname = iv_stream_name
    iv_data        = iv_data
    iv_partitionkey = iv_partition_key ).
  MESSAGE 'Record created.' TYPE 'I'.
CATCH /aws1/cx_knsinvalidargumentex.
  MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_knskmsaccessdeniedex.
  MESSAGE 'You do not have permission to perform this AWS KMS action.' TYPE
'E'.
CATCH /aws1/cx_knskmsdisabledex.
  MESSAGE 'KMS key used is disabled.' TYPE 'E'.
CATCH /aws1/cx_knskmsinvalidstateex.
  MESSAGE 'KMS key used is in an invalid state. ' TYPE 'E'.
CATCH /aws1/cx_knskmsnotfoundex.
  MESSAGE 'KMS key used is not found.' TYPE 'E'.
CATCH /aws1/cx_knskmsoptinrequired.
  MESSAGE 'KMS key option is required.' TYPE 'E'.
CATCH /aws1/cx_knskmsstrottingex.
  MESSAGE 'The rate of requests to AWS KMS is exceeding the request quotas.'
TYPE 'E'.
CATCH /aws1/cx_knsprovthruputexcdex.
  MESSAGE 'The request rate for the stream is too high, or the requested data
is too large for the available throughput.' TYPE 'E'.
CATCH /aws1/cx_knsresourcenotfoundex.
  MESSAGE 'Resource being accessed is not found.' TYPE 'E'.

```

```

ENDTRY.

"Create a shard iterator in order to read the record."
TRY.
    lo_sharditerator = lo_kns->getsharditerator(
        iv_shardid = lo_record_result->get_shardid( )
        iv_sharditeratortype = iv_sharditeratortype
        iv_streamname = iv_stream_name ).
    MESSAGE 'Shard iterator created.' TYPE 'I'.
CATCH /aws1/cx_knsinvalidargumentex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_knsprovthruputexcdex.
    MESSAGE 'The request rate for the stream is too high, or the requested data
is too large for the available throughput.' TYPE 'E'.
CATCH /aws1/cx_sgmresourcenotfound.
    MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
ENDTRY.

"Read the record."
TRY.
    oo_result = lo_kns->getrecords(                " oo_result is returned
for testing purposes. "
        iv_sharditerator = lo_sharditerator->get_sharditerator( ) ).
    MESSAGE 'Shard iterator created.' TYPE 'I'.
CATCH /aws1/cx_knsexpirediteratorex.
    MESSAGE 'Iterator expired.' TYPE 'E'.
CATCH /aws1/cx_knsinvalidargumentex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_knskmsaccessdeniedex.
    MESSAGE 'You do not have permission to perform this AWS KMS action.' TYPE
'E'.
CATCH /aws1/cx_knskmsdisabledex.
    MESSAGE 'KMS key used is disabled.' TYPE 'E'.
CATCH /aws1/cx_knskmsinvalidstateex.
    MESSAGE 'KMS key used is in an invalid state. ' TYPE 'E'.
CATCH /aws1/cx_knskmsnotfoundex.
    MESSAGE 'KMS key used is not found.' TYPE 'E'.
CATCH /aws1/cx_knskmsoptinrequired.
    MESSAGE 'KMS key option is required.' TYPE 'E'.
CATCH /aws1/cx_knskmssthrrottlingex.
    MESSAGE 'The rate of requests to AWS KMS is exceeding the request quotas.'
TYPE 'E'.
CATCH /aws1/cx_knsprovthruputexcdex.

```

```
MESSAGE 'The request rate for the stream is too high, or the requested data
is too large for the available throughput.' TYPE 'E'.
CATCH /aws1/cx_knsresourcenotfoundex.
MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
ENDTRY.

"Delete stream."
TRY.
  lo_kns->deletestream(
    iv_streamname = iv_stream_name ).
  MESSAGE 'Stream deleted.' TYPE 'I'.
  CATCH /aws1/cx_knslimitexceeddex.
  MESSAGE 'The request processing has failed because of a limit exceeded
exception.' TYPE 'E'.
  CATCH /aws1/cx_knsresourceinuseex.
  MESSAGE 'The request processing has failed because the resource is in use.'
TYPE 'E'.
ENDTRY.
```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [CreateStream](#)
  - [DeleteStream](#)
  - [GetRecords](#)
  - [GetShardIterator](#)
  - [PutRecord](#)

## Ações

### CreateStream

O código de exemplo a seguir mostra como usar CreateStream.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_kns->createstream(  
    iv_streamname = iv_stream_name  
    iv_shardcount = iv_shard_count ).  
  MESSAGE 'Stream created.' TYPE 'I'.  
CATCH /aws1/cx_knsinvalidargumentex.  
  MESSAGE 'The specified argument was not valid.' TYPE 'E'.  
CATCH /aws1/cx_knslimitexceeddex.  
  MESSAGE 'The request processing has failed because of a limit exceed  
exception.' TYPE 'E'.  
CATCH /aws1/cx_knsresourceinuseex.  
  MESSAGE 'The request processing has failed because the resource is in use.'  
TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateStream](#) referência da API AWS SDK for SAP ABAP.

## DeleteStream

O código de exemplo a seguir mostra como usar DeleteStream.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

lo_kns->deletestream(
    iv_streamname = iv_stream_name ).
MESSAGE 'Stream deleted.' TYPE 'I'.
CATCH /aws1/cx_knslimitexceeddex.
    MESSAGE 'The request processing has failed because of a limit exceed
exception.' TYPE 'E'.
CATCH /aws1/cx_knsresourceinuseex.
    MESSAGE 'The request processing has failed because the resource is in use.'
TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteStream](#) referência da API AWS SDK for SAP ABAP.

## DescribeStream

O código de exemplo a seguir mostra como usar DescribeStream.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_kns->describestream(
        iv_streamname = iv_stream_name ).
    DATA(lt_stream_description) = oo_result->get_streamdescription( ).
    MESSAGE 'Streams retrieved.' TYPE 'I'.
CATCH /aws1/cx_knslimitexceeddex.
    MESSAGE 'The request processing has failed because of a limit exceed
exception.' TYPE 'E'.
CATCH /aws1/cx_knsresourceinuseex.
    MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeStream](#) referência da API AWS SDK for SAP ABAP.

## GetRecords

O código de exemplo a seguir mostra como usar GetRecords.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_kns->getrecords(           " oo_result is returned for
testing purposes. "
        iv_sharditerator = iv_shard_iterator ).
    DATA(lt_records) = oo_result->get_records( ).
    MESSAGE 'Record retrieved.' TYPE 'I'.
CATCH /aws1/cx_knsexpirediteratorex.
    MESSAGE 'Iterator expired.' TYPE 'E'.
CATCH /aws1/cx_knsinvalidargumentex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_knskmsaccessdeniedex.
    MESSAGE 'You do not have permission to perform this AWS KMS action.' TYPE
'E'.
CATCH /aws1/cx_knskmsdisabledex.
    MESSAGE 'KMS key used is disabled.' TYPE 'E'.
CATCH /aws1/cx_knskmsinvalidstateex.
    MESSAGE 'KMS key used is in an invalid state. ' TYPE 'E'.
CATCH /aws1/cx_knskmsnotfoundex.
    MESSAGE 'KMS key used is not found.' TYPE 'E'.
CATCH /aws1/cx_knskmsoptinrequired.
    MESSAGE 'KMS key option is required.' TYPE 'E'.
CATCH /aws1/cx_knskms throttlingex.
    MESSAGE 'The rate of requests to AWS KMS is exceeding the request quotas.'
TYPE 'E'.
CATCH /aws1/cx_knsprovthruputexcdex.
    MESSAGE 'The request rate for the stream is too high, or the requested data
is too large for the available throughput.' TYPE 'E'.
```

```
CATCH /aws1/cx_knsresourcefoundex.  
    MESSAGE 'Resource being accessed is not found.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetRecords](#) referência da API AWS SDK for SAP ABAP.

## ListStreams

O código de exemplo a seguir mostra como usar ListStreams.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_kns->liststreams(          " oo_result is returned for testing  
purposes. "  
    "Set Limit to specify that a maximum of streams should be returned."  
    iv_limit = iv_limit ).  
    DATA(lt_streams) = oo_result->get_streamnames( ).  
    MESSAGE 'Streams listed.' TYPE 'I'.  
    CATCH /aws1/cx_knslimitexceedex.  
    MESSAGE 'The request processing has failed because of a limit exceed  
exception.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListStreams](#) referência da API AWS SDK for SAP ABAP.

## PutRecord

O código de exemplo a seguir mostra como usar PutRecord.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_kns->putrecord(           " oo_result is returned for
testing purposes. "
        iv_streamname = iv_stream_name
        iv_data       = iv_data
        iv_partitionkey = iv_partition_key ).
    MESSAGE 'Record created.' TYPE 'I'.
CATCH /aws1/cx_knsinvalidargumentex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_knskmsaccessdeniedex.
    MESSAGE 'You do not have permission to perform this AWS KMS action.' TYPE
'E'.
CATCH /aws1/cx_knskmsdisabledex.
    MESSAGE 'KMS key used is disabled.' TYPE 'E'.
CATCH /aws1/cx_knskmsinvalidstateex.
    MESSAGE 'KMS key used is in an invalid state. ' TYPE 'E'.
CATCH /aws1/cx_knskmsnotfoundex.
    MESSAGE 'KMS key used is not found.' TYPE 'E'.
CATCH /aws1/cx_knskmsoptinrequired.
    MESSAGE 'KMS key option is required.' TYPE 'E'.
CATCH /aws1/cx_knskmssthrrottlingex.
    MESSAGE 'The rate of requests to AWS KMS is exceeding the request quotas.'
TYPE 'E'.
CATCH /aws1/cx_knsprovthruputexcdex.
    MESSAGE 'The request rate for the stream is too high, or the requested data
is too large for the available throughput.' TYPE 'E'.
CATCH /aws1/cx_knsresourcenotfoundex.
    MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutRecord](#) referência da API AWS SDK for SAP ABAP.

## RegisterStreamConsumer

O código de exemplo a seguir mostra como usar RegisterStreamConsumer.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_kns->registerstreamconsumer(      " oo_result is returned
for testing purposes. "
        iv_streamarn = iv_stream_arn
        iv_consumername = iv_consumer_name ).
    MESSAGE 'Stream consumer registered.' TYPE 'I'.
CATCH /aws1/cx_knsinvalidargumentex.
    MESSAGE 'The specified argument was not valid.' TYPE 'E'.
CATCH /aws1/cx_sgmresourcecelimitexcd.
    MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
CATCH /aws1/cx_sgmresourceinuse.
    MESSAGE 'Resource being accessed is in use.' TYPE 'E'.
CATCH /aws1/cx_sgmresourcenotfound.
    MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [RegisterStreamConsumer](#) referência da API AWS SDK for SAP ABAP.

## AWS KMS exemplos usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com. AWS KMS

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Conceitos básicos](#)
- [Ações](#)

## Conceitos básicos

### Olá AWS KMS

O exemplo de código a seguir mostra como começar a usar o AWS Key Management Service.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kms->listkeys( ).  
    MESSAGE 'Retrieved KMS keys list.' TYPE 'I'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```


- Para obter detalhes da API, consulte a [ListKeys](#) referência da API AWS SDK for SAP ABAP.

## Ações

### CreateAlias

O código de exemplo a seguir mostra como usar CreateAlias.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  " iv_alias_name = 'alias/my-key-alias'
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  lo_kms->createalias(
    iv_aliasname = iv_alias_name
    iv_targetkeyid = iv_key_id
  ).
  MESSAGE 'Alias created successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsalreadyexistsex.
  MESSAGE 'Alias already exists.' TYPE 'E'.
CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmsinvalidaliasnameex.
  MESSAGE 'Invalid alias name.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateAlias](#) referência da API AWS SDK for SAP ABAP.

## CreateGrant

O código de exemplo a seguir mostra como usar CreateGrant.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  " iv_grantee_principal = 'arn:aws:iam::123456789012:role/my-role'
  " it_operations contains 'Encrypt', 'Decrypt', 'GenerateDataKey'
  oo_result = lo_kms->creategrant(
    iv_keyid = iv_key_id
    iv_granteeprincipal = iv_grantee_principal
    it_operations = it_operations
  ).
  MESSAGE 'Grant created successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsdisabledexception.
  MESSAGE 'The key is disabled.' TYPE 'E'.
CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateGrant](#) referência da API AWS SDK for SAP ABAP.

## CreateKey

O código de exemplo a seguir mostra como usar CreateKey.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_description = 'Created by the AWS SDK for SAP ABAP'
  oo_result = lo_kms->createkey( iv_description = iv_description ).
  MESSAGE 'KMS key created successfully.' TYPE 'I'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
CATCH /aws1/cx_kmslimitexceeddex.
```

```
MESSAGE 'Limit exceeded for KMS resources.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateKey](#) referência da API AWS SDK for SAP ABAP.

## Decrypt

O código de exemplo a seguir mostra como usar Decrypt.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    " iv_ciphertext_blob contains the encrypted data  
    oo_result = lo_kms->decrypt(  
        iv_keyid = iv_key_id  
        iv_ciphertextblob = iv_ciphertext_blob  
    ).  
    MESSAGE 'Text decrypted successfully.' TYPE 'I'.  
    CATCH /aws1/cx_kmsdisabledexception.  
        MESSAGE 'The key is disabled.' TYPE 'E'.  
    CATCH /aws1/cx_kmsincorrectkeyex.  
        MESSAGE 'Incorrect key for decryption.' TYPE 'E'.  
    CATCH /aws1/cx_kmsnotfoundexception.  
        MESSAGE 'Key not found.' TYPE 'E'.  
    CATCH /aws1/cx_kmskmsinternalex.  
        MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a referência da API [Decrypt](#) in AWS SDK for SAP ABAP.

## DeleteAlias

O código de exemplo a seguir mostra como usar DeleteAlias.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_alias_name = 'alias/my-key-alias'  
    lo_kms->deletealias( iv_aliasname = iv_alias_name ).  
    MESSAGE 'Alias deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_kmsnotfoundexception.  
    MESSAGE 'Alias not found.' TYPE 'E'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteAlias](#) referência da API AWS SDK for SAP ABAP.

## DescribeKey

O código de exemplo a seguir mostra como usar DescribeKey.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
```

```
oo_result = lo_kms->describekey( iv_keyid = iv_key_id ).
DATA(lo_key) = oo_result->get_keymetadata( ).
MESSAGE 'Retrieved key information successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsnotfoundexception.
MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeKey](#) referência da API AWS SDK for SAP ABAP.

## DisableKey

O código de exemplo a seguir mostra como usar `DisableKey`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  lo_kms->disablekey( iv_keyid = iv_key_id ).
  MESSAGE 'KMS key disabled successfully.' TYPE 'I'.
  CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
  CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DisableKey](#) referência da API AWS SDK for SAP ABAP.

## EnableKey

O código de exemplo a seguir mostra como usar `EnableKey`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    lo_kms->enablekey( iv_keyid = iv_key_id ).  
    MESSAGE 'KMS key enabled successfully.' TYPE 'I'.  
CATCH /aws1/cx_kmsnotfoundexception.  
    MESSAGE 'Key not found.' TYPE 'E'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [EnableKey](#) referência da API AWS SDK for SAP ABAP.

## EnableKeyRotation

O código de exemplo a seguir mostra como usar `EnableKeyRotation`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

    " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
    lo_kms->enablekeyrotation( iv_keyid = iv_key_id ).
    MESSAGE 'Key rotation enabled successfully.' TYPE 'I'.
    CATCH /aws1/cx_kmsdisabledexception.
    MESSAGE 'The key is disabled.' TYPE 'E'.
    CATCH /aws1/cx_kmsnotfoundexception.
    MESSAGE 'Key not found.' TYPE 'E'.
    CATCH /aws1/cx_kmsunsupportedopex.
    MESSAGE 'Operation not supported for this key.' TYPE 'E'.
    CATCH /aws1/cx_kmskmsinternalex.
    MESSAGE 'An internal error occurred.' TYPE 'E'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [EnableKeyRotation](#) referência da API AWS SDK for SAP ABAP.

## Encrypt

O código de exemplo a seguir mostra como usar Encrypt.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

  TRY.
    " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
    " iv_plaintext contains the data to encrypt
    oo_result = lo_kms->encrypt(
      iv_keyid = iv_key_id
      iv_plaintext = iv_plaintext
    ).
    MESSAGE 'Text encrypted successfully.' TYPE 'I'.
    CATCH /aws1/cx_kmsdisabledexception.
    MESSAGE 'The key is disabled.' TYPE 'E'.
  TRY.

```

```

CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte [Criptografar](#) no AWS SDK para referência da API SAP ABAP.

## GenerateDataKey

O código de exemplo a seguir mostra como usar GenerateDataKey.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  " iv_keyspec = 'AES_256'
  oo_result = lo_kms->generatedatakey(
    iv_keyid = iv_key_id
    iv_keyspec = 'AES_256'
  ).
  MESSAGE 'Data key generated successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsdisabledexception.
  MESSAGE 'The key is disabled.' TYPE 'E'.
CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GenerateDataKey](#) referência da API AWS SDK for SAP ABAP.

## GetKeyPolicy

O código de exemplo a seguir mostra como usar GetKeyPolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    oo_result = lo_kms->getkeypolicy(  
        iv_keyid = iv_key_id  
        iv_policyname = 'default'  
    ).  
    MESSAGE 'Retrieved key policy successfully.' TYPE 'I'.  
CATCH /aws1/cx_kmsnotfoundexception.  
    MESSAGE 'Key not found.' TYPE 'E'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetKeyPolicy](#) referência da API AWS SDK for SAP ABAP.

## ListAliases

O código de exemplo a seguir mostra como usar ListAliases.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_kms->listalias( ).  
    MESSAGE 'Retrieved KMS aliases list.' TYPE 'I'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListAliases](#) referência da API AWS SDK for SAP ABAP.

## ListGrants

O código de exemplo a seguir mostra como usar ListGrants.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    oo_result = lo_kms->listgrants( iv_keyid = iv_key_id ).  
    MESSAGE 'Retrieved grants list.' TYPE 'I'.  
CATCH /aws1/cx_kmsnotfoundexception.  
    MESSAGE 'Key not found.' TYPE 'E'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListGrants](#) referência da API AWS SDK for SAP ABAP.

## ListKeyPolicies

O código de exemplo a seguir mostra como usar ListKeyPolicies.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    oo_result = lo_kms->listkeypolicies( iv_keyid = iv_key_id ).  
    MESSAGE 'Retrieved key policies list.' TYPE 'I'.  
CATCH /aws1/cx_kmsnotfoundexception.  
    MESSAGE 'Key not found.' TYPE 'E'.  
CATCH /aws1/cx_kmskmsinternalex.  
    MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListKeyPolicies](#) referência da API AWS SDK for SAP ABAP.

## ListKeys

O código de exemplo a seguir mostra como usar ListKeys.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_kms->listkeys( ).  
    MESSAGE 'Retrieved KMS keys list.' TYPE 'I'.  
    CATCH /aws1/cx_kmskmsinternalex.  
        MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListKeys](#) referência da API AWS SDK for SAP ABAP.

**PutKeyPolicy**

O código de exemplo a seguir mostra como usar PutKeyPolicy.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    " iv_policy = '{"Version":"2012-10-17",          "Statement": [...]}'  
    lo_kms->putkeypolicy(  
        iv_keyid = iv_key_id  
        iv_policyname = 'default'  
        iv_policy = iv_policy  
    ).  
    MESSAGE 'Key policy updated successfully.' TYPE 'I'.
```

```

CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmsmalformedpolicydocex.
  MESSAGE 'Malformed policy document.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [PutKeyPolicy](#) referência da API AWS SDK for SAP ABAP.

## ReEncrypt

O código de exemplo a seguir mostra como usar ReEncrypt.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " iv_source_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  " iv_destination_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/5678dcba-56cd-78ef-90ab-5678901234cd'
  " iv_ciphertext_blob contains the encrypted data
  oo_result = lo_kms->reencrypt(
    iv_sourcekeyid = iv_source_key_id
    iv_destinationkeyid = iv_destination_key_id
    iv_ciphertextblob = iv_ciphertext_blob
  ).
  MESSAGE 'Ciphertext reencrypted successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsdisabledexception.
  MESSAGE 'The key is disabled.' TYPE 'E'.
CATCH /aws1/cx_kmsincorrectkeyex.
  MESSAGE 'Incorrect source key for decryption.' TYPE 'E'.
CATCH /aws1/cx_kmsnotfoundexception.

```

```
MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ReEncrypt](#) referência da API AWS SDK for SAP ABAP.

## RetireGrant

O código de exemplo a seguir mostra como usar RetireGrant.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  " iv_grant_token = 'AQpAM2RhZ...'
  lo_kms->retiregrant( iv_granttoken = iv_grant_token ).
  MESSAGE 'Grant retired successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Grant not found.' TYPE 'E'.
CATCH /aws1/cx_kmsinvgranttokenex.
  MESSAGE 'Invalid grant token.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [RetireGrant](#) referência da API AWS SDK for SAP ABAP.

## RevokeGrant

O código de exemplo a seguir mostra como usar RevokeGrant.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    " iv_key_id = 'arn:aws:kms:us-  
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'  
    " iv_grant_id = '1a2b3c4d5e6f7g8h9i0j1k2l3m4n5o6p'  
    lo_kms->revokegrant(  
        iv_keyid = iv_key_id  
        iv_grantid = iv_grant_id  
    ).  
    MESSAGE 'Grant revoked successfully.' TYPE 'I'.  
    CATCH /aws1/cx_kmsnotfoundexception.  
        MESSAGE 'Grant or key not found.' TYPE 'E'.  
    CATCH /aws1/cx_kmsinvalidgrantidex.  
        MESSAGE 'Invalid grant ID.' TYPE 'E'.  
    CATCH /aws1/cx_kmskmsinternalex.  
        MESSAGE 'An internal error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [RevokeGrant](#) referência da API AWS SDK for SAP ABAP.

## ScheduleKeyDeletion

O código de exemplo a seguir mostra como usar ScheduleKeyDeletion.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  " iv_pending_window_days = 7
  oo_result = lo_kms->schedulekeydeletion(
    iv_keyid = iv_key_id
    iv_pendingwindowdays = iv_pending_window_days
  ).
  MESSAGE 'Key scheduled for deletion.' TYPE 'I'.
CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ScheduleKeyDeletion](#) referência da API AWS SDK for SAP ABAP.

## Sign

O código de exemplo a seguir mostra como usar Sign.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab' (asymmetric key)
  " iv_message contains the message to sign
  " iv_signing_algorithm = 'RSASSA_PSS_SHA_256'
  oo_result = lo_kms->sign(
    iv_keyid = iv_key_id
    iv_message = iv_message
    iv_signingalgorithm = iv_signing_algorithm
  ).
```

```
MESSAGE 'Message signed successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsdisabledexception.
MESSAGE 'The key is disabled.' TYPE 'E'.
CATCH /aws1/cx_kmsnotfoundexception.
MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmsinvalidkeyusageex.
MESSAGE 'Key cannot be used for signing.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a referência da API [Sign](#) in AWS SDK for SAP ABAP.

## TagResource

O código de exemplo a seguir mostra como usar TagResource.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_tags TYPE /aws1/cl_kmstag=>tt_taglist.

TRY.
  " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab'
  " iv_tag_key = 'Environment'
  " iv_tag_value = 'Production'
  APPEND NEW /aws1/cl_kmstag(
    iv_tagkey = iv_tag_key
    iv_tagvalue = iv_tag_value
  ) TO lt_tags.

  lo_kms->tagresource(
    iv_keyid = iv_key_id
    it_tags = lt_tags
  ).
```

```
MESSAGE 'Tag added to KMS key successfully.' TYPE 'I'.
CATCH /aws1/cx_kmsnotfoundexception.
MESSAGE 'Key not found.' TYPE 'E'.
CATCH /aws1/cx_kmststageexception.
MESSAGE 'Invalid tag format.' TYPE 'E'.
CATCH /aws1/cx_kmskmsinternalex.
MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [TagResource](#) referência da API AWS SDK for SAP ABAP.

## UpdateAlias

O código de exemplo a seguir mostra como usar UpdateAlias.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_alias_name = 'alias/my-key-alias'
  " iv_target_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/5678dcba-56cd-78ef-90ab-5678901234cd'
  lo_kms->updatealias(
    iv_aliasname = iv_alias_name
    iv_targetkeyid = iv_target_key_id
  ).
  MESSAGE 'Alias updated successfully.' TYPE 'I'.
  CATCH /aws1/cx_kmsnotfoundexception.
  MESSAGE 'Alias or key not found.' TYPE 'E'.
  CATCH /aws1/cx_kmskmsinternalex.
  MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateAlias](#) referência da API AWS SDK for SAP ABAP.

## Verify

O código de exemplo a seguir mostra como usar Verify.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    " iv_key_id = 'arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab' (asymmetric key)
    " iv_message contains the original message
    " iv_signature contains the signature to verify
    " iv_signing_algorithm = 'RSASSA_PSS_SHA_256'
    oo_result = lo_kms->verify(
        iv_keyid = iv_key_id
        iv_message = iv_message
        iv_signature = iv_signature
        iv_signingalgorithm = iv_signing_algorithm
    ).
    DATA(lv_valid) = oo_result->get_signaturevalid( ).
    IF lv_valid = abap_true.
        MESSAGE 'Signature is valid.' TYPE 'I'.
    ELSE.
        MESSAGE 'Signature is invalid.' TYPE 'I'.
    ENDIF.
    CATCH /aws1/cx_kmsdisabledexception.
        MESSAGE 'The key is disabled.' TYPE 'E'.
    CATCH /aws1/cx_kmsnotfoundexception.
        MESSAGE 'Key not found.' TYPE 'E'.
    CATCH /aws1/cx_kmskmsinvalidsigex.
        MESSAGE 'Invalid signature.' TYPE 'E'.
    CATCH /aws1/cx_kmskmsinternalex.
        MESSAGE 'An internal error occurred.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte [Verificar](#) no AWS SDK para referência da API SAP ABAP.

## Exemplos do Lambda usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Lambda.

As noções básicas são exemplos de código que mostram como realizar as operações essenciais em um serviço.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Conceitos básicos](#)
- [Ações](#)

## Conceitos básicos


Conheça os conceitos básicos

O exemplo de código a seguir mostra como:

- Criar um perfil do IAM e uma função do Lambda e carregar o código de manipulador.
- Invocar essa função com um único parâmetro e receber resultados.
- Atualizar o código de função e configurar usando uma variável de ambiente.
- Invocar a função com novos parâmetros e receber resultados. Exibir o log de execução retornado.
- Listar as funções para sua conta e limpar os recursos.

Para obter mais informações, consulte [Criar uma função do Lambda no console](#).

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  "Create an AWS Identity and Access Management (IAM) role that grants AWS
  Lambda permission to write to logs."
  DATA(lv_policy_document) = `{` &&
    ` "Version": "2012-10-17",` &&
    ` "Statement": [` &&
      `{` &&
        ` "Effect": "Allow",` &&
        ` "Action": [` &&
          ` "sts:AssumeRole"` &&
        ` ],` &&
        ` "Principal": {` &&
          ` "Service": [` &&
            ` "lambda.amazonaws.com"` &&
          ` ]` &&
        ` }` &&
      ` }` &&
    ` ]` &&
  `}`.

TRY.
  DATA(lo_create_role_output) = lo_iam->createrole(
    iv_rolename = iv_role_name
    iv_assumerolepolicydocument = lv_policy_document
    iv_description = 'Grant lambda permission to write to logs' ).
  DATA(lv_role_arn) = lo_create_role_output->get_role( )->get_arn( ).
  MESSAGE 'IAM role created.' TYPE 'I'.
  WAIT UP TO 10 SECONDS. " Make sure that the IAM role is ready
  for use. "
  CATCH /aws1/cx_iamentityalrдыexex.
    DATA(lo_role) = lo_iam->getrole( iv_rolename = iv_role_name ).
    lv_role_arn = lo_role->get_role( )->get_arn( ).
  CATCH /aws1/cx_iaminvalidinputex.
    MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.

```

```

    CATCH /aws1/cx_iammalformedplydocex.
      MESSAGE 'Policy document in the request is malformed.' TYPE 'E'.
    ENDTRY.

    TRY.
      lo_iam->attachrolepolicy(
        iv_rolename = iv_role_name
        iv_policyarn = 'arn:aws:iam::aws:policy/service-role/
AWSLambdaBasicExecutionRole' ).
      MESSAGE 'Attached policy to the IAM role.' TYPE 'I'.
    CATCH /aws1/cx_iaminvalidinputex.
      MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
    CATCH /aws1/cx_iamnosuchentityex.
      MESSAGE 'The requested resource entity does not exist.' TYPE 'E'.
    CATCH /aws1/cx_iamplynnotattachableex.
      MESSAGE 'Service role policies can only be attached to the service-
linked role for their service.' TYPE 'E'.
    CATCH /aws1/cx_iamunmodableentityex.
      MESSAGE 'Service that depends on the service-linked role is not
modifiable.' TYPE 'E'.
    ENDTRY.

    " Create a Lambda function and upload handler code. "
    " Lambda function performs 'increment' action on a number. "
    TRY.
      lo_lmd->createfunction(
        iv_functionname = iv_function_name
        iv_runtime = `python3.9`
        iv_role = lv_role_arn
        iv_handler = iv_handler
        io_code = io_initial_zip_file
        iv_description = 'AWS Lambda code example' ).
      MESSAGE 'Lambda function created.' TYPE 'I'.
    CATCH /aws1/cx_lmdcodestorageexcex.
      MESSAGE 'Maximum total code size per account exceeded.' TYPE 'E'.
    CATCH /aws1/cx_lmdinvparamvalueex.
      MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
    CATCH /aws1/cx_lmdresourcenotfoundex.
      MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    ENDTRY.

    " Verify the function is in Active state "
    WHILE lo_lmd->getfunction( iv_functionname = iv_function_name )-
>get_configuration( )->ask_state( ) <> 'Active'.

```

```

    IF sy-index = 10.
      EXIT.          " Maximum 10 seconds. "
    ENDIF.
    WAIT UP TO 1 SECONDS.
  ENDWHILE.

  "Invoke the function with a single parameter and get results."
  TRY.
    DATA(lv_json) = /aws1/cl_rt_util=>string_to_xstring(
      `{`  &&
      `  "action": "increment",`  &&
      `  "number": 10`  &&
      `}` ).
    DATA(lo_initial_invoke_output) = lo_lmd->invoke(
      iv_functionname = iv_function_name
      iv_payload = lv_json ).
    ov_initial_invoke_payload = lo_initial_invoke_output->get_payload( ).
    " ov_initial_invoke_payload is returned for testing purposes. "
    DATA(lo_writer_json) = cl_sxml_string_writer=>create( type =
if_sxml=>co_xt_json ).
    CALL TRANSFORMATION id SOURCE XML ov_initial_invoke_payload RESULT XML
lo_writer_json.
    DATA(lv_result) = cl_abap_codepage=>convert_from( lo_writer_json-
>get_output( ) ).
    MESSAGE 'Lambda function invoked.' TYPE 'I'.
    CATCH /aws1/cx_lmdinvparamvalueex.
      MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
    CATCH /aws1/cx_lmdinvrequestcontex.
      MESSAGE 'Unable to parse request body as JSON.' TYPE 'E'.
    CATCH /aws1/cx_lmdresourcenotfoundex.
      MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    CATCH /aws1/cx_lmdunsuppmediatyp00.
      MESSAGE 'Invoke request body does not have JSON as its content type.'
TYPE 'E'.
  ENDTRY.

  " Update the function code and configure its Lambda environment with an
environment variable. "
  " Lambda function is updated to perform 'decrement' action also. "
  TRY.
    lo_lmd->updatefunctioncode(
      iv_functionname = iv_function_name
      iv_zipfile = io_updated_zip_file ).

```

```

        WAIT UP TO 10 SECONDS.           " Make sure that the update is
completed. "
        MESSAGE 'Lambda function code updated.' TYPE 'I'.
        CATCH /aws1/cx_lmdcodestorageexcde.
        MESSAGE 'Maximum total code size per account exceeded.' TYPE 'E'.
        CATCH /aws1/cx_lmdinvparamvalueex.
        MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
        CATCH /aws1/cx_lmdresourcenotfoundex.
        MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    ENENTRY.

    TRY.
        DATA lt_variables TYPE /aws1/
cl_lmdenvironmentvaria00=>tt_environmentvariables.
        DATA ls_variable LIKE LINE OF lt_variables.
        ls_variable-key = 'LOG_LEVEL'.
        ls_variable-value = NEW /aws1/cl_lmdenvironmentvaria00( iv_value =
'info' ).
        INSERT ls_variable INTO TABLE lt_variables.

        lo_lmd->updatefunctionconfiguration(
            iv_functionname = iv_function_name
            io_environment = NEW /aws1/cl_lmdenvironment( it_variables =
lt_variables ) ).
        WAIT UP TO 10 SECONDS.           " Make sure that the update is
completed. "
        MESSAGE 'Lambda function configuration/settings updated.' TYPE 'I'.
        CATCH /aws1/cx_lmdinvparamvalueex.
        MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
        CATCH /aws1/cx_lmdresourceconflictex.
        MESSAGE 'Resource already exists or another operation is in progress.'
TYPE 'E'.
        CATCH /aws1/cx_lmdresourcenotfoundex.
        MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    ENENTRY.

    "Invoke the function with new parameters and get results. Display the
execution log that's returned from the invocation."
    TRY.
        lv_json = /aws1/cl_rt_util=>string_to_xstring(
            `{` ` &&
            `"action": "decrement",` ` &&
            `"number": 10` ` &&
            `}` ).

```

```

        DATA(lo_updated_invoke_output) = lo_lmd->invoke(
            iv_functionname = iv_function_name
            iv_payload = lv_json ).
        ov_updated_invoke_payload = lo_updated_invoke_output->get_payload( ).
        " ov_updated_invoke_payload is returned for testing purposes. "
        lo_writer_json = cl_sxml_string_writer=>create( type =
if_sxml=>co_xt_json ).
        CALL TRANSFORMATION id SOURCE XML ov_updated_invoke_payload RESULT XML
lo_writer_json.
        lv_result = cl_abap_codepage=>convert_from( lo_writer_json-
>get_output( ) ).
        MESSAGE 'Lambda function invoked.' TYPE 'I'.
        CATCH /aws1/cx_lmdinvparamvalueex.
        MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
        CATCH /aws1/cx_lmdinvrequestcontex.
        MESSAGE 'Unable to parse request body as JSON.' TYPE 'E'.
        CATCH /aws1/cx_lmdresourcenotfoundex.
        MESSAGE 'The requested resource does not exist.' TYPE 'E'.
        CATCH /aws1/cx_lmdunsuppmediatyp00.
        MESSAGE 'Invoke request body does not have JSON as its content type.'
TYPE 'E'.
        ENDMETHOD.

        " List the functions for your account. "
        TRY.
            DATA(lo_list_output) = lo_lmd->listfunctions( ).
            DATA(lt_functions) = lo_list_output->get_functions( ).
            MESSAGE 'Retrieved list of Lambda functions.' TYPE 'I'.
            CATCH /aws1/cx_lmdinvparamvalueex.
            MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
        ENDMETHOD.

        " Delete the Lambda function. "
        TRY.
            lo_lmd->deletefunction( iv_functionname = iv_function_name ).
            MESSAGE 'Lambda function deleted.' TYPE 'I'.
            CATCH /aws1/cx_lmdinvparamvalueex.
            MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
            CATCH /aws1/cx_lmdresourcenotfoundex.
            MESSAGE 'The requested resource does not exist.' TYPE 'W'.
        ENDMETHOD.

        " Detach role policy. "
        TRY.

```

```
lo_iam->detachrolepolicy(  
    iv_rolename = iv_role_name  
    iv_policyarn = 'arn:aws:iam::aws:policy/service-role/  
AWSLambdaBasicExecutionRole' ).  
    MESSAGE 'Detached policy from the IAM role.' TYPE 'I'.  
CATCH /aws1/cx_iaminvalidinputex.  
    MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'The requested resource entity does not exist.' TYPE 'W'.  
CATCH /aws1/cx_iamplynottattachableex.  
    MESSAGE 'Service role policies can only be attached to the service-  
linked role for their service.' TYPE 'E'.  
CATCH /aws1/cx_iamunmodableentityex.  
    MESSAGE 'Service that depends on the service-linked role is not  
modifiable.' TYPE 'E'.  
ENDTRY.  
  
" Delete the IAM role. "  
TRY.  
    lo_iam->deleterole( iv_rolename = iv_role_name ).  
    MESSAGE 'IAM role deleted.' TYPE 'I'.  
CATCH /aws1/cx_iamnosuchentityex.  
    MESSAGE 'The requested resource entity does not exist.' TYPE 'W'.  
CATCH /aws1/cx_iamunmodableentityex.  
    MESSAGE 'Service that depends on the service-linked role is not  
modifiable.' TYPE 'E'.  
ENDTRY.  
  
CATCH /aws1/cx_rt_service_generic INTO lo_exception.  
    DATA(lv_error) = lo_exception->get_longtext( ).  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [CreateFunction](#)
  - [DeleteFunction](#)
  - [GetFunction](#)
  - [Invoke](#)
  - [ListFunctions](#)

- [UpdateFunctionCode](#)
- [UpdateFunctionConfiguration](#)

## Ações

### CreateFunction

O código de exemplo a seguir mostra como usar CreateFunction.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  lo_lmd->createfunction(
    iv_functionname = iv_function_name
    iv_runtime = `python3.9`
    iv_role = iv_role_arn
    iv_handler = iv_handler
    io_code = io_zip_file
    iv_description = 'AWS Lambda code example' ).
  MESSAGE 'Lambda function created.' TYPE 'I'.
CATCH /aws1/cx_lmdcodesigningcfgno00.
  MESSAGE 'Code signing configuration does not exist.' TYPE 'E'.
CATCH /aws1/cx_lmdcodestorageexcdex.
  MESSAGE 'Maximum total code size per account exceeded.' TYPE 'E'.
CATCH /aws1/cx_lmdcodeverification00.
  MESSAGE 'Code signature failed one or more validation checks for signature
mismatch or expiration.' TYPE 'E'.
CATCH /aws1/cx_lmdinvalidcodesigex.
  MESSAGE 'Code signature failed the integrity check.' TYPE 'E'.
CATCH /aws1/cx_lmdinvparamvalueex.
  MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
CATCH /aws1/cx_lmdresourceconflictex.
  MESSAGE 'Resource already exists or another operation is in progress.' TYPE
'E'.
CATCH /aws1/cx_lmdresourcenotfoundex.
```

```

    MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    CATCH /aws1/cx_lmdserviceexception.
    MESSAGE 'An internal problem was encountered by the AWS Lambda service.'
TYPE 'E'.
    CATCH /aws1/cx_lmdtoomanyrequestsex.
    MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateFunction](#) referência da API AWS SDK for SAP ABAP.

## DeleteFunction

O código de exemplo a seguir mostra como usar DeleteFunction.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    lo_lmd->deletefunction( iv_functionname = iv_function_name ).
    MESSAGE 'Lambda function deleted.' TYPE 'I'.
    CATCH /aws1/cx_lmdinvparamvalueex.
    MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
    CATCH /aws1/cx_lmdresourceconflictex.
    MESSAGE 'Resource already exists or another operation is in progress.' TYPE
'E'.
    CATCH /aws1/cx_lmdresourcenotfoundex.
    MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    CATCH /aws1/cx_lmdserviceexception.
    MESSAGE 'An internal problem was encountered by the AWS Lambda service.'
TYPE 'E'.
    CATCH /aws1/cx_lmdtoomanyrequestsex.
    MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteFunction](#) referência da API AWS SDK for SAP ABAP.

## GetFunction

O código de exemplo a seguir mostra como usar `GetFunction`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
    oo_result = lo_lmd->getfunction( iv_functionname = iv_function_name ).
" oo_result is returned for testing purposes. "
    MESSAGE 'Lambda function information retrieved.' TYPE 'I'.
    CATCH /aws1/cx_lmdinvparamvalueex.
        MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
    CATCH /aws1/cx_lmdserviceexception.
        MESSAGE 'An internal problem was encountered by the AWS Lambda service.'
TYPE 'E'.
    CATCH /aws1/cx_lmdtoomanyrequestsex.
        MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetFunction](#) referência da API AWS SDK for SAP ABAP.

## Invoke

O código de exemplo a seguir mostra como usar `Invoke`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  DATA(lv_json) = /aws1/cl_rt_util=>string_to_xstring(
    `{`  &&
    `  "action": "increment",`  &&
    `  "number": 10`  &&
    `}` ).
  oo_result = lo_lmd->invoke(
testing purposes. " " oo_result is returned for
    iv_functionname = iv_function_name
    iv_payload = lv_json ).
  MESSAGE 'Lambda function invoked.' TYPE 'I'.
  CATCH /aws1/cx_lmdinvparamvalueex.
    MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
  CATCH /aws1/cx_lmdinvrequestctx.
    MESSAGE 'Unable to parse request body as JSON.' TYPE 'E'.
  CATCH /aws1/cx_lmdinvalidzipfileex.
    MESSAGE 'The deployment package could not be unzipped.' TYPE 'E'.
  CATCH /aws1/cx_lmdrequesttoolargeex.
    MESSAGE 'Invoke request body JSON input limit was exceeded by the request
payload.' TYPE 'E'.
  CATCH /aws1/cx_lmdresourceconflictex.
    MESSAGE 'Resource already exists or another operation is in progress.' TYPE
'E'.
  CATCH /aws1/cx_lmdresourcenotfoundex.
    MESSAGE 'The requested resource does not exist.' TYPE 'E'.
  CATCH /aws1/cx_lmdserviceexception.
    MESSAGE 'An internal problem was encountered by the AWS Lambda service.'
TYPE 'E'.
  CATCH /aws1/cx_lmdtoomanyrequestsex.
    MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.
  CATCH /aws1/cx_lmdunsuppedmediatyp00.
    MESSAGE 'Invoke request body does not have JSON as its content type.' TYPE
'E'.
  ENDTRY.

```

- Para obter os detalhes da API, consulte [Invoke](#) na Referência da API do AWS SDK para SAP ABAP.

## ListFunctions

O código de exemplo a seguir mostra como usar ListFunctions.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_lmd->listfunctions( ).      " oo_result is returned for  
testing purposes. "  
    DATA(lt_functions) = oo_result->get_functions( ).  
    MESSAGE 'Retrieved list of Lambda functions.' TYPE 'I'.  
    CATCH /aws1/cx_lmdinvparamvalueex.  
        MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.  
    CATCH /aws1/cx_lmdserviceexception.  
        MESSAGE 'An internal problem was encountered by the AWS Lambda service.'  
TYPE 'E'.  
    CATCH /aws1/cx_lmdtoomanyrequestsex.  
        MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListFunctions](#) referência da API AWS SDK for SAP ABAP.

## UpdateFunctionCode

O código de exemplo a seguir mostra como usar UpdateFunctionCode.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_lmd->updatefunctioncode(      " oo_result is returned for
testing purposes. "
        iv_functionname = iv_function_name
        iv_zipfile = io_zip_file ).

    MESSAGE 'Lambda function code updated.' TYPE 'I'.
CATCH /aws1/cx_lmdcodesigningcfgno00.
    MESSAGE 'Code signing configuration does not exist.' TYPE 'E'.
CATCH /aws1/cx_lmdcodestorageexcdex.
    MESSAGE 'Maximum total code size per account exceeded.' TYPE 'E'.
CATCH /aws1/cx_lmdcodeverification00.
    MESSAGE 'Code signature failed one or more validation checks for signature
mismatch or expiration.' TYPE 'E'.
CATCH /aws1/cx_lmdinvalidcodesigex.
    MESSAGE 'Code signature failed the integrity check.' TYPE 'E'.
CATCH /aws1/cx_lmdinvparamvalueex.
    MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
CATCH /aws1/cx_lmdresourceconflictex.
    MESSAGE 'Resource already exists or another operation is in progress.' TYPE
'E'.
CATCH /aws1/cx_lmdresourcenotfoundex.
    MESSAGE 'The requested resource does not exist.' TYPE 'E'.
CATCH /aws1/cx_lmdserviceexception.
    MESSAGE 'An internal problem was encountered by the AWS Lambda service.'
TYPE 'E'.
CATCH /aws1/cx_lmdtoomanyrequestsex.
    MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateFunctionCode](#) referência da API AWS SDK for SAP ABAP.

## UpdateFunctionConfiguration

O código de exemplo a seguir mostra como usar UpdateFunctionConfiguration.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_lmd->updatefunctionconfiguration(      " oo_result is returned
for testing purposes. "
        iv_functionname = iv_function_name
        iv_runtime = iv_runtime
        iv_description = 'Updated Lambda function'
        iv_memorysize = iv_memory_size ).

    MESSAGE 'Lambda function configuration/settings updated.' TYPE 'I'.
    CATCH /aws1/cx_lmdcodesigningcfgno00.
    MESSAGE 'Code signing configuration does not exist.' TYPE 'E'.
    CATCH /aws1/cx_lmdcodeverification00.
    MESSAGE 'Code signature failed one or more validation checks for signature
mismatch or expiration.' TYPE 'E'.
    CATCH /aws1/cx_lmdinvalidcodesigex.
    MESSAGE 'Code signature failed the integrity check.' TYPE 'E'.
    CATCH /aws1/cx_lmdinvparamvalueex.
    MESSAGE 'The request contains a non-valid parameter.' TYPE 'E'.
    CATCH /aws1/cx_lmdresourceconflictex.
    MESSAGE 'Resource already exists or another operation is in progress.' TYPE
'E'.
    CATCH /aws1/cx_lmdresourcenotfoundex.
    MESSAGE 'The requested resource does not exist.' TYPE 'E'.
    CATCH /aws1/cx_lmdserviceexception.
    MESSAGE 'An internal problem was encountered by the AWS Lambda service.'
TYPE 'E'.
    CATCH /aws1/cx_lmdtoomanyrequestsex.
    MESSAGE 'The maximum request throughput was reached.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateFunctionConfiguration](#) referência da API AWS SDK for SAP ABAP.

## Exemplos de organizações usando o SDK for SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP with Organizations.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### AttachPolicy

O código de exemplo a seguir mostra como usar AttachPolicy.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_org->attachpolicy(  
    iv_policyid = iv_policy_id  
    iv_targetid = iv_target_id ).  
  MESSAGE 'Policy attached to target.' TYPE 'I'.  
CATCH /aws1/cx_orgaccessdeniedex.
```

```

MESSAGE 'You do not have permission to attach the policy.' TYPE 'E'.
CATCH /aws1/cx_orgpolicynotfoundex.
MESSAGE 'The specified policy does not exist.' TYPE 'E'.
CATCH /aws1/cx_orgtargetnotfoundex.
MESSAGE 'The specified target does not exist.' TYPE 'E'.
CATCH /aws1/cx_orgduplicateplyatta00.
MESSAGE 'The policy is already attached to the target.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [AttachPolicy](#) referência da API AWS SDK for SAP ABAP.

## CreatePolicy

O código de exemplo a seguir mostra como usar CreatePolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_org->createpolicy(           " oo_result is returned for testing
purposes. "
        iv_name          = iv_policy_name
        iv_description   = iv_policy_description
        iv_content       = iv_policy_content
        iv_type          = iv_policy_type ).
    MESSAGE 'Policy created.' TYPE 'I'.
CATCH /aws1/cx_orgaccessdeniedex.
    MESSAGE 'You do not have permission to create a policy.' TYPE 'E'.
CATCH /aws1/cx_orgduplicatepolicyex.
    MESSAGE 'A policy with this name already exists.' TYPE 'E'.
CATCH /aws1/cx_orgmalformedplydocex.
    MESSAGE 'The policy content is malformed.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreatePolicy](#) referência da API AWS SDK for SAP ABAP.

## DeletePolicy

O código de exemplo a seguir mostra como usar DeletePolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_org->deletepolicy(  
    iv_policyid = iv_policy_id ).  
  MESSAGE 'Policy deleted.' TYPE 'I'.  
CATCH /aws1/cx_orgaccessdeniedex.  
  MESSAGE 'You do not have permission to delete the policy.' TYPE 'E'.  
CATCH /aws1/cx_orgpolicynotfoundex.  
  MESSAGE 'The specified policy does not exist.' TYPE 'E'.  
CATCH /aws1/cx_orgpolicyinuseex.  
  MESSAGE 'The policy is still attached to one or more targets.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeletePolicy](#) referência da API AWS SDK for SAP ABAP.

## DescribePolicy

O código de exemplo a seguir mostra como usar DescribePolicy.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_org->describepolicy(      " oo_result is returned for testing  
purposes. "  
    iv_policyid = iv_policy_id ).  
    DATA(lo_policy) = oo_result->get_policy( ).  
    MESSAGE 'Retrieved policy details.' TYPE 'I'.  
    CATCH /aws1/cx_orgaccessdeniedex.  
    MESSAGE 'You do not have permission to describe the policy.' TYPE 'E'.  
    CATCH /aws1/cx_orgpolicynotfoundex.  
    MESSAGE 'The specified policy does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribePolicy](#) referência da API AWS SDK for SAP ABAP.

**DetachPolicy**

O código de exemplo a seguir mostra como usar DetachPolicy.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_org->detachpolicy(  
        iv_policyid = iv_policy_id  
        iv_targetid = iv_target_id ).
```

```

    MESSAGE 'Policy detached from target.' TYPE 'I'.
  CATCH /aws1/cx_orgaccessdeniedex.
    MESSAGE 'You do not have permission to detach the policy.' TYPE 'E'.
  CATCH /aws1/cx_orgpolicynotfoundex.
    MESSAGE 'The specified policy does not exist.' TYPE 'E'.
  CATCH /aws1/cx_orgtargetnotfoundex.
    MESSAGE 'The specified target does not exist.' TYPE 'E'.
  CATCH /aws1/cx_orgpolicynotattex.
    MESSAGE 'The policy is not attached to the target.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DetachPolicy](#) referência da API AWS SDK for SAP ABAP.

## ListPolicies

O código de exemplo a seguir mostra como usar ListPolicies.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_org->listpolicies(      " oo_result is returned for testing
purposes. "
    iv_filter = iv_filter ).
    DATA(lt_policies) = oo_result->get_policies( ).
    MESSAGE 'Retrieved list of policies.' TYPE 'I'.
  CATCH /aws1/cx_orgaccessdeniedex.
    MESSAGE 'You do not have permission to list policies.' TYPE 'E'.
  CATCH /aws1/cx_orgawsorgsnotinuseex.
    MESSAGE 'Your account is not a member of an organization.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListPolicies](#) referência da API AWS SDK for SAP ABAP.

# Exemplos do Amazon Pinpoint usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Pinpoint.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### SendMessage

O código de exemplo a seguir mostra como usar SendMessage.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Envie uma mensagem de e-mail.

```
" Build the addresses map from the list of to_addresses
DATA lt_addresses TYPE /aws1/cl_pptaddressconf=>tt_mapofaddressconfiguration.
LOOP AT it_to_addresses INTO DATA(lo_address).
  INSERT VALUE /aws1/cl_pptaddressconf=>ts_mapofaddressconf_maprow(
    key = lo_address->get_value( )
    value = NEW /aws1/cl_pptaddressconf( iv_channeltype = 'EMAIL' )
  ) INTO TABLE lt_addresses.
ENDLOOP.
```

```

" Send the email message
DATA(lo_result) = lo_ppt->sendmessages(
  iv_applicationid = iv_app_id
  io_messagerequest = NEW /aws1/cl_pptmessagerequest(
    it_addresses = lt_addresses
    io_messageconfiguration = NEW /aws1/cl_pptdirectmessageconf(
      io_emailmessage = NEW /aws1/cl_pptemailmessage(
        iv_fromaddress = iv_sender
        io_simpleemail = NEW /aws1/cl_pptsimpleemail(
          io_subject = NEW /aws1/cl_pptsimpleemailpart(
            iv_charset = iv_char_set
            iv_data = iv_subject
          )
          io_htmlpart = NEW /aws1/cl_pptsimpleemailpart(
            iv_charset = iv_char_set
            iv_data = iv_html_message
          )
          io_textpart = NEW /aws1/cl_pptsimpleemailpart(
            iv_charset = iv_char_set
            iv_data = iv_text_message
          )
        )
      )
    )
  ).

" Extract message IDs from response
DATA(lo_message_response) = lo_result->get_messageresponse( ).
ot_message_ids = lo_message_response->get_result( ).

MESSAGE 'Email message sent successfully.' TYPE 'I'.

```

## Envie uma mensagem SMS.

```

" Build the addresses map for the destination number
DATA lt_addresses TYPE /aws1/cl_pptaddressconf=>tt_mapofaddressconfiguration.
INSERT VALUE /aws1/cl_pptaddressconf=>ts_mapofaddressconf_maprow(
  key = iv_destination_number
  value = NEW /aws1/cl_pptaddressconf( iv_channeltype = 'SMS' )
) INTO TABLE lt_addresses.

```

```

" Send the SMS message
DATA(lo_result) = lo_ppt->sendmessages(
  iv_applicationid = iv_app_id
  io_messagerequest = NEW /aws1/cl_pptmessagerequest(
    it_addresses = lt_addresses
    io_messageconfiguration = NEW /aws1/cl_pptdirectmessageconf(
      io_smsmessage = NEW /aws1/cl_pptsmsmessage(
        iv_body = iv_message
        iv_message_type = iv_message_type
        iv_OriginationNumber = iv_OriginationNumber
      )
    )
  )
).

" Extract message ID from response
DATA(lo_message_response) = lo_result->get_messageresponse( ).
DATA(lt_results) = lo_message_response->get_result( ).
LOOP AT lt_results INTO DATA(ls_result).
  IF ls_result-key = iv_destination_number.
    ov_message_id = ls_result-value->get_messageid( ).
    EXIT.
  ENDIF.
ENDLOOP.

MESSAGE 'SMS message sent successfully.' TYPE 'I'.

```

### Enviar uma mensagem com um modelo de e-mail existente.

```

" Build the addresses map from the list of to_addresses
DATA lt_addresses TYPE /aws1/cl_pptaddressconf=>tt_mapofaddressconfiguration.
LOOP AT it_to_addresses INTO DATA(lo_address).
  INSERT VALUE /aws1/cl_pptaddressconf=>ts_mapofaddressconf_maprow(
    key = lo_address->get_value( )
    value = NEW /aws1/cl_pptaddressconf( iv_channeltype = 'EMAIL' )
  ) INTO TABLE lt_addresses.
ENDLOOP.

" Send the email message using a template
DATA(lo_result) = lo_ppt->sendmessages(
  iv_applicationid = iv_app_id
  io_messagerequest = NEW /aws1/cl_pptmessagerequest(

```

```

it_addresses = lt_addresses
io_messageconfiguration = NEW /aws1/cl_pptdirectmessageconf(
  io_emailmessage = NEW /aws1/cl_pptemailmessage(
    iv_fromaddress = iv_sender
  )
)
io_templateconfiguration = NEW /aws1/cl_ppttemplateconf(
  io_emailtemplate = NEW /aws1/cl_ppttemplate(
    iv_name = iv_template_name
    iv_version = iv_template_version
  )
)
).

" Extract message IDs from response
DATA(lo_message_response) = lo_result->get_messageresponse( ).
ot_message_ids = lo_message_response->get_result( ).

MESSAGE 'Templated email message sent successfully.' TYPE 'I'.

```

Envie uma mensagem de texto com um modelo de SMS existente.

```

" Build the addresses map for the destination number
DATA lt_addresses TYPE /aws1/cl_pptaddressconf=>tt_mapofaddressconfiguration.
INSERT VALUE /aws1/cl_pptaddressconf=>ts_mapofaddressconf_maprow(
  key = iv_destination_number
  value = NEW /aws1/cl_pptaddressconf( iv_channeltype = 'SMS' )
) INTO TABLE lt_addresses.

" Send the SMS message using a template
DATA(lo_result) = lo_ppt->sendmessages(
  iv_applicationid = iv_app_id
  io_messagerequest = NEW /aws1/cl_pptmessagerequest(
    it_addresses = lt_addresses
    io_messageconfiguration = NEW /aws1/cl_pptdirectmessageconf(
      io_smsmessage = NEW /aws1/cl_pptsmsmessage(
        iv_message_type = iv_message_type
        iv_OriginationNumber = iv_OriginationNumber
      )
    )
  )
  io_templateconfiguration = NEW /aws1/cl_ppttemplateconf(

```

```
        io_smstemplate = NEW /aws1/cl_ppttemplate(  
            iv_name = iv_template_name  
            iv_version = iv_template_version  
        )  
    )  
)  
).  
  
" Extract message ID from response  
DATA(lo_message_response) = lo_result->get_messageresponse( ).  
DATA(lt_results) = lo_message_response->get_result( ).  
LOOP AT lt_results INTO DATA(ls_result).  
    IF ls_result-key = iv_destination_number.  
        ov_message_id = ls_result-value->get_messageid( ).  
        EXIT.  
    ENDIF.  
ENDLOOP.  
  
MESSAGE 'Templated SMS message sent successfully.' TYPE 'I'.
```

- Para obter detalhes da API, consulte a [SendMessages](#) referência da API AWS SDK for SAP ABAP.

## Exemplos de API de voz e SMS do Amazon Pinpoint usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Amazon Pinpoint SMS and Voice API.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### CreateConfigurationSet

O código de exemplo a seguir mostra como usar CreateConfigurationSet.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create a new configuration set
  lo_pps->createconfigurationset(
    iv_configurationsetname = iv_configuration_set_name    " e.g., 'my-config-
set'
  ).

  MESSAGE 'Configuration set created successfully.' TYPE 'I'.

  CATCH /aws1/cx_ppsalreadyexistsex INTO DATA(lo_already_exists_ex).
    MESSAGE lo_already_exists_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_already_exists_ex.
  CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_bad_request_ex.
  CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).
    MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_internal_error_ex.
  CATCH /aws1/cx_ppslimitexceededex INTO DATA(lo_limit_exceeded_ex).
    MESSAGE lo_limit_exceeded_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_limit_exceeded_ex.
  CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
    MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_too_many_requests_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateConfigurationSet](#) referência da API AWS SDK for SAP ABAP.

## CreateConfigurationSetEventDestination

O código de exemplo a seguir mostra como usar CreateConfigurationSetEventDestination.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create event destination for the configuration set
  lo_pps->createconfseteventdst(
    iv_configurationsetname = iv_configuration_set_name    " e.g., 'my-config-
set'
    iv_eventdestinationname = iv_event_destination_name    " e.g., 'my-event-
dest'
    io_eventdestination = io_event_destination
  ).

  MESSAGE 'Event destination created successfully.' TYPE 'I'.

CATCH /aws1/cx_ppsalreadyexistsex INTO DATA(lo_already_exists_ex).
  MESSAGE lo_already_exists_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_already_exists_ex.
CATCH /aws1/cx_ppsnotfoundexception INTO DATA(lo_not_found_ex).
  MESSAGE lo_not_found_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_not_found_ex.
CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_bad_request_ex.
CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).
  MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_internal_error_ex.
CATCH /aws1/cx_ppslimitexceededex INTO DATA(lo_limit_exceeded_ex).
  MESSAGE lo_limit_exceeded_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_limit_exceeded_ex.
```

```
CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
  MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_too_many_requests_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateConfigurationSetEventDestination](#) referência da API AWS SDK for SAP ABAP.

## DeleteConfigurationSet

O código de exemplo a seguir mostra como usar DeleteConfigurationSet.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Delete the configuration set
  lo_pps->deleteconfigurationset(
    iv_configurationsetname = iv_configuration_set_name    " e.g., 'my-config-
set'
  ).

  MESSAGE 'Configuration set deleted successfully.' TYPE 'I'.

CATCH /aws1/cx_ppsnotfoundexception INTO DATA(lo_not_found_ex).
  MESSAGE lo_not_found_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_not_found_ex.
CATCH /aws1/cx_ppsbadrequestsex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_bad_request_ex.
CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).
  MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_internal_error_ex.
CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
  MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
```

```
RAISE EXCEPTION lo_too_many_requests_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteConfigurationSet](#) referência da API AWS SDK for SAP ABAP.

## DeleteConfigurationSetEventDestination

O código de exemplo a seguir mostra como usar DeleteConfigurationSetEventDestination.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    " Delete the event destination  
    lo_pps->deleteconfseteventdst(  
        iv_configurationsetname = iv_configuration_set_name    " e.g., 'my-config-  
set'  
        iv_eventdestinationname = iv_event_destination_name    " e.g., 'my-event-  
dest'  
    ).  
  
    MESSAGE 'Event destination deleted successfully.' TYPE 'I'.  
  
CATCH /aws1/cx_ppsnotfoundexception INTO DATA(lo_not_found_ex).  
    MESSAGE lo_not_found_ex->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_not_found_ex.  
CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).  
    MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request_ex.  
CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).  
    MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.  
    RAISE EXCEPTION lo_internal_error_ex.  
CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).  
    MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
```

```

    RAISE EXCEPTION lo_too_many_requests_ex.
  ENDRY.

```

- Para obter detalhes da API, consulte a [DeleteConfigurationSetEventDestination](#) referência da API AWS SDK for SAP ABAP.

## GetConfigurationSetEventDestinations

O código de exemplo a seguir mostra como usar `GetConfigurationSetEventDestinations`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Get event destinations for the configuration set
  oo_result = lo_pps->getconfseteventdestinations(
    iv_configurationsetname = iv_configuration_set_name    " e.g., 'my-config-
set'
  ).

  " Process the event destinations
  LOOP AT oo_result->get_eventdestinations( ) INTO DATA(lo_event_dest).
    DATA(lv_dest_name) = lo_event_dest->get_name( ).
    DATA(lv_enabled) = lo_event_dest->get_enabled( ).

    MESSAGE |Event destination: { lv_dest_name }, Enabled: { lv_enabled }|
TYPE 'I'.

  " Check for CloudWatch Logs destination
  DATA(lo_cloudwatch_dest) = lo_event_dest->get_cloudwatchlogsdst( ).
  IF lo_cloudwatch_dest IS NOT INITIAL.
    DATA(lv_log_group_arn) = lo_cloudwatch_dest->get_loggrouparn( ).
    MESSAGE | CloudWatch Logs destination: { lv_log_group_arn }| TYPE 'I'.
  ENDIF.

```

```

" Check for Kinesis Firehose destination
DATA(lo_firehose_dest) = lo_event_dest->get_kinesisfirehosedst( ).
IF lo_firehose_dest IS NOT INITIAL.
  DATA(lv_delivery_stream) = lo_firehose_dest->get_deliverystreamarn( ).
  MESSAGE | Kinesis Firehose destination: { lv_delivery_stream }| TYPE
'I'.
ENDIF.

" Check for SNS destination
DATA(lo_sns_dest) = lo_event_dest->get_snsdestination( ).
IF lo_sns_dest IS NOT INITIAL.
  DATA(lv_topic_arn) = lo_sns_dest->get_topicarn( ).
  MESSAGE | SNS destination: { lv_topic_arn }| TYPE 'I'.
ENDIF.
ENDLOOP.

CATCH /aws1/cx_ppsnotfoundexception INTO DATA(lo_not_found_ex).
  MESSAGE lo_not_found_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_not_found_ex.
CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_bad_request_ex.
CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).
  MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_internal_error_ex.
CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
  MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_too_many_requests_ex.
ENDTRY.


```

- Para obter detalhes da API, consulte a [GetConfigurationSetEventDestinations](#) referência da API AWS SDK for SAP ABAP.

## ListConfigurationSets

O código de exemplo a seguir mostra como usar ListConfigurationSets.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " List all configuration sets
  oo_result = lo_pps->listconfigurationsets(
    iv_nexttoken = iv_next_token    " Optional: Token for pagination
    iv_pagesize = iv_page_size     " Optional: Number of results per page,
e.g., '10'
  ).

  " Process the configuration sets
  LOOP AT oo_result->get_configurationsets( ) INTO DATA(lo_config_set).
    DATA(lv_config_set_name) = lo_config_set->get_value( ).
    MESSAGE |Configuration set: { lv_config_set_name }| TYPE 'I'.
  ENDLLOOP.

  " Check if there are more results
  DATA(lv_next_token) = oo_result->get_nexttoken( ).
  IF lv_next_token IS NOT INITIAL.
    MESSAGE |More results available. Next token: { lv_next_token }| TYPE 'I'.
  ENDIF.

  CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_bad_request_ex.
  CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).
    MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_internal_error_ex.
  CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
    MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_too_many_requests_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListConfigurationSets](#) referência da API AWS SDK for SAP ABAP.

## SendVoiceMessage

O código de exemplo a seguir mostra como usar SendVoiceMessage.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create SSML message type object with voice parameters
  DATA(lo_ssml_message) = NEW /aws1/cl_ppssmlmessagetype(
    iv_languagecode = iv_language_code      " e.g., 'en-US'
    iv_voiceid = iv_voice_id                " e.g., 'Matthew'
    iv_text = iv_ssml_message               " SSML formatted message text
  ).

  " Create voice message content with the SSML message
  DATA(lo_content) = NEW /aws1/cl_ppsvoicemessagecont(
    io_ssmlmessage = lo_ssml_message
  ).

  " Send the voice message
  DATA(lo_result) = lo_pps->sendvoicemessage(
    iv_Originationphonenummer = iv_Origination_number " e.g., '+12065550110'
    iv_callerid = iv_caller_id                       " e.g., '+12065550199'
    iv_destinationphonenummer = iv_destination_number " e.g., '+12065550142'
    io_content = lo_content
  ).

  " Retrieve the message ID from the response
  ov_message_id = lo_result->get_messageid( ).

  MESSAGE 'Voice message sent successfully.' TYPE 'I'.

  CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_bad_request_ex.
  CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).
    MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
```

```

    RAISE EXCEPTION lo_internal_error_ex.
    CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
    MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
    RAISE EXCEPTION lo_too_many_requests_ex.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [SendVoiceMessage](#) referência da API AWS SDK for SAP ABAP.

## UpdateConfigurationSetEventDestination

O código de exemplo a seguir mostra como usar UpdateConfigurationSetEventDestination.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Update the event destination
  lo_pps->updateconfseteventdst(
    iv_configurationsetname = iv_configuration_set_name    " e.g., 'my-config-
set'
    iv_eventdestinationname = iv_event_destination_name   " e.g., 'my-event-
dest'
    io_eventdestination = io_event_destination
  ).

  MESSAGE 'Event destination updated successfully.' TYPE 'I'.

CATCH /aws1/cx_ppsnotfoundexception INTO DATA(lo_not_found_ex).
  MESSAGE lo_not_found_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_not_found_ex.
CATCH /aws1/cx_ppsbadrequestex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex->get_text( ) TYPE 'I'.
  RAISE EXCEPTION lo_bad_request_ex.
CATCH /aws1/cx_ppsinternalsvcerrorex INTO DATA(lo_internal_error_ex).

```

```
MESSAGE lo_internal_error_ex->get_text( ) TYPE 'I'.
RAISE EXCEPTION lo_internal_error_ex.
CATCH /aws1/cx_ppstoomanyrequestsex INTO DATA(lo_too_many_requests_ex).
MESSAGE lo_too_many_requests_ex->get_text( ) TYPE 'I'.
RAISE EXCEPTION lo_too_many_requests_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateConfigurationSetEventDestination](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon Polly usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Amazon Polly.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

## Ações

### DeleteLexicon

O código de exemplo a seguir mostra como usar DeleteLexicon.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_ply->deletelexicon( iv_name ).  
  MESSAGE 'Lexicon deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_plylexiconnotfoundex.  
  MESSAGE 'Lexicon not found.' TYPE 'E'.  
CATCH /aws1/cx_plyservicefailureex.  
  MESSAGE 'Service failure occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteLexicon](#) referência da API AWS SDK for SAP ABAP.

## DescribeVoices

O código de exemplo a seguir mostra como usar DescribeVoices.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Only pass optional parameters if they have values  
  IF iv_engine IS NOT INITIAL AND iv_language IS NOT INITIAL.  
    oo_result = lo_ply->describevoices(  
      iv_engine = iv_engine  
      iv_languagecode = iv_language ).  
  ELSEIF iv_engine IS NOT INITIAL.  
    oo_result = lo_ply->describevoices(  
      iv_engine = iv_engine ).  
  ELSEIF iv_language IS NOT INITIAL.  
    oo_result = lo_ply->describevoices(  
      iv_languagecode = iv_language ).  
  ELSE.  
    oo_result = lo_ply->describevoices( ).  
  ENDIF.  
  MESSAGE 'Retrieved voice metadata.' TYPE 'I'.  
ENDTRY.
```

```
CATCH /aws1/cx_plyinvalidnexttokenex.  
    MESSAGE 'The NextToken is invalid.' TYPE 'E'.  
CATCH /aws1/cx_plyservicefailureex.  
    MESSAGE 'Service failure occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeVoices](#) referência da API AWS SDK for SAP ABAP.

## GetLexicon

O código de exemplo a seguir mostra como usar GetLexicon.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ply->getlexicon( iv_name ).  
    DATA(lo_lexicon) = oo_result->get_lexicon( ).  
    IF lo_lexicon IS BOUND.  
        DATA(lv_lex_name) = lo_lexicon->get_name( ).  
        MESSAGE |Retrieved lexicon: { lv_lex_name }| TYPE 'I'.  
    ENDIF.  
CATCH /aws1/cx_plylexiconnotfoundex.  
    MESSAGE 'Lexicon not found.' TYPE 'E'.  
CATCH /aws1/cx_plyservicefailureex.  
    MESSAGE 'Service failure occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetLexicon](#) referência da API AWS SDK for SAP ABAP.

## GetSpeechSynthesisTask

O código de exemplo a seguir mostra como usar GetSpeechSynthesisTask.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ply->getspeechsynthesistask( iv_task_id ).  
    DATA(lo_task) = oo_result->get_synthesistask( ).  
    IF lo_task IS BOUND.  
        DATA(lv_status) = lo_task->get_taskstatus( ).  
        MESSAGE |Task status: { lv_status }| TYPE 'I'.  
    ENDIF.  
    CATCH /aws1/cx_plyinvalidtaskidex.  
        MESSAGE 'Invalid task ID.' TYPE 'E'.  
    CATCH /aws1/cx_plyservicefailureex.  
        MESSAGE 'Service failure occurred.' TYPE 'E'.  
    CATCH /aws1/cx_plysynthesistsknotf00.  
        MESSAGE 'Synthesis task not found.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetSpeechSynthesisTask](#) referência da API AWS SDK for SAP ABAP.

**ListLexicons**

O código de exemplo a seguir mostra como usar ListLexicons.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```

oo_result = lo_ply->listlexicons( ).
DATA(lt_lexicons) = oo_result->get_lexicons( ).
DATA(lv_count) = lines( lt_lexicons ).
MESSAGE |Found { lv_count } lexicons| TYPE 'I'.
CATCH /aws1/cx_plyinvalidnexttokenex.
  MESSAGE 'Invalid NextToken.' TYPE 'E'.
CATCH /aws1/cx_plyservicefailureex.
  MESSAGE 'Service failure occurred.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListLexicons](#) referência da API AWS SDK for SAP ABAP.

## ListSpeechSynthesisTasks

O código de exemplo a seguir mostra como usar ListSpeechSynthesisTasks.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Only pass optional parameters if they have values
  IF iv_max_results IS NOT INITIAL AND iv_status IS NOT INITIAL.
    oo_result = lo_ply->listspeechsynthesistasks(
      iv_maxresults = iv_max_results
      iv_status = iv_status ).
  ELSEIF iv_max_results IS NOT INITIAL.
    oo_result = lo_ply->listspeechsynthesistasks(
      iv_maxresults = iv_max_results ).
  ELSEIF iv_status IS NOT INITIAL.
    oo_result = lo_ply->listspeechsynthesistasks(
      iv_status = iv_status ).
  ELSE.
    oo_result = lo_ply->listspeechsynthesistasks( ).
  ENDIF.
DATA(lt_tasks) = oo_result->get_synthesistasks( ).
DATA(lv_count) = lines( lt_tasks ).

```

```

    MESSAGE |Found { lv_count } synthesis tasks| TYPE 'I'.
    CATCH /aws1/cx_plyinvalidnexttokenex.
    MESSAGE 'Invalid NextToken.' TYPE 'E'.
    CATCH /aws1/cx_plyservicefailureex.
    MESSAGE 'Service failure occurred.' TYPE 'E'.
  ENDRY.

```

- Para obter detalhes da API, consulte a [ListSpeechSynthesisTasks](#) referência da API AWS SDK for SAP ABAP.

## PutLexicon

O código de exemplo a seguir mostra como usar PutLexicon.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  lo_ply->putlexicon(
    iv_name = iv_name
    iv_content = iv_content ).
  MESSAGE 'Lexicon created successfully.' TYPE 'I'.
  CATCH /aws1/cx_plyinvalidlexiconex.
  MESSAGE 'Invalid lexicon.' TYPE 'E'.
  CATCH /aws1/cx_plylexiconsizeexcdex.
  MESSAGE 'Lexicon size exceeded.' TYPE 'E'.
  CATCH /aws1/cx_plymaxlexemelengthe00.
  MESSAGE 'Maximum lexeme length exceeded.' TYPE 'E'.
  CATCH /aws1/cx_plymaxlexiconsnoexc00.
  MESSAGE 'Maximum number of lexicons exceeded.' TYPE 'E'.
  CATCH /aws1/cx_plyservicefailureex.
  MESSAGE 'Service failure occurred.' TYPE 'E'.
  CATCH /aws1/cx_plyunsuppedplsalpha00.
  MESSAGE 'Unsupported PLS alphabet.' TYPE 'E'.
  CATCH /aws1/cx_plyunsuppedplslangu00.

```

```
MESSAGE 'Unsupported PLS language.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutLexicon](#) referência da API AWS SDK for SAP ABAP.

## StartSpeechSynthesisTask

O código de exemplo a seguir mostra como usar StartSpeechSynthesisTask.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Only pass optional parameters if they have values  
  IF iv_lang_code IS NOT INITIAL AND iv_s3_key_prefix IS NOT INITIAL.  
    oo_result = lo_ply->startspeechsynthesistask(  
      iv_engine = iv_engine  
      iv_outputformat = iv_audio_format  
      iv_outputs3bucketname = iv_s3_bucket  
      iv_outputs3keyprefix = iv_s3_key_prefix  
      iv_text = iv_text  
      iv_voiceid = iv_voice_id  
      iv_languagecode = iv_lang_code ).  
  ELSEIF iv_lang_code IS NOT INITIAL.  
    oo_result = lo_ply->startspeechsynthesistask(  
      iv_engine = iv_engine  
      iv_outputformat = iv_audio_format  
      iv_outputs3bucketname = iv_s3_bucket  
      iv_text = iv_text  
      iv_voiceid = iv_voice_id  
      iv_languagecode = iv_lang_code ).  
  ELSEIF iv_s3_key_prefix IS NOT INITIAL.  
    oo_result = lo_ply->startspeechsynthesistask(  
      iv_engine = iv_engine  
      iv_outputformat = iv_audio_format  
      iv_outputs3bucketname = iv_s3_bucket
```

```
        iv_outputs3keyprefix = iv_s3_key_prefix
        iv_text = iv_text
        iv_voiceid = iv_voice_id ).
ELSE.
    oo_result = lo_ply->startspeechsynthesistask(
        iv_engine = iv_engine
        iv_outputformat = iv_audio_format
        iv_outputs3bucketname = iv_s3_bucket
        iv_text = iv_text
        iv_voiceid = iv_voice_id ).
ENDIF.
MESSAGE 'Speech synthesis task started.' TYPE 'I'.
CATCH /aws1/cx_plyinvalids3bucketex.
    MESSAGE 'Invalid S3 bucket.' TYPE 'E'.
CATCH /aws1/cx_plyinvalidssmllex.
    MESSAGE 'Invalid SSML.' TYPE 'E'.
CATCH /aws1/cx_plylexiconnotfoundex.
    MESSAGE 'Lexicon not found.' TYPE 'E'.
CATCH /aws1/cx_plyservicefailureex.
    MESSAGE 'Service failure occurred.' TYPE 'E'.
CATCH /aws1/cx_plytextlengthexcdex.
    MESSAGE 'Text length exceeded maximum.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartSpeechSynthesisTask](#) referência da API AWS SDK for SAP ABAP.

## SynthesizeSpeech

O código de exemplo a seguir mostra como usar SynthesizeSpeech.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

TRY.

```
" Only pass optional language code if it has a value
IF iv_lang_code IS NOT INITIAL.
  oo_result = lo_ply->synthesizespeech(
    iv_engine = iv_engine
    iv_outputformat = iv_output_fmt
    iv_text = iv_text
    iv_voiceid = iv_voice_id
    iv_languagecode = iv_lang_code ).
ELSE.
  oo_result = lo_ply->synthesizespeech(
    iv_engine = iv_engine
    iv_outputformat = iv_output_fmt
    iv_text = iv_text
    iv_voiceid = iv_voice_id ).
ENDIF.
MESSAGE 'Speech synthesized successfully.' TYPE 'I'.
CATCH /aws1/cx_plyinvalidssmlx.
  MESSAGE 'Invalid SSML.' TYPE 'E'.
CATCH /aws1/cx_plylexiconnotfoundex.
  MESSAGE 'Lexicon not found.' TYPE 'E'.
CATCH /aws1/cx_plyservicefailureex.
  MESSAGE 'Service failure occurred.' TYPE 'E'.
CATCH /aws1/cx_plytextlengthexcdex.
  MESSAGE 'Text length exceeded maximum.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [SynthesizeSpeech](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon RDS usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Amazon RDS.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Ações](#)

## Ações

### CreateDBParameterGroup

O código de exemplo a seguir mostra como usar CreateDBParameterGroup.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
" iv_dbparametergroupname = 'mydbparametergroup'
" iv_dbparametergroupfamily = 'mysql8.0'
" iv_description           = 'My custom DB parameter group for MySQL 8.0'
TRY.
    oo_result = lo_rds->createdbparametergroup(
        iv_dbparametergroupname = iv_dbparametergroupname
        iv_dbparametergroupfamily = iv_dbparametergroupfamily
        iv_description           = iv_description ).
    MESSAGE 'DB parameter group created.' TYPE 'I'.
CATCH /aws1/cx_rdsdbparmgralrexfault.
    MESSAGE 'DB parameter group already exists.' TYPE 'I'.
CATCH /aws1/cx_rdsdbprmgrquotaexcd00.
    MESSAGE 'DB parameter group quota exceeded.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte [Criar DBParameter grupo](#) no AWS SDK para referência da API SAP ABAP.

### DeleteDBParameterGroup

O código de exemplo a seguir mostra como usar DeleteDBParameterGroup.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
" iv_dbparametergroupname = 'mydbparametergroup'
TRY.
  lo_rds->deletedbparametergroup(
    iv_dbparametergroupname = iv_dbparametergroupname ).
  MESSAGE 'DB parameter group deleted.' TYPE 'I'.
CATCH /aws1/cx_rdsdbprmgrnotfndfault.
  MESSAGE 'DB parameter group not found.' TYPE 'I'.
CATCH /aws1/cx_rdsinvdbprmgrstatef00.
  MESSAGE 'DB parameter group is in an invalid state.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte [Excluir DBParameter grupo](#) no AWS SDK para referência da API SAP ABAP.

## DescribeDBEngineVersions

O código de exemplo a seguir mostra como usar DescribeDBEngineVersions.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" iv_engine          = 'mysql'
" iv_dbparametergroupfamily = 'mysql8.0' (optional - filters by parameter group
family)
TRY.
```

```

oo_result = lo_rds->describedbengineversions(
  iv_engine          = iv_engine
  iv_dbparametergroupfamily = iv_dbparametergroupfamily ).
DATA(lv_version_count) = lines( oo_result->get_dbengineversions( ) ).
MESSAGE |Retrieved { lv_version_count } engine versions.| TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte [Descrição DBEngine das versões](#) no AWS SDK para referência da API SAP ABAP.

## DescribeDBParameterGroups

O código de exemplo a seguir mostra como usar DescribeDBParameterGroups.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" iv_dbparametergroupname = 'mydbparametergroup'
TRY.
  oo_result = lo_rds->describedbparametergroups(
    iv_dbparametergroupname = iv_dbparametergroupname ).
  MESSAGE 'DB parameter group retrieved.' TYPE 'I'.
CATCH /aws1/cx_rdsdbprimgrnotfndfault.
  MESSAGE 'DB parameter group not found.' TYPE 'I'.
ENDTRY.


```

- Para obter detalhes da API, consulte [Descrever DBParameter grupos](#) no AWS SDK para referência da API SAP ABAP.

## DescribeDBParameters

O código de exemplo a seguir mostra como usar DescribeDBParameters.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
" iv_dbparametergroupname = 'mydbparametergroup'
" iv_source                 = 'user' (optional - filters by parameter source)
TRY.
  oo_result = lo_rds->describedbparameters(
    iv_dbparametergroupname = iv_dbparametergroupname
    iv_source                 = iv_source ).
  DATA(lv_param_count) = lines( oo_result->get_parameters( ) ).
  MESSAGE |Retrieved { lv_param_count } parameters.| TYPE 'I'.
CATCH /aws1/cx_rdsdbprmgrnotfndfault.
  MESSAGE 'DB parameter group not found.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte [Descrição DBParameters](#) na referência da API ABAP do AWS SDK for SAP.

## DescribeOrderableDBInstanceOptions

O código de exemplo a seguir mostra como usar DescribeOrderableDBInstanceOptions.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" iv_engine                 = 'mysql'
" iv_engineversion         = '8.0.35'
TRY.
```

```

oo_result = lo_rds->descrororderabledbinstoptions(
  iv_engine      = iv_engine
  iv_engineversion = iv_engineversion ).
DATA(lv_option_count) = lines( oo_result->get_orderabledbinstoptions( ) ).
MESSAGE |Retrieved { lv_option_count } orderable DB instance options.| TYPE
'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte [DescribeOrderableDBInstanceOpções](#) no AWS SDK para referência da API SAP ABAP.

## ModifyDBParameterGroup

O código de exemplo a seguir mostra como usar ModifyDBParameterGroup.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" iv_dbparametergroupname = 'mydbparametergroup'
" it_parameters - table containing parameter objects with:
"   - parametername = 'max_connections'
"   - parametervalue = '100'
"   - applymethod = 'immediate' or 'pending-reboot'
TRY.
  oo_result = lo_rds->modifydbparametergroup(
    iv_dbparametergroupname = iv_dbparametergroupname
    it_parameters            = it_parameters ).
  MESSAGE 'DB parameter group modified.' TYPE 'I'.
CATCH /aws1/cx_rdsdbprmgrnotfndfault.
  MESSAGE 'DB parameter group not found.' TYPE 'I'.
CATCH /aws1/cx_rdsinldbprmgrstatef00.
  MESSAGE 'DB parameter group is in an invalid state.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte [Modificar DBParameter grupo](#) no AWS SDK para referência da API SAP ABAP.

## Exemplos do Amazon Redshift usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Redshift.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateCluster

O código de exemplo a seguir mostra como usar CreateCluster.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie o cluster.

```
TRY.  
  " Example values: iv_cluster_identifier = 'my-redshift-cluster'  
  " Example values: iv_node_type = 'ra3.4xlarge'  
  " Example values: iv_master_username = 'awsuser'
```

```

" Example values: iv_master_password = 'AwsUser1000'
" Example values: iv_publicly_accessible = abap_true
" Example values: iv_number_of_nodes = 2
oo_result = lo_rsh->createcluster(
  iv_clusteridentifier = iv_cluster_identifier
  iv_nodetype = iv_node_type
  iv_masterusername = iv_master_username
  iv_masteruserpassword = iv_master_password
  iv_publiclyaccessible = iv_publicly_accessible
  iv_numberofnodes = iv_number_of_nodes
).
MESSAGE 'Redshift cluster created successfully.' TYPE 'I'.
CATCH /aws1/cx_rshclustalrddyexfault.
  MESSAGE 'Cluster already exists.' TYPE 'I'.
CATCH /aws1/cx_rshclstquotaexcdfault.
  MESSAGE 'Cluster quota exceeded.' TYPE 'I'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateCluster](#) referência da API AWS SDK for SAP ABAP.

## DeleteCluster

O código de exemplo a seguir mostra como usar DeleteCluster.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Excluir o cluster.

```

TRY.
" Example values: iv_cluster_identifier = 'my-redshift-cluster'
lo_rsh->deletecluster(
  iv_clusteridentifier = iv_cluster_identifier
  iv_skipfinalclustersnapshot = abap_true

```

```

    ).
    MESSAGE 'Redshift cluster deleted successfully.' TYPE 'I'.
  CATCH /aws1/cx_rshclustnotfoundfault.
    MESSAGE 'Cluster not found.' TYPE 'I'.
  CATCH /aws1/cx_rshinvcluststatefault.
    MESSAGE 'Invalid cluster state for deletion.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteCluster](#) referência da API AWS SDK for SAP ABAP.

## DescribeClusters

O código de exemplo a seguir mostra como usar DescribeClusters.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Descrever o cluster.

```

  TRY.
    " Example values: iv_cluster_identififier = 'my-redshift-cluster' (optional)
    oo_result = lo_rsh->describeclusters(
      iv_clusteridentififier = iv_cluster_identififier
    ).
    lt_clusters = oo_result->get_clusters( ).
    lv_cluster_count = lines( lt_clusters ).
    MESSAGE |Retrieved { lv_cluster_count } cluster(s).| TYPE 'I'.
  CATCH /aws1/cx_rshclustnotfoundfault.
    MESSAGE 'Cluster not found.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeClusters](#) referência da API AWS SDK for SAP ABAP.

## DescribeStatement

O código de exemplo a seguir mostra como usar DescribeStatement.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Example values: iv_statement_id = 'xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx'  
  oo_result = lo_rsd->describestatement(  
    iv_id = iv_statement_id  
  ).  
  lv_status = oo_result->get_status( ).  
  MESSAGE |Statement status: { lv_status }| TYPE 'I'.  
CATCH /aws1/cx_rsdresourcenotfoundex.  
  MESSAGE 'Statement not found.' TYPE 'I'.  
CATCH /aws1/cx_rsdinternalserverex.  
  MESSAGE 'Internal server error.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeStatement](#) referência da API AWS SDK for SAP ABAP.

## ExecuteStatement

O código de exemplo a seguir mostra como usar ExecuteStatement.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Example values: iv_cluster_identifier = 'redshift-cluster-movies'
  " Example values: iv_database_name = 'dev'
  " Example values: iv_user_name = 'awsuser'
  " Example values: iv_sql = 'SELECT * FROM movies WHERE year = :year'
  " Example values: it_parameter_list - SQL parameters for parameterized
queries

  " Only pass parameters if the list is not empty
  IF it_parameter_list IS NOT INITIAL.
    oo_result = lo_rsd->executestatement(
      iv_clusteridentifier = iv_cluster_identifier
      iv_database = iv_database_name
      iv_dbuser = iv_user_name
      iv_sql = iv_sql
      it_parameters = it_parameter_list
    ).
  ELSE.
    oo_result = lo_rsd->executestatement(
      iv_clusteridentifier = iv_cluster_identifier
      iv_database = iv_database_name
      iv_dbuser = iv_user_name
      iv_sql = iv_sql
    ).
  ENDIF.

  lv_statement_id = oo_result->get_id( ).
  MESSAGE |Statement executed. ID: { lv_statement_id }| TYPE 'I'.
  CATCH /aws1/cx_rsdexecutestatementex.
    MESSAGE 'Statement execution error.' TYPE 'I'.
  CATCH /aws1/cx_rsdresourcenotfoundex.
    MESSAGE 'Resource not found.' TYPE 'I'.
ENDTRY.


```

- Para obter detalhes da API, consulte a [ExecuteStatement](#) referência da API AWS SDK for SAP ABAP.

## GetStatementResult

O código de exemplo a seguir mostra como usar GetStatementResult.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

## Verifique o resultado da instrução

```
TRY.
  " Example values: iv_statement_id = 'xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx'
  " Handle pagination for large result sets

DO.
  lo_result_page = lo_rsd->getstatementresult(
    iv_id = iv_statement_id
    iv_nexttoken = lv_next_token
  ).

  " Collect records from this page
  lt_page_records = lo_result_page->get_records( ).
  APPEND LINES OF lt_page_records TO lt_all_records.

  " Check if there are more pages
  lv_next_token = lo_result_page->get_nexttoken( ).
  IF lv_next_token IS INITIAL.
    EXIT. " No more pages
  ENDIF.
ENDDO.

" For the last call, set oo_result for return value
oo_result = lo_result_page.
lv_record_count = lines( lt_all_records ).
MESSAGE |Retrieved { lv_record_count } record(s).| TYPE 'I'.
CATCH /aws1/cx_rsdresourcenotfoundex.
  MESSAGE 'Statement not found or results not available.' TYPE 'I'.
CATCH /aws1/cx_rsdinternalserverex.
  MESSAGE 'Internal server error.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetStatementResult](#) referência da API AWS SDK for SAP ABAP.

## ListDatabases

O código de exemplo a seguir mostra como usar ListDatabases.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  " Example values: iv_cluster_identififier = 'redshift-cluster-movies'
  " Example values: iv_database_name = 'dev'
  " Example values: iv_database_user = 'awsuser'
  oo_result = lo_rsd->listdatabases(
    iv_clusteridentififier = iv_cluster_identififier
    iv_database = iv_database_name
    iv_dbuser = iv_database_user
  ).
  lt_databases = oo_result->get_databases( ).
  lv_db_count = lines( lt_databases ).
  MESSAGE |Retrieved { lv_db_count } database(s).| TYPE 'I'.
CATCH /aws1/cx_rsddatabaseconnex.
  MESSAGE 'Database connection error.' TYPE 'I'.
CATCH /aws1/cx_rsdresourcenotfoundex.
  MESSAGE 'Cluster not found.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListDatabases](#) referência da API AWS SDK for SAP ABAP.

## ModifyCluster

O código de exemplo a seguir mostra como usar ModifyCluster.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Modificar um cluster.

```
TRY.  
  " Example values: iv_cluster_identifier = 'my-redshift-cluster'  
  " Example values: iv_pref_maintenance_wn = 'wed:07:30-wed:08:00'  
  lo_rsh->modifycluster(  
    iv_clusteridentifier = iv_cluster_identifier  
    iv_preferredmaintenancewin00 = iv_pref_maintenance_wn  
  ).  
  MESSAGE 'Redshift cluster modified successfully.' TYPE 'I'.  
CATCH /aws1/cx_rshclustnotfoundfault.  
  MESSAGE 'Cluster not found.' TYPE 'I'.  
CATCH /aws1/cx_rshinvcluststatefault.  
  MESSAGE 'Invalid cluster state for modification.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ModifyCluster](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon Rekognition usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Rekognition.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Ações](#)

## Ações

### CompareFaces

O código de exemplo a seguir mostra como usar CompareFaces.

Para obter mais informações, consulte [Comparação de faces em imagens](#).

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Create S3 object reference for the source image  
  DATA(lo_source_s3obj) = NEW /aws1/cl_reks3object(  
    iv_bucket = iv_source_s3_bucket  
    iv_name = iv_source_s3_key ).  
  
  " Create source image object  
  DATA(lo_source_image) = NEW /aws1/cl_rekimage(  
    io_s3object = lo_source_s3obj ).  
  
  " Create S3 object reference for the target image  
  DATA(lo_target_s3obj) = NEW /aws1/cl_reks3object(  
    iv_bucket = iv_target_s3_bucket  
    iv_name = iv_target_s3_key ).  
  
  " Create target image object  
  DATA(lo_target_image) = NEW /aws1/cl_rekimage(  
    io_s3object = lo_target_s3obj ).  
  
  " Compare faces  
  oo_result = lo_rek->comparefaces(  

```

```
io_sourceimage = lo_source_image
io_targetimage = lo_target_image
iv_similaritythreshold = iv_similarity ).

DATA(lt_face_matches) = oo_result->get_facematches( ).
DATA(lt_unmatched_faces) = oo_result->get_unmatchedfaces( ).

" Get counts of matched and unmatched faces
DATA(lv_matched_count) = lines( lt_face_matches ).
DATA(lv_unmatched_count) = lines( lt_unmatched_faces ).

" Output detailed comparison results
DATA(lv_message) = |Face comparison completed: | &&
                  |{ lv_matched_count } matched face(s), | &&
                  |{ lv_unmatched_count } unmatched face(s).|.
MESSAGE lv_message TYPE 'I'.
CATCH /aws1/cx_rekinvalids3objectex.
  MESSAGE 'Invalid S3 object.' TYPE 'E'.
CATCH /aws1/cx_rekinvalidparameterex.
  MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CompareFaces](#) referência da API AWS SDK for SAP ABAP.

## CreateCollection

O código de exemplo a seguir mostra como usar CreateCollection.

Para obter mais informações, consulte [Criar uma coleção](#).

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

TRY.

```
oo_result = lo_rek->createcollection(  
    iv_collectionid = iv_collection_id ).  
MESSAGE 'Collection created successfully.' TYPE 'I'.  
CATCH /aws1/cx_rekresrcalrdyexistsex.  
MESSAGE 'Collection already exists.' TYPE 'E'.  
CATCH /aws1/cx_rekinvalidparameterex.  
MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateCollection](#) referência da API AWS SDK for SAP ABAP.

## DeleteCollection

O código de exemplo a seguir mostra como usar DeleteCollection.

Para obter mais informações, consulte [Excluir uma coleção](#).

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_rek->deletecollection(  
        iv_collectionid = iv_collection_id ).  
MESSAGE 'Collection deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_rekresourcenotfoundex.  
MESSAGE 'Collection not found.' TYPE 'E'.  
CATCH /aws1/cx_rekinvalidparameterex.  
MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteCollection](#) referência da API AWS SDK for SAP ABAP.

## DeleteFaces

O código de exemplo a seguir mostra como usar DeleteFaces.

Para obter mais informações, consulte [Excluir faces de uma coleção](#).

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_rek->deletefaces(  
    iv_collectionid = iv_collection_id  
    it_faceids = it_face_ids ).  
  
  DATA(lt_deleted_faces) = oo_result->get_deletedfaces( ).  
  DATA(lv_deleted_count) = lines( lt_deleted_faces ).  
  DATA(lv_msg6) = |{ lv_deleted_count } face(s) deleted successfully.|.  
  MESSAGE lv_msg6 TYPE 'I'.  
CATCH /aws1/cx_rekresourcenotfoundex.  
  MESSAGE 'Collection not found.' TYPE 'E'.  
CATCH /aws1/cx_rekinvalidparameterex.  
  MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteFaces](#) referência da API AWS SDK for SAP ABAP.

## DescribeCollection

O código de exemplo a seguir mostra como usar DescribeCollection.

Para obter mais informações, consulte [Descrever uma coleção](#).

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_rek->describecollection(  
        iv_collectionid = iv_collection_id ).  
    DATA(lv_face_count) = oo_result->get_facecount( ).  
    DATA(lv_msg) = |Collection described: { lv_face_count } face(s) indexed.|.  
    MESSAGE lv_msg TYPE 'I'.  
    CATCH /aws1/cx_rekresourcenotfoundex.  
        MESSAGE 'Collection not found.' TYPE 'E'.  
    CATCH /aws1/cx_rekinvalidparameterex.  
        MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeCollection](#) referência da API AWS SDK for SAP ABAP.

**DetectFaces**

O código de exemplo a seguir mostra como usar DetectFaces.

Para obter mais informações, consulte [Detectar faces em uma imagem](#).

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
" Create S3 object reference for the image
DATA(lo_s3object) = NEW /aws1/cl_reks3object(
  iv_bucket = iv_s3_bucket
  iv_name = iv_s3_key ).

" Create image object
DATA(lo_image) = NEW /aws1/cl_rekimage(
  io_s3object = lo_s3object ).

" Detect faces in the image with all attributes
DATA(lt_attributes) = VALUE /aws1/cl_rekattributes_w=>tt_attributes( ).
DATA(lo_attr_wrapper) = NEW /aws1/cl_rekattributes_w( iv_value = 'ALL' ).
INSERT lo_attr_wrapper INTO TABLE lt_attributes.

oo_result = lo_rek->detectfaces(
  io_image = lo_image
  it_attributes = lt_attributes ).

DATA(lt_face_details) = oo_result->get_facedetails( ).
DATA(lv_detected_count) = lines( lt_face_details ).
DATA(lv_msg8) = ||{ lv_detected_count } face(s) detected in image.|.
MESSAGE lv_msg8 TYPE 'I'.
CATCH /aws1/cx_rekinvalids3objectex.
  MESSAGE 'Invalid S3 object.' TYPE 'E'.
CATCH /aws1/cx_rekinvalidparameterex.
  MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.
```


- Para obter detalhes da API, consulte a [DetectFaces](#) referência da API AWS SDK for SAP ABAP.

## DetectLabels

O código de exemplo a seguir mostra como usar DetectLabels.

Para obter mais informações, consulte [Detectar rótulos em uma imagem](#).

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create S3 object reference for the image
  DATA(lo_s3object) = NEW /aws1/cl_reks3object(
    iv_bucket = iv_s3_bucket
    iv_name = iv_s3_key ).

  " Create image object
  DATA(lo_image) = NEW /aws1/cl_rekimage(
    io_s3object = lo_s3object ).

  " Detect labels in the image
  oo_result = lo_rek->detectlabels(
    io_image = lo_image
    iv_maxlabels = iv_max_labels ).

  DATA(lt_labels) = oo_result->get_labels( ).
  DATA(lv_label_count) = lines( lt_labels ).
  DATA(lv_msg9) = |{ lv_label_count } label(s) detected in image.|.
  MESSAGE lv_msg9 TYPE 'I'.
CATCH /aws1/cx_rekinvalids3objectex.
  MESSAGE 'Invalid S3 object.' TYPE 'E'.
CATCH /aws1/cx_rekinvalidparameterex.
  MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectLabels](#) referência da API AWS SDK for SAP ABAP.

## DetectModerationLabels

O código de exemplo a seguir mostra como usar DetectModerationLabels.

Para obter mais informações, consulte [Detectar imagens impróprias](#).

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create S3 object reference for the image
  DATA(lo_s3object) = NEW /aws1/cl_reks3object(
    iv_bucket = iv_s3_bucket
    iv_name = iv_s3_key ).

  " Create image object
  DATA(lo_image) = NEW /aws1/cl_rekimage(
    io_s3object = lo_s3object ).

  " Detect moderation labels
  oo_result = lo_rek->detectmoderationlabels(
    io_image = lo_image ).

  DATA(lt_moderation_labels) = oo_result->get_moderationlabels( ).
  DATA(lv_mod_count) = lines( lt_moderation_labels ).
  DATA(lv_msg10) = |{ lv_mod_count } moderation label(s) detected.|.
  MESSAGE lv_msg10 TYPE 'I'.
CATCH /aws1/cx_rekinvalids3objectex.
  MESSAGE 'Invalid S3 object.' TYPE 'E'.
CATCH /aws1/cx_rekinvalidparameterex.
  MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DetectModerationLabels](#) referência da API AWS SDK for SAP ABAP.

## DetectText

O código de exemplo a seguir mostra como usar DetectText.

Para obter mais informações, consulte [Detectar texto em uma imagem](#).

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Create S3 object reference for the image  
  DATA(lo_s3object) = NEW /aws1/cl_reks3object(  
    iv_bucket = iv_s3_bucket  
    iv_name = iv_s3_key ).  
  
  " Create image object  
  DATA(lo_image) = NEW /aws1/cl_rekimage(  
    io_s3object = lo_s3object ).  
  
  " Detect text in the image  
  oo_result = lo_rek->detecttext(  
    io_image = lo_image ).  
  
  DATA(lt_text_detections) = oo_result->get_textdetections( ).  
  DATA(lv_text_count) = lines( lt_text_detections ).  
  DATA(lv_msg11) = |{ lv_text_count } text detection(s) found.|.  
  MESSAGE lv_msg11 TYPE 'I'.  
  CATCH /aws1/cx_rekinvalids3objectex.  
    MESSAGE 'Invalid S3 object.' TYPE 'E'.  
  CATCH /aws1/cx_rekinvalidparameterex.  
    MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```


- Para obter detalhes da API, consulte a [DetectText](#) referência da API AWS SDK for SAP ABAP.

## IndexFaces

O código de exemplo a seguir mostra como usar IndexFaces.

Para obter mais informações, consulte [Adicionar faces a uma coleção](#).

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Create S3 object reference for the image
  DATA(lo_s3object) = NEW /aws1/cl_reks3object(
    iv_bucket = iv_s3_bucket
    iv_name = iv_s3_key ).

  " Create image object
  DATA(lo_image) = NEW /aws1/cl_rekimage(
    io_s3object = lo_s3object ).

  " Index faces in the image
  oo_result = lo_rek->indexfaces(
    iv_collectionid = iv_collection_id
    io_image = lo_image
    iv_externalimageid = iv_external_id
    iv_maxfaces = iv_max_faces ).

  DATA(lt_face_records) = oo_result->get_facerecords( ).
  DATA(lv_indexed_count) = lines( lt_face_records ).
  DATA(lv_msg2) = |{ lv_indexed_count } face(s) indexed successfully.|.
  MESSAGE lv_msg2 TYPE 'I'.
CATCH /aws1/cx_rekresourcenotfoundex.
  MESSAGE 'Collection not found.' TYPE 'E'.
CATCH /aws1/cx_rekinvalids3objectex.
  MESSAGE 'Invalid S3 object.' TYPE 'E'.
CATCH /aws1/cx_rekinvalidparameterex.
  MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [IndexFaces](#) referência da API AWS SDK for SAP ABAP.

## ListCollections

O código de exemplo a seguir mostra como usar `ListCollections`.

Para obter mais informações, consulte [Listar coleções](#).

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_rek->listcollections(  
        iv_maxresults = iv_max_results ).  
  
    DATA(lt_collection_ids) = oo_result->get_collectionids( ).  
    DATA(lv_coll_count) = lines( lt_collection_ids ).  
    DATA(lv_msg7) = |{ lv_coll_count } collection(s) found.|.  
    MESSAGE lv_msg7 TYPE 'I'.  
CATCH /aws1/cx_rekinvalidparameterex.  
    MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```


- Para obter detalhes da API, consulte a [ListCollections](#) referência da API AWS SDK for SAP ABAP.

## ListFaces

O código de exemplo a seguir mostra como usar `ListFaces`.

Para obter mais informações, consulte [Listar faces em uma coleção](#).

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_rek->listfaces(  
    iv_collectionid = iv_collection_id  
    iv_maxresults = iv_max_results ).  
  
  DATA(lt_faces) = oo_result->get_faces( ).  
  DATA(lv_face_count2) = lines( lt_faces ).  
  DATA(lv_msg3) = |{ lv_face_count2 } face(s) found in collection.|.  
  MESSAGE lv_msg3 TYPE 'I'.  
  CATCH /aws1/cx_rekresourcenotfoundex.  
    MESSAGE 'Collection not found.' TYPE 'E'.  
  CATCH /aws1/cx_rekinvalidparameterex.  
    MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```


- Para obter detalhes da API, consulte a [ListFaces](#) referência da API AWS SDK for SAP ABAP.

## RecognizeCelebrities

O código de exemplo a seguir mostra como usar RecognizeCelebrities.

Para obter mais informações, consulte [Reconhecer celebridades em uma imagem](#).

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Create S3 object reference for the image  
  DATA(lo_s3object) = NEW /aws1/cl_reks3object(  
    iv_bucket = iv_s3_bucket  
    iv_name = iv_s3_key ).  
  
  " Create image object  
  DATA(lo_image) = NEW /aws1/cl_rekimage(  
    io_s3object = lo_s3object ).  
  
  " Recognize celebrities  
  oo_result = lo_rek->recognizecelebrities(  
    io_image = lo_image ).  
  
  DATA(lt_celebrity_faces) = oo_result->get_celebrityfaces( ).  
  DATA(lv_celeb_count) = lines( lt_celebrity_faces ).  
  DATA(lv_msg12) = |{ lv_celeb_count } celebrity/celebrities recognized.|.  
  MESSAGE lv_msg12 TYPE 'I'.  
  CATCH /aws1/cx_rekinvalids3objectex.  
    MESSAGE 'Invalid S3 object.' TYPE 'E'.  
  CATCH /aws1/cx_rekinvalidparameterex.  
    MESSAGE 'Invalid parameter value.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [RecognizeCelebrities](#) referência da API AWS SDK for SAP ABAP.

## SearchFaces

O código de exemplo a seguir mostra como usar SearchFaces.

Para obter mais informações, consulte [Pesquisar uma face \(face ID\)](#).

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  oo_result = lo_rek->searchfaces(
    iv_collectionid = iv_collection_id
    iv_faceid = iv_face_id
    iv_facematchthreshold = iv_threshold
    iv_maxfaces = iv_max_faces ).

  DATA(lt_face_matches) = oo_result->get_facematches( ).
  DATA(lv_match_count2) = lines( lt_face_matches ).
  DATA(lv_msg5) = |Face search completed: { lv_match_count2 } match(es)
found.|.
  MESSAGE lv_msg5 TYPE 'I'.
  CATCH /aws1/cx_rekresourcenotfoundex.
    MESSAGE 'Collection or face not found.' TYPE 'E'.
  CATCH /aws1/cx_rekinvalidparameterex.
    MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [SearchFaces](#) referência da API AWS SDK for SAP ABAP.

## SearchFacesByImage

O código de exemplo a seguir mostra como usar SearchFacesByImage.

Para obter mais informações, consulte [Pesquisar uma face \(imagem\)](#).

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Create S3 object reference for the image
  DATA(lo_s3object) = NEW /aws1/cl_reks3object(
    iv_bucket = iv_s3_bucket
    iv_name = iv_s3_key ).

```

```
" Create image object
DATA(lo_image) = NEW /aws1/cl_rekimage(
  io_s3object = lo_s3object ).

" Search for matching faces
oo_result = lo_rek->searchfacesbyimage(
  iv_collectionid = iv_collection_id
  io_image = lo_image
  iv_facematchthreshold = iv_threshold
  iv_maxfaces = iv_max_faces ).

DATA(lt_face_matches) = oo_result->get_facematches( ).
DATA(lv_match_count) = lines( lt_face_matches ).
DATA(lv_msg4) = |Face search completed: { lv_match_count } match(es)
found.|.
MESSAGE lv_msg4 TYPE 'I'.
CATCH /aws1/cx_rekresourcenotfoundex.
MESSAGE 'Collection not found.' TYPE 'E'.
CATCH /aws1/cx_rekinvalids3objectex.
MESSAGE 'Invalid S3 object.' TYPE 'E'.
CATCH /aws1/cx_rekinvalidparameterex.
MESSAGE 'Invalid parameter value.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [SearchFacesByImage](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon S3 usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon S3.

As noções básicas são exemplos de código que mostram como realizar as operações essenciais em um serviço.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Conceitos básicos](#)
- [Ações](#)
- [Cenários](#)

## Conceitos básicos

Conheça os conceitos básicos

O exemplo de código a seguir mostra como:

- Criar um bucket e fazer upload de um arquivo para ele.
- Baixar um objeto de um bucket.
- Copiar um objeto em uma subpasta em um bucket.
- Listar os objetos em um bucket.
- Exclua os objetos do bucket e o bucket.

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA(lo_session) = /aws1/cl_rt_session_aws=>create( cv_pfl ).
DATA(lo_s3) = /aws1/cl_s3_factory=>create( lo_session ).

" Create an Amazon Simple Storage Service (Amazon S3) bucket. "
TRY.
    " determine our region from our session
```

```
DATA(lv_region) = CONV /aws1/s3_bucketlocationcnstrnt( lo_session-
>get_region( ) ).
DATA lo_constraint TYPE REF TO /aws1/cl_s3_createbucketconf.
" When in the us-east-1 region, you must not specify a constraint
" In all other regions, specify the region as the constraint
IF lv_region = 'us-east-1'.
  CLEAR lo_constraint.
ELSE.
  lo_constraint = NEW /aws1/cl_s3_createbucketconf( lv_region ).
ENDIF.

lo_s3->createbucket(
  iv_bucket = iv_bucket_name
  io_createbucketconfiguration = lo_constraint ).
MESSAGE 'S3 bucket created.' TYPE 'I'.
CATCH /aws1/cx_s3_bucketalrddyexists.
  MESSAGE 'Bucket name already exists.' TYPE 'E'.
CATCH /aws1/cx_s3_bktalrddyownedbyyou.
  MESSAGE 'Bucket already exists and is owned by you.' TYPE 'E'.
ENDTRY.

"Upload an object to an S3 bucket."
TRY.
  "Get contents of file from application server."
  DATA lv_file_content TYPE xstring.
  OPEN DATASET iv_key FOR INPUT IN BINARY MODE.
  READ DATASET iv_key INTO lv_file_content.
  CLOSE DATASET iv_key.

  lo_s3->putobject(
    iv_bucket = iv_bucket_name
    iv_key = iv_key
    iv_body = lv_file_content ).
  MESSAGE 'Object uploaded to S3 bucket.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
ENDTRY.

" Get an object from a bucket. "
TRY.
  DATA(lo_result) = lo_s3->getobject(
    iv_bucket = iv_bucket_name
    iv_key = iv_key ).
```

```
DATA(lv_object_data) = lo_result->get_body( ).
MESSAGE 'Object retrieved from S3 bucket.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
MESSAGE 'Bucket does not exist.' TYPE 'E'.
CATCH /aws1/cx_s3_nosuchkey.
MESSAGE 'Object key does not exist.' TYPE 'E'.
ENDTRY.

" Copy an object to a subfolder in a bucket. "
TRY.
  lo_s3->copyobject(
    iv_bucket = iv_bucket_name
    iv_key = |{ iv_copy_to_folder }/{ iv_key }|
    iv_copysource = |{ iv_bucket_name }/{ iv_key }| ).
  MESSAGE 'Object copied to a subfolder.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
  MESSAGE 'Bucket does not exist.' TYPE 'E'.
  CATCH /aws1/cx_s3_nosuchkey.
  MESSAGE 'Object key does not exist.' TYPE 'E'.
ENDTRY.

" List objects in the bucket. "
TRY.
  DATA(lo_list) = lo_s3->listobjects(
    iv_bucket = iv_bucket_name ).
  MESSAGE 'Retrieved list of objects in S3 bucket.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
  MESSAGE 'Bucket does not exist.' TYPE 'E'.
ENDTRY.
DATA text TYPE string VALUE 'Object List - '.
DATA lv_object_key TYPE /aws1/s3_objectkey.
LOOP AT lo_list->get_contents( ) INTO DATA(lo_object).
  lv_object_key = lo_object->get_key( ).
  CONCATENATE lv_object_key ', ' INTO text.
ENDLOOP.
MESSAGE text TYPE 'I'.

" Delete the objects in a bucket. "
TRY.
  lo_s3->deleteobject(
    iv_bucket = iv_bucket_name
    iv_key = iv_key ).
  lo_s3->deleteobject(
    iv_bucket = iv_bucket_name
```

```
        iv_key = |{ iv_copy_to_folder }/{ iv_key }| ).
    MESSAGE 'Objects deleted from S3 bucket.' TYPE 'I'.
    CATCH /aws1/cx_s3_nosuchbucket.
        MESSAGE 'Bucket does not exist.' TYPE 'E'.
    ENDTRY.

" Delete the bucket. "
TRY.
    lo_s3->deletebucket(
        iv_bucket = iv_bucket_name ).
    MESSAGE 'Deleted S3 bucket.' TYPE 'I'.
    CATCH /aws1/cx_s3_nosuchbucket.
        MESSAGE 'Bucket does not exist.' TYPE 'E'.
    ENDTRY.
```


- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [CopyObject](#)
  - [CreateBucket](#)
  - [DeleteBucket](#)
  - [DeleteObjects](#)
  - [GetObject](#)
  - [ListObjectsV2](#)
  - [PutObject](#)

## Ações

### CopyObject

O código de exemplo a seguir mostra como usar CopyObject.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_s3->copyobject(  
    iv_bucket = iv_dest_bucket  
    iv_key = iv_dest_object  
    iv_copysource = |{ iv_src_bucket }/{ iv_src_object }| ).  
  MESSAGE 'Object copied to another bucket.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
CATCH /aws1/cx_s3_nosuchkey.  
  MESSAGE 'Object key does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CopyObject](#) referência da API AWS SDK for SAP ABAP.

**CreateBucket**

O código de exemplo a seguir mostra como usar CreateBucket.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " determine our region from our session  
  DATA(lv_region) = CONV /aws1/s3_bucketlocationcnstrnt( lo_session-  
>get_region( ) ).  
  DATA lo_constraint TYPE REF TO /aws1/cl_s3_createbucketconf.
```

```
" When in the us-east-1 region, you must not specify a constraint
" In all other regions, specify the region as the constraint
IF lv_region = 'us-east-1'.
  CLEAR lo_constraint.
ELSE.
  lo_constraint = NEW /aws1/cl_s3_createbucketconf( lv_region ).
ENDIF.

lo_s3->createbucket(
  iv_bucket = iv_bucket_name
  io_createbucketconfiguration = lo_constraint ).
MESSAGE 'S3 bucket created.' TYPE 'I'.
CATCH /aws1/cx_s3_bucketalrddyexists.
  MESSAGE 'Bucket name already exists.' TYPE 'E'.
CATCH /aws1/cx_s3_bktalrddyownedbyyou.
  MESSAGE 'Bucket already exists and is owned by you.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateBucket](#) referência da API AWS SDK for SAP ABAP.

## DeleteBucket

O código de exemplo a seguir mostra como usar DeleteBucket.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
```

```
lo_s3->deletebucket(
  iv_bucket = iv_bucket_name ).
MESSAGE 'Deleted S3 bucket.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
  MESSAGE 'Bucket does not exist.' TYPE 'E'.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteBucket](#) referência da API AWS SDK for SAP ABAP.

## DeleteBucketCors

O código de exemplo a seguir mostra como usar DeleteBucketCors.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_s3->deletebucketcors(  
    iv_bucket = iv_bucket_name ).  
  MESSAGE 'Bucket CORS configuration deleted.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteBucketCors](#) referência da API AWS SDK for SAP ABAP.

## DeleteBucketLifecycle

O código de exemplo a seguir mostra como usar DeleteBucketLifecycle.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_s3->deletebucketlifecycle(  
    iv_bucket = iv_bucket_name ).  
  MESSAGE 'Bucket lifecycle configuration deleted.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteBucketLifecycle](#) referência da API AWS SDK for SAP ABAP.

**DeleteBucketPolicy**

O código de exemplo a seguir mostra como usar DeleteBucketPolicy.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_s3->deletebucketpolicy(  
    iv_bucket = iv_bucket_name ).  
  MESSAGE 'Bucket policy deleted.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteBucketPolicy](#) referência da API AWS SDK for SAP ABAP.

## DeleteObject

O código de exemplo a seguir mostra como usar DeleteObject.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_s3->deleteobject(  
    iv_bucket = iv_bucket_name  
    iv_key = iv_object_key ).  
  MESSAGE 'Object deleted from S3 bucket.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteObject](#) referência da API AWS SDK for SAP ABAP.

## DeleteObjects

O código de exemplo a seguir mostra como usar DeleteObjects.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_s3->deleteobjects(           " oo_result is returned for
testing purposes. "
    iv_bucket = iv_bucket_name
    io_delete = NEW /aws1/cl_s3_delete( it_objects = it_object_keys ) ).
    MESSAGE 'Objects deleted from S3 bucket.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteObjects](#) referência da API AWS SDK for SAP ABAP.

## GetBucketAcl

O código de exemplo a seguir mostra como usar GetBucketAcl.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_s3->getbucketacl(           " oo_result is returned for testing
purposes. "
    iv_bucket = iv_bucket_name ).
    MESSAGE 'Retrieved bucket ACL.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [GetBucketAcl](#) referência da API AWS SDK for SAP ABAP.

## GetBucketCors

O código de exemplo a seguir mostra como usar `GetBucketCors`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_s3->getbucketcors(          " oo_result is returned for  
testing purposes. "  
    iv_bucket = iv_bucket_name ).  
    MESSAGE 'Retrieved bucket CORS configuration.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetBucketCors](#) referência da API AWS SDK for SAP ABAP.

## GetBucketLifecycleConfiguration

O código de exemplo a seguir mostra como usar `GetBucketLifecycleConfiguration`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_s3->getbucketlifecycleconf(          " oo_result is returned  
for testing purposes. "
```

```
        iv_bucket = iv_bucket_name ).  
        MESSAGE 'Retrieved bucket lifecycle configuration.' TYPE 'I'.  
    CATCH /aws1/cx_s3_nosuchbucket.  
        MESSAGE 'Bucket does not exist.' TYPE 'E'.  
    ENDTRY.
```

- Para obter detalhes da API, consulte a [GetBucketLifecycleConfiguration](#) referência da API AWS SDK for SAP ABAP.

## GetBucketPolicy

O código de exemplo a seguir mostra como usar `GetBucketPolicy`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
    TRY.  
        oo_result = lo_s3->getbucketpolicy(           " oo_result is returned for  
testing purposes. "  
        iv_bucket = iv_bucket_name ).  
        DATA(lv_policy) = oo_result->get_policy( ).  
        MESSAGE 'Retrieved bucket policy.' TYPE 'I'.  
    CATCH /aws1/cx_s3_nosuchbucket.  
        MESSAGE 'Bucket does not exist.' TYPE 'E'.  
    ENDTRY.
```

- Para obter detalhes da API, consulte a [GetBucketPolicy](#) referência da API AWS SDK for SAP ABAP.

## GetObject

O código de exemplo a seguir mostra como usar `GetObject`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
    oo_result = lo_s3->getobject(          " oo_result is returned for testing
purposes. "
        iv_bucket = iv_bucket_name
        iv_key = iv_object_key ).
    DATA(lv_object_data) = oo_result->get_body( ).
    MESSAGE 'Object retrieved from S3 bucket.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
CATCH /aws1/cx_s3_nosuchkey.
    MESSAGE 'Object key does not exist.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetObject](#) referência da API AWS SDK for SAP ABAP.

## GetObjectAcl

O código de exemplo a seguir mostra como usar GetObjectAcl.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_s3->getobjectacl(      " oo_result is returned for testing
purposes. "
```

```
        iv_bucket = iv_bucket_name
        iv_key = iv_object_key ).
    MESSAGE 'Retrieved object ACL.' TYPE 'I'.
    CATCH /aws1/cx_s3_nosuchbucket.
        MESSAGE 'Bucket does not exist.' TYPE 'E'.
    CATCH /aws1/cx_s3_nosuchkey.
        MESSAGE 'Object key does not exist.' TYPE 'E'.
    ENDRY.
```

- Para obter detalhes da API, consulte a [GetObjectAcl](#) referência da API AWS SDK for SAP ABAP.

## GetObjectLegalHold

O código de exemplo a seguir mostra como usar `GetObjectLegalHold`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
    TRY.
        oo_result = lo_s3->getobjectlegalhold(           " oo_result is returned for
testing purposes. "
        iv_bucket = iv_bucket_name
        iv_key = iv_object_key ).
        MESSAGE 'Retrieved object legal hold status.' TYPE 'I'.
    CATCH /aws1/cx_s3_nosuchbucket.
        MESSAGE 'Bucket does not exist.' TYPE 'E'.
    CATCH /aws1/cx_s3_nosuchkey.
        MESSAGE 'Object key does not exist.' TYPE 'E'.
    ENDRY.
```

- Para obter detalhes da API, consulte a [GetObjectLegalHold](#) referência da API AWS SDK for SAP ABAP.

## GetObjectLockConfiguration

O código de exemplo a seguir mostra como usar `GetObjectLockConfiguration`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_s3->getobjectlockconfiguration(           " oo_result is  
returned for testing purposes. "  
    iv_bucket = iv_bucket_name ).  
    MESSAGE 'Retrieved object lock configuration.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetObjectLockConfiguration](#) referência da API AWS SDK for SAP ABAP.

## HeadBucket

O código de exemplo a seguir mostra como usar `HeadBucket`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_s3->headbucket(           " oo_result is returned for testing  
purposes. "
```

```
        iv_bucket = iv_bucket_name ).
    MESSAGE 'Bucket exists and you have access to it.' TYPE 'I'.
    CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
    ENDTRY.
```

- Para obter detalhes da API, consulte a [HeadBucket](#) referência da API AWS SDK for SAP ABAP.

## ListBuckets

O código de exemplo a seguir mostra como usar ListBuckets.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
    TRY.
        oo_result = lo_s3->listbuckets(          " oo_result is returned for testing
purposes. "
        ).
        DATA(lv_bucket_count) = lines( oo_result->get_buckets( ) ).
        MESSAGE |Retrieved { lv_bucket_count } buckets in all regions.| TYPE 'I'.
    CATCH /aws1/cx_rt_generic.
        MESSAGE 'Unable to list buckets.' TYPE 'E'.
    ENDTRY.
```

- Para obter detalhes da API, consulte a [ListBuckets](#) referência da API AWS SDK for SAP ABAP.

## ListObjectVersions

O código de exemplo a seguir mostra como usar ListObjectVersions.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_s3->listobjectversions(           " oo_result is returned for  
testing purposes. "  
        iv_bucket = iv_bucket_name  
        iv_prefix = iv_prefix ).  
    MESSAGE 'Retrieved object versions.' TYPE 'I'.  
CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListObjectVersions](#) referência da API AWS SDK for SAP ABAP.

## ListObjectsV2

O código de exemplo a seguir mostra como usar ListObjectsV2.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_s3->listobjectsv2(           " oo_result is returned for  
testing purposes. "  
        iv_bucket = iv_bucket_name ).  
    MESSAGE 'Retrieved list of objects in S3 bucket.' TYPE 'I'.
```

```
CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte [ListObjectsV2 na referência](#) da API ABAP do AWS SDK para SAP.

## PutBucketAcl

O código de exemplo a seguir mostra como usar PutBucketAcl.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " Example: Grant log delivery access to a bucket  
  " iv_grantwrite = 'uri=http://acs.amazonaws.com/groups/s3/LogDelivery'  
  lo_s3->putbucketacl(  
    iv_bucket = iv_bucket_name  
    iv_grantwrite = iv_grantwrite ).  
  MESSAGE 'Bucket ACL updated.' TYPE 'I'.  
  CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
  ENDTRY.
```

- Para obter detalhes da API, consulte a [PutBucketAcl](#) referência da API AWS SDK for SAP ABAP.

## PutBucketCors

O código de exemplo a seguir mostra como usar PutBucketCors.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  " Example: Allow PUT, POST, DELETE methods from http://www.example.com  
  lo_s3->putbucketcors(  
    iv_bucket = iv_bucket_name  
    io_corsconfiguration = NEW /aws1/cl_s3_corsconfiguration(  
      it_corsrules = it_cors_rules ) ).  
  MESSAGE 'Bucket CORS configuration set.' TYPE 'I'.  
  CATCH /aws1/cx_s3_nosuchbucket.  
  MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutBucketCors](#) referência da API AWS SDK for SAP ABAP.

## PutBucketLifecycleConfiguration

O código de exemplo a seguir mostra como usar PutBucketLifecycleConfiguration.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Example: Expire objects with prefix 'logs/' after 30 days  
  lo_s3->putbucketlifecycleconf(  
    iv_bucket = iv_bucket_name
```

```

    io_lifecycleconfiguration = NEW /aws1/cl_s3_bucketlconf(
      it_rules = it_lifecycle_rule ) ).
  MESSAGE 'Bucket lifecycle configuration set.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
  MESSAGE 'Bucket does not exist.' TYPE 'E'.
  ENDRY.

```

- Para obter detalhes da API, consulte a [PutBucketLifecycleConfiguration](#) referência da API AWS SDK for SAP ABAP.

## PutBucketPolicy

O código de exemplo a seguir mostra como usar PutBucketPolicy.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

  TRY.
    " Example policy JSON string
    " iv_policy = '{"Version":"2012-10-17",          "Statement":
  [{"Effect":"Allow","Principal":{"AWS":"arn:aws:iam::123456789012:user/
  user"},"Action":["s3:GetObject"],"Resource":["arn:aws:s3:::bucketname/*"]}]}'
    lo_s3->putbucketpolicy(
      iv_bucket = iv_bucket_name
      iv_policy = iv_policy ).
    MESSAGE 'Bucket policy set.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
  ENDRY.

```

- Para obter detalhes da API, consulte a [PutBucketPolicy](#) referência da API AWS SDK for SAP ABAP.

## PutBucketVersioning

O código de exemplo a seguir mostra como usar PutBucketVersioning.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Example: Enable versioning on a bucket  
  " iv_status = 'Enabled'  
  lo_s3->putbucketversioning(  
    iv_bucket = iv_bucket_name  
    io_versioningconfiguration = NEW /aws1/cl_s3_versioningconf(  
      iv_status = iv_status ) ).  
  MESSAGE 'Bucket versioning enabled.' TYPE 'I'.  
  CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
  ENDTRY.
```

- Para obter detalhes da API, consulte a [PutBucketVersioning](#) referência da API AWS SDK for SAP ABAP.

## PutObject

O código de exemplo a seguir mostra como usar PutObject.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Get contents of file from application server."
DATA lv_body TYPE xstring.
OPEN DATASET iv_file_name FOR INPUT IN BINARY MODE.
READ DATASET iv_file_name INTO lv_body.
CLOSE DATASET iv_file_name.

"Upload/put an object to an S3 bucket."
TRY.
    lo_s3->putobject(
        iv_bucket = iv_bucket_name
        iv_key = iv_file_name
        iv_body = lv_body ).
    MESSAGE 'Object uploaded to S3 bucket.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutObject](#) referência da API AWS SDK for SAP ABAP.

## PutObjectAc1

O código de exemplo a seguir mostra como usar PutObjectAc1.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    " Example: Grant read access to an AWS user
    " iv_grantread = 'emailAddress=user@example.com'
    lo_s3->putobjectacl(
        iv_bucket = iv_bucket_name
        iv_key = iv_object_key
        iv_grantread = iv_grantread ).
```

```

    MESSAGE 'Object ACL updated.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
  CATCH /aws1/cx_s3_nosuchkey.
    MESSAGE 'Object key does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [PutObjectAcl](#) referência da API AWS SDK for SAP ABAP.

## PutObjectLegalHold

O código de exemplo a seguir mostra como usar PutObjectLegalHold.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Example: Set legal hold status to ON
  " iv_status = 'ON'
  lo_s3->putobjectlegalhold(
    iv_bucket = iv_bucket_name
    iv_key = iv_object_key
    io_legalhold = NEW /aws1/cl_s3_objlocklegalhold(
      iv_status = iv_status ) ).
  MESSAGE 'Object legal hold status set.' TYPE 'I'.
  CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
  CATCH /aws1/cx_s3_nosuchkey.
    MESSAGE 'Object key does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [PutObjectLegalHold](#) referência da API AWS SDK for SAP ABAP.

## PutObjectLockConfiguration

O código de exemplo a seguir mostra como usar PutObjectLockConfiguration.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Example: Enable object lock with default retention  
  " iv_enabled = 'Enabled'  
  lo_s3->putobjectlockconfiguration(  
    iv_bucket = iv_bucket_name  
    io_objectlockconfiguration = NEW /aws1/cl_s3_objectlockconf(  
      iv_objectlockenabled = iv_enabled ) ).  
  MESSAGE 'Object lock configuration set.' TYPE 'I'.  
  CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutObjectLockConfiguration](#) referência da API AWS SDK for SAP ABAP.

## PutObjectRetention

O código de exemplo a seguir mostra como usar PutObjectRetention.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " Example: Set retention mode to GOVERNANCE for 30 days  
  " iv_mode = 'GOVERNANCE'  
  " iv_retain_date should be a timestamp in the future  
  lo_s3->putobjectretention(  
    iv_bucket = iv_bucket_name  
    iv_key = iv_object_key  
    io_retention = NEW /aws1/cl_s3_objectlockret(  
      iv_mode = iv_mode  
      iv_retainuntildate = iv_retain_date )  
    iv_bypassgovernanceretention = abap_true ).  
  MESSAGE 'Object retention set.' TYPE 'I'.  
  CATCH /aws1/cx_s3_nosuchbucket.  
    MESSAGE 'Bucket does not exist.' TYPE 'E'.  
  CATCH /aws1/cx_s3_nosuchkey.  
    MESSAGE 'Object key does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [PutObjectRetention](#) referência da API AWS SDK for SAP ABAP.

## Cenários

### Criar um URL pré-assinado

O exemplo de código a seguir mostra como criar um URL pré-assinado para o Amazon S3 e fazer upload de um objeto.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie solicitações pré-assinadas para objetos GET do S3.

```
" iv_bucket_name is the bucket name  
" iv_key is the object name like "myfile.txt"
```

```
DATA(lo_session) = /aws1/cl_rt_session_aws=>create( cv_pfl ).
DATA(lo_s3) = /aws1/cl_s3_factory=>create( lo_session ).

"Upload a nice Hello World file to an S3 bucket."
TRY.
    DATA(lv_contents) = cl_abap_codepage=>convert_to( 'Hello, World' ).
    lo_s3->putobject(
        iv_bucket = iv_bucket_name
        iv_key = iv_key
        iv_body = lv_contents
        iv_contenttype = 'text/plain' ).
    MESSAGE 'Object uploaded to S3 bucket.' TYPE 'I'.
CATCH /aws1/cx_s3_nosuchbucket.
    MESSAGE 'Bucket does not exist.' TYPE 'E'.
ENDTRY.

" now generate a presigned URL with a 600-second expiration
DATA(lo_presigner) = lo_s3->get_presigner( iv_expires_sec = 600 ).
" the presigner getobject() method has the same signature as
" lo_s3->getobject(), but it doesn't actually make the call.
" to the service. It just prepares a presigned URL for a future call
DATA(lo_presigned_req) = lo_presigner->getobject(
    iv_bucket = iv_bucket_name
    iv_key = iv_key ).

" You can provide this URL to a web page, user, email etc so they
" can retrieve the file. The URL will expire in 10 minutes.
ov_url = lo_presigned_req->get_url( ).
```

## SageMaker Exemplos de IA usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com SageMaker IA.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Ações](#)
- [Cenários](#)

## Ações

### CreateEndpoint

O código de exemplo a seguir mostra como usar CreateEndpoint.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_production_variants TYPE /aws1/
cl_sgmproductionvariant=>tt_productionvariantlist.
DATA lo_production_variants TYPE REF TO /aws1/cl_sgmproductionvariant.
DATA oo_ep_config_result TYPE REF TO /aws1/cl_sgmcreateendptcfgout.

"Create a production variant as an ABAP object."
"Identifies a model that you want to host and the resources chosen to deploy for
hosting it."
lo_production_variants = NEW #( iv_variantname = iv_variant_name
                               iv_modelname = iv_model_name
                               iv_initialinstancecount =
iv_initial_instance_count
                               iv_instancetype = iv_instance_type ).

INSERT lo_production_variants INTO TABLE lt_production_variants.

"Create an endpoint configuration."
TRY.
```

```

        oo_ep_config_result = lo_sgm->createendpointconfig(
            iv_endpointconfigname = iv_endpoint_config_name
            it_productionvariants = lt_production_variants ).
        MESSAGE 'Endpoint configuration created.' TYPE 'I'.
    CATCH /aws1/cx_sgmresourcecelimitexcd.
        MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
    ENDTRY.

    "Create an endpoint."
    TRY.
        oo_result = lo_sgm->createendpoint(      " oo_result is returned for testing
purposes. "
            iv_endpointconfigname = iv_endpoint_config_name
            iv_endpointname = iv_endpoint_name ).
        MESSAGE 'Endpoint created.' TYPE 'I'.
    CATCH /aws1/cx_sgmresourcecelimitexcd.
        MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateEndpoint](#) referência da API AWS SDK for SAP ABAP.

## CreateModel

O código de exemplo a seguir mostra como usar CreateModel.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lo_primarycontainer TYPE REF TO /aws1/cl_sgmcontainerdefn.
```

```
"Create an ABAP object for the container image based on input variables."
lo_primarycontainer = NEW #( iv_image = iv_container_image
```

```

        iv_modeldataurl = iv_model_data_url ).

"Create an Amazon SageMaker model."
TRY.
    oo_result = lo_sgm->createmodel(      " oo_result is returned for testing
purposes. "
        iv_executionrolearn = iv_execution_role_arn
        iv_modelname = iv_model_name
        io_primarycontainer = lo_primarycontainer ).
    MESSAGE 'Model created.' TYPE 'I'.
CATCH /aws1/cx_sgmresourcecelimitexcd.
    MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateModel](#) referência da API AWS SDK for SAP ABAP.

## CreateTrainingJob

O código de exemplo a seguir mostra como usar CreateTrainingJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA lo_hyperparameters_w TYPE REF TO /aws1/cl_sgmhyperparameters_w.
DATA lt_hyperparameters TYPE /aws1/cl_sgmhyperparameters_w=>tt_hyperparameters.
DATA lt_input_data_config TYPE /aws1/cl_sgmchannel=>tt_inputdataconfig.
DATA lo_trn_channel TYPE REF TO /aws1/cl_sgmchannel.
DATA lo_trn_datasource TYPE REF TO /aws1/cl_sgmdatasource.
DATA lo_trn_s3datasource TYPE REF TO /aws1/cl_sgms3datasource.
DATA lo_val_channel TYPE REF TO /aws1/cl_sgmchannel.
DATA lo_val_datasource TYPE REF TO /aws1/cl_sgmdatasource.
DATA lo_val_s3datasource TYPE REF TO /aws1/cl_sgms3datasource.
DATA lo_algorithm_specification TYPE REF TO /aws1/cl_sgmalgorithmsspec.

```

```
DATA lo_resource_config TYPE REF TO /aws1/cl_sgmresourceconfig.  
DATA lo_output_data_config TYPE REF TO /aws1/cl_sgmoutputdataconfig.  
DATA lo_stopping_condition TYPE REF TO /aws1/cl_sgmstoppingcondition.
```

"Create ABAP internal table for hyperparameters based on input variables."

"These hyperparameters are based on the Amazon SageMaker built-in algorithm, XGBoost."

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_max_depth ).  
INSERT VALUE #( key = 'max_depth' value = lo_hyperparameters_w ) INTO TABLE  
lt_hyperparameters.
```

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_eta ).  
INSERT VALUE #( key = 'eta' value = lo_hyperparameters_w ) INTO TABLE  
lt_hyperparameters.
```

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_eval_metric ).  
INSERT VALUE #( key = 'eval_metric' value = lo_hyperparameters_w ) INTO TABLE  
lt_hyperparameters.
```

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_scale_pos_weight ).  
INSERT VALUE #( key = 'scale_pos_weight' value = lo_hyperparameters_w ) INTO  
TABLE lt_hyperparameters.
```

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_subsample ).  
INSERT VALUE #( key = 'subsample' value = lo_hyperparameters_w ) INTO TABLE  
lt_hyperparameters.
```

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_objective ).  
INSERT VALUE #( key = 'objective' value = lo_hyperparameters_w ) INTO TABLE  
lt_hyperparameters.
```

```
lo_hyperparameters_w = NEW #( iv_value = iv_hp_num_round ).  
INSERT VALUE #( key = 'num_round' value = lo_hyperparameters_w ) INTO TABLE  
lt_hyperparameters.
```

"Create ABAP objects for training data sources."

```
lo_trn_s3datasource = NEW #( iv_s3datatype = iv_trn_data_s3datatype  
                             iv_s3datadistributiontype =  
iv_trn_data_s3datadistribution  
                             iv_s3uri = iv_trn_data_s3uri ).
```

```
lo_trn_datasource = NEW #( io_s3datasource = lo_trn_s3datasource ).
```

```
lo_trn_channel = NEW #( iv_channelname = 'train'
```

```
        io_datasource = lo_trn_datasource
        iv_compressiontype = iv_trn_data_compressiontype
        iv_contenttype = iv_trn_data_contenttype ).

INSERT lo_trn_channel INTO TABLE lt_input_data_config.

"Create ABAP objects for validation data sources."
lo_val_s3datasource = NEW #( iv_s3datatype = iv_val_data_s3datatype
                             iv_s3datadistributiontype =
iv_val_data_s3datadistribution
                             iv_s3uri = iv_val_data_s3uri ).

lo_val_datasource = NEW #( io_s3datasource = lo_val_s3datasource ).

lo_val_channel = NEW #( iv_channelname = 'validation'
                        io_datasource = lo_val_datasource
                        iv_compressiontype = iv_val_data_compressiontype
                        iv_contenttype = iv_val_data_contenttype ).

INSERT lo_val_channel INTO TABLE lt_input_data_config.

"Create an ABAP object for algorithm specification."
lo_algorithm_specification = NEW #( iv_trainingimage = iv_training_image
                                     iv_traininginputmode =
iv_training_input_mode ).

"Create an ABAP object for resource configuration."
lo_resource_config = NEW #( iv_instancecount = iv_instance_count
                            iv_instancetype = iv_instance_type
                            iv_volumesizeingb = iv_volume_sizeingb ).

"Create an ABAP object for output data configuration."
lo_output_data_config = NEW #( iv_s3outputpath = iv_s3_output_path ).

"Create an ABAP object for stopping condition."
lo_stopping_condition = NEW #( iv_maxruntimeinseconds =
iv_max_runtime_in_seconds ).

"Create a training job."
TRY.
    oo_result = lo_sgm->createtrainingjob( " oo_result is returned for
testing purposes. "
        iv_trainingjobname          = iv_training_job_name
        iv_rolearn                  = iv_role_arn
```

```

        it_hyperparameters      = lt_hyperparameters
        it_inputdataconfig      = lt_input_data_config
        io_algorithmspecification = lo_algorithm_specification
        io_outputdataconfig      = lo_output_data_config
        io_resourceconfig        = lo_resource_config
        io_stoppingcondition     = lo_stopping_condition ).
    MESSAGE 'Training job created.' TYPE 'I'.
    CATCH /aws1/cx_sgmresourceinuse.
        MESSAGE 'Resource being accessed is in use.' TYPE 'E'.
    CATCH /aws1/cx_sgmresourceNotFound.
        MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
    CATCH /aws1/cx_sgmresourceLimitExcd.
        MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateTrainingJob](#) referência da API AWS SDK for SAP ABAP.

## CreateTransformJob

O código de exemplo a seguir mostra como usar CreateTransformJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA lo_transforminput TYPE REF TO /aws1/cl_sgmtransforminput.
DATA lo_transformoutput TYPE REF TO /aws1/cl_sgmtransformoutput.
DATA lo_transformresources TYPE REF TO /aws1/cl_sgmtransformresources.
DATA lo_datasource TYPE REF TO /aws1/cl_sgmtransformdatasrc.
DATA lo_s3datasource TYPE REF TO /aws1/cl_sgmtransforms3datasrc.

```

"Create an ABAP object for an Amazon Simple Storage Service (Amazon S3) data source."

```
lo_s3datasource = NEW #( iv_s3uri = iv_tf_data_s3uri
```

```

        iv_s3datatype = iv_tf_data_s3datatype ).

"Create an ABAP object for data source."
lo_datasource = NEW #( io_s3datasource = lo_s3datasource ).

"Create an ABAP object for transform data source."
lo_transforminput = NEW #( io_datasource = lo_datasource
                           iv_contenttype = iv_tf_data_contenttype
                           iv_compressiontype = iv_tf_data_compressiontype ).

"Create an ABAP object for resource configuration."
lo_transformresources = NEW #( iv_instancecount = iv_instance_count
                               iv_instancetype = iv_instance_type ).

"Create an ABAP object for output data configuration."
lo_transformoutput = NEW #( iv_s3outputpath = iv_s3_output_path ).

"Create a transform job."
TRY.
    oo_result = lo_sgm->createtransformjob(      " oo_result is returned for
testing purposes. "
        iv_modelname = iv_tf_model_name
        iv_transformjobname = iv_tf_job_name
        io_transforminput = lo_transforminput
        io_transformoutput = lo_transformoutput
        io_transformresources = lo_transformresources ).
    MESSAGE 'Transform job created.' TYPE 'I'.
CATCH /aws1/cx_sgmresourceinuse.
    MESSAGE 'Resource being accessed is in use.' TYPE 'E'.
CATCH /aws1/cx_sgmresourceisnotfound.
    MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
CATCH /aws1/cx_sgmresourceusageexceeded.
    MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
ENDTRY.


```

- Para obter detalhes da API, consulte a [CreateTransformJob](#) referência da API AWS SDK for SAP ABAP.

## DeleteEndpoint

O código de exemplo a seguir mostra como usar DeleteEndpoint.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Delete an endpoint."
TRY.
  lo_sgm->deleteendpoint(
    iv_endpointname = iv_endpoint_name ).
  MESSAGE 'Endpoint configuration deleted.' TYPE 'I'.
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_endpoint_exception).
  DATA(lv_endpoint_error) = |"{ lo_endpoint_exception->av_err_code }" -
{ lo_endpoint_exception->av_err_msg }|.
  MESSAGE lv_endpoint_error TYPE 'E'.
ENDTRY.


"Delete an endpoint configuration."
TRY.
  lo_sgm->deleteendpointconfig(
    iv_endpointconfigname = iv_endpoint_config_name ).
  MESSAGE 'Endpoint deleted.' TYPE 'I'.
  CATCH /aws1/cx_rt_service_generic INTO DATA(lo_endpointconfig_exception).
  DATA(lv_endpointconfig_error) = |"{ lo_endpointconfig_exception-
>av_err_code }" - { lo_endpointconfig_exception->av_err_msg }|.
  MESSAGE lv_endpointconfig_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteEndpoint](#) referência da API AWS SDK for SAP ABAP.

## DeleteModel

O código de exemplo a seguir mostra como usar DeleteModel.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    lo_sgm->deletemodel(  
        iv_modelname = iv_model_name ).  
    MESSAGE 'Model deleted.' TYPE 'I'.  
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).  
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception->  
>av_err_msg }|.  
    MESSAGE lv_error TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteModel](#) referência da API AWS SDK for SAP ABAP.

## DescribeTrainingJob

O código de exemplo a seguir mostra como usar DescribeTrainingJob.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_sgm->describetrainingjob( " oo_result is returned for  
testing purposes. "  
        iv_trainingjobname = iv_training_job_name ).  
    MESSAGE 'Retrieved description of training job.' TYPE 'I'.
```

```

    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
      DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
      MESSAGE lv_error TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeTrainingJob](#) referência da API AWS SDK for SAP ABAP.

## ListAlgorithms

O código de exemplo a seguir mostra como usar ListAlgorithms.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

    TRY.
      oo_result = lo_sgm->listalgorithms(          " oo_result is returned for
testing purposes. "
      iv_namecontains = iv_name_contains ).
      MESSAGE 'Retrieved list of algorithms.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
      DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
      MESSAGE lv_error TYPE 'E'.
    ENDTRY.


```

- Para obter detalhes da API, consulte a [ListAlgorithms](#) referência da API AWS SDK for SAP ABAP.

## ListModels

O código de exemplo a seguir mostra como usar ListModels.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
    oo_result = lo_sgm->listmodels(           " oo_result is returned for
testing purposes. "
        iv_namecontains = iv_name_contains ).
    MESSAGE 'Retrieved list of models.' TYPE 'I'.
    CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
    DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
    MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListModels](#) referência da API AWS SDK for SAP ABAP.

## ListNotebookInstances

O código de exemplo a seguir mostra como usar ListNotebookInstances.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_sgm->listnotebookinstances(           " oo_result is returned
for testing purposes. "
        iv_namecontains = iv_name_contains ).
    MESSAGE 'Retrieved list of notebook instances.' TYPE 'I'.
```

```
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListNotebookInstances](#) referência da API AWS SDK for SAP ABAP.

## ListTrainingJobs

O código de exemplo a seguir mostra como usar ListTrainingJobs.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  oo_result = lo_sgm->listtrainingjobs(      " oo_result is returned for
testing purposes. "
  iv_namecontains = iv_name_contains
  iv_maxresults = iv_max_results ).
  MESSAGE 'Retrieved list of training jobs.' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_exception).
  DATA(lv_error) = |"{ lo_exception->av_err_code }" - { lo_exception-
>av_err_msg }|.
  MESSAGE lv_error TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListTrainingJobs](#) referência da API AWS SDK for SAP ABAP.

# Cenários

## Conceitos básicos de usar modelos e endpoints

O exemplo de código a seguir mostra como:

- Comece um trabalho de treinamento e crie um modelo de SageMaker IA.
- Criar uma configuração de endpoint.
- Criar um endpoint e, em seguida, limpar os recursos.

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lo_hyperparameters_w TYPE REF TO /aws1/cl_sgmhyperparameters_w.
DATA lo_trn_channel TYPE REF TO /aws1/cl_sgmchannel.
DATA lo_trn_datasource TYPE REF TO /aws1/cl_sgmdatasource.
DATA lo_trn_s3datasource TYPE REF TO /aws1/cl_sgms3datasource.
DATA lo_val_channel TYPE REF TO /aws1/cl_sgmchannel.
DATA lo_val_datasource TYPE REF TO /aws1/cl_sgmdatasource.
DATA lo_val_s3datasource TYPE REF TO /aws1/cl_sgms3datasource.
DATA lo_algorithm_specification TYPE REF TO /aws1/cl_sgmalgorithm_spec.
DATA lo_resource_config TYPE REF TO /aws1/cl_sgmresourceconfig.
DATA lo_output_data_config TYPE REF TO /aws1/cl_sgmoutputdataconfig.
DATA lo_stopping_condition TYPE REF TO /aws1/cl_sgmstoppingcondition.
DATA lo_primarycontainer TYPE REF TO /aws1/cl_sgmcontainerdefn.
DATA lo_production_variants TYPE REF TO /aws1/cl_sgmproductionvariant.
DATA lo_ep_config_result TYPE REF TO /aws1/cl_sgmcreateendptcfgout.
DATA lo_training_result TYPE REF TO /aws1/cl_sgmdescrtrnjobrsp.
DATA lt_production_variants TYPE /aws1/
cl_sgmproductionvariant=>tt_productionvariantlist.
DATA lt_input_data_config TYPE /aws1/cl_sgmchannel=>tt_inputdataconfig.
DATA lt_hyperparameters TYPE /aws1/cl_sgmhyperparameters_w=>tt_hyperparameters.
DATA lv_model_data_url TYPE /aws1/sgmurl.
```

```
lv_model_data_url = iv_s3_output_path && iv_training_job_name && '/output/
model.tar.gz'.

"Create ABAP internal table for hyperparameters based on input variables."
"These hyperparameters are based on Amazon SageMaker built-in algorithm -
XGBoost"
lo_hyperparameters_w = NEW #( iv_value = iv_hp_max_depth ).
INSERT VALUE #( key = 'max_depth' value = lo_hyperparameters_w ) INTO TABLE
lt_hyperparameters.

lo_hyperparameters_w = NEW #( iv_value = iv_hp_eta ).
INSERT VALUE #( key = 'eta' value = lo_hyperparameters_w ) INTO TABLE
lt_hyperparameters.

lo_hyperparameters_w = NEW #( iv_value = iv_hp_eval_metric ).
INSERT VALUE #( key = 'eval_metric' value = lo_hyperparameters_w ) INTO TABLE
lt_hyperparameters.

lo_hyperparameters_w = NEW #( iv_value = iv_hp_scale_pos_weight ).
INSERT VALUE #( key = 'scale_pos_weight' value = lo_hyperparameters_w ) INTO
TABLE lt_hyperparameters.

lo_hyperparameters_w = NEW #( iv_value = iv_hp_subsample ).
INSERT VALUE #( key = 'subsample' value = lo_hyperparameters_w ) INTO TABLE
lt_hyperparameters.

lo_hyperparameters_w = NEW #( iv_value = iv_hp_objective ).
INSERT VALUE #( key = 'objective' value = lo_hyperparameters_w ) INTO TABLE
lt_hyperparameters.

lo_hyperparameters_w = NEW #( iv_value = iv_hp_num_round ).
INSERT VALUE #( key = 'num_round' value = lo_hyperparameters_w ) INTO TABLE
lt_hyperparameters.

"Create ABAP internal table for data based on input variables."
"Training data."
lo_trn_s3datasource = NEW #( iv_s3datatype = iv_trn_data_s3datatype
                             iv_s3datadistributiontype =
iv_trn_data_s3datadistribution
                             iv_s3uri = iv_trn_data_s3uri ).

lo_trn_datasource = NEW #( io_s3datasource = lo_trn_s3datasource ).

lo_trn_channel = NEW #( iv_channelname = 'train'
```

```
        io_datasource = lo_trn_datasource
        iv_compressiontype = iv_trn_data_compressiontype
        iv_contenttype = iv_trn_data_contenttype ).
INSERT lo_trn_channel INTO TABLE lt_input_data_config.

"Validation data."
lo_val_s3datasource = NEW #( iv_s3datatype = iv_val_data_s3datatype
                             iv_s3datadistributiontype =
iv_val_data_s3datadistribution
                             iv_s3uri = iv_val_data_s3uri ).

lo_val_datasource = NEW #( io_s3datasource = lo_val_s3datasource ).

lo_val_channel = NEW #( iv_channelname = 'validation'
                       io_datasource = lo_val_datasource
                       iv_compressiontype = iv_val_data_compressiontype
                       iv_contenttype = iv_val_data_contenttype ).
INSERT lo_val_channel INTO TABLE lt_input_data_config.

"Create an ABAP object for algorithm specification based on input variables."
lo_algorithm_specification = NEW #( iv_trainingimage = iv_training_image
                                     iv_traininginputmode =
iv_training_input_mode ).

"Create an ABAP object for resource configuration."
lo_resource_config = NEW #( iv_instancecount = iv_instance_count
                            iv_instancetype = iv_instance_type
                            iv_volumesizeingb = iv_volume_sizeingb ).

"Create an ABAP object for output data configuration."
lo_output_data_config = NEW #( iv_s3outputpath = iv_s3_output_path ).

"Create an ABAP object for stopping condition."
lo_stopping_condition = NEW #( iv_maxruntimeinseconds =
iv_max_runtime_in_seconds ).

TRY.
    lo_sgm->createtrainingjob(
        iv_trainingjobname      = iv_training_job_name
        iv_rolearn              = iv_role_arn
        it_hyperparameters      = lt_hyperparameters
        it_inputdataconfig      = lt_input_data_config
        io_algorithmspecification = lo_algorithm_specification
        io_outputdataconfig     = lo_output_data_config
```

```
        io_resourceconfig          = lo_resource_config
        io_stoppingcondition       = lo_stopping_condition ).
    MESSAGE 'Training job created.' TYPE 'I'.
    CATCH /aws1/cx_sgmresourceinuse.
        MESSAGE 'Resource being accessed is in use.' TYPE 'E'.
    CATCH /aws1/cx_sgmresourceNotFound.
        MESSAGE 'Resource being accessed is not found.' TYPE 'E'.
    CATCH /aws1/cx_sgmresourceLimitExcd.
        MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
    ENDMETHOD.

    "Wait for training job to be completed."
    lo_training_result = lo_sgm->describetrainingjob( iv_trainingjobname =
iv_training_job_name ).
    WHILE lo_training_result->get_trainingjobstatus( ) <> 'Completed'.
        IF sy-index = 30.
            EXIT.                "Maximum 900 seconds."
        ENDIF.
        WAIT UP TO 30 SECONDS.
        lo_training_result = lo_sgm->describetrainingjob( iv_trainingjobname =
iv_training_job_name ).
    ENDWHILE.

    "Create ABAP object for the container image based on input variables."
    lo_primarycontainer = NEW #( iv_image = iv_training_image
                                iv_modeldataurl = lv_model_data_url ).

    "Create an Amazon SageMaker model."
    TRY.
        lo_sgm->createmodel(
            iv_executionrolearn = iv_role_arn
            iv_modelname = iv_model_name
            io_primarycontainer = lo_primarycontainer ).
        MESSAGE 'Model created.' TYPE 'I'.
    CATCH /aws1/cx_sgmresourceLimitExcd.
        MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
    ENDMETHOD.

    "Create an endpoint production variant."
    lo_production_variants = NEW #( iv_variantname = iv_ep_variant_name
                                    iv_modelname = iv_model_name
                                    iv_initialinstancecount =
iv_ep_initial_instance_count
                                    iv_instancetype = iv_ep_instance_type ).
```

```
INSERT lo_production_variants INTO TABLE lt_production_variants.

TRY.
    "Create an endpoint configuration."
    lo_ep_config_result = lo_sgm->createendpointconfig(
        iv_endpointconfigname = iv_ep_cfg_name
        it_productionvariants = lt_production_variants ).
    MESSAGE 'Endpoint configuration created.' TYPE 'I'.

    "Create an endpoint."
    oo_ep_output = lo_sgm->createendpoint(          " oo_ep_output is returned for
testing purposes. "
        iv_endpointconfigname = iv_ep_cfg_name
        iv_endpointname = iv_ep_name ).
    MESSAGE 'Endpoint created.' TYPE 'I'.
    CATCH /aws1/cx_sgmresourcecelimitexcd.
        MESSAGE 'You have reached the limit on the number of resources.' TYPE 'E'.
    ENDTRY.

    "Wait for endpoint creation to be completed."
    DATA(lo_endpoint_result) = lo_sgm->describeendpoint( iv_endpointname =
iv_ep_name ).
    WHILE lo_endpoint_result->get_endpointstatus( ) <> 'InService'.
        IF sy-index = 30.
            EXIT.          "Maximum 900 seconds."
        ENDIF.
        WAIT UP TO 30 SECONDS.
        lo_endpoint_result = lo_sgm->describeendpoint( iv_endpointname = iv_ep_name ).
    ENDWHILE.

TRY.
    "Delete an endpoint."
    lo_sgm->deleteendpoint(
        iv_endpointname = iv_ep_name ).
    MESSAGE 'Endpoint deleted' TYPE 'I'.

    "Delete an endpoint configuration."
    lo_sgm->deleteendpointconfig(
        iv_endpointconfigname = iv_ep_cfg_name ).
    MESSAGE 'Endpoint configuration deleted.' TYPE 'I'.

    "Delete model."
    lo_sgm->deletemodel(
        iv_modelname = iv_model_name ).
```

```
MESSAGE 'Model deleted.' TYPE 'I'.
CATCH /aws1/cx_rt_service_generic INTO DATA(lo_endpointconfig_exception).
DATA(lv_endpointconfig_error) = |"{ lo_endpointconfig_exception-
>av_err_code }" - { lo_endpointconfig_exception->av_err_msg }|.
MESSAGE lv_endpointconfig_error TYPE 'E'.
ENDTRY.
```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [CreateEndpoint](#)
  - [CreateEndpointConfig](#)
  - [CreateModel](#)
  - [CreateTrainingJob](#)
  - [DeleteEndpoint](#)
  - [DeleteEndpointConfig](#)
  - [DeleteModel](#)
  - [DescribeEndpoint](#)
  - [DescribeTrainingJob](#)

## Exemplos de Secrets Manager usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Secrets Manager.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)

# Ações

## BatchGetSecretValue

O código de exemplo a seguir mostra como usar BatchGetSecretValue.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " iv_filter_name = 'mySecret'
  DATA(lo_result) = lo_smr->batchgetsecretvalue(
    it_filters = VALUE /aws1/cl_smrfilter=>tt_filterslisttype(
      (
        NEW /aws1/cl_smrfilter(
          iv_key = 'name'
          it_values = VALUE /aws1/
cl_smrfiltvalsstrlist_w=>tt_filtervaluesstringlist(
          ( NEW /aws1/cl_smrfiltvalsstrlist_w( iv_value = iv_filter_name ) )
        )
      )
    )
  ).
  ot_secret_values = lo_result->get_secretvalues( ).
  MESSAGE 'Secrets retrieved successfully.' TYPE 'I'.
CATCH /aws1/cx_smrresourcenotfoundex.
  MESSAGE 'One or more requested secrets were not found.' TYPE 'E'.
CATCH /aws1/cx_smrdecryptionfailure.
  MESSAGE 'Failed to decrypt one or more secrets.' TYPE 'E'.
CATCH /aws1/cx_smrinvalidparameterex.
  MESSAGE 'Invalid parameter provided.' TYPE 'E'.
CATCH /aws1/cx_smrinvalidrequestex.
  MESSAGE 'Invalid request.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [BatchGetSecretValue](#) referência da API AWS SDK for SAP ABAP.

## GetSecretValue

O código de exemplo a seguir mostra como usar GetSecretValue.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  " iv_secret_name = 'MySecretName'  
  DATA(lo_result) = lo_smr->getsecretvalue( iv_secretid = iv_secret_name ).  
  ov_secret_value = lo_result->get_secretstring( ).  
  MESSAGE 'Secret value retrieved successfully.' TYPE 'I'.  
CATCH /aws1/cx_smrresourcefoundex.  
  MESSAGE 'The requested secret was not found.' TYPE 'E'.  
CATCH /aws1/cx_smrdecryptionfailure.  
  MESSAGE 'Failed to decrypt the secret.' TYPE 'E'.  
CATCH /aws1/cx_smrinvalidparameterex.  
  MESSAGE 'Invalid parameter provided.' TYPE 'E'.  
CATCH /aws1/cx_smrinvalidrequestex.  
  MESSAGE 'Invalid request.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetSecretValue](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon SES usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon SES.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateReceiptFilter

O código de exemplo a seguir mostra como usar CreateReceiptFilter.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
" iv_allow = abap_true means 'Allow', abap_false means 'Block'
DATA(lv_policy) = COND /aws1/sesreceiptfilterpolicy(
  WHEN iv_allow = abap_true THEN 'Allow'
  ELSE 'Block'
).

DATA(lo_ip_filter) = NEW /aws1/cl_sesreceiptipfilter(
  iv_policy = lv_policy
  iv_cidr = iv_ip_address_or_range
).

DATA(lo_filter) = NEW /aws1/cl_sesreceiptfilter(
  iv_name = iv_filter_name
  io_ipfilter = lo_ip_filter
).

TRY.
```

```

    lo_ses->createreceiptfilter( io_filter = lo_filter ).
    MESSAGE 'Receipt filter created successfully' TYPE 'I'.
    CATCH /aws1/cx_sesalreadyexistsex INTO DATA(lo_ex1).
    DATA(lv_error) = |Filter already exists: { lo_ex1->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex1.
    CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
    lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateReceiptFilter](#) referência da API AWS SDK for SAP ABAP.

## CreateReceiptRule

O código de exemplo a seguir mostra como usar CreateReceiptRule.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" Create S3 action for copying emails to S3
DATA(lo_s3_action) = NEW /aws1/cl_sess3action(
    iv_bucketname = iv_bucket_name
    iv_objectkeyprefix = iv_prefix
).

" Create receipt action with S3 action
DATA(lo_action) = NEW /aws1/cl_sesreceiptaction(
    io_s3action = lo_s3_action
).

" Create list of actions
DATA lt_actions TYPE /aws1/cl_sesreceiptaction=>tt_receiptactionslist.

```

```
APPEND lo_action TO lt_actions.

" Create receipt rule
DATA(lo_rule) = NEW /aws1/cl_sesrecepitrule(
  iv_name = iv_rule_name
  iv_enabled = abap_true
  it_recipients = it_recipients
  it_actions = lt_actions
).

TRY.
  lo_ses->createrecepitrule(
    iv_rulesetname = iv_rule_set_name
    io_rule = lo_rule
  ).
  MESSAGE 'Receipt rule created successfully' TYPE 'I'.
CATCH /aws1/cx_sesinvalids3confex INTO DATA(lo_ex1).
  DATA(lv_error) = |Invalid S3 configuration: { lo_ex1->get_text( ) }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_ex1.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
  lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_ex_generic.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateReceiptRule](#) referência da API AWS SDK for SAP ABAP.

## CreateReceiptRuleSet

O código de exemplo a seguir mostra como usar CreateReceiptRuleSet.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  lo_ses->createreciptruleset( iv_rulesetname = iv_rule_set_name ).
  MESSAGE 'Receipt rule set created successfully' TYPE 'I'.
CATCH /aws1/cx_sesalreadyexistsex INTO DATA(lo_ex1).
  DATA(lv_error) = |Rule set already exists: { lo_ex1->get_text( ) }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_ex1.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
  lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateReceiptRuleSet](#) referência da API AWS SDK for SAP ABAP.

## CreateTemplate

O código de exemplo a seguir mostra como usar CreateTemplate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA(lo_template) = NEW /aws1/cl_sestemplate(
  iv_templatename = iv_name
  iv_subjectpart = iv_subject
  iv_textpart = iv_text
  iv_htmlpart = iv_html
).

TRY.
  lo_ses->createtemplate( io_template = lo_template ).
  MESSAGE 'Template created successfully' TYPE 'I'.
CATCH /aws1/cx_sesalreadyexistsex INTO DATA(lo_ex1).
  DATA(lv_error) = |Template already exists: { lo_ex1->get_text( ) }|.

```

```
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex1.
CATCH /aws1/cx_sesinvalidtemplateex INTO DATA(lo_ex2).
lv_error = |Invalid template: { lo_ex2->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex2.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex_generic.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateTemplate](#) referência da API AWS SDK for SAP ABAP.

## DeleteIdentity

O código de exemplo a seguir mostra como usar DeleteIdentity.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
lv_ses->deleteidentity( iv_identity = iv_identity ).
MESSAGE 'Identity deleted successfully' TYPE 'I'.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteIdentity](#) referência da API AWS SDK for SAP ABAP.

## DeleteReceiptFilter

O código de exemplo a seguir mostra como usar DeleteReceiptFilter.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ses->deletereciptfilter( iv_filtername = iv_filter_name ).  
    MESSAGE 'Receipt filter deleted successfully' TYPE 'I'.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).  
    DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteReceiptFilter](#) referência da API AWS SDK for SAP ABAP.

## DeleteReceiptRule

O código de exemplo a seguir mostra como usar DeleteReceiptRule.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ses->deletereciptrule(  
        iv_rulesetname = iv_rule_set_name
```

```

        iv_rulename = iv_rule_name
    ).
    MESSAGE 'Receipt rule deleted successfully' TYPE 'I'.
    CATCH /aws1/cx_sesrulesetdoesnotexist INTO DATA(lo_ex1).
        DATA(lv_error) = |Rule set does not exist: { lo_ex1->get_text( ) }|.
        MESSAGE lv_error TYPE 'I'.
        RAISE EXCEPTION lo_ex1.
    CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
        lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
        MESSAGE lv_error TYPE 'I'.
        RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteReceiptRule](#) referência da API AWS SDK for SAP ABAP.

## DeleteReceiptRuleSet

O código de exemplo a seguir mostra como usar DeleteReceiptRuleSet.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    lo_ses->deleterecepitruleset( iv_rulesetname = iv_rule_set_name ).
    MESSAGE 'Receipt rule set deleted successfully' TYPE 'I'.
    CATCH /aws1/cx_sescannotdelete INTO DATA(lo_ex1).
        DATA(lv_error) = |Cannot delete rule set: { lo_ex1->get_text( ) }|.
        MESSAGE lv_error TYPE 'I'.
        RAISE EXCEPTION lo_ex1.
    CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
        lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
        MESSAGE lv_error TYPE 'I'.
        RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteReceiptRuleSet](#) referência da API AWS SDK for SAP ABAP.

## DeleteTemplate

O código de exemplo a seguir mostra como usar DeleteTemplate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ses->deletetemplate( iv_templatename = iv_template_name ).  
    MESSAGE 'Template deleted successfully' TYPE 'I'.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).  
    DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteTemplate](#) referência da API AWS SDK for SAP ABAP.

## DescribeReceiptRuleSet

O código de exemplo a seguir mostra como usar DescribeReceiptRuleSet.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ses->describerecepitruleset(  
        iv_rulesetname = iv_rule_set_name  
    ).  
    MESSAGE 'Receipt rule set described successfully' TYPE 'I'.  
CATCH /aws1/cx_sesrulesetdoesnotexist INTO DATA(lo_ex1).  
    DATA(lv_error) = |Rule set does not exist: { lo_ex1->get_text( ) }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_ex1.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).  
    lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_ex_generic.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeReceiptRuleSet](#) referência da API AWS SDK for SAP ABAP.

**GetIdentityVerificationAttributes**

O código de exemplo a seguir mostra como usar `GetIdentityVerificationAttributes`.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lt_identities TYPE /aws1/cl_sesidentitylist_w=>tt_identitylist.
```

```
APPEND NEW /aws1/cl_sesidentitylist_w( iv_value = iv_identity ) TO
lt_identities.

TRY.
  DATA(lo_result) = lo_ses->getidentityverificationattrs(
    it_identities = lt_identities
  ).

  DATA(lt_attrs) = lo_result->get_verificationattributes( ).
  IF lt_attrs IS NOT INITIAL.
    LOOP AT lt_attrs ASSIGNING FIELD-SYMBOL(<ls_attr>).
      ov_status = <ls_attr>-value->get_verificationstatus( ).
      EXIT.
    ENDLLOOP.
  ELSE.
    ov_status = 'NotFound'.
  ENDIF.
  CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
    DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetIdentityVerificationAttributes](#) referência da API AWS SDK for SAP ABAP.

## GetTemplate

O código de exemplo a seguir mostra como usar GetTemplate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  DATA(lo_result) = lo_ses->gettemplate( iv_templatename = iv_template_name ).
```

```
oo_template = lo_result->get_template( ).
MESSAGE 'Template retrieved successfully' TYPE 'I'.
CATCH /aws1/cx_sestmpldoesnotexistex INTO DATA(lo_ex1).
DATA(lv_error) = |Template does not exist: { lo_ex1->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex1.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex_generic.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetTemplate](#) referência da API AWS SDK for SAP ABAP.

## ListIdentities

O código de exemplo a seguir mostra como usar ListIdentities.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
DATA(lo_result) = lo_ses->listidentities(
    iv_identitytype = iv_identity_type
    iv_maxitems = iv_max_items
).
ot_identities = lo_result->get_identities( ).
MESSAGE 'Identities retrieved successfully' TYPE 'I'.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListIdentities](#) referência da API AWS SDK for SAP ABAP.

## ListReceiptFilters

O código de exemplo a seguir mostra como usar ListReceiptFilters.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  DATA(lo_result) = lo_ses->listreceiptfilters( ).  
  ot_filters = lo_result->get_filters( ).  
  MESSAGE 'Receipt filters retrieved successfully' TYPE 'I'.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).  
  DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.  
  MESSAGE lv_error TYPE 'I'.  
  RAISE EXCEPTION lo_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListReceiptFilters](#) referência da API AWS SDK for SAP ABAP.

## ListTemplates

O código de exemplo a seguir mostra como usar ListTemplates.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  DATA(lo_result) = lo_ses->listtemplates( iv_maxitems = iv_max_items ).
  ot_templates = lo_result->get_templatesmetadata( ).
  MESSAGE 'Templates retrieved successfully' TYPE 'I'.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
  DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.
  MESSAGE lv_error TYPE 'I'.
  RAISE EXCEPTION lo_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListTemplates](#) referência da API AWS SDK for SAP ABAP.

## SendEmail

O código de exemplo a seguir mostra como usar SendEmail.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" Create message object
DATA(lo_subject) = NEW /aws1/cl_sescontent( iv_data = iv_subject ).
DATA(lo_text_body) = NEW /aws1/cl_sescontent( iv_data = iv_text ).
DATA(lo_html_body) = NEW /aws1/cl_sescontent( iv_data = iv_html ).
DATA(lo_body) = NEW /aws1/cl_sesbody(
  io_text = lo_text_body
  io_html = lo_html_body
).
DATA(lo_message) = NEW /aws1/cl_sesmessage(
  io_subject = lo_subject
  io_body = lo_body
).

TRY.
  " Send email

```

```

DATA(lo_result) = lo_ses->sendemail(
  iv_source = iv_source
  io_destination = io_destination
  io_message = lo_message
  it_replytoaddresses = it_reply_tos
).
ov_msg_id = lo_result->get_messageid( ).
MESSAGE 'Email sent successfully' TYPE 'I'.
CATCH /aws1/cx_sesacctsendingpause00 INTO DATA(lo_ex1).
DATA(lv_error) = |Account sending paused: { lo_ex1->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex1.
CATCH /aws1/cx_sesmessagerejected INTO DATA(lo_ex2).
lv_error = |Message rejected: { lo_ex2->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex2.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
MESSAGE lv_error TYPE 'I'.
RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [SendEmail](#) referência da API AWS SDK for SAP ABAP.

## SendTemplatedEmail

O código de exemplo a seguir mostra como usar SendTemplatedEmail.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Send templated email
  DATA(lo_result) = lo_ses->sendtemplatedemail(
    iv_source = iv_source
    io_destination = io_destination

```

```

        iv_template = iv_template_name
        iv_templatedata = iv_template_data
        it_replytoaddresses = it_reply_tos
    ).
    ov_msg_id = lo_result->get_messageid( ).
    MESSAGE 'Templated email sent successfully' TYPE 'I'.
CATCH /aws1/cx_sestmpldoesnotexistex INTO DATA(lo_ex1).
    DATA(lv_error) = |Template does not exist: { lo_ex1->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex1.
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
    lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [SendTemplatedEmail](#) referência da API AWS SDK for SAP ABAP.

## UpdateTemplate

O código de exemplo a seguir mostra como usar UpdateTemplate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

DATA(lo_template) = NEW /aws1/cl_sestemplate(
    iv_templatename = iv_name
    iv_subjectpart = iv_subject
    iv_textpart = iv_text
    iv_htmlpart = iv_html
).

TRY.
    lo_ses->updatetemplate( io_template = lo_template ).

```

```

    MESSAGE 'Template updated successfully' TYPE 'I'.
    CATCH /aws1/cx_sestmpldoesnotexistex INTO DATA(lo_ex1).
    DATA(lv_error) = |Template does not exist: { lo_ex1->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex1.
    CATCH /aws1/cx_sesinvalidtemplateex INTO DATA(lo_ex2).
    lv_error = |Invalid template: { lo_ex2->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex2.
    CATCH /aws1/cx_rt_generic INTO DATA(lo_ex_generic).
    lv_error = |An error occurred: { lo_ex_generic->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex_generic.
ENDTRY.

```

- Para obter detalhes da API, consulte a [UpdateTemplate](#) referência da API AWS SDK for SAP ABAP.

## VerifyDomainIdentity

O código de exemplo a seguir mostra como usar VerifyDomainIdentity.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    DATA(lo_result) = lo_ses->verifydomainidentity( iv_domain =
iv_domain_name ).
    ov_token = lo_result->get_verificationtoken( ).
    MESSAGE 'Domain verification initiated' TYPE 'I'.
    CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).
    DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.
    MESSAGE lv_error TYPE 'I'.
    RAISE EXCEPTION lo_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [VerifyDomainIdentity](#) referência da API AWS SDK for SAP ABAP.

## VerifyEmailIdentity

O código de exemplo a seguir mostra como usar `VerifyEmailIdentity`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ses->verifyemailidentity( iv_emailaddress = iv_email_address ).  
    MESSAGE 'Email verification initiated' TYPE 'I'.  
CATCH /aws1/cx_rt_generic INTO DATA(lo_ex).  
    DATA(lv_error) = |An error occurred: { lo_ex->get_text( ) }|.  
    MESSAGE lv_error TYPE 'I'.  
    RAISE EXCEPTION lo_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [VerifyEmailIdentity](#) referência da API AWS SDK for SAP ABAP.

## Exemplos da API v2 do Amazon SES usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com a API v2 do Amazon SES.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateContact

O código de exemplo a seguir mostra como usar CreateContact.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_se2->createcontact(  
    iv_contactlistname = iv_contact_list_name  
    iv_emailaddress = iv_email_address ).  
  MESSAGE 'Contact created successfully.' TYPE 'I'.  
CATCH /aws1/cx_se2alreadyexistsex.  
  MESSAGE 'Contact already exists.' TYPE 'I'.  
CATCH /aws1/cx_se2badrequestex.  
  MESSAGE 'Bad request.' TYPE 'E'.  
CATCH /aws1/cx_se2notfoundexception.  
  MESSAGE 'Contact list not found.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateContact](#) referência da API AWS SDK for SAP ABAP.

### CreateContactList

O código de exemplo a seguir mostra como usar CreateContactList.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_se2->createcontactlist(  
    iv_contactlistname = iv_contact_list_name ).  
  MESSAGE 'Contact list created successfully.' TYPE 'I'.  
CATCH /aws1/cx_se2alreadyexistsex.  
  MESSAGE 'Contact list already exists.' TYPE 'I'.  
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).  
  MESSAGE 'Bad request - contact list limit may be reached.' TYPE 'I'.  
  " Re-raise the exception so the caller can handle it  
  RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_se2limitexceededex INTO DATA(lo_limit_exceeded).  
  MESSAGE 'Limit exceeded - contact list limit reached.' TYPE 'I'.  
  " Re-raise the exception so the caller can handle it  
  RAISE EXCEPTION lo_limit_exceeded.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateContactList](#) referência da API AWS SDK for SAP ABAP.

## CreateEmailIdentity

O código de exemplo a seguir mostra como usar CreateEmailIdentity.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  lo_se2->createemailidentity(
    iv_emailidentity = iv_email_identity ).
  MESSAGE 'Email identity created successfully.' TYPE 'I'.
CATCH /aws1/cx_se2alreadyexistsex.
  MESSAGE 'Email identity already exists.' TYPE 'I'.
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).
  MESSAGE lo_bad_request TYPE 'I' DISPLAY LIKE 'E'.
CATCH /aws1/cx_se2limitexceededex INTO DATA(lo_limit_exceeded).
  MESSAGE lo_limit_exceeded TYPE 'I' DISPLAY LIKE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateEmailIdentity](#) referência da API AWS SDK for SAP ABAP.

## CreateEmailTemplate

O código de exemplo a seguir mostra como usar CreateEmailTemplate.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  DATA(lo_template_content) = NEW /aws1/cl_se2emailtplcontent(
    iv_subject = iv_subject
    iv_html = iv_html
    iv_text = iv_text ).

  lo_se2->createemailtemplate(
    iv_templatename = iv_template_name
    io_templatecontent = lo_template_content ).
  MESSAGE 'Email template created successfully.' TYPE 'I'.
CATCH /aws1/cx_se2alreadyexistsex.
  MESSAGE 'Email template already exists.' TYPE 'I'.
CATCH /aws1/cx_se2badrequestex.
```

```
MESSAGE 'Bad request.' TYPE 'E'.
CATCH /aws1/cx_se2limitexceededException.
MESSAGE 'Limit exceeded.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateEmailTemplate](#) referência da API AWS SDK for SAP ABAP.

## DeleteContactList

O código de exemplo a seguir mostra como usar DeleteContactList.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  lo_se2->deletecontactlist(
    iv_contactlistname = iv_contact_list_name ).
  MESSAGE 'Contact list deleted successfully.' TYPE 'I'.
CATCH /aws1/cx_se2notfoundexception.
  MESSAGE 'Contact list not found.' TYPE 'I'.
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).
  MESSAGE 'Bad request.' TYPE 'I'.
  RAISE EXCEPTION lo_bad_request.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteContactList](#) referência da API AWS SDK for SAP ABAP.

## DeleteEmailIdentity

O código de exemplo a seguir mostra como usar DeleteEmailIdentity.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_se2->deleteemailidentity(  
    iv_emailidentity = iv_email_identity ).  
  MESSAGE 'Email identity deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_se2notfoundexception.  
  MESSAGE 'Email identity not found.' TYPE 'I'.  
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).  
  MESSAGE 'Bad request.' TYPE 'I'.  
  RAISE EXCEPTION lo_bad_request.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteEmailIdentity](#) referência da API AWS SDK for SAP ABAP.

**DeleteEmailTemplate**

O código de exemplo a seguir mostra como usar DeleteEmailTemplate.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_se2->deleteemailtemplate(  
    iv_templatename = iv_template_name ).  
  MESSAGE 'Email template deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_se2notfoundexception.
```

```
MESSAGE 'Email template not found.' TYPE 'I'.
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).
MESSAGE 'Bad request.' TYPE 'I'.
RAISE EXCEPTION lo_bad_request.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteEmailTemplate](#) referência da API AWS SDK for SAP ABAP.

## ListContacts

O código de exemplo a seguir mostra como usar ListContacts.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  oo_result = lo_se2->listcontacts(
    iv_contactlistname = iv_contact_list_name ).
  DATA(lv_count) = lines( oo_result->get_contacts( ) ).
  MESSAGE |Retrieved { lv_count } contacts from list.| TYPE 'I'.
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).
  MESSAGE 'Bad request.' TYPE 'I'.
  RAISE EXCEPTION lo_bad_request.
CATCH /aws1/cx_se2notfoundexception INTO DATA(lo_not_found).
  MESSAGE 'Contact list not found.' TYPE 'I'.
  RAISE EXCEPTION lo_not_found.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListContacts](#) referência da API AWS SDK for SAP ABAP.

## SendEmail

O código de exemplo a seguir mostra como usar SendEmail.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Envia uma mensagem.

```
TRY.
  " Create destination with recipient address
  DATA lt_to_addresses TYPE /aws1/
cl_se2emailaddresslist_w=>tt_emailaddresslist.
  APPEND NEW /aws1/cl_se2emailaddresslist_w( iv_value = iv_to_email_address )
TO lt_to_addresses.
  DATA(lo_destination) = NEW /aws1/cl_se2destination(
    it_toaddresses = lt_to_addresses ).

  " Create message content
  DATA(lo_subject) = NEW /aws1/cl_se2content( iv_data = iv_subject ).
  DATA(lo_text_body) = NEW /aws1/cl_se2content( iv_data = iv_text_body ).
  DATA(lo_html_body) = NEW /aws1/cl_se2content( iv_data = iv_html_body ).
  DATA(lo_body) = NEW /aws1/cl_se2body(
    io_text = lo_text_body
    io_html = lo_html_body ).
  DATA(lo_message) = NEW /aws1/cl_se2message(
    io_subject = lo_subject
    io_body = lo_body ).

  DATA(lo_content) = NEW /aws1/cl_se2emailcontent(
    io_simple = lo_message ).

  " Send the email
  lo_se2->sendemail(
    iv_fromemailaddress = iv_from_email_address
    io_destination = lo_destination
    io_content = lo_content ).
  MESSAGE 'Email sent successfully.' TYPE 'I'.
CATCH /aws1/cx_se2accountsuspendedex INTO DATA(lo_account_suspended).
  MESSAGE 'Account suspended.' TYPE 'I'.
  RAISE EXCEPTION lo_account_suspended.
CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).
```

```

MESSAGE 'Bad request.' TYPE 'I'.
RAISE EXCEPTION lo_bad_request.
CATCH /aws1/cx_se2messagerejected INTO DATA(lo_message_rejected).
MESSAGE 'Message rejected - check email verification.' TYPE 'I'.
RAISE EXCEPTION lo_message_rejected.
ENDTRY.

```

Envia uma mensagem usando um modelo.

```

TRY.
  " Create destination with recipient address
  DATA lt_to_addresses TYPE /aws1/
cl_se2emailaddresslist_w=>tt_emailaddresslist.
  APPEND NEW /aws1/cl_se2emailaddresslist_w( iv_value = iv_to_email_address )
TO lt_to_addresses.
  DATA(lo_destination) = NEW /aws1/cl_se2destination(
    it_toaddresses = lt_to_addresses ).

  " Create template reference
  DATA(lo_template) = NEW /aws1/cl_se2template(
    iv_templatename = iv_template_name
    iv_templatedata = iv_template_data ).

  DATA(lo_content) = NEW /aws1/cl_se2emailcontent(
    io_template = lo_template ).

  " Create list management options
  DATA(lo_list_mgmt) = NEW /aws1/cl_se2listmanagementopts(
    iv_contactlistname = iv_contact_list_name ).

  " Send the email using template
  lo_se2->sendemail(
    iv_fromemailaddress = iv_from_email_address
    io_destination = lo_destination
    io_content = lo_content
    io_listmanagementoptions = lo_list_mgmt ).
  MESSAGE 'Email sent using template successfully.' TYPE 'I'.
  CATCH /aws1/cx_se2accountsuspendedex INTO DATA(lo_account_suspended).
  MESSAGE 'Account suspended.' TYPE 'I'.
  RAISE EXCEPTION lo_account_suspended.
  CATCH /aws1/cx_se2badrequestex INTO DATA(lo_bad_request).
  MESSAGE 'Bad request.' TYPE 'I'.

```

```
RAISE EXCEPTION lo_bad_request.  
CATCH /aws1/cx_se2messagerejected INTO DATA(lo_message_rejected).  
MESSAGE 'Message rejected - check email verification.' TYPE 'I'.  
RAISE EXCEPTION lo_message_rejected.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [SendEmail](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon SNS usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon SNS.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos


- [Ações](#)
- [Cenários](#)

## Ações

### CreateTopic

O código de exemplo a seguir mostra como usar CreateTopic.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_sns->createtopic( iv_name = iv_topic_name ). " oo_result is  
returned for testing purposes. "  
    MESSAGE 'SNS topic created' TYPE 'I'.  
    CATCH /aws1/cx_snstopiclimitexcdex.  
        MESSAGE 'Unable to create more topics. You have reached the maximum number  
of topics allowed.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateTopic](#) referência da API AWS SDK for SAP ABAP.

## DeleteTopic

O código de exemplo a seguir mostra como usar DeleteTopic.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_sns->deletetopic( iv_topicarn = iv_topic_arn ).  
    MESSAGE 'SNS topic deleted.' TYPE 'I'.  
    CATCH /aws1/cx_snsnotfoundexception.  
        MESSAGE 'Topic does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteTopic](#) referência da API AWS SDK for SAP ABAP.

## GetTopicAttributes

O código de exemplo a seguir mostra como usar GetTopicAttributes.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_sns->gettopicattributes( iv_topicarn = iv_topic_arn ). "  
oo_result is returned for testing purposes. "  
    DATA(lt_attributes) = oo_result->get_attributes( ).  
    MESSAGE 'Retrieved attributes/properties of a topic.' TYPE 'I'.  
    CATCH /aws1/cx_snsnotfoundexception.  
        MESSAGE 'Topic does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetTopicAttributes](#) referência da API AWS SDK for SAP ABAP.

## ListSubscriptions

O código de exemplo a seguir mostra como usar ListSubscriptions.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_sns->listsubscriptions( ).           " oo_result is  
returned for testing purposes. "  
    DATA(lt_subscriptions) = oo_result->get_subscriptions( ).  
    MESSAGE 'Retrieved list of subscribers.' TYPE 'I'.  
    CATCH /aws1/cx_rt_generic.  
        MESSAGE 'Unable to list subscribers.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListSubscriptions](#) referência da API AWS SDK for SAP ABAP.

## ListTopics

O código de exemplo a seguir mostra como usar ListTopics.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_sns->listtopics( ).                 " oo_result is returned for  
testing purposes. "  
    DATA(lt_topics) = oo_result->get_topics( ).  
    MESSAGE 'Retrieved list of topics.' TYPE 'I'.  
    CATCH /aws1/cx_rt_generic.  
        MESSAGE 'Unable to list topics.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListTopics](#) referência da API AWS SDK for SAP ABAP.

## Publish

O código de exemplo a seguir mostra como usar Publish.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    oo_result = lo_sns->publish(           " oo_result is returned for
testing purposes. "
    iv_topicarn = iv_topic_arn
    iv_message = iv_message ).
    MESSAGE 'Message published to SNS topic.' TYPE 'I'.
CATCH /aws1/cx_snsnotfoundexception.
    MESSAGE 'Topic does not exist.' TYPE 'E'.
ENDTRY.

```

Publique uma mensagem com atributos para um tópico.

```

TRY.
    oo_result = lo_sns->publish(           " oo_result is returned for
testing purposes. "
    iv_topicarn = iv_topic_arn
    iv_message = iv_message
    it_messageattributes = it_msg_attrs ).
    MESSAGE 'Message with attributes published to SNS topic.' TYPE 'I'.
CATCH /aws1/cx_snsnotfoundexception.
    MESSAGE 'Topic does not exist.' TYPE 'E'.
ENDTRY.

```

Publique uma mensagem em vários formatos em um tópico.

```

" Build JSON message structure for multi-format message
DATA(lv_json_message) = |\{ "default": "{ iv_default_message }", "sms":
"{ iv_sms_message }", "email": "{ iv_email_message }" \}|.

TRY.

```

```

        oo_result = lo_sns->publish(
testing purposes. " " oo_result is returned for
        iv_topicarn = iv_topic_arn
        iv_message = lv_json_message
        iv_subject = iv_subject
        iv_messagestructure = 'json' ).
MESSAGE 'Multi-format message published to SNS topic.' TYPE 'I'.
CATCH /aws1/cx_snsnotfoundexception.
MESSAGE 'Topic does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter os detalhes da API, consulte [Publicar](#) na Referência da API do AWS SDK para SAP ABAP.

## SetSubscriptionAttributes

O código de exemplo a seguir mostra como usar SetSubscriptionAttributes.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    lo_sns->setsubscriptionattributes(
        iv_subscriptionarn = iv_subscription_arn
        iv_attributename = 'FilterPolicy'
        iv_attributevalue = iv_filter_policy ).
MESSAGE 'Added filter policy to subscription.' TYPE 'I'.
CATCH /aws1/cx_snsnotfoundexception.
MESSAGE 'Subscription does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [SetSubscriptionAttributes](#) referência da API AWS SDK for SAP ABAP.

## SetTopicAttributes

O código de exemplo a seguir mostra como usar SetTopicAttributes.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_sns->settopicattributes(  
        iv_topicarn = iv_topic_arn  
        iv_attributename = iv_attribute_name  
        iv_attributevalue = iv_attribute_value ).  
    MESSAGE 'Set/updated SNS topic attributes.' TYPE 'I'.  
CATCH /aws1/cx_snsnotfoundexception.  
    MESSAGE 'Topic does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [SetTopicAttributes](#) referência da API AWS SDK for SAP ABAP.

## Subscribe

O código de exemplo a seguir mostra como usar Subscribe.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Inscrever um endereço de e-mail em um tópico.

```
TRY.
    oo_result = lo_sns->subscribe(
for testing purposes."
        iv_topicarn = iv_topic_arn
        iv_protocol = 'email'
        iv_endpoint = iv_email_address
        iv_returnsubscriptionarn = abap_true ).
    MESSAGE 'Email address subscribed to SNS topic.' TYPE 'I'.
CATCH /aws1/cx_snsnotfoundexception.
    MESSAGE 'Topic does not exist.' TYPE 'E'.
CATCH /aws1/cx_snssubscriptionlmt00.
    MESSAGE 'Unable to create subscriptions. You have reached the maximum number
of subscriptions allowed.' TYPE 'E'.
ENDTRY.
```

- Consulte detalhes da API em [Inscrever-se](#) na Referência da API do AWS SDK para SAP ABAP.

## Unsubscribe

O código de exemplo a seguir mostra como usar Unsubscribe.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    lo_sns->unsubscribe( iv_subscriptionarn = iv_subscription_arn ).
    MESSAGE 'Subscription deleted.' TYPE 'I'.
CATCH /aws1/cx_snsnotfoundexception.
    MESSAGE 'Subscription does not exist.' TYPE 'E'.
CATCH /aws1/cx_snsinvalidparameterex.
    MESSAGE 'Subscription with "PendingConfirmation" status cannot be deleted/
unsubscribed. Confirm subscription before performing unsubscribe operation.' TYPE
'E'.
ENDTRY.
```

- Consulte detalhes da API em [Cancelar assinatura](#) na Referência da API do AWS SDK para SAP ABAP.

## Cenários

Criar e publicar em um tópico FIFO

O exemplo de código a seguir mostra como criar e publicar em um tópico FIFO do Amazon SNS.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie um tópico FIFO, inscreva uma fila FIFO do Amazon SQS no tópico e publique uma mensagem em um tópico do Amazon SNS.

```
" Creates a FIFO topic. "
DATA lt_tpc_attributes TYPE /aws1/cl_snstopicattrsmw=>tt_topicattributesmap.
DATA ls_tpc_attributes TYPE /aws1/
cl_snstopicattrsmw=>ts_topicattributesmap_maprow.
ls_tpc_attributes-key = 'FifoTopic'.
ls_tpc_attributes-value = NEW /aws1/cl_snstopicattrsmw( iv_value = 'true' ).
INSERT ls_tpc_attributes INTO TABLE lt_tpc_attributes.

TRY.
  DATA(lo_create_result) = lo_sns->createtopic(
    iv_name = iv_topic_name
    it_attributes = lt_tpc_attributes ).
  DATA(lv_topic_arn) = lo_create_result->get_topicarn( ).
  ov_topic_arn = lv_topic_arn.
ov_topic_arn is returned for testing purposes. "
  MESSAGE 'FIFO topic created' TYPE 'I'.
CATCH /aws1/cx_snstopicclimitexcdex.
```

```

    MESSAGE 'Unable to create more topics. You have reached the maximum number
of topics allowed.' TYPE 'E'.
    ENDRTRY.

```

```

    " Subscribes an endpoint to an Amazon Simple Notification Service (Amazon SNS)
topic. "

```

```

    " Only Amazon Simple Queue Service (Amazon SQS) FIFO queues can be subscribed to
an SNS FIFO topic. "

```

```

    TRY.

```

```

        DATA(lo_subscribe_result) = lo_sns->subscribe(
            iv_topicarn = lv_topic_arn
            iv_protocol = 'sqs'
            iv_endpoint = iv_queue_arn ).

```

```

        DATA(lv_subscription_arn) = lo_subscribe_result->get_subscriptionarn( ).
        ov_subscription_arn = lv_subscription_arn.

```

```

    ov_subscription_arn is returned for testing purposes. "

```

```

    MESSAGE 'SQS queue was subscribed to SNS topic.' TYPE 'I'.

```

```

    CATCH /aws1/cx_snsnotfoundexception.

```

```

    MESSAGE 'Topic does not exist.' TYPE 'E'.

```

```

    CATCH /aws1/cx_snssubscriptionlmt00.

```

```

    MESSAGE 'Unable to create subscriptions. You have reached the maximum number
of subscriptions allowed.' TYPE 'E'.

```

```

    ENDRTRY.

```

```

    " Publish message to SNS topic. "

```

```

    TRY.

```

```

        DATA lt_msg_attributes TYPE /aws1/

```

```

    cl_snsmessageattrvalue=>tt_messageattributemap.

```

```

        DATA ls_msg_attributes TYPE /aws1/

```

```

    cl_snsmessageattrvalue=>ts_messageattributemap_maprow.

```

```

        ls_msg_attributes-key = 'Importance'.

```

```

        ls_msg_attributes-value = NEW /aws1/cl_snsmessageattrvalue( iv_datatype =
'String'

```

```

                                                                    iv_stringvalue =

```

```

'String' ).

```

```

        INSERT ls_msg_attributes INTO TABLE lt_msg_attributes.

```

```

        DATA(lo_result) = lo_sns->publish(

```

```

            iv_topicarn = lv_topic_arn

```

```

            iv_message = 'The price of your mobile plan has been increased from $19
to $23'

```

```

            iv_subject = 'Changes to mobile plan'

```

```

            iv_messagegroupid = 'Update-2'

```

```

            iv_messagededuplicationid = 'Update-2.1'

```

```

        it_messageattributes = lt_msg_attributes ).
        ov_message_id = lo_result->get_messageid( ).
        ov_message_id is returned for testing purposes. "
        MESSAGE 'Message was published to SNS topic.' TYPE 'I'.
        CATCH /aws1/cx_snsnotfoundexception.
        MESSAGE 'Topic does not exist.' TYPE 'E'.
    ENDTRY.

```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK para SAP ABAP.
  - [CreateTopic](#)
  - [Publicar](#)
  - [Assinar](#)

## Publicar uma mensagem de texto SMS

O exemplo de código a seguir mostra como publicar mensagens SMS usando o Amazon SNS.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

" iv_phone_number = '+12065550101' - Phone number in E.164 format
TRY.
    oo_result = lo_sns->publish(
        testing purposes. "
        iv_phonenumber = iv_phone_number
        iv_message = iv_message ).
    MESSAGE 'Message published to phone number.' TYPE 'I'.
    CATCH /aws1/cx_snsnotfoundexception.
    MESSAGE 'Phone number does not exist.' TYPE 'E'.
ENDTRY.

```

- Para obter os detalhes da API, consulte [Publicar](#) na Referência da API do AWS SDK para SAP ABAP.

## Exemplos do Amazon SQS usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon SQS.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)
- [Cenários](#)

## Ações

### CreateQueue

O código de exemplo a seguir mostra como usar CreateQueue.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie uma fila padrão do Amazon SQS.

TRY.

```

        oo_result = lo_sqs->createqueue( iv_queue_name = iv_queue_name ).
oo_result is returned for testing purposes. "
        MESSAGE 'SQS queue created.' TYPE 'I'.
        CATCH /aws1/cx_sqsqueuedeletedrecently.
        MESSAGE 'After deleting a queue, wait 60 seconds before creating another
queue with the same name.' TYPE 'E'.
        CATCH /aws1/cx_sqsqueueexists.
        MESSAGE 'A queue with this name already exists.' TYPE 'E'.
    ENDDY.

```

Crie uma fila do Amazon SQS que aguarda a chegada de uma mensagem.

```

    TRY.
        DATA lt_attributes TYPE /aws1/cl_sqsqueueattribmap_w=>tt_queueattributemap.
        DATA ls_attribute TYPE /aws1/
cl_sqsqueueattribmap_w=>ts_queueattributemap_maprow.
        ls_attribute-key = 'ReceiveMessageWaitTimeSeconds'. " Time in
seconds for long polling, such as how long the call waits for a message to arrive
in the queue before returning. "
        ls_attribute-value = NEW /aws1/cl_sqsqueueattribmap_w( iv_value =
iv_wait_time ).
        INSERT ls_attribute INTO TABLE lt_attributes.
        oo_result = lo_sqs->createqueue( " oo_result is returned
for testing purposes. "
            iv_queue_name = iv_queue_name
            it_attributes = lt_attributes ).
        MESSAGE 'SQS queue created.' TYPE 'I'.
        CATCH /aws1/cx_sqsqueuedeletedrecently.
        MESSAGE 'After deleting a queue, wait 60 seconds before creating another
queue with the same name.' TYPE 'E'.
        CATCH /aws1/cx_sqsqueueexists.
        MESSAGE 'A queue with this name already exists.' TYPE 'E'.
    ENDDY.


```

- Para obter detalhes da API, consulte a [CreateQueue](#) referência da API AWS SDK for SAP ABAP.

## DeleteMessage

O código de exemplo a seguir mostra como usar DeleteMessage.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_sqs->deletemessage(  
    iv_queueurl = iv_queue_url  
    iv_receipthandle = iv_receipt_handle ).  
  MESSAGE 'Message deleted from SQS queue.' TYPE 'I'.  
CATCH /aws1/cx_sqsinvalididformat.  
  MESSAGE 'The specified receipt handle is not valid.' TYPE 'E'.  
CATCH /aws1/cx_sqsreceipthandleisinv.  
  MESSAGE 'The specified receipt handle is not valid for the current version.'  
TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteMessage](#) referência da API AWS SDK for SAP ABAP.

## DeleteMessageBatch

O código de exemplo a seguir mostra como usar DeleteMessageBatch.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_sqs->deletemessagebatch( " oo_result is returned for  
testing purposes. "
```

```
        iv_queueurl = iv_queue_url
        it_entries = it_entries ).
    MESSAGE 'Messages deleted from SQS queue.' TYPE 'I'.
    CATCH /aws1/cx_sqsbcentidsnotdist00.
        MESSAGE 'Two or more batch entries in the request have the same ID.' TYPE
'E'.
    CATCH /aws1/cx_sqsemptybatchrequest.
        MESSAGE 'The batch request does not contain any entries.' TYPE 'E'.
    CATCH /aws1/cx_sqsinvbatchentryid.
        MESSAGE 'The ID of a batch entry in a batch request is not valid.' TYPE 'E'.
    CATCH /aws1/cx_sqstoomanyentriesin00.
        MESSAGE 'The batch request contains more entries than allowed.' TYPE 'E'.
    ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteMessageBatch](#) referência da API AWS SDK for SAP ABAP.

## DeleteQueue

O código de exemplo a seguir mostra como usar DeleteQueue.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
    TRY.
        lo_sqs->deletequeue( iv_queueurl = iv_queue_url ).
        MESSAGE 'SQS queue deleted' TYPE 'I'.
    ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteQueue](#) referência da API AWS SDK for SAP ABAP.

## GetQueueUrl

O código de exemplo a seguir mostra como usar `GetQueueUrl`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_sqs->getqueueurl( iv_queuename = iv_queue_name ).      "  
oo_result is returned for testing purposes. "  
    MESSAGE 'Queue URL retrieved.' TYPE 'I'.  
    CATCH /aws1/cx_sqsqueuedoesnotexist.  
        MESSAGE 'The requested queue does not exist.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetQueueUrl](#) referência da API AWS SDK for SAP ABAP.

## ListQueues

O código de exemplo a seguir mostra como usar `ListQueues`.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_sqs->listqueues( ).      " oo_result is returned for  
testing purposes. "
```

```
MESSAGE 'Retrieved list of queues.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListQueues](#) referência da API AWS SDK for SAP ABAP.

## ReceiveMessage

O código de exemplo a seguir mostra como usar ReceiveMessage.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Receba uma mensagem de uma fila do Amazon SQS.

```
TRY.
    oo_result = lo_sqs->receivemessage( iv_queueurl = iv_queue_url ).    "
oo_result is returned for testing purposes. "
    DATA(lt_messages) = oo_result->get_messages( ).
    MESSAGE 'Message received from SQS queue.' TYPE 'I'.
CATCH /aws1/cx_sqsoverlimit.
    MESSAGE 'Maximum number of in-flight messages reached.' TYPE 'E'.
ENDTRY.
```

Receba uma mensagem de uma fila do Amazon SQS usando o suporte de sondagem longa.

```
TRY.
    oo_result = lo_sqs->receivemessage(           " oo_result is returned for
testing purposes. "
        iv_queueurl = iv_queue_url
        iv_waittimeseconds = iv_wait_time ).    " Time in seconds for long
polling, such as how long the call waits for a message to arrive in the queue
before returning. " ).
    DATA(lt_messages) = oo_result->get_messages( ).
    MESSAGE 'Message received from SQS queue.' TYPE 'I'.
```

```
CATCH /aws1/cx_sqsoverlimit.
  MESSAGE 'Maximum number of in-flight messages reached.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [ReceiveMessage](#) referência da API AWS SDK for SAP ABAP.

## SendMessage

O código de exemplo a seguir mostra como usar SendMessage.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.
  oo_result = lo_sqs->sendmessage(           " oo_result is returned for
testing purposes. "
  iv_queueurl = iv_queue_url
  iv_messagebody = iv_message ).
  MESSAGE 'Message sent to SQS queue.' TYPE 'I'.
CATCH /aws1/cx_sqsinvalidmsgcnts.
  MESSAGE 'Message contains non-valid characters.' TYPE 'E'.
CATCH /aws1/cx_sqsunsupportedop.
  MESSAGE 'Operation not supported.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [SendMessage](#) referência da API AWS SDK for SAP ABAP.

## SendMessageBatch

O código de exemplo a seguir mostra como usar SendMessageBatch.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    oo_result = lo_sqs->sendmessagebatch(          " oo_result is returned for
testing purposes. "
        iv_queueurl = iv_queue_url
        it_entries = it_messages ).
    MESSAGE 'Messages sent to SQS queue.' TYPE 'I'.
CATCH /aws1/cx_sqsbtcentidsnotdist00.
    MESSAGE 'Two or more batch entries in the request have the same ID.' TYPE
'E'.
CATCH /aws1/cx_sqsbatchreqtoolong.
    MESSAGE 'The length of all the messages put together is more than the
limit.' TYPE 'E'.
CATCH /aws1/cx_sqsemptybatchrequest.
    MESSAGE 'The batch request does not contain any entries.' TYPE 'E'.
CATCH /aws1/cx_sqsinvbatchentryid.
    MESSAGE 'The ID of a batch entry in a batch request is not valid.' TYPE 'E'.
CATCH /aws1/cx_sqstoomanyentriesin00.
    MESSAGE 'The batch request contains more entries than allowed.' TYPE 'E'.
CATCH /aws1/cx_sqsunsupportedop.
    MESSAGE 'Operation not supported.' TYPE 'E'.
ENDTRY.
```


- Para obter detalhes da API, consulte a [SendMessageBatch](#) referência da API AWS SDK for SAP ABAP.

## Cenários

### Criar e publicar em um tópico FIFO

O exemplo de código a seguir mostra como criar e publicar em um tópico FIFO do Amazon SNS.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie um tópico FIFO, inscreva uma fila FIFO do Amazon SQS no tópico e publique uma mensagem em um tópico do Amazon SNS.

```

" Creates a FIFO topic. "
DATA lt_tpc_attributes TYPE /aws1/cl_snstopicattrsmw=>tt_topicattributesmap.
DATA ls_tpc_attributes TYPE /aws1/
cl_snstopicattrsmw=>ts_topicattributesmap_maprow.
ls_tpc_attributes-key = 'FifoTopic'.
ls_tpc_attributes-value = NEW /aws1/cl_snstopicattrsmw( iv_value = 'true' ).
INSERT ls_tpc_attributes INTO TABLE lt_tpc_attributes.

TRY.
  DATA(lo_create_result) = lo_sns->createtopic(
    iv_name = iv_topic_name
    it_attributes = lt_tpc_attributes ).
  DATA(lv_topic_arn) = lo_create_result->get_topicarn( ).
  ov_topic_arn = lv_topic_arn.
ov_topic_arn is returned for testing purposes. "
  MESSAGE 'FIFO topic created' TYPE 'I'.
CATCH /aws1/cx_snstopiclimitexcdex.
  MESSAGE 'Unable to create more topics. You have reached the maximum number
of topics allowed.' TYPE 'E'.
ENDTRY.

" Subscribes an endpoint to an Amazon Simple Notification Service (Amazon SNS)
topic. "
" Only Amazon Simple Queue Service (Amazon SQS) FIFO queues can be subscribed to
an SNS FIFO topic. "
TRY.
  DATA(lo_subscribe_result) = lo_sns->subscribe(
    iv_topicarn = lv_topic_arn
    iv_protocol = 'sqs'
    iv_endpoint = iv_queue_arn ).

```

```

        DATA(lv_subscription_arn) = lo_subscribe_result->get_subscriptionarn( ).
        ov_subscription_arn = lv_subscription_arn.
    "
    ov_subscription_arn is returned for testing purposes. "
    MESSAGE 'SQS queue was subscribed to SNS topic.' TYPE 'I'.
    CATCH /aws1/cx_snsnotfoundexception.
        MESSAGE 'Topic does not exist.' TYPE 'E'.
    CATCH /aws1/cx_snssubscriptionlmte00.
        MESSAGE 'Unable to create subscriptions. You have reached the maximum number
of subscriptions allowed.' TYPE 'E'.
    ENDRY.

" Publish message to SNS topic. "
TRY.
    DATA lt_msg_attributes TYPE /aws1/
cl_snsmessageattrvalue=>tt_messageattributemap.
    DATA ls_msg_attributes TYPE /aws1/
cl_snsmessageattrvalue=>ts_messageattributemap_maprow.
    ls_msg_attributes-key = 'Importance'.
    ls_msg_attributes-value = NEW /aws1/cl_snsmessageattrvalue( iv_datatype =
'String'
                                                                    iv_stringvalue =
'High' ).
    INSERT ls_msg_attributes INTO TABLE lt_msg_attributes.

    DATA(lo_result) = lo_sns->publish(
        iv_topicarn = lv_topic_arn
        iv_message = 'The price of your mobile plan has been increased from $19
to $23'
        iv_subject = 'Changes to mobile plan'
        iv_messagegroupid = 'Update-2'
        iv_messagededuplicationid = 'Update-2.1'
        it_messageattributes = lt_msg_attributes ).
    ov_message_id = lo_result->get_messageid( ).
    "
    ov_message_id is returned for testing purposes. "
    MESSAGE 'Message was published to SNS topic.' TYPE 'I'.
    CATCH /aws1/cx_snsnotfoundexception.
        MESSAGE 'Topic does not exist.' TYPE 'E'.
    ENDRY.

```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK para SAP ABAP.

- [CreateTopic](#)
- [Publicar](#)
- [Assinar](#)

## Exemplos de Step Functions usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Step Functions.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateActivity

O código de exemplo a seguir mostra como usar CreateActivity.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    DATA(lo_result) = lo_sfn->createactivity(  
        iv_name = iv_name  
    ).  
    ov_activity_arn = lo_result->get_activityarn( ).
```

```

    MESSAGE 'Activity created successfully.' TYPE 'I'.
  CATCH /aws1/cx_sfnactivityalrddyex.
    MESSAGE 'Activity already exists.' TYPE 'E'.
  CATCH /aws1/cx_sfninvalidname.
    MESSAGE 'Invalid activity name.' TYPE 'E'.
  CATCH /aws1/cx_sfnactivitylimitexcd.
    MESSAGE 'Activity limit exceeded.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [CreateActivity](#) referência da API AWS SDK for SAP ABAP.

## CreateStateMachine

O código de exemplo a seguir mostra como usar CreateStateMachine.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  DATA(lo_result) = lo_sfn->createstatemachine(
    iv_name = iv_name
    iv_definition = iv_definition
    iv_rolearn = iv_role_arn
  ).
  ov_state_machine_arn = lo_result->get_statemachearn( ).
  MESSAGE 'State machine created successfully.' TYPE 'I'.
  CATCH /aws1/cx_sfnstatemachearlex.
    MESSAGE 'State machine already exists.' TYPE 'E'.
  CATCH /aws1/cx_sfninvaliddefinition.
    MESSAGE 'Invalid state machine definition.' TYPE 'E'.
  CATCH /aws1/cx_sfninvalidname.
    MESSAGE 'Invalid state machine name.' TYPE 'E'.
  CATCH /aws1/cx_sfninvalidarn.
    MESSAGE 'Invalid role ARN.' TYPE 'E'.

```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateStateMachine](#) referência da API AWS SDK for SAP ABAP.

## DeleteActivity

O código de exemplo a seguir mostra como usar DeleteActivity.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_sfn->deleteactivity(  
    iv_activityarn = iv_activity_arn  
  ).  
  MESSAGE 'Activity deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_sfninvalidarn.  
  MESSAGE 'Invalid activity ARN.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteActivity](#) referência da API AWS SDK for SAP ABAP.

## DeleteStateMachine

O código de exemplo a seguir mostra como usar DeleteStateMachine.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_sfn->deletestatemachine(  
    iv_statemachearn = iv_state_machine_arn  
  ).  
  MESSAGE 'State machine deleted successfully.' TYPE 'I'.  
CATCH /aws1/cx_sfninvalidarn.  
  MESSAGE 'Invalid state machine ARN.' TYPE 'E'.  
CATCH /aws1/cx_sfnvalidationex.  
  MESSAGE 'Validation error occurred.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteStateMachine](#) referência da API AWS SDK for SAP ABAP.

**DescribeExecution**

O código de exemplo a seguir mostra como usar DescribeExecution.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  oo_result = lo_sfn->describeexecution(  
    iv_executionarn = iv_execution_arn  
  ).  
  MESSAGE 'Execution described successfully.' TYPE 'I'.  
ENDTRY.
```

```
CATCH /aws1/cx_sfnextdoesnotexist.  
    MESSAGE 'Execution does not exist.' TYPE 'E'.  
CATCH /aws1/cx_sfninvalidarn.  
    MESSAGE 'Invalid execution ARN.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeExecution](#) referência da API AWS SDK for SAP ABAP.

## DescribeStateMachine

O código de exemplo a seguir mostra como usar DescribeStateMachine.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_sfn->describestatemachine(  
        iv_statemachinarn = iv_state_machine_arn  
    ).  
    MESSAGE 'State machine described successfully.' TYPE 'I'.  
CATCH /aws1/cx_sfnstatemachinedoes00.  
    MESSAGE 'State machine does not exist.' TYPE 'E'.  
CATCH /aws1/cx_sfninvalidarn.  
    MESSAGE 'Invalid state machine ARN.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeStateMachine](#) referência da API AWS SDK for SAP ABAP.

## GetActivityTask

O código de exemplo a seguir mostra como usar GetActivityTask.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_sfn->getactivitytask(  
        iv_activityarn = iv_activity_arn  
    ).  
    MESSAGE 'Activity task retrieved successfully.' TYPE 'I'.  
CATCH /aws1/cx_sfnactivitydoesnotex.  
    MESSAGE 'Activity does not exist.' TYPE 'E'.  
CATCH /aws1/cx_sfninvalidarn.  
    MESSAGE 'Invalid activity ARN.' TYPE 'E'.  
CATCH /aws1/cx_sfnactivityworkerlm00.  
    MESSAGE 'Activity worker limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetActivityTask](#) referência da API AWS SDK for SAP ABAP.

## ListActivities

O código de exemplo a seguir mostra como usar ListActivities.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    DATA(lo_result) = lo_sfn->listactivities( ).
```

```

DATA(lt_activities) = lo_result->get_activities( ).
LOOP AT lt_activities INTO DATA(lo_activity).
  IF lo_activity->get_name( ) = iv_name.
    ov_activity_arn = lo_activity->get_activityarn( ).
    EXIT.
  ENDIF.
ENDLOOP.
MESSAGE 'Activities listed successfully.' TYPE 'I'.
CATCH /aws1/cx_sfninvalidtoken.
  MESSAGE 'Invalid pagination token.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListActivities](#) referência da API AWS SDK for SAP ABAP.

## ListStateMachines

O código de exemplo a seguir mostra como usar ListStateMachines.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  DATA(lo_result) = lo_sfn->liststatemachines( ).
  DATA(lt_state_machines) = lo_result->get_statemachines( ).
  LOOP AT lt_state_machines INTO DATA(lo_state_machine).
    IF lo_state_machine->get_name( ) = iv_name.
      ov_state_machine_arn = lo_state_machine->get_statemachinearn( ).
      EXIT.
    ENDIF.
  ENDLOOP.
  MESSAGE 'State machines listed successfully.' TYPE 'I'.
  CATCH /aws1/cx_sfninvalidtoken.
    MESSAGE 'Invalid pagination token.' TYPE 'E'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [ListStateMachines](#) referência da API AWS SDK for SAP ABAP.

## SendTaskSuccess

O código de exemplo a seguir mostra como usar SendTaskSuccess.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_sfn->sendtasksuccess(  
    iv_tasktoken = iv_task_token  
    iv_output = iv_task_response  
  ).  
  MESSAGE 'Task success sent successfully.' TYPE 'I'.  
CATCH /aws1/cx_sfninvalidtoken.  
  MESSAGE 'Invalid task token.' TYPE 'E'.  
CATCH /aws1/cx_sfntaskdoesnotexist.  
  MESSAGE 'Task does not exist.' TYPE 'E'.  
CATCH /aws1/cx_sfninvalidoutput.  
  MESSAGE 'Invalid task output.' TYPE 'E'.  
CATCH /aws1/cx_sfntasktimedout.  
  MESSAGE 'Task timed out.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [SendTaskSuccess](#) referência da API AWS SDK for SAP ABAP.

## StartExecution

O código de exemplo a seguir mostra como usar StartExecution.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  DATA(lo_result) = lo_sfn->startexecution(  
    iv_statemachinearn = iv_state_machine_arn  
    iv_input = iv_input  
  ).  
  ov_execution_arn = lo_result->get_executionarn( ).  
  MESSAGE 'Execution started successfully.' TYPE 'I'.  
CATCH /aws1/cx_sfnstatemachinedoes00.  
  MESSAGE 'State machine does not exist.' TYPE 'E'.  
CATCH /aws1/cx_sfninvalidarn.  
  MESSAGE 'Invalid state machine ARN.' TYPE 'E'.  
CATCH /aws1/cx_sfninvalidexecinput.  
  MESSAGE 'Invalid execution input.' TYPE 'E'.  
CATCH /aws1/cx_sfnexeclimitexceeded.  
  MESSAGE 'Execution limit exceeded.' TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartExecution](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Systems Manager usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com Systems Manager.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

## Tópicos

- [Ações](#)

## Ações

### CreateDocument

O código de exemplo a seguir mostra como usar CreateDocument.

#### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  lo_ssm->createdocument(  
    iv_name = iv_name  
    iv_content = iv_content  
    iv_documenttype = 'Command' ).  
  MESSAGE 'Document created.' TYPE 'I'.  
CATCH /aws1/cx_ssmdocalreadyexists.  
  MESSAGE 'Document already exists.' TYPE 'I'.  
CATCH /aws1/cx_ssminvaliddoccontent.  
  MESSAGE 'Invalid document content.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateDocument](#) referência da API AWS SDK for SAP ABAP.

### CreateMaintenanceWindow

O código de exemplo a seguir mostra como usar CreateMaintenanceWindow.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
    oo_result = lo_ssm->createmaintenancewindow(  
        iv_name = iv_name  
        iv_schedule = iv_schedule  
        iv_duration = iv_duration  
        iv_cutoff = iv_cutoff  
        iv_allowunassociatedtargets = iv_allow_unassociated_targets ).  
    MESSAGE 'Maintenance window created.' TYPE 'I'.  
CATCH /aws1/cx_ssmresrclimitexcdex.  
    MESSAGE 'Resource limit exceeded.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateMaintenanceWindow](#) referência da API AWS SDK for SAP ABAP.

## CreateOpsItem

O código de exemplo a seguir mostra como usar CreateOpsItem.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_ssm->createopsitem(  
        iv_title = iv_title
```

```
        iv_source = iv_source
        iv_category = iv_category
        iv_severity = iv_severity
        iv_description = iv_description ).
    MESSAGE 'OpsItem created.' TYPE 'I'.
CATCH /aws1/cx_ssmopsitemlimitexcdex.
    MESSAGE 'You have exceeded your open OpsItem limit.' TYPE 'I'.
CATCH /aws1/cx_ssmopsitemalrddyexex.
    MESSAGE 'OpsItem already exists.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateOpsItem](#) referência da API AWS SDK for SAP ABAP.

## DeleteDocument

O código de exemplo a seguir mostra como usar DeleteDocument.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    lo_ssm->deletedocument( iv_name = iv_name ).
    MESSAGE 'Document deleted.' TYPE 'I'.
CATCH /aws1/cx_ssminvaliddocument.
    MESSAGE 'Invalid document.' TYPE 'I'.
CATCH /aws1/cx_ssmassocinstances.
    MESSAGE 'Document has associated instances.' TYPE 'I'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteDocument](#) referência da API AWS SDK for SAP ABAP.

## DeleteMaintenanceWindow

O código de exemplo a seguir mostra como usar DeleteMaintenanceWindow.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ssm->deletemaintenancewindow( iv_windowid = iv_window_id ).  
    MESSAGE 'Maintenance window deleted.' TYPE 'I'.  
CATCH /aws1/cx_ssminternalservererr.  
    MESSAGE 'Internal server error occurred.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteMaintenanceWindow](#) referência da API AWS SDK for SAP ABAP.

## DeleteOpsItem

O código de exemplo a seguir mostra como usar DeleteOpsItem.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    lo_ssm->deleteopsitem( iv_opsitemid = iv_ops_item_id ).  
    MESSAGE 'OpsItem deleted.' TYPE 'I'.  
CATCH /aws1/cx_ssmopsiteminvparamex.
```

```
MESSAGE 'Invalid OpsItem parameter.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DeleteOpsItem](#) referência da API AWS SDK for SAP ABAP.

## DescribeDocument

O código de exemplo a seguir mostra como usar DescribeDocument.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
TRY.  
  DATA(lo_result) = lo_ssm->describdocument( iv_name = iv_name ).  
  DATA(lo_document) = lo_result->get_document( ).  
  IF lo_document IS BOUND.  
    rv_status = lo_document->get_status( ).  
    MESSAGE |Document status: { rv_status }| TYPE 'I'.  
  ENDIF.  
  CATCH /aws1/cx_ssminvaliddocument.  
    MESSAGE 'Invalid document.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeDocument](#) referência da API AWS SDK for SAP ABAP.

## DescribeOpsItems

O código de exemplo a seguir mostra como usar DescribeOpsItems.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  " Create filter for OpsItem ID
  DATA(lt_filters) = VALUE /aws1/cl_ssmopsitemfilter=>tt_opsitemfilters(
    ( NEW /aws1/cl_ssmopsitemfilter(
      iv_key = 'OpsItemId'
      it_values = VALUE /aws1/
cl_ssmopsitemfiltvals_w=>tt_opsitemfiltervalues(
      ( NEW /aws1/cl_ssmopsitemfiltvals_w( iv_value = iv_ops_item_id ) )
    )
    iv_operator = 'Equal'
  ) )
  ).

  " Use paginator to get all results
  DATA(lo_paginator) = lo_ssm->get_paginator( ).
  DATA(lo_iterator) = lo_paginator->describeopsitems(
    it_opsitemfilters = lt_filters ).

  rv_found = abap_false.

  WHILE lo_iterator->has_next( ).
    DATA(lo_result) = CAST /aws1/cl_ssmdescropsitemrsp( lo_iterator-
>get_next( ) ).
    LOOP AT lo_result->get_opsitemsummaries( ) INTO DATA(lo_item).
      DATA(lv_title) = lo_item->get_title( ).
      DATA(lv_status) = lo_item->get_status( ).
      MESSAGE |The OpsItem title is { lv_title } and the status is
{ lv_status }| TYPE 'I'.
      rv_found = abap_true.
    ENDLOOP.
  ENDWHILE.
  CATCH /aws1/cx_ssminternalservererr.
    MESSAGE 'Internal server error occurred.' TYPE 'I'.
  ENDTRY.

```

- Para obter detalhes da API, consulte a [DescribeOpsItems](#) referência da API AWS SDK for SAP ABAP.

## ListCommandInvocations

O código de exemplo a seguir mostra como usar ListCommandInvocations.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
  " Use paginator to get all results
  DATA(lo_paginator) = lo_ssm->get_paginator( ).
  DATA(lo_iterator) = lo_paginator->listcommandinvocations(
    iv_instanceid = iv_instance_id ).

  DATA lv_count TYPE i VALUE 0.

  WHILE lo_iterator->has_next( ).
    DATA(lo_result) = CAST /aws1/cl_ssmlistcmdinvcsresult( lo_iterator-
>get_next( ) ).
    LOOP AT lo_result->get_commandinvocations( ) INTO DATA(lo_invocation).
      lv_count = lv_count + 1.
      DATA(lv_requested_datetime) = lo_invocation->get_requesteddatetime( ).
      MESSAGE |Command invocation requested at: { lv_requested_datetime }|
TYPE 'I'.
    ENDLLOOP.
  ENDWHILE.

  MESSAGE |{ lv_count } command invocation(s) found for instance
{ iv_instance_id }.| TYPE 'I'.
  CATCH /aws1/cx_ssminvalidinstanceid.
    MESSAGE 'Invalid instance ID.' TYPE 'I'.
  CATCH /aws1/cx_ssminvalidcommandid.
```

```
MESSAGE 'Invalid command ID.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [ListCommandInvocations](#) referência da API AWS SDK for SAP ABAP.

## SendCommand

O código de exemplo a seguir mostra como usar SendCommand.

### SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  DATA(lo_result) = lo_ssm->sendcommand(  
    it_instanceids = it_instance_ids  
    iv_documentname = iv_document_name  
    iv_timeoutseconds = 3600 ).  
  DATA(lo_command) = lo_result->get_command( ).  
  IF lo_command IS BOUND.  
    rv_command_id = lo_command->get_commandid( ).  
    MESSAGE 'Command sent successfully.' TYPE 'I'.  
  ENDIF.  
  CATCH /aws1/cx_ssminvaliddocument.  
    MESSAGE 'Invalid document.' TYPE 'I'.  
  CATCH /aws1/cx_ssminvalidinstanceid.  
    MESSAGE 'Invalid instance ID.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [SendCommand](#) referência da API AWS SDK for SAP ABAP.

## UpdateMaintenanceWindow

O código de exemplo a seguir mostra como usar UpdateMaintenanceWindow.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_ssm->updatemaintenancewindow(  
    iv_windowid = iv_window_id  
    iv_name = iv_name  
    iv_enabled = iv_enabled  
    iv_schedule = iv_schedule  
    iv_duration = iv_duration  
    iv_cutoff = iv_cutoff  
    iv_allowunassociatedtargets = iv_allow_unassociated_targets ).  
  MESSAGE 'Maintenance window updated.' TYPE 'I'.  
CATCH /aws1/cx_ssmdoesnotexistex.  
  MESSAGE 'Maintenance window does not exist.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateMaintenanceWindow](#) referência da API AWS SDK for SAP ABAP.

## UpdateOpsItem

O código de exemplo a seguir mostra como usar UpdateOpsItem.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  lo_ssm->updateopsitem(  
    iv_opsitemid = iv_ops_item_id  
    iv_title = iv_title  
    iv_description = iv_description  
    iv_status = iv_status ).  
  MESSAGE 'OpsItem updated.' TYPE 'I'.  
CATCH /aws1/cx_ssmopsitemnotfoundex.  
  MESSAGE 'OpsItem not found.' TYPE 'I'.  
CATCH /aws1/cx_ssmopsiteminvparamex.  
  MESSAGE 'Invalid OpsItem parameter.' TYPE 'I'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateOpsItem](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon Textract usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Textract.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos

- [Ações](#)
- [Cenários](#)

# Ações

## AnalyzeDocument

O código de exemplo a seguir mostra como usar AnalyzeDocument.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Detects text and additional elements, such as forms or tables,"
"in a local image file or from in-memory byte data."
"The image must be in PNG or JPG format."

"Create ABAP objects for feature type."
"Add TABLES to return information about the tables."
"Add FORMS to return detected form data."
"To perform both types of analysis, add TABLES and FORMS to FeatureTypes."

DATA(lt_featuretypes) = VALUE /aws1/cl_texfeaturetypes_w=>tt_featuretypes(
  ( NEW /aws1/cl_texfeaturetypes_w( iv_value = 'FORMS' ) )
  ( NEW /aws1/cl_texfeaturetypes_w( iv_value = 'TABLES' ) ) ).

"Create an ABAP object for the Amazon Simple Storage Service (Amazon S3)
object."
DATA(lo_s3object) = NEW /aws1/cl_texs3object( iv_bucket = iv_s3bucket
  iv_name   = iv_s3object ).

"Create an ABAP object for the document."
DATA(lo_document) = NEW /aws1/cl_texdocument( io_s3object = lo_s3object ).

"Analyze document stored in Amazon S3."
TRY.
  oo_result = lo_tex->analyzedocument(           "oo_result is returned for testing
purposes."
  io_document           = lo_document
```

```
        it_featuretypes = lt_featuretypes ).
    LOOP AT oo_result->get_blocks( ) INTO DATA(lo_block).
        IF lo_block->get_text( ) = 'INGREDIENTS: POWDERED SUGAR* (CANE SUGAR, '.
            MESSAGE 'Found text in the doc: ' && lo_block->get_text( ) TYPE 'I'.
        ENDIF.
    ENDLLOOP.
    MESSAGE 'Analyze document completed.' TYPE 'I'.
    CATCH /aws1/cx_texaccessdeniedex.
        MESSAGE 'You do not have permission to perform this action.' TYPE 'E'.
    CATCH /aws1/cx_texbaddocumentex.
        MESSAGE 'Amazon Textract is not able to read the document.' TYPE 'E'.
    CATCH /aws1/cx_texdocumenttoolargeex.
        MESSAGE 'The document is too large.' TYPE 'E'.
    CATCH /aws1/cx_texhlquotaexceededex.
        MESSAGE 'Human loop quota exceeded.' TYPE 'E'.
    CATCH /aws1/cx_texinternalservererr.
        MESSAGE 'Internal server error.' TYPE 'E'.
    CATCH /aws1/cx_texinvalidparameterex.
        MESSAGE 'Request has non-valid parameters.' TYPE 'E'.


    CATCH /aws1/cx_texinvalids3objectex.
        MESSAGE 'Amazon S3 object is not valid.' TYPE 'E'.
    CATCH /aws1/cx_texpvthruputexcdex.
        MESSAGE 'Provisioned throughput exceeded limit.' TYPE 'E'.
    CATCH /aws1/cx_texthrottlingex.
        MESSAGE 'The request processing exceeded the limit.' TYPE 'E'.
    CATCH /aws1/cx_texunsupporteddocex.
        MESSAGE 'The document is not supported.' TYPE 'E'.
    ENDTRY.
```

- Para obter detalhes da API, consulte a [AnalyzeDocument](#) referência da API AWS SDK for SAP ABAP.

## DetectDocumentText

O código de exemplo a seguir mostra como usar DetectDocumentText.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Detects text in the input document."
"Amazon Textract can detect lines of text and the words that make up a line of
text."
"The input document must be in one of the following image formats: JPEG, PNG,
PDF, or TIFF."

"Create an ABAP object for the Amazon S3 object."
DATA(lo_s3object) = NEW /aws1/cl_texs3object( iv_bucket = iv_s3bucket
      iv_name   = iv_s3object ).

"Create an ABAP object for the document."
DATA(lo_document) = NEW /aws1/cl_texdocument( io_s3object = lo_s3object ).
"Analyze document stored in Amazon S3."
TRY.
    oo_result = lo_tex->detectdocumenttext( io_document = lo_document ).
"oo_result is returned for testing purposes."
    LOOP AT oo_result->get_blocks( ) INTO DATA(lo_block).
        IF lo_block->get_text( ) = 'INGREDIENTS: POWDERED SUGAR* (CANE SUGAR, '
            MESSAGE 'Found text in the doc: ' && lo_block->get_text( ) TYPE 'I'.
        ENDIF.
    ENDLOOP.
    DATA(lo_metadata) = oo_result->get_documentmetadata( ).
    MESSAGE 'The number of pages in the document is ' && lo_metadata->ask_pages( ) TYPE 'I'.
    MESSAGE 'Detect document text completed.' TYPE 'I'.
CATCH /aws1/cx_texaccessdeniedex.
    MESSAGE 'You do not have permission to perform this action.' TYPE 'E'.
CATCH /aws1/cx_texbaddocumentex.
    MESSAGE 'Amazon Textract is not able to read the document.' TYPE 'E'.
CATCH /aws1/cx_texdocumenttoolargeex.
    MESSAGE 'The document is too large.' TYPE 'E'.
CATCH /aws1/cx_texinternalservererr.
    MESSAGE 'Internal server error.' TYPE 'E'.
```

```

CATCH /aws1/cx_texinvalidparameterex.
  MESSAGE 'Request has non-valid parameters.' TYPE 'E'.
CATCH /aws1/cx_texinvalids3objectex.
  MESSAGE 'Amazon S3 object is not valid.' TYPE 'E'.
CATCH /aws1/cx_texpvthruputexcdex.
  MESSAGE 'Provisioned throughput exceeded limit.' TYPE 'E'.
CATCH /aws1/cx_texthrottlingex.
  MESSAGE 'The request processing exceeded the limit' TYPE 'E'.
CATCH /aws1/cx_texunsupporteddocex.
  MESSAGE 'The document is not supported.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DetectDocumentText](#) referência da API AWS SDK for SAP ABAP.

## GetDocumentAnalysis

O código de exemplo a seguir mostra como usar GetDocumentAnalysis.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

"Gets the results for an Amazon Textract"
"asynchronous operation that analyzes text in a document."
TRY.
  oo_result = lo_tex->getdocumentanalysis( iv_jobid = iv_jobid ).
"oo_result is returned for testing purposes."
  WHILE oo_result->get_jobstatus( ) <> 'SUCCEEDED'.
    IF sy-index = 10.
      EXIT.          "Maximum 300 seconds.
    ENDIF.
    WAIT UP TO 30 SECONDS.
    oo_result = lo_tex->getdocumentanalysis( iv_jobid = iv_jobid ).
  ENDWHILE.

```

```
DATA(lt_blocks) = oo_result->get_blocks( ).
LOOP AT lt_blocks INTO DATA(lo_block).
  IF lo_block->get_text( ) = 'INGREDIENTS: POWDERED SUGAR* (CANE SUGAR, '.
    MESSAGE 'Found text in the doc: ' && lo_block->get_text( ) TYPE 'I'.
  ENDIF.
ENDLOOP.
MESSAGE 'Document analysis retrieved.' TYPE 'I'.
CATCH /aws1/cx_texaccessdeniedex.
  MESSAGE 'You do not have permission to perform this action.' TYPE 'E'.
CATCH /aws1/cx_texinternalservererr.
  MESSAGE 'Internal server error.' TYPE 'E'.
CATCH /aws1/cx_texinvalidjobidex.
  MESSAGE 'Job ID is not valid.' TYPE 'E'.
CATCH /aws1/cx_texinvalidkmskeyex.
  MESSAGE 'AWS KMS key is not valid.' TYPE 'E'.
CATCH /aws1/cx_texinvalidparameterex.
  MESSAGE 'Request has non-valid parameters.' TYPE 'E'.
CATCH /aws1/cx_texinvalids3objectex.
  MESSAGE 'Amazon S3 object is not valid.' TYPE 'E'.
CATCH /aws1/cx_texpvthruputexcdex.
  MESSAGE 'Provisioned throughput exceeded limit.' TYPE 'E'.
CATCH /aws1/cx_textrhottlingex.
  MESSAGE 'The request processing exceeded the limit.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetDocumentAnalysis](#) referência da API AWS SDK for SAP ABAP.

## StartDocumentAnalysis

O código de exemplo a seguir mostra como usar StartDocumentAnalysis.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

"Starts the asynchronous analysis of an input document for relationships  
"between detected items such as key-value pairs, tables, and selection  
elements."

"Create ABAP objects for feature type."

"Add TABLES to return information about the tables."

"Add FORMS to return detected form data."

"To perform both types of analysis, add TABLES and FORMS to FeatureTypes."

```
DATA(lt_featuretypes) = VALUE /aws1/cl_texfeaturetypes_w=>tt_featuretypes(
  ( NEW /aws1/cl_texfeaturetypes_w( iv_value = 'FORMS' ) )
  ( NEW /aws1/cl_texfeaturetypes_w( iv_value = 'TABLES' ) ) ).
```

"Create an ABAP object for the Amazon S3 object."

```
DATA(lo_s3object) = NEW /aws1/cl_texs3object( iv_bucket = iv_s3bucket
  iv_name = iv_s3object ).
```

"Create an ABAP object for the document."

```
DATA(lo_documentlocation) = NEW /aws1/cl_texdocumentlocation( io_s3object =
lo_s3object ).
```

"Start async document analysis."

TRY.

```
oo_result = lo_tex->startdocumentanalysis( "oo_result is returned for
testing purposes."
```

```
io_documentlocation = lo_documentlocation
```

```
it_featuretypes = lt_featuretypes ).
```

```
DATA(lv_jobid) = oo_result->get_jobid( ).
```

```
MESSAGE 'Document analysis started.' TYPE 'I'.
```

```
CATCH /aws1/cx_texaccessdeniedex.
```

```
MESSAGE 'You do not have permission to perform this action.' TYPE 'E'.
```

```
CATCH /aws1/cx_texbaddocumentex.
```

```
MESSAGE 'Amazon Textract is not able to read the document.' TYPE 'E'.
```

```
CATCH /aws1/cx_texdocumenttoolargeex.
```

```
MESSAGE 'The document is too large.' TYPE 'E'.
```

```
CATCH /aws1/cx_texidempotentprmmis00.
```

```
MESSAGE 'Idempotent parameter mismatch exception.' TYPE 'E'.
```

```
CATCH /aws1/cx_texinternalservererr.
```

```
MESSAGE 'Internal server error.' TYPE 'E'.
```

```
CATCH /aws1/cx_texinvalidkmskeyex.
```

```
MESSAGE 'AWS KMS key is not valid.' TYPE 'E'.
```

```
CATCH /aws1/cx_texinvalidparameterex.
```

```
MESSAGE 'Request has non-valid parameters.' TYPE 'E'.
```

```

CATCH /aws1/cx_texinvalids3objectex.
  MESSAGE 'Amazon S3 object is not valid.' TYPE 'E'.
CATCH /aws1/cx_texlimitexceeddex.
  MESSAGE 'An Amazon Textract service limit was exceeded.' TYPE 'E'.
CATCH /aws1/cx_texprovthruputexcdex.
  MESSAGE 'Provisioned throughput exceeded limit.' TYPE 'E'.
CATCH /aws1/cx_texthrottlingex.
  MESSAGE 'The request processing exceeded the limit.' TYPE 'E'.
CATCH /aws1/cx_texunsupporteddocex.
  MESSAGE 'The document is not supported.' TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [StartDocumentAnalysis](#) referência da API AWS SDK for SAP ABAP.

## StartDocumentTextDetection

O código de exemplo a seguir mostra como usar StartDocumentTextDetection.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

"Starts the asynchronous detection of text in a document."
"Amazon Textract can detect lines of text and the words that make up a line of
text."

"Create an ABAP object for the Amazon S3 object."
DATA(lo_s3object) = NEW /aws1/cl_texs3object( iv_bucket = iv_s3bucket
  iv_name   = iv_s3object ).
"Create an ABAP object for the document."
DATA(lo_documentlocation) = NEW /aws1/cl_texdocumentlocation( io_s3object =
lo_s3object ).
"Start document analysis."
TRY.

```

```
oo_result = lo_tex->startdocumenttextdetection( io_documentlocation =
lo_documentlocation ).
DATA(lv_jobid) = oo_result->get_jobid( ).           "oo_result is returned
for testing purposes."
MESSAGE 'Document analysis started.' TYPE 'I'.
CATCH /aws1/cx_texaccessdeniedex.
MESSAGE 'You do not have permission to perform this action.' TYPE 'E'.
CATCH /aws1/cx_texbaddocumentex.
MESSAGE 'Amazon Textract is not able to read the document.' TYPE 'E'.
CATCH /aws1/cx_texdocumenttoolargeex.
MESSAGE 'The document is too large.' TYPE 'E'.
CATCH /aws1/cx_texidempotentprmmis00.
MESSAGE 'Idempotent parameter mismatch exception.' TYPE 'E'.
CATCH /aws1/cx_texinternalservererr.
MESSAGE 'Internal server error.' TYPE 'E'.
CATCH /aws1/cx_texinvalidkmskeyex.
MESSAGE 'AWS KMS key is not valid.' TYPE 'E'.
CATCH /aws1/cx_texinvalidparameterex.
MESSAGE 'Request has non-valid parameters.' TYPE 'E'.
CATCH /aws1/cx_texinvalids3objectex.
MESSAGE 'Amazon S3 object is not valid.' TYPE 'E'.
CATCH /aws1/cx_texlimitexceeddex.
MESSAGE 'An Amazon Textract service limit was exceeded.' TYPE 'E'.
CATCH /aws1/cx_texprovthruputexcdex.
MESSAGE 'Provisioned throughput exceeded limit.' TYPE 'E'.
CATCH /aws1/cx_texthrottlingex.
MESSAGE 'The request processing exceeded the limit.' TYPE 'E'.
CATCH /aws1/cx_texunsupporteddocex.
MESSAGE 'The document is not supported.' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartDocumentTextDetection](#) referência da API AWS SDK for SAP ABAP.

## Cenários

### Conceitos básicos de usar análise de documentos

O exemplo de código a seguir mostra como:

- Iniciar análise assíncrona.

- Obter análise de documento.

## SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Create ABAP objects for feature type."
"Add TABLES to return information about the tables."
"Add FORMS to return detected form data."
"To perform both types of analysis, add TABLES and FORMS to FeatureTypes."

DATA(lt_featuretypes) = VALUE /aws1/cl_texfeaturetypes_w=>tt_featuretypes(
  ( NEW /aws1/cl_texfeaturetypes_w( iv_value = 'FORMS' ) )
  ( NEW /aws1/cl_texfeaturetypes_w( iv_value = 'TABLES' ) ) ).

"Create an ABAP object for the Amazon Simple Storage Service (Amazon S3)
object."
DATA(lo_s3object) = NEW /aws1/cl_texs3object( iv_bucket = iv_s3bucket
  iv_name   = iv_s3object ).

"Create an ABAP object for the document."
DATA(lo_documentlocation) = NEW /aws1/cl_texdocumentlocation( io_s3object =
lo_s3object ).

"Start document analysis."
TRY.
  DATA(lo_start_result) = lo_tex->startdocumentanalysis(
    io_documentlocation   = lo_documentlocation
    it_featuretypes       = lt_featuretypes ).
  MESSAGE 'Document analysis started.' TYPE 'I'.
CATCH /aws1/cx_texaccessdeniedex.
  MESSAGE 'You do not have permission to perform this action.' TYPE 'E'.
CATCH /aws1/cx_texbaddocumentex.
  MESSAGE 'Amazon Textract is not able to read the document.' TYPE 'E'.
CATCH /aws1/cx_texdocumenttoolargeex.
  MESSAGE 'The document is too large.' TYPE 'E'.
CATCH /aws1/cx_texidempotentprmmis00.
```

```

    MESSAGE 'Idempotent parameter mismatch exception.' TYPE 'E'.
  CATCH /aws1/cx_texinternalservererr.
    MESSAGE 'Internal server error.' TYPE 'E'.
  CATCH /aws1/cx_texinvalidkmskeyex.
    MESSAGE 'AWS KMS key is not valid.' TYPE 'E'.
  CATCH /aws1/cx_texinvalidparameterex.
    MESSAGE 'Request has non-valid parameters.' TYPE 'E'.
  CATCH /aws1/cx_texinvalids3objectex.
    MESSAGE 'Amazon S3 object is not valid.' TYPE 'E'.
  CATCH /aws1/cx_texlimitexceededx.
    MESSAGE 'An Amazon Textract service limit was exceeded.' TYPE 'E'.
  CATCH /aws1/cx_texprovthruputexcdex.
    MESSAGE 'Provisioned throughput exceeded limit.' TYPE 'E'.
  CATCH /aws1/cx_texthrottlingex.
    MESSAGE 'The request processing exceeded the limit.' TYPE 'E'.
  CATCH /aws1/cx_texunsupporteddocex.
    MESSAGE 'The document is not supported.' TYPE 'E'.
ENDTRY.

"Get job ID from the output."
DATA(lv_jobid) = lo_start_result->get_jobid( ).

"Wait for job to complete."
oo_result = lo_tex->getdocumentanalysis( iv_jobid = lv_jobid ).      " oo_result
is returned for testing purposes. "
WHILE oo_result->get_jobstatus( ) <> 'SUCCEEDED'.
  IF sy-index = 10.
    EXIT.                  "Maximum 300 seconds."
  ENDIF.
  WAIT UP TO 30 SECONDS.
  oo_result = lo_tex->getdocumentanalysis( iv_jobid = lv_jobid ).
ENDWHILE.

DATA(lt_blocks) = oo_result->get_blocks( ).
LOOP AT lt_blocks INTO DATA(lo_block).
  IF lo_block->get_text( ) = 'INGREDIENTS: POWDERED SUGAR* (CANE SUGAR, '.
    MESSAGE 'Found text in the doc: ' && lo_block->get_text( ) TYPE 'I'.
  ENDIF.
ENDLOOP.

```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.

- [GetDocumentAnalysis](#)
- [StartDocumentAnalysis](#)

## Exemplos do Amazon Transcribe usando SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Transcribe.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Tópicos

- [Ações](#)

## Ações

### CreateVocabulary

O código de exemplo a seguir mostra como usar CreateVocabulary.

SDK para SAP ABAP

#### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
  IF it_phrases IS NOT INITIAL.  
    oo_result = lo_tnb->createvocabulary(  
      iv_vocabularyname = iv_vocabulary_name  
      iv_languagecode = iv_language_code
```

```
        it_phrases = it_phrases ).
ELSEIF iv_vocab_file_uri IS NOT INITIAL.
    oo_result = lo_tnb->createvocabulary(
        iv_vocabularyname = iv_vocabulary_name
        iv_languagecode = iv_language_code
        iv_vocabularyfileuri = iv_vocab_file_uri ).
ENDIF.
MESSAGE 'Custom vocabulary created.' TYPE 'I'.
CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex TYPE 'I'.
    RAISE EXCEPTION lo_bad_request_ex.
CATCH /aws1/cx_tnblimitexceededex INTO DATA(lo_limit_ex).
    MESSAGE lo_limit_ex TYPE 'I'.
    RAISE EXCEPTION lo_limit_ex.
CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).
    MESSAGE lo_internal_ex TYPE 'I'.
    RAISE EXCEPTION lo_internal_ex.
CATCH /aws1/cx_tnbconflictexception INTO DATA(lo_conflict_ex).
    MESSAGE lo_conflict_ex TYPE 'I'.
    RAISE EXCEPTION lo_conflict_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [CreateVocabulary](#) referência da API AWS SDK for SAP ABAP.

## DeleteTranscriptionJob

O código de exemplo a seguir mostra como usar DeleteTranscriptionJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.
    lo_tnb->deletetranscriptionjob( iv_job_name ).
    MESSAGE 'Transcription job deleted.' TYPE 'I'.
```

```

CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex TYPE 'I'.
  RAISE EXCEPTION lo_bad_request_ex.
CATCH /aws1/cx_tnblimitexceededex INTO DATA(lo_limit_ex).
  MESSAGE lo_limit_ex TYPE 'I'.
  RAISE EXCEPTION lo_limit_ex.
CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).
  MESSAGE lo_internal_ex TYPE 'I'.
  RAISE EXCEPTION lo_internal_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteTranscriptionJob](#) referência da API AWS SDK for SAP ABAP.

## DeleteVocabulary

O código de exemplo a seguir mostra como usar DeleteVocabulary.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  lo_tnb->deletevocabulary( iv_vocabulary_name ).
  MESSAGE 'Vocabulary deleted.' TYPE 'I'.
CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex TYPE 'I'.
CATCH /aws1/cx_tnblimitexceededex INTO DATA(lo_limit_ex).
  MESSAGE lo_limit_ex TYPE 'I'.
  RAISE EXCEPTION lo_limit_ex.
CATCH /aws1/cx_tnbnotfoundexception INTO DATA(lo_not_found_ex).
  MESSAGE lo_not_found_ex TYPE 'I'.
CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).
  MESSAGE lo_internal_ex TYPE 'I'.
  RAISE EXCEPTION lo_internal_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [DeleteVocabulary](#) referência da API AWS SDK for SAP ABAP.

## GetTranscriptionJob

O código de exemplo a seguir mostra como usar GetTranscriptionJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_tnb->gettranscriptionjob( iv_job_name ).  
    DATA(lo_job) = oo_result->get_transcriptionjob( ).  
    MESSAGE 'Retrieved transcription job details.' TYPE 'I'.  
    CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).  
    MESSAGE lo_bad_request_ex TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request_ex.  
    CATCH /aws1/cx_tnbnotfoundexception INTO DATA(lo_not_found_ex).  
    MESSAGE lo_not_found_ex TYPE 'I'.  
    RAISE EXCEPTION lo_not_found_ex.  
    CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).  
    MESSAGE lo_internal_ex TYPE 'I'.  
    RAISE EXCEPTION lo_internal_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetTranscriptionJob](#) referência da API AWS SDK for SAP ABAP.

## GetVocabulary

O código de exemplo a seguir mostra como usar GetVocabulary.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    oo_result = lo_tnb->getvocabulary( iv_vocabulary_name ).  
    MESSAGE 'Retrieved vocabulary details.' TYPE 'I'.  
    CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).  
    MESSAGE lo_bad_request_ex TYPE 'I'.  
    RAISE EXCEPTION lo_bad_request_ex.  
    CATCH /aws1/cx_tnbnotfoundexception INTO DATA(lo_not_found_ex).  
    MESSAGE lo_not_found_ex TYPE 'I'.  
    RAISE EXCEPTION lo_not_found_ex.  
    CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).  
    MESSAGE lo_internal_ex TYPE 'I'.  
    RAISE EXCEPTION lo_internal_ex.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [GetVocabulary](#) referência da API AWS SDK for SAP ABAP.

**ListTranscriptionJobs**

O código de exemplo a seguir mostra como usar ListTranscriptionJobs.

## SDK para SAP ABAP

**Note**

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
TRY.  
    IF iv_job_filter IS NOT INITIAL.
```

```

        oo_result = lo_tnb->listtranscriptionjobs( iv_jobnamecontains =
iv_job_filter ).
    ELSE.
        oo_result = lo_tnb->listtranscriptionjobs( ).
    ENDIF.
    MESSAGE 'Retrieved transcription jobs list.' TYPE 'I'.
    CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex TYPE 'I'.
    RAISE EXCEPTION lo_bad_request_ex.
    CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).
    MESSAGE lo_internal_ex TYPE 'I'.
    RAISE EXCEPTION lo_internal_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListTranscriptionJobs](#) referência da API AWS SDK for SAP ABAP.

## ListVocabularies

O código de exemplo a seguir mostra como usar ListVocabularies.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
    IF iv_vocab_filter IS NOT INITIAL.
        oo_result = lo_tnb->listvocabularies( iv_namecontains = iv_vocab_filter ).
    ELSE.
        oo_result = lo_tnb->listvocabularies( ).
    ENDIF.
    MESSAGE 'Retrieved vocabularies list.' TYPE 'I'.
    CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex TYPE 'I'.
    RAISE EXCEPTION lo_bad_request_ex.
    CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).

```

```

MESSAGE lo_internal_ex TYPE 'I'.
RAISE EXCEPTION lo_internal_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListVocabularies](#) referência da API AWS SDK for SAP ABAP.

## StartTranscriptionJob

O código de exemplo a seguir mostra como usar StartTranscriptionJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  DATA(lo_media) = NEW /aws1/cl_tnbmedia( iv_mediafileuri = iv_media_uri ).
  DATA(lo_settings) = NEW /aws1/cl_tnbsettings( ).
  IF iv_vocabulary_name IS NOT INITIAL.
    lo_settings = NEW /aws1/cl_tnbsettings( iv_vocabularyname =
iv_vocabulary_name ).
  ENDIF.

  oo_result = lo_tnb->starttranscriptionjob(
    iv_transcriptionjobname = iv_job_name
    io_media = lo_media
    iv_mediaformat = iv_media_format
    iv_languagecode = iv_language_code
    io_settings = lo_settings ).

  MESSAGE 'Transcription job started.' TYPE 'I'.
  CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
  MESSAGE lo_bad_request_ex TYPE 'I'.
  RAISE EXCEPTION lo_bad_request_ex.
  CATCH /aws1/cx_tnblimitexceededex INTO DATA(lo_limit_ex).
  MESSAGE lo_limit_ex TYPE 'I'.

```

```

    RAISE EXCEPTION lo_limit_ex.
  CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).
    MESSAGE lo_internal_ex TYPE 'I'.
    RAISE EXCEPTION lo_internal_ex.
  CATCH /aws1/cx_tnbconflictexception INTO DATA(lo_conflict_ex).
    MESSAGE lo_conflict_ex TYPE 'I'.
    RAISE EXCEPTION lo_conflict_ex.
ENDTRY.

```

- Para obter detalhes da API, consulte a [StartTranscriptionJob](#) referência da API AWS SDK for SAP ABAP.

## UpdateVocabulary

O código de exemplo a seguir mostra como usar UpdateVocabulary.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

TRY.
  IF it_phrases IS NOT INITIAL.
    oo_result = lo_tnb->updatevocabulary(
      iv_vocabularyname = iv_vocabulary_name
      iv_languagecode = iv_language_code
      it_phrases = it_phrases ).
  ELSEIF iv_vocab_file_uri IS NOT INITIAL.
    oo_result = lo_tnb->updatevocabulary(
      iv_vocabularyname = iv_vocabulary_name
      iv_languagecode = iv_language_code
      iv_vocabularyfileuri = iv_vocab_file_uri ).
  ENDIF.
  MESSAGE 'Vocabulary updated.' TYPE 'I'.
  CATCH /aws1/cx_tnbbadrequestex INTO DATA(lo_bad_request_ex).
    MESSAGE lo_bad_request_ex TYPE 'I'.
  CATCH /aws1/cx_tnblimitexceededx INTO DATA(lo_limit_ex).

```

```
MESSAGE lo_limit_ex TYPE 'I'.
RAISE EXCEPTION lo_limit_ex.
CATCH /aws1/cx_tnbnotfoundexception INTO DATA(lo_not_found_ex).
MESSAGE lo_not_found_ex TYPE 'I'.
CATCH /aws1/cx_tnbinternalfailureex INTO DATA(lo_internal_ex).
MESSAGE lo_internal_ex TYPE 'I'.
RAISE EXCEPTION lo_internal_ex.
CATCH /aws1/cx_tnbconflictexception INTO DATA(lo_conflict_ex).
MESSAGE lo_conflict_ex TYPE 'I'.
RAISE EXCEPTION lo_conflict_ex.
ENDTRY.
```

- Para obter detalhes da API, consulte a [UpdateVocabulary](#) referência da API AWS SDK for SAP ABAP.

## Exemplos do Amazon Translate usando o SDK para SAP ABAP

Os exemplos de código a seguir mostram como realizar ações e implementar cenários comuns usando o AWS SDK para SAP ABAP com o Amazon Translate.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Cada exemplo inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

### Tópicos


- [Ações](#)
- [Cenários](#)

## Ações

### **DescribeTextTranslationJob**

O código de exemplo a seguir mostra como usar `DescribeTextTranslationJob`.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).


```
"Gets the properties associated with an asynchronous batch translation job."  
"Includes properties such as name, ID, status, source and target languages, and  
input/output Amazon Simple Storage Service (Amazon S3) buckets."  
TRY.  
    oo_result = lo_xl8->describetexttranslationjob(      "oo_result is returned  
for testing purposes."  
    iv_jobid      = iv_jobid ).  
    MESSAGE 'Job description retrieved.' TYPE 'I'.  
CATCH /aws1/cx_xl8internalserverex.  
    MESSAGE 'An internal server error occurred. Retry your request.' TYPE 'E'.  
CATCH /aws1/cx_xl8resourcenotfoundex.  
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.  
CATCH /aws1/cx_xl8toomanyrequestsex.  
    MESSAGE 'You have made too many requests within a short period of time.'  
TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [DescribeTextTranslationJob](#) referência da API AWS SDK for SAP ABAP.

## ListTextTranslationJobs

O código de exemplo a seguir mostra como usar ListTextTranslationJobs.

## SDK para SAP ABAP

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

"Gets a list of the batch translation jobs that you have submitted."

DATA lo_filter TYPE REF TO /aws1/cl_xl8textxl8translationjobfilt.

"Create an ABAP object for filtering using jobname."
lo_filter = NEW #( iv_jobname = iv_jobname ).

TRY.
    oo_result = lo_xl8->listtexttranslationjobs(      "oo_result is returned for
testing purposes."
        io_filter      = lo_filter ).
    MESSAGE 'Jobs retrieved.' TYPE 'I'.
    CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred. Retry your request.' TYPE 'E'.
    CATCH /aws1/cx_xl8invalidfilterex.
        MESSAGE 'The filter specified for the operation is not valid. Specify a
different filter.' TYPE 'E'.
    CATCH /aws1/cx_xl8invalidrequestex.
        MESSAGE 'The request that you made is not valid.' TYPE 'E'.
    CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
ENDTRY.

```

- Para obter detalhes da API, consulte a [ListTextTranslationJobs](#) referência da API AWS SDK for SAP ABAP.

## StartTextTranslationJob

O código de exemplo a seguir mostra como usar StartTextTranslationJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```

"Starts an asynchronous batch translation job."

```

"Use batch translation jobs to translate large volumes of text across multiple documents at once."

```
DATA lo_inputdataconfig TYPE REF TO /aws1/cl_xl8inputdataconfig.
DATA lo_outputdataconfig TYPE REF TO /aws1/cl_xl8outputdataconfig.
DATA lt_targetlanguagecodes TYPE /aws1/
cl_xl8tgtlanguagecodes00=>tt_targetlanguagecodestrlist.
DATA lo_targetlanguagecodes TYPE REF TO /aws1/cl_xl8tgtlanguagecodes00.

"Create an ABAP object for the input data config."
lo_inputdataconfig = NEW #( iv_s3uri = iv_input_data_s3uri
                           iv_contenttype = iv_input_data_contenttype ).

"Create an ABAP object for the output data config."
lo_outputdataconfig = NEW #( iv_s3uri = iv_output_data_s3uri ).

"Create an internal table for target languages."
lo_targetlanguagecodes = NEW #( iv_value = iv_targetlanguagecode ).
INSERT lo_targetlanguagecodes INTO TABLE lt_targetlanguagecodes.

TRY.
    oo_result = lo_xl8->starttexttranslationjob(      "oo_result is returned for
testing purposes."
    io_inputdataconfig = lo_inputdataconfig
    io_outputdataconfig = lo_outputdataconfig
    it_targetlanguagecodes = lt_targetlanguagecodes
    iv_dataaccessrolelearn = iv_dataaccessrolelearn
    iv_jobname = iv_jobname
    iv_sourcelanguagecode = iv_sourcelanguagecode ).
    MESSAGE 'Translation job started.' TYPE 'I'.
CATCH /aws1/cx_xl8internalserverex.
    MESSAGE 'An internal server error occurred. Retry your request.' TYPE 'E'.
CATCH /aws1/cx_xl8invparamvalueex.
    MESSAGE 'The value of the parameter is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8invalidrequestex.
    MESSAGE 'The request that you made is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8resourcenotfoundex.
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
    MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
CATCH /aws1/cx_xl8unsuppdedlanguage00.
    MESSAGE 'Amazon Translate does not support translation from the language of
the source text into the requested target language.' TYPE 'E'.
```

```
ENDTRY.
```

- Para obter detalhes da API, consulte a [StartTextTranslationJob](#) referência da API AWS SDK for SAP ABAP.

## StopTextTranslationJob

O código de exemplo a seguir mostra como usar StopTextTranslationJob.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Stops an asynchronous batch translation job that is in progress."
```

```
TRY.  
    oo_result = lo_xl8->stoptexttranslationjob(      "oo_result is returned for  
testing purposes."  
        iv_jobid      = iv_jobid ).  
    MESSAGE 'Translation job stopped.' TYPE 'I'.  
    CATCH /aws1/cx_xl8internalserverex.  
        MESSAGE 'An internal server error occurred.' TYPE 'E'.  
    CATCH /aws1/cx_xl8resourcenotfoundex.  
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.  
    CATCH /aws1/cx_xl8toomanyrequestsex.  
        MESSAGE 'You have made too many requests within a short period of time.'  
TYPE 'E'.  
ENDTRY.
```

- Para obter detalhes da API, consulte a [StopTextTranslationJob](#) referência da API AWS SDK for SAP ABAP.

## TranslateText

O código de exemplo a seguir mostra como usar TranslateText.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
"Translates input text from the source language to the target language."
TRY.
    oo_result = lo_xl8->translatetext(      "oo_result is returned for testing
purposes."
        iv_text          = iv_text
        iv_sourcelanguagecode = iv_sourcelanguagecode
        iv_targetlanguagecode = iv_targetlanguagecode ).
    MESSAGE 'Translation completed.' TYPE 'I'.
    CATCH /aws1/cx_xl8detectedlanguage00.
        MESSAGE 'The confidence that Amazon Comprehend accurately detected the
source language is low.' TYPE 'E'.
    CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred.' TYPE 'E'.
    CATCH /aws1/cx_xl8invalidrequestex.
        MESSAGE 'The request that you made is not valid.' TYPE 'E'.
    CATCH /aws1/cx_xl8resourcenotfoundex.
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
    CATCH /aws1/cx_xl8serviceunavailex.
        MESSAGE 'The Amazon Translate service is temporarily unavailable.' TYPE 'E'.
    CATCH /aws1/cx_xl8textsizegmtexcdex.
        MESSAGE 'The size of the text you submitted exceeds the size limit. ' TYPE
'E'.
    CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
    CATCH /aws1/cx_xl8unsuppdedlanguage00.
        MESSAGE 'Amazon Translate does not support translation from the language of
the source text into the requested target language. ' TYPE 'E'.
ENDTRY.
```

- Para obter detalhes da API, consulte a [TranslateText](#) referência da API AWS SDK for SAP ABAP.

## Cenários

Conceitos básicos de usar trabalhos de tradução

O exemplo de código a seguir mostra como:

- Iniciar um trabalho assíncrono de tradução em lote.
- Aguardar a conclusão do trabalho assíncrono.
- Descrever o trabalho assíncrono.

SDK para SAP ABAP

### Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

```
DATA lo_inputdataconfig TYPE REF TO /aws1/cl_xl8inputdataconfig.
DATA lo_outputdataconfig TYPE REF TO /aws1/cl_xl8outputdataconfig.
DATA lt_targetlanguagecodes TYPE /aws1/
cl_xl8tgtlanguagecodes00=>tt_targetlanguagecodestrlist.
DATA lo_targetlanguagecodes TYPE REF TO /aws1/cl_xl8tgtlanguagecodes00.

"Create an ABAP object for the input data config."
lo_inputdataconfig = NEW #( iv_s3uri = iv_input_data_s3uri
                           iv_contenttype = iv_input_data_contenttype ).

"Create an ABAP object for the output data config."
lo_outputdataconfig = NEW #( iv_s3uri = iv_output_data_s3uri ).

"Create an internal table for target languages."
lo_targetlanguagecodes = NEW #( iv_value = iv_targetlanguagecode ).
INSERT lo_targetlanguagecodes INTO TABLE lt_targetlanguagecodes.

TRY.
```

```

DATA(lo_translationjob_result) = lo_xl8->starttexttranslationjob(
  io_inputdataconfig = lo_inputdataconfig
  io_outputdataconfig = lo_outputdataconfig
  it_targetlanguagecodes = lt_targetlanguagecodes
  iv_dataaccessrolelearn = iv_dataaccessrolelearn
  iv_jobname = iv_jobname
  iv_sourcelanguagecode = iv_sourcelanguagecode ).
MESSAGE 'Translation job started.' TYPE 'I'.
CATCH /aws1/cx_xl8internalserverex.
  MESSAGE 'An internal server error occurred. Retry your request.' TYPE 'E'.
CATCH /aws1/cx_xl8invparamvalueex.
  MESSAGE 'The value of the parameter is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8invalidrequestex.
  MESSAGE 'The request that you made is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8resourcenotfoundex.
  MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
  MESSAGE 'You have made too many requests within a short period of time. '
TYPE 'E'.
CATCH /aws1/cx_xl8unsuppdedlanguage00.
  MESSAGE 'Amazon Translate does not support translation from the language of
the source text into the requested target language.' TYPE 'E'.
ENDTRY.

"Get the job ID."
DATA(lv_jobid) = lo_translationjob_result->get_jobid( ).

"Wait for translate job to complete."
DATA(lo_des_translation_result) = lo_xl8->describetexttranslationjob( iv_jobid =
lv_jobid ).
WHILE lo_des_translation_result->get_textxlationjobproperties( )-
>get_jobstatus( ) <> 'COMPLETED'.
  IF sy-index = 30.
    EXIT.          "Maximum 900 seconds."
  ENDIF.
  WAIT UP TO 30 SECONDS.
  lo_des_translation_result = lo_xl8->describetexttranslationjob( iv_jobid =
lv_jobid ).
ENDWHILE.

TRY.
  oo_result = lo_xl8->describetexttranslationjob(      "oo_result is returned
for testing purposes."
  iv_jobid      = lv_jobid ).

```

```
    MESSAGE 'Job description retrieved.' TYPE 'I'.
  CATCH /aws1/cx_xl8internalserverex.
    MESSAGE 'An internal server error occurred. Retry your request.' TYPE 'E'.
  CATCH /aws1/cx_xl8resourcenotfoundex.
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
  CATCH /aws1/cx_xl8toomanyrequestsex.
    MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
  ENDTRY.
```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para SAP ABAP.
  - [DescribeTextTranslationJob](#)
  - [StartTextTranslationJob](#)

# Usando o AWS SDK para SAP ABAP Knowledge MCP Server

O AWS SDK para SAP ABAP Knowledge MCP Server é um recurso de conhecimento especializado que permite que assistentes de codificação de IA gerem código ABAP preciso para integrações. Ao conectar seu IDE habilitado para IA a esse servidor, você pode descrever o que deseja criar em linguagem natural e receber um código ABAP sintaticamente correto que invoca adequadamente o SDK para SAP ABAP. AWS

O servidor MCP usa o [Model Context Protocol \(MCP\)](#), um padrão aberto para conectar assistentes de IA a fontes externas de conhecimento. O servidor é atualizado diariamente em sincronia com as versões do AWS SDK for SAP ABAP, para que seu IDE sempre tenha acesso às informações atuais e precisas do SDK.

O AWS SDK para SAP ABAP Knowledge MCP Server está disponível sem custo adicional. Você paga apenas pelos AWS recursos e serviços que consome em seus aplicativos SAP com o AWS SDK para SAP ABAP.

## Tópicos

- [Benefícios](#)
- [Configurando o servidor MCP](#)
- [O que o servidor MCP pode fazer](#)
- [Considerações importantes](#)

## Benefícios

O AWS SDK para SAP ABAP Knowledge MCP Server oferece os seguintes benefícios para desenvolvedores de ABAP.

- Geração precisa de código — Seu assistente de codificação de IA obtém um conhecimento profundo dos padrões específicos do AWS SDK para SAP ABAP, assinaturas de métodos, tipos de dados e tratamento de exceções. Isso elimina erros comuns e reduz significativamente os erros de compilação e o tempo de depuração.

- **Descoberta instantânea do SDK** — O servidor MCP expõe a base de conhecimento completa do AWS SDK para SAP ABAP, abrangendo mais de 200 serviços. AWS Você pode descobrir serviços, operações e tipos de dados disponíveis sem pesquisar manualmente a documentação.
- **Sempre atualizada** — A base de conhecimento é atualizada diariamente em sincronia com os lançamentos do SDK. Novos serviços e assinaturas de métodos atualizadas estão imediatamente disponíveis para seu IDE.
- **Instalação zero** — A configuração requer apenas colar uma URL na configuração MCP do seu IDE. Nenhuma instalação de software local, nenhum gerenciamento de dependências e nenhum processo de aprovação de TI é necessário.

## Configurando o servidor MCP

Para conectar seu IDE ao AWS SDK do SAP ABAP Knowledge MCP Server, adicione a seguinte configuração às configurações MCP do seu IDE. O formato da configuração e a localização do arquivo podem variar de acordo com o IDE; consulte a documentação do MCP do seu IDE para obter detalhes.

O exemplo a seguir mostra a configuração do Kiro. Adicione isso ao arquivo de configuração MCP do seu espaço de trabalho `.kiro/settings/mcp.json` ou ao seu arquivo de configuração em nível de usuário: `~/.kiro/settings/mcp.json`

```
{
  "mcpServers": {
    "abap-sdk-knowledge": {
      "url": "https://sdk-for-sap-abap-knowledge-mcp.global.api.aws/mcp",
      "disabled": false,
      "autoApprove": ["*"]
    }
  }
}
```

A `autoApprove` configuração permite que seu IDE use as ferramentas do servidor MCP sem solicitar aprovação em cada solicitação. Isso é seguro porque o servidor MCP é somente para leitura e não modifica nenhum dado nem executa código.

Depois de salvar a configuração, seu IDE se conectará automaticamente ao servidor MCP. Nenhuma reinicialização é necessária na maioria IDEs. Você pode verificar a conexão fazendo uma pergunta

ao seu assistente de IA sobre o AWS SDK para SAP ABAP, como “Quais AWS serviços estão disponíveis no AWS SDK para SAP ABAP?”

Se o seu IDE não se conectar ao servidor MCP, verifique se o IDE suporta o Protocolo de Contexto do Modelo e se a sintaxe do arquivo de configuração corresponde aos requisitos do seu IDE.

#### Note

O endpoint do servidor MCP usa HTTPS e não exige autenticação. Seu IDE se comunica com o servidor somente quando você faz perguntas ao assistente de IA relacionadas ao AWS SDK para uso do SAP ABAP. Sua lógica comercial proprietária e o código do aplicativo SAP permanecem em seu ambiente de desenvolvimento.

## O que o servidor MCP pode fazer

Uma vez conectado, seu assistente de codificação de IA pode usar os seguintes recursos fornecidos pelo servidor MCP.

- **Descoberta de serviços** — liste e pesquise todos os AWS serviços disponíveis no AWS SDK para SAP ABAP, incluindo metadados de serviço e nomes de classes de fábrica.
- **Detalhes da operação** — Recupere assinaturas do método ABAP, parâmetros de entrada, tipos de retorno e tipos de exceção para qualquer operação do SDK.
- **Definições de tipo de dados** — pesquise definições de estrutura ABAP para tipos de dados do SDK, incluindo nomes e tipos de campo.
- **Exemplos de código** — Recupere exemplos de código ABAP funcionais para operações específicas, abrangendo casos de uso comuns para cada serviço.
- **Criação de sessões e clientes** — Gere código ABAP padronizado para criar sessões de SDK, inicializar clientes de serviço e chamar operações com tratamento correto de exceções.

Por exemplo, você pode solicitar ao seu assistente de IA solicitações como as seguintes.

- “Escreva um código ABAP para ler uma mensagem de uma fila SQS.”
- “Gere código ABAP para invocar um modelo Amazon Bedrock para geração de texto.”
- “Mostre-me como fazer upload de um arquivo para o Amazon S3 a partir do ABAP.”

O assistente de IA usa o conhecimento do servidor MCP para gerar um código ABAP sintaticamente correto que segue as convenções do AWS SDK for SAP ABAP, incluindo a criação adequada de sessões, inicialização do cliente e padrões de tratamento de exceções.

## Considerações importantes

Lembre-se do seguinte ao usar o AWS SDK para o SAP ABAP Knowledge MCP Server.

- O servidor MCP fornece acesso somente de leitura ao conhecimento do SDK. Ele não executa código ABAP nem se conecta aos sistemas SAP.
- O servidor MCP cobre somente o AWS SDK para SAP ABAP. Ele não fornece assistência geral de programação ABAP não relacionada ao AWS SDK para uso do SAP ABAP.
- O servidor MCP não oferece suporte a bibliotecas ABAP personalizadas ou de terceiros.
- O servidor MCP fornece documentação com base na versão mais recente do SDK, que pode não corresponder à versão instalada em seu sistema SAP.
- Exemplos de código gerado são pontos de partida. Revise e teste todo o código gerado antes de usá-lo na produção.
- A qualidade e o formato do código gerado podem variar dependendo dos recursos do assistente de IA do seu IDE.

Separadamente do AWS SDK para SAP ABAP, o Knowledge MCP Server descrito nesta página, AWS também oferece o [AWS Knowledge MCP Server](#), que fornece informações abrangentes sobre AWS serviços, orientação arquitetônica e solução de problemas em todos os serviços. AWS Os dois servidores se complementam e são especialmente eficazes quando usados juntos.

# Segurança em AWS SDK para SAP ABAP

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de data centers e arquiteturas de rede criados para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que é executada Serviços da AWS no Nuvem AWS. AWS também fornece serviços que você pode usar com segurança. Auditores terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de [AWS](#) de . Para saber mais sobre os programas de conformidade aplicáveis AWS SDK para SAP ABAP, consulte [Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS em Escopo por Programa](#) .
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS service (Serviço da AWS) que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta seção abrange os seguintes tópicos.

## Tópicos

- [Autenticação do sistema SAP em AWS](#)
- [Práticas recomendadas de segurança do IAM](#)
- [Autorizações da SAP](#)
- [Operações seguras](#)
- [Usando a autenticação de chave de acesso secreta com criptografia SSF](#)
- [Usar certificados com o IAM Roles Anywhere](#)
- [Usando o perfil de origem para acesso entre contas](#)
- [Usando o SAP Credential Store](#)

# Autenticação do sistema SAP em AWS

Antes que um sistema SAP possa fazer chamadas AWS em nome dos usuários do SAP, o sistema SAP deve se autenticar AWS em. AWS SDK para SAP ABAP suporta os três métodos básicos de autenticação a seguir, selecionados nas configurações do perfil do SDK emIMG.

AWS A edição SDK para SAP ABAP - BTP só pode ser autenticada com o [the section called “Autenticação da chave de acesso secreta”](#) método usando o SAP Credential Store.

Para cenários de acesso entre contas, o SDK para SAP ABAP também oferece suporte ao perfil de origem, o que permite encadear várias suposições de função do IAM em todas as contas usando qualquer um dos métodos básicos de autenticação. Para obter mais informações, consulte [the section called “Autenticação do perfil de origem para acesso entre contas”](#).

## Tópicos

- [Autenticação de metadados da instância do Amazon EC2](#)
- [Autenticação da chave de acesso secreta](#)
- [Autenticação baseada em certificado usando o IAM Roles Anywhere](#)
- [Autenticação do perfil de origem para acesso entre contas](#)
- [Próxima etapa](#)

## Autenticação de metadados da instância do Amazon EC2

Os sistemas SAP executados no Amazon EC2 podem adquirir credenciais de curta duração e rotativas automaticamente a partir dos metadados da instância do Amazon EC2. Para obter mais informações, consulte [Como usar credenciais para metadados de instância do Amazon EC2](#).

É altamente recomendável esse método de autenticação ao usar o SDK para SAP ABAP. Para habilitar, o administrador do Basis deve habilitar a comunicação HTTP de saída. Nenhuma outra configuração é necessária.

### Note

Esse método de autenticação se aplica somente se seus sistemas SAP estiverem em execução no Amazon EC2. Os sistemas SAP hospedados on-premises ou em outros ambientes de nuvem não podem ser autenticados usando esse método.

## Autenticação da chave de acesso secreta

Com esse método, você usa um ID de chave de acesso e uma chave de acesso secreta para autenticar seu sistema SAP na AWS. O sistema SAP faz login AWS usando um usuário do IAM. Para obter mais informações, consulte [Gerenciar chaves de acesso para usuários do IAM](#).

O administrador do Basis recebe um ID de chave de acesso e uma chave de acesso secreta do administrador do IAM da AWS. Seu sistema SAP deve ser configurado para armazenar o ID da chave de acesso e a chave de acesso secreta.

- Proteja, armazene e encaminhe (SSF)
  - Use a funcionalidade SSF para autenticar o AWS SDK para SAP ABAP. Para obter mais informações, consulte [Assinaturas digitais e criptografia](#).
  - Você também pode testar as funcionalidades envelope e develop de SSFs com o relatório SSF02. Para obter mais informações, consulte [Teste da instalação do SSF](#).
  - As etapas para configurar o SSF para SDK para SAP ABAP estão descritas na transação / AWS1/IMG. Vá para Pré-requisitos técnicos e selecione Configurações adicionais para sistemas on-premises. Para obter etapas detalhadas de configuração, consulte [Usando a autenticação por chave de acesso secreta com criptografia SSF](#).
- Armazenamento de credenciais SAP
  - Use o SAP Credential Store para autenticar o AWS SDK para a edição SAP ABAP - BTP. Para obter mais informações, consulte [O que é o SAP Credential Store?](#)
  - Consulte [Usando o SAP Credential Store](#) para ver as etapas de configuração.

## Autenticação baseada em certificado usando o IAM Roles Anywhere

Um certificado X.509 emitido pela sua autoridade de certificação (CA) pode ser usado para autenticação com o AWS Identity and Access Management Roles Anywhere. O certificado deve ser configurado em STRUST. A CA deve ser registrada com o IAM Roles Anywhere como uma âncora de confiança e um perfil deve ser criado para especificar os perfis e as políticas que o IAM Roles Anywhere assumiria. Para obter mais informações, consulte [Criação de uma âncora de confiança e um perfil no AWS Identity and Access Management Roles Anywhere](#).

Consulte etapas detalhadas sobre como usar o IAM Roles Anywhere com o SDK para SAP ABAP em [Using certificates with IAM Roles Anywhere](#).

**Note**

A revogação de certificado só é permitida por meio do uso de listas de revogação de certificados importadas. Consulte mais informações em [Revocation](#).

## Autenticação do perfil de origem para acesso entre contas

O perfil de origem é um recurso avançado que permite encadear várias suposições de função do IAM em todas AWS as contas. Com esse método, um perfil assume uma função, que então assume outra função e assim por diante, semelhante ao `source_profile` parâmetro na CLI AWS .

O perfil de origem funciona com qualquer um dos três métodos básicos de autenticação (metadados de instância, chave de acesso secreta ou baseado em certificado). O primeiro perfil na cadeia deve usar um desses métodos básicos, e os perfis subsequentes na cadeia usam as credenciais do perfil anterior para assumir a próxima função.

Isso é útil para cenários de acesso entre contas em que você precisa percorrer várias AWS contas para alcançar seus recursos de destino. Para obter etapas detalhadas de configuração, consulte [Usando o perfil de origem para acesso entre contas](#).

## Próxima etapa

Depois de autenticar seu sistema SAP AWS, o SDK for SAP ABAP executa automaticamente e assume `sts:assumeRole` a função IAM apropriada para a função comercial do usuário SAP.

## Práticas recomendadas de segurança do IAM

O administrador do IAM será responsável pelas três áreas principais a seguir.

- Garantir que o sistema SAP possa se autenticar com metadados do Amazon EC2 ou credenciais de chave secreta.
- Garantir que o sistema SAP tenha as permissões necessárias para se elevar com o `sts:assumeRole`.
- Para cada perfil lógico do IAM, criar um perfil do IAM para usuários do SAP com as permissões necessárias para realizar as funções de negócios (por exemplo, as permissões necessárias para Amazon S3, DynamoDB ou outros serviços). Essas são os perfis que o usuário assumirá.

Para obter mais informações, consulte o capítulo [Segurança](#) no SAP Lens: AWS Well-Architected Framework.

## Tópicos

- [Melhores práticas para perfil de instância do Amazon EC2](#)
- [Perfil do IAM para usuários do SAP](#)
- [Considerações sobre a segurança do perfil de origem](#)

## Melhores práticas para perfil de instância do Amazon EC2

A instância do Amazon EC2 na qual seu sistema SAP é executado tem um conjunto de autorizações com base em seu perfil de instância. Geralmente, o perfil da instância só precisa ter permissões para chamar `sts:assumeRole`, para permitir que o sistema SAP assuma perfis do IAM específicos da empresa, conforme o necessário. Essa elevação para outros perfis garante que um programa ABAP possa assumir um perfil que dê ao usuário o privilégio mínimo necessário para realizar o trabalho. Por exemplo, um perfil de instância pode conter a seguinte declaração.

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": [
        "arn:aws:iam::012345678912:role/finance-cfo",
        "arn:aws:iam::012345678912:role/finance-auditor",
        "arn:aws:iam::012345678912:role/finance-reporting"
      ]
    }
  ]
}
```

Esse exemplo anterior permite que o sistema SAP assuma as funções do IAM para o usuário CFO, AUDITOR ou REPORTING. O SDK escolherá a função correta do IAM para o usuário com base na função PFCG do usuário no SAP.

O perfil de instância do Amazon EC2 também pode ser usado para outras funções.

- [AWS Backint Agent para SAP HANA](#)
- [SAP em alta disponibilidade da AWS com roteamento de endereço IP de sobreposição](#)

Essas soluções também podem exigir permissões `sts:assumeRole` para perfis específicos de backup ou failover ou podem exigir que as permissões sejam atribuídas diretamente ao perfil de instância.

## Perfil do IAM para usuários do SAP

O programa ABAP precisa de permissões para realizar o trabalho do usuário: ler uma tabela do DynamoDB, invocar o Amazon Textract em um objeto PDF no Amazon S3, executar uma função. O mesmo modelo de segurança é usado em todos os AWS SDKs. Você pode usar um perfil do IAM existente que foi usado para outro AWS SDK.

O analista de negócios da SAP solicitará ao administrador do IAM o perfil `arn:aws:` de um perfil do IAM para cada perfil lógico necessário. Por exemplo, em um cenário financeiro, o analista de negócios pode definir os seguintes perfis lógicos do IAM.

- CFO
- AUDITOR
- REPORTING

O administrador do IAM definirá os perfis do IAM para cada perfil lógico do IAM.

### CFO

- `arn:aws:iam::0123456789:role/finance-cfo`
- permissões de gravação de dados em um bucket do Amazon S3.
- permissões de leitura e gravação em um banco de dados do DynamoDB

### AUDITOR

- `arn:aws:iam::0123456789:role/finance-auditor`

- permissões de leitura em um bucket do Amazon S3
- permissões de leitura em um banco de dados do DynamoDB

## REPORTING

- `arn:aws:iam::0123456789:role/finance-reporting`
- permissões de leitura em um banco de dados do DynamoDB
- nenhuma permissão para o bucket do Amazon S3

O analista de negócios inserirá os perfis do IAM em uma tabela de mapeamento para mapear os perfis lógicos do IAM com os perfis físicos do IAM.

Os perfis do IAM para usuários do SAP precisam permitir a ação `sts:assumeRole` de entidades principais confiáveis. As entidades principais confiáveis podem variar de acordo com a forma como o sistema SAP é autenticado na AWS. Para obter mais informações, consulte [Especificar uma entidade principal](#).

Veja alguns dos cenários de exemplo comuns do SAP:

- Sistema SAP em execução no Amazon EC2 com um perfil de instância atribuído: aqui, um perfil de instância do Amazon EC2 é anexado a um perfil do IAM.

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sts:AssumeRole"
      ],
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/SapInstanceProfile"
      }
    }
  ]
}
```

- Sistemas SAP em execução no Amazon EC2 sem um perfil de instância; aqui o Amazon EC2 assume perfis para usuários SAP.

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sts:AssumeRole"
      ],
      "Principal": {
        "Service": [ "ec2.amazonaws.com" ]
      }
    }
  ]
}
```

- Sistemas SAP executados on-premises: os sistemas SAP executados on-premises só podem ser autenticados usando a chave de acesso secreta. Para obter mais informações, consulte [Autenticação do sistema do SAP em AWS](#).

Aqui qualquer perfil do IAM assumido por um usuário do SAP deve ter uma relação de confiança que confie no usuário do SAP.

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sts:AssumeRole"
      ],
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/SAP_SYSTEM_S4H"
      }
    }
  ]
}
```

## Considerações sobre a segurança do perfil de origem

Ao usar o perfil de origem:

### Gerenciamento de perfis do IAM

Crítico: as funções do IAM nas cadeias de perfis de origem devem ser rigorosamente gerenciadas para evitar acesso não autorizado e escalonamento de privilégios:

- Aplique o princípio do menor privilégio - conceda somente as permissões mínimas necessárias para a finalidade específica de cada função
- Audite regularmente as permissões da função - revise e atualize as políticas de função trimestralmente ou quando os requisitos mudarem
- Monitore o uso da função - Use para rastrear chamadas de AssumeRole API e identificar padrões incomuns
- Limite as relações de confiança - restrinja quais diretores podem assumir cada função somente àqueles que realmente precisam de acesso
- Use condições em políticas de confiança - adicione condições como IP de origem, requisitos de MFA ou restrições baseadas em tempo, quando apropriado
- Documente os objetivos da função - mantenha uma documentação clara do caso de uso pretendido de cada função e das permissões necessárias

### Autorização e controle de acesso

- Garanta que todos os perfis intermediários na cadeia tenham as políticas de confiança apropriadas configuradas
- Os usuários devem ter /AWS1/SESS autorização para TODOS os perfis da cadeia, incluindo perfis intermediários
- Cada função do IAM deve confiar explicitamente na função anterior na cadeia

### Salvaguardas técnicas

- O SDK impõe uma profundidade máxima de cadeia de 5 perfis para evitar chamadas excessivas à API STS
- Referências circulares são automaticamente detectadas e evitadas

- O método de autenticação do perfil básico é validado para garantir que ele use um método padrão (INST, SSF ou RLA)

Para obter mais informações sobre como configurar o perfil de origem, consulte [Usando o perfil de origem para acesso entre contas](#).

## Autorizações da SAP

A autorização necessária para configurar o SDK depende da edição do SDK.

Tópicos

- [Autorizações para configuração](#)
- [Autorizações SAP para usuários finais](#)

## Autorizações para configuração

Consulte as guias a seguir para obter mais detalhes.

SDK for SAP ABAP

As seguintes autorizações são necessárias para configurar o SDK para SAP ABAP.

- S\_TCODE
  - TCD = /AWS1/IMG
- S\_TABU\_DIS
  - ACTVT = 02, 03
- DICBERCLS

Escolha um dos seguintes grupos de autorização.

- /AWS1/CFG- AWS SDK para SAP ABAP v1 - Config
- /AWS1/MOD- AWS SDK para SAP ABAP v1 - Tempo de execução
- /AWS1/PFL- AWS SDK para SAP ABAP v1 - Perfil do SDK
- /AWS1/RES- AWS SDK para SAP ABAP v1 - Recursos lógicos
- /AWS1/TRC- AWS SDK para SAP ABAP v1 - Rastreamento

## SDK for SAP ABAP - BTP edition

Use as etapas a seguir para permitir que o SDK for SAP ABAP - edição BTP acesse a configuração.

1. Crie uma nova função comercial usando o modelo SAP\_BR\_BPC\_EXPERT de função comercial. Esse modelo fornece acesso ao aplicativo Custom Business Configuration.
2. Em Detalhes gerais da função, acesse Categorias de acesso e escolha Irrestrito para ajuda de leitura, gravação e valor.
3. Vá até a guia Catálogo comercial e atribua o catálogo /AWS1/RTBTP\_BCAT comercial para fornecer acesso à configuração do SDK.
4. Vá até a guia Usuários corporativos e designe usuários corporativos para conceder acesso à configuração do SDK.

## Autorizações SAP para usuários finais

Pré-requisito: definir perfis do SDK

Antes que o administrador de segurança da SAP possa definir suas funções, o analista de negócios definirá os perfis do SDK em transação /AWS1/IMG para o AWS SDK for SAP ABAP ou o aplicativo Custom Business Configuration for SDK for SAP ABAP - edição BTP. Normalmente, um perfil SDK será nomeado de acordo com sua função comercial: ZFINANCE, ZBILLING, ZMFG, ZPAYROLL etc. Para cada perfil do SDK, o analista de negócios definirá perfis lógicos do IAM com nomes curtos, como CFO, AUDITOR, REPORTING. Eles serão mapeados para os perfis reais do IAM pelo administrador de segurança do IAM.

Defina PFCG ou funções comerciais

### Note

As funções do PFCG são chamadas de funções de negócios no ambiente SAP, BTP e ABAP.

O administrador de segurança do SAP então adicionará um objeto de autorização da /AWS1/SESS para conceder acesso a um perfil do SDK.

Objeto de autenticação da /AWS1/SESS

- Campo /AWS1/PROF = ZFINANCE

Os usuários também devem ser mapeados para perfis lógicos do IAM para cada perfil do SDK, dependendo da função do trabalho. Por exemplo, um auditor financeiro com acesso a relatórios pode ser autorizado para um perfil lógico do IAM chamado AUDITOR.

Objeto de autenticação da /AWS1/LROL

- Campo /AWS1/PROF = ZFINANCE
- Campo /AWS1/LROL = AUDITOR

Enquanto isso, o CFO, com autorizações de leitura/gravação, pode ter um perfil de PFCG autorizando-o o perfil lógico de CFO.

Objeto de autenticação da /AWS1/LROL

- Campo /AWS1/PROF = ZFINANCE
- Campo /AWS1/LROL = CFO

Em geral, um usuário deve ser autorizado para apenas um perfil lógico do IAM por perfil do SDK. Se um usuário for autorizado para mais de uma função do IAM (por exemplo, se o CFO estiver autorizado para ambas as funções CFO e funções AUDITOR lógicas do IAM), o AWS SDK rompe o empate garantindo que a função de maior prioridade (menor número de sequência) entre em vigor.

Como em todos os cenários de segurança, os usuários devem ter privilégios mínimos para realizar suas funções de trabalho. Uma estratégia simples para gerenciar funções de PFCG seria nomear funções únicas de PFCG de acordo com o perfil do SDK e a função lógica que elas autorizam. Por exemplo, o perfil Z\_AWS\_PFL\_ZFINANCE\_CFO concede acesso ao perfil ZFINANCE e ao perfil lógico do IAM do CFO. Esses perfis únicos podem então ser atribuídas a perfis compostos que definem as funções do trabalho. Cada empresa tem sua própria estratégia para gerenciamento de perfis, e nós encorajamos você a definir uma estratégia de PFCG que funcione para a sua empresa.

# Operações seguras

## Criptografia de dados em repouso

AWS As chaves de acesso secretas são usadas para autenticar o SDK. Eles são criptografados usando a funcionalidade SSF ou Credential Store da SAP.

## Criptografia de dados em trânsito

Todas as chamadas para Serviços da AWS são criptografadas com HTTPS. O SAP ICM gerencia a conexão HTTPS. AWS os certificados devem ser confiáveis no STRUST.

## Uso da API

Quando um usuário fABAP assume um perfil usando `sts:assumeRole`, o nome da sessão é intitulado `USERID-SID-MANDT`, onde:

- USERID é o usuário ABAP da variável SY-UNAME.
- SID é o ID do sistema ABAP da variável SY-SYSID.
- MANDT é o cliente ABAP da variável SY-MANDT.

O nome da sessão aparece CloudTrail como nome de usuário. Isso garante que as chamadas de API de um usuário ABAP possam ser rastreadas até o sistema, o cliente e o usuário que iniciou a chamada. Para obter mais informações, consulte [O que é o AWS CloudTrail?](#)

## Usando a autenticação de chave de acesso secreta com criptografia SSF

Os sistemas SAP locais (ou sistemas executados em outras nuvens) podem ser autenticados usando a autenticação AWS por chave de acesso secreta com. AWS Identity and Access Management O [Secure Store and Forward Mechanism](#) (SSF) da SAP é usado para criptografar e armazenar com segurança AWS as credenciais (ID da chave de acesso e chave de acesso secreta) de um usuário do IAM. O sistema SAP faz login AWS usando um usuário do IAM. Consulte [Gerenciamento de chaves de acesso para usuários do IAM](#) para obter informações.

## Pré-requisitos

Os seguintes pré-requisitos devem ser atendidos antes de iniciar a configuração:

- Os perfis do IAM para usuários do SAP devem ser criados pelo administrador do IAM. As funções devem ter permissões para chamar o necessário Serviços da AWS. Para obter mais informações, consulte [Práticas recomendadas de segurança no IAM](#).
- Crie autorização para executar a transação /AWS1/IMG. Consulte mais informações em [Autorizações para configuração](#).

## Procedimento

Siga estas instruções para configurar o armazenamento de credenciais criptografado por SSF:

### Etapas

- [Etapa 1 — Definir um aplicativo SSF para armazenamento de credenciais](#)
- [Etapa 2 — Defina os parâmetros de criptografia para o aplicativo SSF](#)
- [Etapa 3 — Criar PSE para o aplicativo SSF](#)
- [Etapa 4 — Atribuir um aplicativo SSF ao AWS SDK para SAP ABAP](#)
- [Etapa 5 — Configurar o perfil do SDK para usar credenciais criptografadas por SSF](#)

### Etapa 1 — Definir um aplicativo SSF para armazenamento de credenciais

1. Execute o código da transação SE16 para definir um aplicativo SSF.
2. Insira o nome da tabela SSFAPPLIC e selecione Novas entradas.
3. Insira os seguintes detalhes:
  - APPLIC: ZAWS1 (nome do aplicativo SSF).
  - DESCRIÇÃO: SSF Encryption for the AWS SDK para SAP ABAP (descrição).
  - Escolha Selected(X) a opção para os campos restantes.
4. Selecione Salvar.

### Etapa 2 — Defina os parâmetros de criptografia para o aplicativo SSF

1. Execute o código da transação /n/AWS1/IMG para iniciar o Guia de Implementação (IMG) do AWS SDK para SAP ABAP
2. Expanda o nó IMG > AWS SDK para SAP ABAP Configurações > Pré-requisitos técnicos > Configurações adicionais para sistemas locais.

3. Execute a atividade Set SSF Parameters IMG.
4. Selecione Novas entradas e escolha a aplicação de SSF criada na etapa anterior. Selecione Salvar.
5. Modifique o algoritmo de hash para SHA256(ou superior) e o algoritmo de criptografia para AES256-CBC. Mantenha as outras configurações como padrão e selecione Salvar.

Essas configurações de criptografia serão usadas para AWS criptografar credenciais com segurança.

### Etapa 3 — Criar PSE para o aplicativo SSF

1. Execute a /n/AWS1/IMG transação e selecione AWS SDK para SAP ABAP Configurações > Pré-requisitos técnicos > Configurações adicionais para sistemas locais.
2. Execute a atividade IMG Create PSE for SSF Application, que o direcionará para o. STRUST transaction Selecione Editar.
3. Selecione com o botão direito a aplicação de SSF criado na [the section called “Etapa 1”](#) e escolha Criar. Mantenha todas as outras configurações padrão e selecione Continuar. Certifique-se de escolher o algoritmo RSA e não o DSA.

### Etapa 4 — Atribuir um aplicativo SSF ao AWS SDK para SAP ABAP

1. Execute a /n/AWS1/IMG transação e selecione AWS SDK para SAP ABAP Configurações > Pré-requisitos técnicos > Configurações adicionais para sistemas locais.
2. Execute a atividade IMG Atribua um aplicativo SSF ao. AWS SDK para SAP ABAP
3. Selecione Novas entradas e insira o aplicativo SSF criado em [the section called “Etapa 1”](#). Selecione Salvar.

### Etapa 5 — Configurar o perfil do SDK para usar credenciais criptografadas por SSF

1. Execute a /n/AWS1/IMG transação e selecione AWS SDK para SAP ABAP Configurações > Configurações do aplicativo.
2. Execute o perfil do SDK da atividade IMG.
3. Selecione Novas entradas. Insira o nome e a descrição do perfil. Selecione Salvar.
4. Destaque a entrada que você criou e clique na ramificação de árvore Autenticação e Configurações.

## 5. Selecione Novas entradas e insira os seguintes detalhes:

- SID: O ID do sistema SAP.
- Cliente: O cliente do sistema SAP.
- ID do cenário: selecione o DEFAULT cenário criado pelo administrador do Basis.
- AWS Região: AWS região para a qual você deseja fazer chamadas.
- Método de autenticação: **Credentials from SSF Storage** selecione na lista suspensa e selecione Salvar. Selecione Definir credenciais e insira o ID da chave de acesso e a chave de acesso secreta do usuário do IAM.
- Desative as funções do IAM: mantenha isso como padrão, ou seja, desmarcado.
- Selecione Salvar.

## 6. Clique na ramificação da árvore de Mapeamento de perfis do IAM. Selecione Novas entradas. Insira o número de sequência, o nome da função lógica do IAM e o ARN da função do IAM fornecido pelo administrador do AWS IAM. Selecione Salvar.

Consulte mais informações em [Configuração da aplicação](#).

# Usar certificados com o IAM Roles Anywhere

O sistema SAP pode ser autenticado AWS usando a autenticação baseada em certificados com AWS Identity and Access Management o Roles Anywhere. Você deve configurar o certificado no STRUST e configurar o perfil do SDK em /AWS1/IMG.

## Pré-requisitos

Os pré-requisitos a seguir devem ser atendidos antes de começar a configuração para certificação.

- O certificado X.509 emitido pela autoridade de certificação (CA) deve atender aos requisitos a seguir.
  - O certificado de assinatura deve ser um certificado v3.
  - A cadeia não deve exceder cinco certificados.
  - O certificado deve permitir algoritmos RSA ou ECDSA.
- Registre sua CA com o IAM Roles Anywhere como uma âncora confiável e crie um perfil para especificar o roles/políticas IAM Roles Anywhere. Para obter mais informações, consulte [Criação de uma âncora de confiança e um perfil no AWS Identity and Access Management Roles Anywhere](#).

- Os perfis do IAM para usuários do SAP devem ser criados pelo administrador do IAM. As funções devem ter permissões para chamar o necessário Serviços da AWS. Para obter mais informações, consulte [Práticas recomendadas de segurança no IAM](#).
- Crie autorização para executar a transação /AWS1/IMG. Consulte mais informações em [Autorizações para configuração](#).

## Procedimento

Siga estas instruções para configurar a autenticação baseada em certificado.

### Etapas

- [Etapa 1: definir uma aplicação de SSF usando o Secure Store and Forward \(SSF\) de SAP](#)
- [Etapa 2: definir os parâmetros de SSF](#)
- [Etapa 3: criar o PSE e a solicitação de certificado](#)
- [Etapa 4: importar a resposta do certificado para o PSE relevante](#)
- [Etapa 5: configurar o perfil do SDK para usar o IAM Roles Anywhere](#)

### Etapa 1: definir uma aplicação de SSF usando o Secure Store and Forward (SSF) de SAP

1. Execute o código da transação SE16 para definir uma aplicação de SSF.
2. Insira o nome da tabela SSFAPPLIC e selecione Novas entradas.
3. Insira um nome para a aplicação de SSF no campo APPLIC, uma descrição no campo DESCRIPT e selecione a opção Selected (X) para os campos restantes.

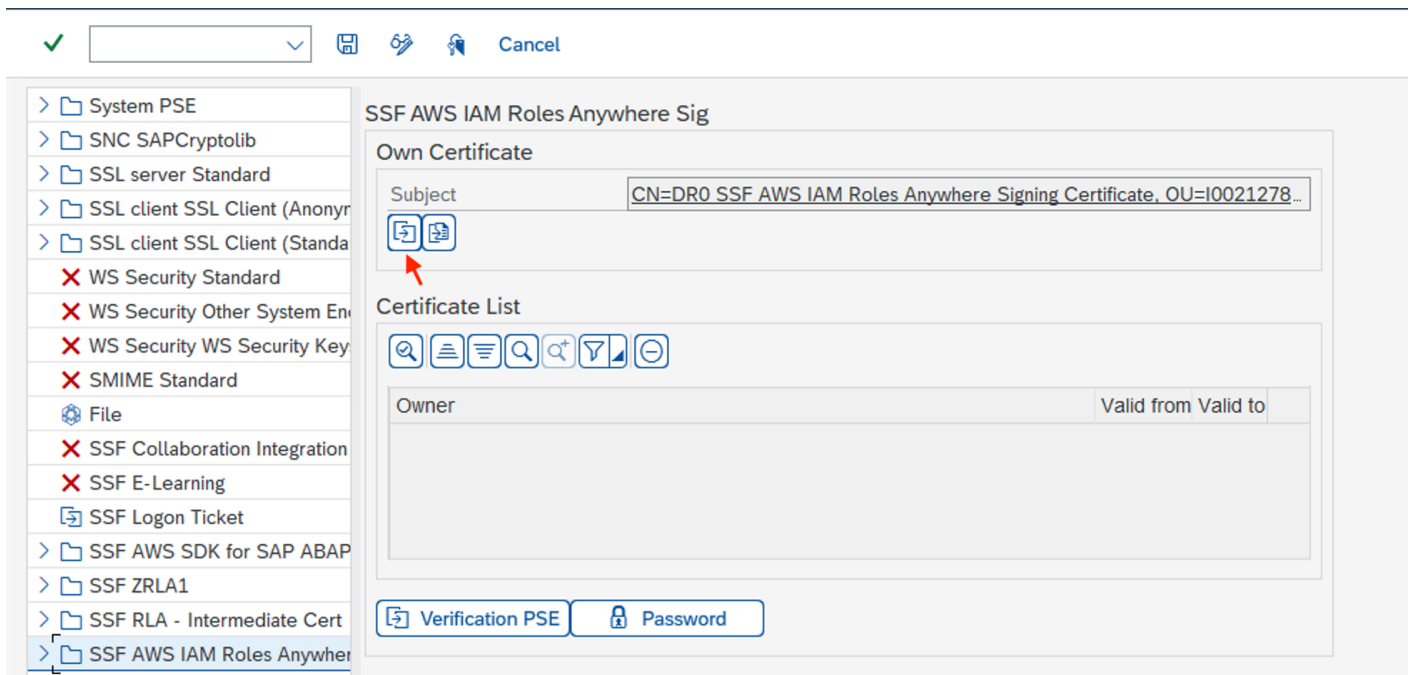
### Etapa 2: definir os parâmetros de SSF

1. Execute o Guia /n/AWS1/IMG de AWS SDK para SAP ABAP Implementação (IMG) para iniciar.
2. Selecione Configurações do AWS SDK para SAP ABAP > Pré-requisitos técnicos > Configurações adicionais para sistemas on-premises.
3. Execute a atividade do IMG Definir parâmetros de SSF.
4. Selecione Novas entradas e escolha a aplicação de SSF criada na etapa anterior. Selecione Salvar.

5. Modifique o algoritmo de hash para SHA256, e o algoritmo de criptografia para AES256-CBC. Mantenha as outras configurações como padrão e selecione Salvar.

### Etapa 3: criar o PSE e a solicitação de certificado

1. Execute a transação /n/AWS1/IMG e selecione Configurações do AWS SDK para SAP ABAP > Pré-requisitos técnicos > Configurações adicionais para sistemas on-premises.
2. Execute a atividade Create PSE for SSF Application do IMG.
3. Selecione Editar para a transação STRUST.
4. Selecione com o botão direito a aplicação de SSF criado na [the section called “Etapa 1”](#) e escolha Criar. Mantenha todas as outras configurações padrão e selecione Continuar.
5. Selecione Criar solicitação de certificado. Veja a imagem a seguir. Mantenha as opções padrão e selecione Continuar. Copie ou exporte a solicitação de certificado gerada e forneça-a à CA. A CA verifica a solicitação e responde com um certificado de chave pública assinado.



O processo de assinatura varia de acordo com a CA e a tecnologia usada por ela. Consulte [Emissão de certificados de entidade final privada](#) com a Autoridade de Certificação AWS Privada para obter um exemplo.

## Etapa 4: importar a resposta do certificado para o PSE relevante

1. Execute a transação /n/AWS1/IMG e selecione Configurações do AWS SDK para SAP ABAP > Pré-requisitos técnicos > Configurações adicionais para sistemas on-premises.
2. Execute a atividade Create PSE for SSF Application do IMG.
3. Selecione Editar para a transação STRUST.
4. Escolha a aplicação de SSF e selecione Importar resposta do certificado, localizada na seção de PSE abaixo do assunto. Copie e cole a resposta do certificado na caixa de texto ou importe o arquivo do sistema de arquivos. Selecione Continuar > Salvar.
5. Os detalhes do certificado podem ser visualizados selecionando o assunto duas vezes. As informações são exibidas na seção de certificados.

## Etapa 5: configurar o perfil do SDK para usar o IAM Roles Anywhere

1. Execute a /n/AWS1/IMG transação e selecione AWS SDK para SAP ABAP Configurações > Configurações do aplicativo.
2. Crie um perfil do SDK e dê um nome a ele.
3. Escolha o IAM Roles Anywhere como método de autenticação.
  - No painel esquerdo, escolha Autenticação e configurações.
  - Crie uma entrada e insira as informações do sistema de SAP e da Região da AWS.
  - Selecione IAM Roles Anywhere para o método de autenticação e selecione Salvar.
  - Selecione Inserir detalhes e, na janela pop-up, escolha a aplicação de SSF criada na [the section called “Etapa 1”](#). Insira o ARN da âncora de confiança e o ARN do perfil que foram criados em [the section called “Pré-requisitos”](#). Veja a imagem a seguir. Selecione Continuar.

Select Signing Certificate issued by your certificate authority (CA) from SSF

Certificate (SSF Application)

Enter your IAM Roles Anywhere details

Trust Anchor ARN

Profile ARN

4. No painel esquerdo, selecione Mapeamento de perfis do IAM. Insira um nome e forneça o ARN do perfil do IAM fornecido pelo administrador do IAM.

Consulte mais informações em [Configuração da aplicação](#).

## Usando o perfil de origem para acesso entre contas

O perfil de origem permite que os sistemas SAP acessem AWS recursos em várias contas, encadeando as suposições da função do IAM. Um perfil assume uma função, que então assume outra função, e assim por diante, semelhante ao `source_profile` parâmetro na CLI AWS. Isso é útil para cenários de acesso entre contas em que você precisa percorrer várias AWS contas para alcançar seus recursos de destino.

Exemplo: Seu sistema SAP é executado na Conta A (111111111111) e precisa acessar os buckets do Amazon S3 na Conta C (333333333333). Você configura três perfis:

1. DEV\_BASE obtém credenciais básicas dos metadados da instância do Amazon EC2 e assume a função P na Conta A
2. SHARED\_SERVICES usa DEV\_BASE credenciais para assumir a função Q na conta B (222222222222)
3. PROD\_S3\_ACCESS usa SHARED\_SERVICES credenciais para assumir a Função R na Conta C

Quando seu aplicativo usa `PROD_S3_ACCESS`, o SDK executa automaticamente a cadeia: obtenha credenciais dos metadados da instância → assuma a função P → assuma a função Q → assuma a função R.

## Pré-requisitos

Os seguintes pré-requisitos devem ser atendidos antes de configurar o perfil de origem:

- As funções do IAM para cada etapa da cadeia devem ser criadas pelo administrador do IAM. Cada função deve ter:
  - Permissões para ligar para o necessário Serviços da AWS
  - Relação de confiança configurada para permitir que a função anterior na cadeia a assuma

Para obter mais informações, consulte [Práticas recomendadas de segurança no IAM](#).

- Crie autorização para executar a transação `/AWS1/IMG`. Consulte mais informações em [Autorizações para configuração](#).
- Os usuários devem ter `/AWS1/SESS` autorização para TODOS os perfis da cadeia, incluindo perfis intermediários.

## Procedimento

Siga estas instruções para configurar o perfil de origem.


### Etapas

- [Etapa 1 — Configurar o perfil básico](#)
- [Etapa 2 — Configurar perfis encadeados](#)

### Etapa 1 — Configurar o perfil básico

O perfil básico é o primeiro perfil na cadeia e deve usar um método de autenticação padrão.

1. Execute a `/n/AWS1/IMG` transação para iniciar o Guia de AWS SDK para SAP ABAP Implementação (IMG).
2. Selecione AWS SDK for SAP ABAP Settings > Application Configurations > SDK Profile.
3. Crie um novo perfil para usar como seu perfil base selecionando Novas entradas e inserindo o nome e a descrição do perfil. Selecione Salvar.

 Note

Se você estiver usando um perfil existente que já esteja configurado com um método de autenticação padrão (INST, SSF ou RLA), você pode pular as etapas restantes desta seção e prosseguir diretamente para. [the section called “Etapa 2”](#)

4. Selecione o perfil que você criou e, em seguida, selecione Autenticação e configurações > Novas entradas e insira os seguintes detalhes:

- SID: O ID do sistema SAP
- Cliente: O cliente do sistema SAP
- ID do cenário: selecione o DEFAULT cenário criado pelo administrador do Basis
- AWS Região: AWS região para a qual você deseja fazer chamadas
- Método de autenticação: selecione uma das seguintes opções:
  - Função da instância via metadados para sistemas SAP executados no Amazon EC2
  - Credenciais do SSF Storage para sistemas locais ou outros sistemas em nuvem
  - IAM Roles Anywhere para autenticação baseada em certificados

Selecione Salvar.

5. Selecione IAM Role Mapping > New Entries e insira:

- Número de sequência: 1
- Função lógica do IAM: um nome descritivo (por exemplo, DEV\_BASE\_ROLE)
- ARN da função do IAM: o ARN da função do IAM na primeira conta (por exemplo,) `arn:aws:iam::111111111111:role/DevBaseRole`

Selecione Salvar.

## Etapa 2 — Configurar perfis encadeados

Configure cada perfil intermediário e final na cadeia.

Para **SHARED\_SERVICES** perfil (correntes de **DEV\_BASE**):

1. Execute a `/n/AWS1/IMG` transação.
2. Selecione AWS SDK for SAP ABAP Settings > Application Configurations > SDK Profile.

3. Selecione Novas entradas. Insira o nome do perfil (por exemplo,SHARED\_SERVICES) e a descrição. Selecione Salvar.
4. Selecione o perfil que você criou e, em seguida, selecione Autenticação e configurações > Novas entradas e insira os seguintes detalhes:
  - SID: O ID do sistema SAP
  - Cliente: O cliente do sistema SAP
  - ID do cenário: selecione o DEFAULT cenário criado pelo administrador do Basis
  - AWS Região: AWS região para a qual você deseja fazer chamadas
  - Método de autenticação: selecione Perfil de origem no menu suspenso
  - ID do perfil de origem: insira o ID do perfil base (por exemplo,DEV\_BASE)

Selecione Salvar.

5. Selecione IAM Role Mapping > New Entries e insira:
  - Número de sequência: 1
  - Função lógica do IAM: um nome descritivo (por exemplo,SHARED\_ROLE)
  - ARN da função do IAM: `arn:aws:iam::222222222222:role/SharedServicesRole`

Selecione Salvar.

Para **PROD\_S3\_ACCESS** perfil (correntes de**SHARED\_SERVICES**):

Repita as mesmas etapasSHARED\_SERVICES, mas:

- Use PROD\_S3\_ACCESS como nome
- Definir ID do perfil de origem como SHARED\_SERVICES
- Use PROD\_S3\_ROLE e `arn:aws:iam::333333333333:role/ProdS3AccessRole` no mapeamento de funções do IAM

Para ver as melhores práticas de segurança, incluindo gerenciamento de funções do IAM, configuração de políticas de confiança e requisitos de autorização, consulte [Melhores práticas para a segurança do IAM](#).

# Usando o SAP Credential Store

O SAP Credential Store é usado na SAP Business Technology Platform para armazenar com segurança as credenciais para autenticação de chave de acesso secreta. AWS Você deve ter uma assinatura para usar o serviço.

As instruções a seguir pressupõem que você já tenha configurado um perfil do SDK. Para obter mais informações, consulte [Configurando. AWS SDK para SAP ABAP](#)

Antes de iniciar a configuração, verifique se você atende aos pré-requisitos. Para obter mais informações, consulte [SAP Credential Store](#).

## Tópicos

- [Etapas de configuração](#)
- [Usando o SAP Credential Store com o SDK](#)

## Etapas de configuração

### Etapas

- [Etapa 1: definir as configurações para autenticação](#)
- [Etapa 2: criar uma chave de serviço](#)
- [Etapa 3: converter a chave de serviço em .p12 formato](#)
- [Etapa 4: Conecte-se ao SAP BTP, ambiente ABAP](#)

### Etapa 1: definir as configurações para autenticação

Use as etapas a seguir para definir as configurações do Credential Store para autenticação.

1. Navegue até a guia Configurações da instância do SAP Credential Store.
2. Selecione Editar configurações:
  - Escolha TLS mútuo como o tipo de autenticação padrão.
  - Selecione Desativado para o status de criptografia de carga útil. A carga é criptografada em trânsito com HTTPS. No entanto, a carga útil atualmente não pode ser criptografada duas vezes.
3. Selecione Salvar.

## Etapa 2: criar uma chave de serviço

Use as etapas a seguir para criar uma chave de serviço para o Credential Store.

1. No painel esquerdo do aplicativo SAP Credential Store, navegue até Service Keys.
2. Selecione Criar chave de serviço.
3. Insira um nome para a chave de serviço e selecione Criar.

A chave de serviço é criada com base no tipo de autenticação escolhido. Faça o download da chave de serviço e mantenha-a segura para uso posterior.

## Etapa 3: converter a chave de serviço em **.p12** formato

É necessário um certificado de cliente no **.p12** formato para criar um usuário de saída para o sistema de comunicação. Use as etapas a seguir para gerar um **.p12** certificado a partir dos detalhes do certificado fornecidos na chave do Serviço de Armazenamento de Credenciais.

1. Baixe o certificado SAP Cloud Root CA (exigido pela SAP) do [SAP Trust Center Services](#).
2. Abra o certificado SAP Cloud Root CA em qualquer formato de arquivo de texto. No final do arquivo, pressione Enter e copie e cole o certificado do campo certificado da chave de serviço. Substitua os caracteres \n da nova linha pela nova linha real (Enter) e salve o certificado inteiro em formato de **.cer** arquivo.
3. Copie a chave do campo chave da chave de serviço. Essa chave privada deve ser tratada como dados confidenciais. Cole-o em um arquivo de texto e substitua os caracteres \n da nova linha pela nova linha real (Enter). Salve a chave privada em um arquivo de texto.
4. Com o certificado e a chave privada gerados nas etapas anteriores, execute o comando a seguir para gerar um **.p12** certificado.

```
openssl pkcs12 -export -out <.p12_<filename>> -inkey <private_key.key> -in  
<certificate.cer>
```

O comando exigia a verificação da senha de exportação. Guarde a senha para uso posterior.

Exclua o arquivo de **.key** texto salvo em sua chave privada.

## Etapa 4: Conecte-se ao SAP BTP, ambiente ABAP

Configure o ambiente SAP BTP, ABAP para se conectar ao SAP Credential Store.

### Tópicos

- [Sistema de comunicação](#)
- [Arranjo de comunicação](#)

### Sistema de comunicação

Use as etapas a seguir para criar um sistema de comunicação que permita a comunicação do ambiente SAP BTP e ABAP com o SAP Credential Store.

1. Abra a barra de lançamento Fiori do sistema de ambiente ABAP.
2. Selecione o quadro Sistemas de comunicação para abrir o aplicativo.
3. Selecione Novo.
4. Insira um nome e uma ID para o sistema de comunicação e selecione Criar. Por exemplo, você pode nomear o sistema ZSAP\_CREDSTORE.
5. Insira outras informações necessárias:
  - Nome do host: copie o nome do host do URL da chave de serviço. Por exemplo, se o URL for `https://credstore.mesh.cf.us10.hana.ondemand.com/api/v1/credentials`, o nome do host será `credstore.mesh.cf.us10.hana.ondemand.com`.
  - Usuários para comunicação externa: selecione + para adicionar um novo usuário.
    - a. Selecione Certificado de cliente SSL como mecanismo de autenticação.
    - b. Selecione Carregar novo certificado:
      - Procure o .p12 certificado gerado na etapa anterior.
      - Insira uma descrição.
      - Insira a senha de exportação usada para gerar o .p12 certificado.
      - Selecione Carregar.
    - c. Selecione Criar para criar um usuário externo.
6. Selecione Salvar.
7. Exclua a chave de serviço baixada na etapa anterior.

## Arranjo de comunicação

Use as etapas a seguir para criar um acordo de comunicação para fornecer um cenário de comunicação para comunicação externa.

1. Abra a barra de lançamento Fiori do sistema de ambiente ABAP.
2. Selecione o quadro Arranjos de comunicação para abrir o aplicativo.
3. Selecione Novo.
4. Selecione o cenário /AWS1/CRED\_COMM\_SCENARIO de comunicação e insira um nome para o arranjo de comunicação. Por exemplo, .Z\_AWS\_SDK\_TO\_SAP\_CREDSTORE
5. Escolha Criar.
6. No campo Sistema de Comunicação, procure o Sistema de Comunicação criado na etapa anterior. Outras informações são preenchidas automaticamente após a seleção do sistema.
7. Selecione Salvar.
8. Selecione Verificar conexão para testar sua conexão.

Quando essa configuração estiver concluída, o ambiente ABAP poderá usar o arranjo de comunicação para usar o serviço SAP Credential Store via serviço de saída (HTTP).

## Usando o SAP Credential Store com o SDK

### Etapas

- [Etapa 1: criar um namespace e credencial \(s\)](#)
- [Etapa 2: Configurar o aplicativo Custom Business Configuration](#)

### Etapa 1: criar um namespace e credencial (s)

Crie um namespace e uma credencial no SAP Credential Store com a ajuda da SAP — [Crie, edite e exclua uma credencial](#).

Insira os detalhes a seguir para criar uma credencial do tipo Key.

- Namespace — insira um nome para o namespace e agrupe as credenciais relacionadas.
- Nome — Insira um nome para a chave. Recomendamos `saws-0123456789012-username`, onde:
  - `0123456789012` é o Conta da AWS ID ao qual a credencial concede acesso

- `username` é o nome de usuário do IAM ao qual a credencial pertence
- **Valor** — Insira uma chave de acesso secreta codificada em base 64. Use o comando a seguir para codificar em base 64 sua chave de acesso secreta.

```
xargs echo -n | base64 # just press enter, do not enter arguments on the command line
MySecretAccessKey
Ctrl-D
```

O comando lê a chave de acesso secreta da entrada padrão e a passa para base64 sem uma nova linha à direita. Ele envia a chave de acesso secreta codificada em base 64 para a tela. Limpe ou feche seu terminal depois de copiar o valor no SAP Credential Store.

- **Nome de usuário** — Insira o ID da sua chave de acesso.
- Escolha Criar.

Um novo namespace com uma credencial é criado e as credenciais podem ser adicionadas, excluídas ou modificadas nesse namespace.

Siga o princípio do privilégio mínimo para gerenciar o acesso às credenciais armazenadas no namespace.

## Etapa 2: Configurar o aplicativo Custom Business Configuration

Use as etapas a seguir para configurar o aplicativo Custom Business Configuration para definir a credencial a ser usada para autenticação pelo SDK.

1. Abra a barra de lançamento Fiori do sistema de ambiente ABAP.
2. Navegue pelo quadro Configuração Empresarial Personalizada para abrir o aplicativo.
3. Abra a configuração comercial do SDK Profile.
4. Selecione o perfil do SDK para o qual as configurações de autenticação devem ser definidas para o SAP Credential Store.
5. Na guia Autenticação e configurações do perfil selecionado, selecione Editar e insira os seguintes detalhes:
  - **Método de autenticação** — Selecione as credenciais do SAP Credential Store.
  - **Namespace** — Insira o namespace criado no SAP Credential Store. Para obter mais informações, consulte [the section called “Etapa 1: criar um namespace e credencial \(s\)”](#).

- Nome da chave — insira o nome da credencial da chave criada. Para obter mais informações, consulte [the section called “Etapa 1: criar um namespace e credencial \(s\)”](#).
  - Arranjo de comunicação — insira o nome do arranjo de comunicação criado. Para obter mais informações, consulte [the section called “Arranjo de comunicação”](#).
6. Selecione Aplicar para acessar a tela Perfil do AWS SDK.
  7. Selecione Selecionar transporte para selecionar o transporte usando a ajuda de valores.
  8. Selecione Salvar.

# Solucionar problemas AWS SDK para SAP ABAP

Esta seção fornece etapas de solução de problemas para possíveis cenários de erro.

## Tópicos

- [Falha de importação](#)
- [Restrição de localização não especificada](#)
- [Erros SSL](#)
- [Configuração do perfil](#)
- [Autorização do IAM](#)
- [Autorização para realizar as ações necessárias](#)
- [Cenário ativo](#)
- [Caracteres especiais no código](#)
- [Conectividade](#)

## Falha de importação

Problema — A classe 'CL\_SYSTEM\_UUID' não contém uma interface 'IF\_SYSTEM\_UUID\_\_STATIC RFC4122

Causa: a nota SAP 0002619546 está ausente em seu sistema.

Resolução: verifique se o [SAP Note 0002619546](#) foi aplicado ao seu sistema.

## Restrição de localização não especificada

Problema: a restrição de localização não especificada é incompatível com o endpoint de region específico para o qual a solicitação foi enviada

Causa — Seu bucket do Amazon S3 não tem a AWS região no `io_createbucketconfiguration` parâmetro.

Resolução: ao criar um bucket em qualquer região, exceto `us-east-1`, especifique a região do bucket do Amazon S3 usando o parâmetro `io_createbucketconfiguration` em `createbucket()`. Não é necessário especificar uma restrição para `us-east-1`.

Veja a seguir um exemplo de um parâmetro `io_createbucketconfiguration` configurado corretamente:

```
createbucket(  
  iv_bucket = 'amzn-s3-demo-bucket'  
  io_createbucketconfiguration = NEW /aws1/c1_s3_createbucketconf( 'us-west-1' )  
).
```

## Erros SSL

Problema: falha na incompatibilidade do nome de host do certificado do servidor SSL ou no handshake SSL com docs.aws.amazon.com:443: SSSLERR\_NO\_SSL\_RESPONSE

Causa: o parâmetro `icm/HTTPS/client_sni_enabled` não está definido como TRUE no perfil DEFAULT.

Resolução: use as etapas a seguir para solucionar os problemas em questão ou qualquer outro problema relacionado ao SSL.

1. Abra a SAPGUI e vá até a barra de comando.
2. Execute a transação RZ10.
3. Vá para Perfil e escolha o perfil DEFAULT. A versão é preenchida automaticamente.
4. Na seção Editar perfil, selecione Manutenção estendida e, em seguida, selecione Alterar.
5. Pesquise o parâmetro `icm/HTTPS/client_sni_enabled`.
  - Se o parâmetro existir, edite o valor do parâmetro e defina-o como TRUE.
  - Se o parâmetro não existir, crie um parâmetro usando as etapas a seguir.

1. Selecione Parâmetro.

### Note

Selecione o parâmetro para criação e não edição (ícone de lápis).

2. Insira `icm/HTTPS/client_sni_enabled` no campo Nome do parâmetro.
3. Insira TRUE no campo Valor do parâmetro.
4. Selecione Salvar.

6. Salve essas alterações no perfil DEFAULT e saia.

## Configuração do perfil

Problema: não foi possível encontrar a configuração no perfil <profile\_name> com o cenário DEFAULT para <sid>:<client>

Causas: <profile\_name> está incorreto ou não foi configurado.

Resolução: use as etapas a seguir para configurar o perfil.

1. Abra a SAPGUI e execute a transação /n/AWS1/IMG.
2. Vá para Configuração do aplicativo > Perfil do SDK.
  - Se seu perfil estiver configurado, verifique se o nome do perfil está correto.
  - Se seu perfil não estiver configurado, siga as etapas para configurar um perfil.
3. Selecione Novas entradas.
  - a. Insira um Nome e uma Descrição para o perfil.
  - b. Selecione Salvar.
4. Escolha a entrada criada na etapa anterior e selecione Autenticação e configurações.
5. Selecione Novas entradas, insira os detalhes a seguir e selecione Salvar.
  - SID
  - Cliente
  - ID do cenário
  - AWS Região
  - Método de autenticação
    - Selecione a função da instância por meio de metadados para sistemas SAP em AWS execução.
    - Selecione Credenciais do armazenamento SSF para sistemas SAP executados on-premises ou em outra nuvem.
6. Selecione Mapeamento de perfis do IAM > Novas entradas, insira os detalhes a seguir e selecione Salvar.
  - Número de sequência

- Perfil lógico do IAM
- ARN do perfil do IAM

## Autorização do IAM

Problema — Não foi possível assumir a função <iam\_role\_arn> ou o usuário: <user\_arn> não está autorizado a executar: sts: AssumeRole no recurso: <iam\_role\_arn>

Causas: os seguintes podem ser os possíveis motivos desse erro.

- O ARN incorreto do perfil do IAM foi especificado
- O usuário do IAM não tem permissão para acessar o perfil do IAM
- Relação de falta de confiança entre o perfil do IAM assumido e o perfil do IAM assumido ou usuário do IAM

Resolução: use as etapas a seguir para garantir que o ARN do perfil do IAM esteja correto.

1. Abra a SAPGUI e execute a transação /n/AWS1/IMG.
2. Vá para Configuração do aplicativo > Perfil do SDK e escolha o perfil que foi configurado com seu perfil do IAM.
3. Selecione Mapeamento de perfis do IAM e verifique ou corrija o ARN do perfil do IAM.
  - Se o ARN do seu perfil do IAM estiver correto, verifique se seu perfil do IAM foi configurado corretamente. Para obter mais informações, consulte [Solução de problemas dos perfis do IAM](#).

## Autorização para realizar as ações necessárias

Problema: o usuário <user\_arn> não está autorizado a executar: <action> no recurso: <resource\_arn>

Causa: o usuário não tem permissões para realizar uma ação.

Resolução: user\_arn deve ser configurado com as permissões necessárias do resource\_arn para realizar uma determinada action. Para obter mais informações, consulte [Permissões necessárias para acessar recursos do IAM](#).

## Cenário ativo

Problema: nenhum cenário ativo configurado

Causa: a configuração do cenário ativo foi perdida.

Resolução: consulte [Configurações de runtime](#) para configurar um cenário ativo.

## Caracteres especiais no código

Aviso: o caractere 0x00A0 não pode fazer parte de uma palavra ABAP

### Note

Esse aviso pode ser precedido por várias mensagens de erro.

Causa: copiar e colar código de fontes diferentes pode inserir caracteres especiais em seu código.

Resolução: quando você cola qualquer código no editor de código-fonte ABAP, você vê o seguinte pop-up.

Caracteres de espaço ininterruptos foram detectados. Converter em espaços?

Escolha Sim para responder a essa pergunta. Além disso, recomendamos selecionar o código para copiá-lo, em vez de usar o botão copiar nas caixas de código.

## Conectividade

Problema: SCLNT\_HTTP (411): Falha na conexão direta com tla.region.amazonaws.com:443: NIECONN\_REFUSED (-10)

Causa: o sistema SAP não tem conectividade com a Internet e não pode estabelecer uma conexão TCP/IP com a porta 443 de tla.region.amazonaws.com.

Resolução — O sistema SAP deve ser capaz de estabelecer conexão com AWS endpoints na porta HTTPS 443, diretamente ou por meio de um servidor proxy. Você pode estabelecer/verificar a conectividade com a Internet com uma das seguintes opções:

- Conexão de saída direta à Internet por meio de um NAT ou gateway da Internet

- Conexão por meio de um servidor proxy

Para obter mais informações, consulte [Conexão por meio de um servidor de proxy](#).

## Tópicos adicionais

Esta seção abrange os seguintes tópicos.

Tópicos

- [AWS SDK para SAP ABAP lançamentos](#)
- [Licenciamento SAP](#)

## AWS SDK para SAP ABAP lançamentos

AWS O SDK para SAP ABAP é entregue em transportes, e o AWS SDK para SAP ABAP - edição BTP é entregue como complementos. O mecanismo para importar transportes e complementos é diferente, mas a funcionalidade técnica é a mesma. Para obter mais informações, consulte [Configurar](#).

Tópicos

- [Estratégia de lançamento](#)
- [Práticas recomendadas](#)
- [Patches de SDK para SAP ABAP](#)
- [Instalação de um módulo adicional](#)
- [Desinstalando o SDK para SAP ABAP](#)

## Estratégia de lançamento

A versão 1 do AWS SDK para SAP ABAP é atualizada com frequência. Novos patches são lançados semanalmente ou diariamente com base nos lançamentos e atualizações do Serviços da AWS. Os patches para Serviços da AWS podem incluir correções de bugs e outras alterações que atualizam o nível de patch do SDK. Para obter mais informações, consulte AWS SDKs a política de [manutenção de ferramentas](#).

## Práticas recomendadas

Recomendamos reter o mesmo nível de patch do SDK para SAP ABAP para todos os sistemas SAP (desenvolvimento, controle de qualidade e produção).

Ao corrigir o SDK, importe a versão mais recente em sua sandbox. Em seguida, você pode importá-lo para os sistemas de desenvolvimento, controle de qualidade e produção, seguindo seus procedimentos normais de controle de alterações.

## Patches de SDK para SAP ABAP

Cada versão do SDK para SAP ABAP é entregue como um conjunto de transportes cumulativos, incluindo todas as correções de erros, atributos e atualizações. Não há diferença entre um patch e um transporte de instalação. Você deve importar os transportes mais recentes para corrigir o SDK para SAP ABAP.

Devido às dependências dos módulos `core` Runtime e API, os módulos de API individuais não podem ser corrigidos separadamente. Ao aplicar patches, você deve atualizar o `core` módulo e todos os outros módulos de API que foram instalados, mesmo que determinados módulos não estejam mais em uso.

Os cenários a seguir descrevem o processo de aplicação de patches:

- Cenário 1: atualização regular da versão do SDK

Ao aplicar o patch para a versão mais recente do SDK:

- Importe todos os transportes (módulos API principais e instalados) simultaneamente
  - Nenhuma sequência de importação específica é necessária
  - Por exemplo, se você importou os transportes `core`, `ec2` e `lmd` ao instalar o SDK, deverá importar os transportes mais recentes para `core`, `ec2` e `lmd` durante a aplicação de patches.
- Cenário 2: Adicionar um novo módulo de API

Ao importar um novo módulo de API (por exemplo, `Amazon tex Textract`) de uma versão de SDK diferente da dos módulos atualmente instalados:

- Todos os módulos existentes devem ser corrigidos para corresponder à versão de lançamento do SDK do novo módulo
  - Importe simultaneamente (sem sequência específica):
    - Transportes mais recentes para todos os módulos existentes (`core` e módulos de API instalados)
    - Transportes para o novo módulo de API `tex`
- Cenário 3: atualização de um módulo de API existente

Ao atualizar um módulo de API existente (por exemplo, Amazon Translatex18) para a versão mais recente do SDK, as atualizações de módulos individuais não são suportadas devido a interdependências. Você deve corrigir todo o SDK atualizando o `core` módulo e todos os módulos de API instalados juntos, seguindo o processo descrito em [Regular SDK Version Update \(p. 589\)](#)

## Instalação de um módulo adicional

Importe o transporte para o novo módulo no mesmo nível de patch que o existente `core` e os módulos para instalar um módulo de API adicional em seu sistema SAP. Siga as diretrizes em [the section called “Patches de SDK para SAP ABAP”](#) se quiser importar uma versão mais recente do módulo. Isso garante que os níveis de patch sejam compatíveis em todos os módulos do SDK.

## Desinstalando o SDK para SAP ABAP

[Para desinstalar o SDK para SAP ABAP, você deve baixar um kit de transporte de exclusão do release/uninstall-abapsdk-LATEST.zip. <https://sdk-for-sapabap.aws.amazon.com/awsSdkSapabapV1/>](#)

```
curl "https://sdk-for-sapabap.aws.amazon.com/awsSdkSapabapV1/release/uninstall-abapsdk-LATEST.zip" -o "uninstall-abapsdk-LATEST.zip"
```

Você pode baixar um arquivo de assinatura em <https://sdk-for-sapabap.aws.amazon.com/awsSdkSapabapV1/release/uninstall-abapsdk-latest.sig>. Para validar o arquivo, consulte [Verificar SDK para SAP ABAP](#).

Para cada módulo SDK instalado em seu sistema SAP, o transporte de exclusão correspondente deve ser importado do arquivo ZIP anterior. Você pode remover um único módulo sem desinstalar o SDK inteiro. Você pode fazer isso importando somente o transporte de exclusão do módulo que você deseja remover. Se você estiver desinstalando o SDK inteiro com todos os seus módulos, o transporte de exclusão principal deverá ser importado por último.

Recomendamos que você teste a desinstalação em um sandbox antes de tentar nos sistemas de desenvolvimento, controle de qualidade ou produção.

### Considerações

Antes de desinstalar o SDK, consulte as considerações a seguir.

- As configurações do SDK de serão perdidas. O IMG deve ser reconfigurado na instalação.
- Se você tiver programas Z que dependem do SDK, eles gerarão erros de sintaxe após a remoção do SDK.
- As funções PFCG ou Business que contêm referências de autorização do SDK terão autorizações inválidas após a remoção do SDK. Remova as referências de autorização do SDK dos perfis do PFCG antes de desinstalar o SDK.

### Note

AWS O SDK para SAP ABAP - edição BTP não pode ser desinstalado durante a versão prévia para desenvolvedores.

## Licenciamento SAP

O uso do software da SAP está sujeito aos termos da SAP. Você é responsável por cumprir os termos de licenciamento da SAP, incluindo distribuição de software e condições de licenciamento indireto. Qualquer informação fornecida não é aconselhamento jurídico e não deve ser usada para fins de conformidade com o licenciamento. Se tiver dúvidas sobre o licenciamento ou os direitos ao software da Microsoft, consulte sua equipe jurídica, a Microsoft ou seu revendedor da Microsoft.

Pergunta: o uso do SDK para SAP ABAP afetará minha licença SAP?

Resposta: AWS SDK para SAP ABAP permite que você consuma Serviços da AWS com seu próprio código ABAP. É usado em cenários de integração entre um sistema SAP e os Serviços da AWS. Qualquer cenário em que os dados do sistema SAP sejam enviados para um sistema terceirizado (não SAP) ou criados por esse sistema pode ter implicações no licenciamento indireto. A SAP tem várias abordagens para definir o acesso indireto, como cálculos baseados no usuário e cálculos baseados em resultados. A metodologia para definir o acesso indireto depende do seu contrato com a SAP. Você deve estar ciente das orientações fornecidas em seu contrato com a SAP e pode discutir mais sobre isso com a SAP ou seu revendedor.

Em 2018, a SAP lançou dois documentos, Guia de acesso indireto para clientes da base instalada da SAP e Preços do SAP ERP para a era digital, abordando o acesso indireto/digital. Esses documentos podem ser encontrados nos sites da SAP e são exemplos de abordagens de licenciamento indireto. No entanto, esses documentos não refletem seu contrato específico com a SAP.

# Histórico do documento para AWS SDK para SAP ABAP

## Guia do desenvolvedor

A tabela a seguir descreve as versões de documentação do AWS SDK para SAP ABAP.

Alteração	Descrição	Data
<a href="#">Novo conteúdo</a>	Foi adicionada a <a href="#">substituição do servidor Per-service proxy</a> .	15 de maio de 2026
<a href="#">Novo conteúdo</a>	Adicionado <a href="#">usando o AWS SDK para o SAP ABAP Knowledge MCP</a> Server.	24 de abril de 2026
<a href="#">Novo conteúdo</a>	Adicionado <a href="#">usando o perfil de origem para Cross-Account acesso</a> .	4 de março de 2026
<a href="#">Novo conteúdo</a>	Adicionado <a href="#">usando autenticação de chave de acesso secreta com criptografia SSF</a> .	12 de janeiro de 2025
<a href="#">Novo conteúdo</a>	Prévia para desenvolvedores do SDK para SAP ABAP - edição BTP.	31 de maio de 2024
<a href="#">Novo conteúdo</a>	Adição de <a href="#">Using certificates with IAM Roles Anywhere</a> .	1.º de dezembro de 2023
<a href="#">Novo conteúdo</a>	Adição de <a href="#">Building products with SDK</a> .	1.º de dezembro de 2023
<a href="#">Novo conteúdo</a>	Adição de <a href="#">Retry behavior</a> .	1.º de dezembro de 2023
<a href="#">Novo conteúdo</a>	Adição de <a href="#">Licenciamento SAP</a> .	22 de setembro de 2023

---

<a href="#">Versão pública</a>	Relançamento inicial do AWS SDK para SAP ABAP Developer Guide.	30 de junho de 2023
<a href="#">Novo conteúdo</a>	Atributos <a href="#">adicionados do AWS SDK para SAP ABAP</a> .	30 de maio de 2023
<a href="#">Novo conteúdo</a>	Solução de <a href="#">problemas adicionada do AWS SDK para SAP ABAP</a> .	17 de fevereiro de 2023
<a href="#">Pré-visualização para desenvolvedores</a>	Prévia do AWS SDK para SAP ABAP Developer Guide para desenvolvedores.	17 de novembro de 2022

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.