Unable to locate subtitle

AWS Glue DataBrew Guida per gli sviluppatori

AWS Glue DataBrew Guida per gli sviluppatori: ***Unable to locate subtitle***

Table of Contents

Che cos'èAWS Glue DataBrew?

AWS Glue DataBrewè uno strumento di preparazione visiva dei dati che consente agli utenti di pulire e normalizzare i dati senza scrivere alcun codice. L'utilizzo DataBrew aiuta a ridurre il tempo necessario per preparare i dati per l'analisi e l'apprendimento automatico (ML) fino all'80%, rispetto alla preparazione dei dati sviluppata su misura. Puoi scegliere tra oltre 250 trasformazioni già pronte per automatizzare le attività di preparazione dei dati, come il filtraggio delle anomalie, la conversione dei dati in formati standard e la correzione di valori non validi.

In questo modo DataBrew, analisti aziendali, data scientist e data engineer possono collaborare più facilmente per ottenere informazioni dai dati grezzi. Poiché DataBrew è serverless, indipendentemente dal livello tecnico, è possibile esplorare e trasformare terabyte di dati grezzi senza dover creare cluster o gestire alcuna infrastruttura.

Con l' DataBrew interfaccia intuitiva, puoi scoprire, visualizzare, pulire e trasformare in modo interattivo i dati grezzi. DataBrew fornisce suggerimenti intelligenti per aiutarti a identificare i problemi di qualità dei dati che possono essere difficili da individuare e che richiedono molto tempo per essere risolti. Con la DataBrew preparazione dei dati, puoi impiegare il tuo tempo per agire in base ai risultati e iterare più rapidamente. Puoi salvare la trasformazione come passaggi di una ricetta, che puoi aggiornare o riutilizzare in seguito con altri set di dati e distribuirla su base continuativa.

L'immagine seguente mostra come DataBrew funziona ad alto livello.

Per utilizzarlo DataBrew, devi creare un progetto e connetterti ai tuoi dati. Nell'area di lavoro del progetto, i dati vengono visualizzati in un'interfaccia visiva simile a una griglia. Qui puoi esplorare i dati e vedere le distribuzioni dei valori e i grafici per comprenderne il profilo.

Per preparare i dati, puoi scegliere tra più di 250 trasformazioni punta e clicca. Queste includono la rimozione di valori nulli, la sostituzione dei valori mancanti, la correzione delle incongruenze dello schema, la creazione di colonne basate su funzioni e molto altro. È inoltre possibile utilizzare le trasformazioni per applicare tecniche di elaborazione del linguaggio naturale (NLP) per suddividere le frasi in frasi. Le anteprime immediate mostrano una parte dei dati prima e dopo la trasformazione, in modo da poter modificare la ricetta prima di applicarla all'intero set di dati.

Dopo DataBrew aver eseguito la ricetta sul set di dati, l'output viene archiviato in Amazon Simple Storage Service (Amazon S3). Una volta che il set di dati pulito e preparato si trova in Amazon S3, un altro dei tuoi sistemi di storage o gestione dei dati può importarlo.

Concetti e termini fondamentali in AWS Glue DataBrew

Di seguito, è possibile trovare una panoramica dei concetti e della terminologia principali in AWS Glue DataBrew. Dopo aver letto questa sezione, vedeteNozioni di base su AWS Glue DataBrew, che illustra il processo di creazione di progetti, connessione dei set di dati e esecuzione dei job.

Argomenti

- Progetto
- Set di dati
- Recipe
- Processo
- Data lineage
- Profilo dei dati

Progetto

L'area di lavoro interattiva per la preparazione dei dati in DataBrew si chiama progetto. Utilizzando un progetto di dati, gestisci una raccolta di elementi correlati: dati, trasformazioni e processi pianificati. Come parte della creazione di un progetto, scegli o crei un set di dati su cui lavorare. Successivamente, crei una ricetta, che è un insieme di istruzioni o passaggi su cui DataBrew vuoi

agire. Queste azioni trasformano i dati grezzi in un modulo pronto per essere utilizzato dalla pipeline di dati.

Set di dati

Per set di dati si intende semplicemente un insieme di dati, ovvero righe o record suddivisi in colonne o campi. Quando crei un DataBrew progetto, ti connetti o carichi i dati che desideri trasformare o preparare. DataBrew può lavorare con dati provenienti da qualsiasi fonte, importati da file formattati, e si collega direttamente a un elenco crescente di archivi dati.

Infatti DataBrew, un set di dati è una connessione di sola lettura ai dati. DataBrew raccoglie una serie di metadati descrittivi per fare riferimento ai dati. Nessun dato effettivo può essere modificato o archiviato da. DataBrew Per semplicità, utilizziamo il set di dati per fare riferimento sia al set di dati effettivo che agli usi dei metadati. DataBrew

Recipe

In DataBrew, una ricetta è un insieme di istruzioni o passaggi relativi ai dati su cui si desidera DataBrew agire. Una ricetta può contenere molti passaggi e ogni passaggio può contenere molte azioni. Utilizzi gli strumenti di trasformazione sulla barra degli strumenti per impostare tutte le modifiche che desideri apportare ai tuoi dati. Successivamente, quando sei pronto per vedere il prodotto finito della tua ricetta, assegni questo lavoro DataBrew e lo pianifichi. DataBrew memorizza le istruzioni sulla trasformazione dei dati, ma non memorizza nessuno dei dati effettivi. Puoi scaricare e riutilizzare le ricette in altri progetti. Puoi anche pubblicare più versioni di una ricetta.

Processo

DataBrew si occupa di trasformare i dati eseguendo le istruzioni impostate durante la preparazione di una ricetta. Il processo di esecuzione di queste istruzioni è chiamato processo. Un job può mettere in atto le tue ricette di dati in base a una pianificazione preimpostata. Ma non sei limitato a un programma. Puoi anche eseguire lavori su richiesta. Se vuoi profilare alcuni dati, non hai bisogno di una ricetta. In tal caso, puoi semplicemente impostare un processo di profilazione per creare un profilo di dati.

Data lineage

DataBrew tiene traccia dei dati in un'interfaccia visiva per determinarne l'origine, chiamata derivazione dei dati. Questa visualizzazione mostra come i dati fluiscono attraverso diverse entità da

dove provenivano originariamente. Puoi vederne l'origine, le altre entità da cui è stato influenzato, cosa gli è successo nel tempo e dove sono stati archiviati.

Profilo dei dati

Quando crei il profilo dei tuoi dati, DataBrew crea un rapporto chiamato profilo dati. Questo riepilogo descrive la forma esistente dei dati, incluso il contesto del contenuto, la struttura dei dati e le relative relazioni. È possibile creare un profilo dati per qualsiasi set di dati eseguendo un processo di profilo dati.

Integrazioni di prodotti e servizi

Utilizza questa sezione per sapere con quali prodotti e servizi si integrano DataBrew.

DataBrew funziona con i seguenti AWS servizi per il networking, la gestione e la governance:

- Amazon CloudFront
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Step Functions

DataBrew funziona con i seguenti AWS data lake e data store:

- AWS Lake Formation
- Amazon S3

DataBrew supporta i seguenti formati di file ed estensioni per il caricamento dei dati.

Format (Formato)	Estensione del file (opzionale)	Estensioni per file compressi (obbligatorie)

Comma-separated valori	`.csv`	`.gz`
		`.snappy`
		`.lz4`

Format (Formato)	Estensione del file (opzionale)	Estensioni per file compressi (obbligatorie)
		`.bz2` `.deflate`
Cartella di lavoro Microsoft Excel	`.xlsx`	Nessun supporto per la compressione

JSON (documento JSON e righe JSON)	`.json, .jsonl`	`.gz`
		`.snappy`
		`.lz4`
		`.bz2`
		`.deflate`

Apache ORC	`.orc`	`.zlib` `.snappy`

Apache Parquet	`.parquet`	`.gz`
		`.snappy`
		`.lz4`

DataBrew scrive file di output su Amazon S3 e supporta i seguenti formati di file ed estensioni.

Format (Formato)	Estensione del file (non compressa)	Estensioni di file (compresse)
Comma-separated valori	`.csv`	`.csv.snappy` , `.csv.gz`, `.csv.lz4`, `csv.bz2`, `.csv.deflate` , `csv.br`

Format (Formato)	Estensione del file (non compressa)	Estensioni di file (compresse)
Tab-separated valori	`.csv`	`.tsv.snappy` , `.tsv.gz`, `.tsv.lz4`, `tsv.bz2`, `.tsv.deflate` , `tsv.br`
Apache Parquet	`.parquet`	`.parquet.snappy` , `.parquet.gz` , `.parquet.lz4` , `.parquet.lzo` , `.parquet.br`
AWS Glue parquet	Non supportata	`.glue.parquet.snappy`
Apache Avro	`.avro`	`.avro.snappy` , `.avro.gz`, `.avro.lz4` , `.avro.bz2` , `.avro.deflate` , `.avro.br`
Apache ORC	`.orc`	`.orc.snappy` , `.orc.lzo`, `.orc.zlib`
XML	`.xml`	`.xml.snappy` , `.xml.gz`, `.xml.lz4`, `.xml.bz2`, `.xml.deflate` , `.xml.br`
JSON (solo formato JSON Lines)	`.json`	`.json.snappy` , `.json.gz`, `.json.lz4` , `json.bz2`, `.json.deflate` , `.json.br`
Tableau Hyper	Non supportata	Non applicabile

Configurazione AWS Glue DataBrew

Prima di iniziare AWS Glue DataBrew, devi configurare alcune autorizzazioni, un utente e un ruolo. Inizia eseguendo i seguenti passaggi:

1. Registrazione di un AWS account in base alle esigenze e creazione di policy AWS Identity and Access Management(IAM) per consentire agli utenti di eseguire DataBrew:
   - Registrazione di un nuovo AWS account e aggiunta di un utente. Per ulteriori informazioni, consulta Configurare un nuovo AWS account.
   - Aggiungere una policy IAM per un utente della console. Un utente con queste autorizzazioni può accedere DataBrew a.Console di gestione AWS
   - Aggiungere autorizzazioni per le risorse di dati per un ruolo IAM. Un ruolo IAM con queste autorizzazioni può accedere ai dati per conto dell'utente.

   Devi essere un amministratore IAM per creare utenti, ruoli e policy.
2. Aggiungere utenti o gruppi per DataBrew. Un utente o un gruppo con le autorizzazioni corrette può accedere DataBrew alla console.
3. Aggiungere un ruolo con autorizzazioni per l'accesso ai dati. DataBrew Un ruolo con le autorizzazioni corrette può accedere ai dati per conto dell'utente.

Configurare un nuovo AWS account

Se non disponi di un AWS account, crea un AWS account e crea un utente amministratore IAM.

Se non ne possiedi uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Aprire https://portal.aws.amazon.com/billing/signup.
2. Segui le istruzioni online.

   Nel corso della procedura di registrazione riceverai una telefonata o un messaggio di testo e ti verrà chiesto di inserire un codice di verifica attraverso la tastiera del telefono.

   Quando ti iscrivi a un Account AWS, Utente root dell'account AWSviene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di

sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire attività che richiedono l'accesso di un utente root.

Per creare un utente amministratore, scegli una delle seguenti opzioni.

Scelta di un modo per gestire il tuo amministratore	Per	Come	Puoi anche
In IAM Identity Center (Consigliato)	Usa credenziali a breve termine per accedere a AWS. Ciò è in linea con le best practice per la sicurezza. Per informazioni sulle best practice, consulta Best practice per la sicurezza in IAM nella Guida per l'utente di IAM.	Segui le istruzioni riportate in Nozioni di base nella Guida per l'utente di Centro identità AWS IAM.	Configura l'accesso programmatico configurando l'uso Centro identità AWS IAM nella Guida AWS CLI per l'AWS Command Line Interface utente.
In IAM (Non consigliato)	Usa credenziali a lungo termine per accedere a AWS.	Segui le istruzioni in Creare un utente IAM per l'accesso di emergenza nella Guida per l'utente di IAM.	Configura l'accesso programmatico seguendo quanto riportato in Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente di IAM.

Per ulteriori informazioni, consulta gli argomenti seguenti nella Guida per l'utente IAM:

- Che cos'è IAM?

- Configurazione con IAM
- Creazione di un utente e un gruppo di amministrazione (console)

Configurazione del AWS CLI

Se prevedi di utilizzare JupyterLab o l' DataBrew API, assicurati di installare AWS Command Line Interface(AWS CLI). Non è necessario per utilizzare la DataBrew console o eseguire i passaggi degli esercizi introduttivi.

Per configurare il AWS CLI

1. Scarica e configura il file AWS CLI utilizzando i passaggi seguenti:
   - Installazione dell'AWS CLI
   - Nozioni di base sulla configurazione
2. Verifica la configurazione immettendo il seguente DataBrew comando al prompt dei comandi.

   ```
   aws databrew help
   ```

   Se questa istruzione restituisce l'errore "`aws: error: argument command: Invalid choice`" seguito da un lungo elenco di servizi, disinstallatelo e reinstallatelo.AWS CLI Questa azione non sovrascrive la configurazione esistente.

AWS CLI i comandi utilizzano la AWS regione predefinita della configurazione, a meno che non venga impostata con un parametro o un profilo. È possibile aggiungere il `--region` parametro a ciascun comando.

Se preferisci, puoi aggiungere un profilo denominato in `~/.aws/config` o `%UserProfile%/.aws/config` (in Microsoft Windows). I profili denominati possono inoltre conservare altre impostazioni, come illustrato nell'esempio seguente.

```
[profile databrew]
aws_access_key_id = ACCESS-KEY-ID-OF-IAM-USER
aws_secret_access_key = SECRET-ACCESS-KEY-ID-OF-IAM-USER
region = us-east-1
output = text
```

Configurazione AWS Identity and Access Management Autorizzazioni (IAM)

Prima di iniziare, devi configurare alcune cose in IAM. Devi essere un amministratore o farti aiutare da uno di loro. Tuttavia, se disponi di un account con accesso da amministratore, puoi eseguire queste attività da solo. In questa sezione puoi trovare semplici istruzioni per ogni attività.

Di seguito è riportata una panoramica di ciò che devi fare:

- Come parte di questo processo, aggiungi un utente. Non è necessario aggiungere un nuovo utente, è possibile utilizzarne uno esistente. Alleghi DataBrew le autorizzazioni in modo che l'utente possa aprire la DataBrew console.
- Crea un ruolo IAM. Un ruolo consente determinate azioni e concede autorizzazioni quando viene utilizzato, entro certi limiti. Ad esempio, funziona solo per gli utenti del tuo AWS account. Puoi aggiungere altre limitazioni in un secondo momento.
- Crea la policy o le policy IAM di cui hai bisogno. Una policy è un elenco di cose che un utente è autorizzato a fare. Per creare una policy, apri un'altra pagina della console e incolli il testo da un file scaricato.

> **Note**
>
> Qui forniamo informazioni di configurazione di base. Ti consigliamo di dedicare del tempo alla personalizzazione delle autorizzazioni in modo che soddisfino le tue esigenze di sicurezza e conformità. Se hai bisogno di assistenza, contatta l'amministratore o il servizio di AWS assistenza.

Per aggiungere le autorizzazioni richieste

1. Crea policy IAM per consentire agli utenti di eseguire le operazioni DataBrew procedendo come segue:

   - Aggiungi una policy IAM personalizzata per un utente della console. Se non hai bisogno di una policy personalizzata, puoi scegliere invece la policy AWS gestita. Basta aggiungerla all'utente nel passaggio 2. Un utente con queste autorizzazioni può accedere alla console DataBrew di servizio.

- Aggiungi le autorizzazioni per le risorse di dati. Un ruolo IAM con queste autorizzazioni può accedere ai dati per conto dell'utente.

Devi essere un amministratore per creare utenti, ruoli e politiche.

2. Aggiungi utenti o gruppi per DataBrew. Un utente o un gruppo con le autorizzazioni corrette può accedere alla DataBrew console.
3. Aggiungi un ruolo con autorizzazioni per l'accesso ai dati. DataBrew Un ruolo con le autorizzazioni corrette può accedere ai dati per conto dell'utente.

Configurazione delle politiche IAM per DataBrew

Utilizzi le politiche IAM per gestire le autorizzazioni. Una policy semplifica l'aggiunta delle autorizzazioni correlate tutte in una volta, anziché una alla volta.

Ti consigliamo di creare le politiche utilizzando gli stessi nomi che forniamo. Utilizziamo i nomi riportati di seguito per queste politiche in tutta la documentazione. L'uso di questi nomi semplifica anche la necessità di contattare l'AWS assistenza. Tuttavia, puoi scegliere di modificare sia i nomi delle politiche che il loro contenuto. Per ulteriori informazioni sulle policy IAM, consulta Create a customer managed policy nella IAM User Guide.

Dopo aver creato le politiche necessarie per l'uso DataBrew, le alleghi a utenti e ruoli. La procedura per eseguire questa operazione è illustrata più avanti in questa sezione.

Argomenti

- Aggiungere una policy IAM per un utente della console
- Aggiungere autorizzazioni per le risorse di dati per un ruolo IAM
- Configurazione delle politiche IAM per DataBrew

Aggiungere una policy IAM per un utente della console

La configurazione delle autorizzazioni per un utente per Console di gestione AWSè facoltativa, ma se hai bisogno dell'accesso alla console, esegui prima questo passaggio.

Per configurare le autorizzazioni di accesso DataBrew alla console, scegli una delle seguenti opzioni:

- Utilizza la politica gestita da AWS:`AwsGlueDataBrewFullAccessPolicy`. Se scegli questa opzione, passa alla politica successiva,Aggiungere autorizzazioni per le risorse di dati per un ruolo IAM.
- Crea la politica descritta in questa sezione,`AwsGlueDataBrewCustomUserPolicy`. Questa opzione consente di personalizzare la politica con requisiti di sicurezza personalizzati aggiuntivi.

La seguente politica concede le autorizzazioni necessarie per eseguire la DataBrew console. Fornisci tali autorizzazioni utilizzando IAM.

Per definire la policy AwsGlueDataBrewCustomUserPolicy IAM per DataBrew (console)

1. Scarica il codice JSON per la policy `AwsGlueDataBrewCustomUserPolicy`IAM.
2. Accedi Console di gestione AWS e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
3. Nel riquadro di navigazione, scegli Policy.
4. Per ogni policy, scegli Crea policy.
5. Nella schermata Crea policy, vai alla scheda JSON.
6. Copia l'istruzione JSON della policy che hai scaricato. Incollala sull'istruzione di esempio nell'editor.
7. Verifica che la politica sia personalizzata in base al tuo account, ai requisiti di sicurezza e alle AWS risorse richieste. Se è necessario apportare modifiche, è possibile apportarle nell'editor.
8. Scegliere Esamina policy.

Per definire la politica AwsGlueDataBrewCustomUserPolicy IAM per DataBrew (AWS CLI)

1. Scarica il codice JSON per la policy `AwsGlueDataBrewCustomUserPolicy`IAM.
2. Personalizza la policy come descritto nel primo passaggio della procedura precedente.
3. Esegui il comando seguente per creare la politica.

```
aws iam create-policy --policy-name AwsGlueDataBrewCustomUserPolicy --policy-
document file://iam-policy-AwsGlueDataBrewCustomUserPolicy.json
```

Aggiungere autorizzazioni per le risorse di dati per un ruolo IAM

Per connettersi ai dati,AWS Glue DataBrew deve disporre di un ruolo IAM che possa passare per conto dell'utente. Di seguito, puoi scoprire come creare la policy da allegare successivamente a un ruolo IAM.

La `AwsGlueDataBrewDataResourcePolicy` policy concede le autorizzazioni necessarie per connettersi ai dati utilizzando. DataBrew Per qualsiasi operazione che accede ai dati in un'altra AWS risorsa, come l'accesso ai tuoi oggetti in Amazon S3 DataBrew , è necessaria l'autorizzazione per accedere alla risorsa per tuo conto.

Per definire la policy AwsGlueDataBrewDataResourcePolicy IAM per DataBrew (console)

1. Scarica il file JSON per `AwsGlueDataBrewDataResourcePolicy`.
2. Accedi Console di gestione AWS e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
3. Nel riquadro di navigazione, scegli Policy.
4. Per ogni policy, scegli Crea policy.
5. Nella schermata Crea policy, vai alla scheda JSON.
6. Copia l'istruzione JSON della policy che hai scaricato. Incollala sull'istruzione di esempio nell'editor.
7. Verifica che la politica sia personalizzata in base al tuo account, ai requisiti di sicurezza e alle AWS risorse richieste. Se è necessario apportare modifiche, è possibile apportarle nell'editor.
8. Scegliere Esamina policy.

Per definire la politica AwsGlueDataBrewDataResourcePolicy IAM per DataBrew (AWS CLI)

1. Scarica il file JSON per `AwsGlueDataBrewDataResourcePolicy`.
2. Personalizzate la policy come descritto nel primo passaggio della procedura precedente.
3. Esegui il comando seguente per creare la politica.

```
aws iam create-policy --policy-name AwsGlueDataBrewDataResourcePolicy --policy-
document file://iam-policy-AwsGlueDataBrewDataResourcePolicy.json
```

Configurazione delle politiche IAM per DataBrew

Di seguito, puoi trovare dettagli ed esempi sulle politiche IAM che puoi utilizzare. DataBrew I dettagli sulle politiche di base sono disponibili qui. Inoltre, ci sono altri esempi che non è necessario utilizzare DataBrew. Sono configurazioni aggiuntive che è possibile utilizzare in determinate situazioni.

Argomenti

AwsGlueDataBrewCustomUserPolicy

La `AwsGlueDataBrewCustomUserPolicy` politica concede la maggior parte delle autorizzazioni necessarie per utilizzare la console. DataBrew Alcune delle risorse specificate in questa politica si riferiscono ai servizi utilizzati da. DataBrew Questi includono nomi per AWS Glue Data Catalog bucket Amazon S3, Amazon CloudWatch Logs e risorse.AWS KMSÈ simile alla policy gestita AWS denominata. `AwsGlueDataBrewFullAccessPolicy`

La tabella seguente descrive le autorizzazioni concesse dalla policy.

Azione	Risorsa	Descrizione
`"databrew:*"`	`"*"`	Concede l'autorizzazione a eseguire tutte le operazioni DataBrew API.

`"glue:GetDatabases"`
`"glue:GetPartitions"`
`"glue:GetTable"`
`"glue:GetTables"`
`"glue:GetDataCatalogEncryptionSettings"`
	`"*"`	Consente l'elenco di AWS Glue database e tabelle.

Azione	Risorsa	Descrizione
"dataexchange:ListDataSets" "dataexchange:ListDataSetRevisions" "dataexchange:ListRevisionAssets" "dataexchange:CreateJob" "dataexchange:StartJob" "dataexchange:GetJob"	"*"	Consente l'elenco delle risorse AWS Data Exchange nei set di dati.
"kms:DescribeKey" "kms:ListKeys" "kms:ListAliases"	"*"	Consente l'elenco delle AWS KMS chiavi da utilizzare per la crittografia dell'output del lavoro.
"kms:GenerateDataKey"	"arn:aws:kms:::key/key_ids"	Consente la crittografia dell'output del lavoro.
"s3:ListAllMyBuckets" "s3:GetBucketCORS" "s3:GetBucketLocation" "s3:GetEncryptionConfiguration"	"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"	Consente l'elenco di bucket Amazon S3 per progetti, set di dati e lavori. Consente l'invio di file di output a S3.
"sts:GetCallerIdentity"	"*"	Ottieni informazioni sul chiamante corrente.

Azione	Risorsa	Descrizione
`"cloudtrail:LookupEvents",`	`"*"`	Consenti l'elenco AWS CloudTrail degli eventi per i set di dati (data lineage).
`"iam:ListRoles"` `"iam:GetRole"`	`"*"`	Consente di elencare i ruoli IAM da utilizzare per progetti e lavori.

AwsGlueDataBrewDataResourcePolicy

La `AwsGlueDataBrewDataResourcePolicy` policy concede le autorizzazioni necessarie per connettersi ai dati e configurarli. DataBrew

La tabella seguente descrive le autorizzazioni concesse dalla policy.

Azione	Risorsa	Descrizione
`"s3:GetObject"`	`"arn:aws:s3:::bucket_name/*",` `"arn:aws:s3:::bucket_name"`	Consente di visualizzare in anteprima i file.
`"s3:PutObject"` `"s3:PutBucketCORS"`	`"arn:aws:s3:::bucket_name/*",` `"arn:aws:s3:::bucket_name"`	Consente l'invio di file di output a S3.
`"s3:DeleteObject"`	`"arn:aws:s3:::bucket_name/*",` `"arn:aws:s3:::bucket_name"`	Consente di eliminare un oggetto creato da. DataBrew
`"s3:ListBucket"`	`"arn:aws:s3:::bucket_name/*",` `"arn:aws:s3:::bucket_name"`	Consente l'elenco di bucket Amazon S3 da progetti, set di dati e lavori.

Azione	Risorsa	Descrizione
`"kms:Decrypt"`	`"arn:aws:kms:::key/key_ids"`	Consente la decrittografia di set di dati crittografati.
`"kms:GenerateDataKey"`	`"arn:aws:kms:::key/key_ids"`	Consente la crittografia dell'output del lavoro.

`"ec2:DescribeVpcEndpoints"`
`"ec2:DescribeRouteTables"`
`"ec2:DeleteNetworkInterface"`
`"ec2:DescribeNetworkInterfaces"`
`"ec2:DescribeSecurityGroups"`
`"ec2:DescribeSubnets"`
`"ec2:DescribeVpcAttribute"`
`"ec2:CreateNetworkInterface"`
	`"*"`	Consente la configurazione di elementi di rete Amazon EC2, come i cloud privati virtuali (VPC), durante l'esecuzione di lavori e progetti.

`"ec2:DeleteNetworkInterface"`	`"*"`	Consente di eliminare un'interfaccia di rete in un VPC.

Azione	Risorsa	Descrizione

`"ec2:CreateTags"`
 	
`"ec2:DeleteTags"`
	`"arn:aws:ec2:::network-interface/*", "arn:aws:ec2:::security-group/*"`	Consente la creazione e l'eliminazione di tag.
		
		Queste autorizzazioni sono necessarie se utilizzi un catalogo AWS Glue dati con un VPC abilitato . DataBrew passa i dati AWS Glue per eseguire lavori e progetti. Queste autorizzazioni consentono di etichettare le risorse Amazon EC2 create per gli endpoint di sviluppo.AWS Glue etichetta le interfacce di rete, i gruppi di sicurezza e le istanze di Amazon EC2 con. `aws-glue-service-resource`

`"logs:CreateLogGroup"`
 	
`"logs:CreateLogStream"`
 	
`"logs:PutLogEvents"`
	`"arn:aws:logs:::log-group:/aws-glue-databrew/*"`	Consente la scrittura di log su Amazon CloudWatch Logs
		
		DataBrew scrive i log in gruppi di log i cui nomi iniziano con. `aws-glue-databrew`

Azione	Risorsa	Descrizione
`"lakeformation:GetDataAccess"`	`"*"`	Consente l'accesso a AWS Lake Formation, a condizione che `"Glue":"GetTable"` sia consentito anche L'uso di Lake Formation richiede un'ulteriore configurazione nella console Lake Formation.

Policy IAM per l'utilizzo di oggetti Amazon S3 con DataBrew

La `AwsGlueDataBrewSpecificS3BucketPolicy` policy concede le autorizzazioni necessarie per accedere a S3 per conto di utenti non amministrativi.

Personalizza la policy come segue:

1. Sostituisci i percorsi di Amazon S3 nella policy in modo che indichino i percorsi che desideri utilizzare. Nel testo di esempio, *`BUCKET-NAME-1/SPECIFIC-OBJECT-NAME`* rappresenta un oggetto o un file specifico. *`BUCKET-NAME-2/`*rappresenta tutti gli oggetti (*) il cui nome di percorso inizia con`BUCKET-NAME-2/`. Aggiornali per assegnare un nome ai bucket che stai utilizzando.
2. (Facoltativo) Utilizza i caratteri jolly nei percorsi di Amazon S3 per limitare ulteriormente le autorizzazioni. Per ulteriori informazioni, consulta Elementi delle policy IAM: variabili e tag nella Guida per l'utente di IAM.

   Best practice di sicurezza: per impedire l'accesso non autorizzato ai bucket Amazon S3 con nomi simili in AWS altri account, includi la chiave di condizione `aws:ResourceAccount` nella tua policy. Ciò garantisce che sia DataBrew possibile accedere solo ai bucket all'interno del proprio AWS account, anche quando si utilizzano ARN di risorse wildcard. Aggiungi la seguente condizione alle tue dichiarazioni politiche:

```
"Condition": {
"StringEquals": {
"aws:ResourceAccount": "123456789012"
```

```
}
}
```

`123456789012`Sostituiscilo con l'ID AWS del tuo account effettivo.

A tale scopo, potresti limitare le autorizzazioni per le azioni `s3:PutObject` e`s3:PutBucketCORS`. Queste azioni sono necessarie solo per gli utenti che creano DataBrew progetti, poiché tali utenti devono essere in grado di inviare file di output a S3.

Per ulteriori informazioni e per vedere alcuni esempi di cosa puoi aggiungere a una policy IAM per Amazon S3, consulta Bucket Policy Examples nella Amazon S3 Developer Guide.

La tabella seguente descrive le autorizzazioni concesse dalla policy.

Azione	Risorsa	Descrizione
`"s3:GetObject"`	`"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"`	Consente di visualizzare in anteprima i file.
`"s3:PutObject"` `"s3:PutBucketCORS"`	`"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"`	Consente l'invio di file di output a S3.
`"s3:DeleteObject"`	`"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"`	Consente di eliminare un oggetto.

Per definire la policy AwsGlueDataBrewSpecificS3BucketPolicy IAM per DataBrew (console)

1. Scarica il codice JSON per la policy `AwsGlueDataBrewSpecificS3BucketPolicy`IAM.
2. Accedi Console di gestione AWS e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.

3. Nel riquadro di navigazione, scegli Policy.
4. Per ogni policy, scegli Crea policy.
5. Nella schermata Crea policy, vai alla scheda JSON.
6. Incolla l'istruzione JSON della policy sull'istruzione di esempio nell'editor.
7. Verifica che la politica sia personalizzata in base al tuo account, ai requisiti di sicurezza e alle AWS risorse richieste. Se è necessario apportare modifiche, è possibile apportarle nell'editor.
8. Scegliere Esamina policy.

Per definire la politica AwsGlueDataBrewSpecificS3BucketPolicy IAM per DataBrew (AWS CLI)

1. Scarica il file JSON per `AwsGlueDataBrewSpecificS3BucketPolicy`.
2. Personalizzate la policy come descritto nel primo passaggio della procedura precedente.
3. Esegui il comando seguente per creare la politica.

```
aws iam create-policy --policy-name AwsGlueDataBrewSpecificS3BucketPolicy --policy-
document file://iam-policy-AwsGlueDataBrewSpecificS3BucketPolicy.json
```

Politica IAM con cui utilizzare la crittografia DataBrew

La `AwsGlueDataBrewS3EncryptedPolicy` policy concede le autorizzazioni necessarie per accedere agli oggetti S3 crittografati con AWS Key Management Service(AWS KMS) per conto di utenti non amministrativi.

Personalizza la policy come segue:

1. Sostituisci i percorsi di Amazon S3 nella policy in modo che indichino i percorsi che desideri utilizzare. Nel testo di esempio, *`BUCKET-NAME-1/SPECIFIC-OBJECT-NAME`* rappresenta un oggetto o un file specifico. *`BUCKET-NAME-2/`*rappresenta tutti gli oggetti (`*`) il cui nome di percorso inizia con`BUCKET-NAME-2/`. Aggiornali per assegnare un nome ai bucket che stai utilizzando.
2. (Facoltativo) Utilizza i caratteri jolly nei percorsi di Amazon S3 per limitare ulteriormente le autorizzazioni. Per ulteriori informazioni, consulta Elementi delle policy IAM: variabili e tag.

A tale scopo, potresti limitare le autorizzazioni per le azioni e. `s3:PutObject` `s3:PutBucketCORS` Queste azioni sono necessarie solo per gli utenti che creano DataBrew progetti, poiché tali utenti devono essere in grado di inviare file di output a S3.

Per ulteriori informazioni e per vedere alcuni esempi di cosa puoi aggiungere a una policy IAM per Amazon S3, consulta Bucket Policy Examples.

3. Trova i seguenti ARN di risorse nel file. `ToUseKms`

```
  "arn:aws:kms:AWS-REGION-NAME:AWS-ACCOUNT-ID-WITHOUT-DASHES:key/KEY-IDS",
"arn:aws:kms:AWS-REGION-NAME:AWS-ACCOUNT-ID-WITHOUT-DASHES:key/KEY-IDS"
```

4. Cambia l'AWS account di esempio con il tuo numero di AWS account (senza trattini).
5. Modifica l'elenco di esempio per elencare invece i ruoli IAM che desideri utilizzare. Ti consigliamo di definire l'ambito delle tue policy IAM in base al set di autorizzazioni più piccolo possibile. Tuttavia, puoi consentire al tuo utente di accedere a tutti i ruoli IAM, ad esempio se utilizzi un account di apprendimento personale con dati di esempio. Per consentire all'elenco di accedere a tutti i ruoli IAM, modifica l'elenco di esempio con una sola voce:`"arn:aws:iam::111122223333:role/*"`.

La tabella seguente descrive le autorizzazioni concesse dalla policy.

Azione	Risorsa	Descrizione
`"s3:GetObject"`	`"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"`	Consente di visualizzare in anteprima i file.
`"s3:ListBucket"`	`"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"`	Consente l'elenco di bucket Amazon S3 da progetti, set di dati e lavori.
`"s3:PutObject"`	`"arn:aws:s3:::bucket_name/*",`	Consente l'invio di file di output a S3.

Azione	Risorsa	Descrizione
	`"arn:aws:s3:::bucket_name"`	
`"s3:DeleteObject"`	`"arn:aws:s3:::bucket_name/*", "arn:aws:s3:::bucket_name"`	Consente di eliminare un oggetto creato da. DataBrew
`"kms:Decrypt"`	`"arn:aws:kms:::key/key_ids"`	Consente la decrittografia di set di dati crittografati.
`"kms:GenerateDataKey*"`	`"arn:aws:kms:::key/key_ids"`	Consente la crittografia dell'output del lavoro.

Per definire la policy AwsGlueDataBrewS3EncryptedPolicy IAM per DataBrew (console)

1. Scarica il codice JSON per la policy `AwsGlueDataBrewS3EncryptedPolicy`IAM.
2. Accedi Console di gestione AWS e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
3. Nel riquadro di navigazione, scegli Policy.
4. Per ogni policy, scegli Crea policy.
5. Nella schermata Crea policy, vai alla scheda JSON.
6. Incolla l'istruzione JSON della policy sull'istruzione di esempio nell'editor.
7. Verifica che la politica sia personalizzata in base al tuo account, ai requisiti di sicurezza e alle AWS risorse richieste. Se è necessario apportare modifiche, è possibile apportarle nell'editor.
8. Scegliere Esamina policy.

Per definire la politica AwsGlueDataBrewS3EncryptedPolicy IAM per DataBrew (AWS CLI)

1. Scarica il file JSON per `AwsGlueDataBrewS3EncryptedPolicy`.
2. Personalizzate la policy come descritto nel primo passaggio della procedura precedente.
3. Esegui il comando seguente per creare la politica.

```
aws iam create-policy --policy-name AwsGlueDataBrewS3EncryptedPolicy --policy-
document file://iam-policy-AwsGlueDataBrewS3EncryptedPolicy.json
```

Aggiungere utenti o gruppi con DataBrew autorizzazioni

Assegni politiche ai ruoli e ruoli a utenti e gruppi per gestire le autorizzazioni. Per ulteriori informazioni, consulta IAM Identities (users, groups and roles) nella IAM User Guide.

Prima di iniziare, devi avere almeno un utente a cui assegnare le autorizzazioni.

Utilizza la procedura seguente per configurare DataBrew le autorizzazioni per gli utenti che devono lavorare nella DataBrew console o eseguire DataBrew comandi nella CLI.

Per configurare le autorizzazioni DataBrew

1. Crea una chiave di accesso per consentire all'utente di utilizzare AWS CLI for DataBrew e altri strumenti di sviluppo.
2. Abilita Console di gestione AWS l'accesso per consentire all'utente di utilizzare la AWS console.
3. Crea un ruolo per DataBrew utenti o gruppi.
4. Scegli la politica che stai utilizzando. Esegui una delle seguenti operazioni:
   - Se l'hai creata`AwsGlueDataBrewCustomUserPolicy`, selezionala dall'elenco.
   - Per utilizzare la AWS-managed politica, `AwsGlueDataBrewFullAccessPolicy` selezionala dall'elenco.
5. Assegna quella politica al ruolo.
6. Imposta le relazioni di trust per il ruolo in modo che un utente o un gruppo possa assumere il ruolo pertinente.
   - Se non utilizzi i gruppi, affida il ruolo all'utente.
   - Se utilizzi i gruppi, affida il ruolo al gruppo e aggiungi l'utente al gruppo.

Aggiungere un ruolo IAM con autorizzazioni per le risorse di dati

Utilizzi i ruoli IAM per gestire le policy assegnate insieme. Un ruolo IAM può essere utilizzato da qualcuno che ricopre un ruolo particolare, come un DataBrew utente o DataBrew se stesso. Per ulteriori informazioni, consulta Ruoli IAM nella Guida per l'utente di IAM.

Utilizza la seguente procedura per creare un ruolo IAM necessario per consentire ai DataBrew progetti di accedere ai dati.

Per collegare la policy IAM richiesta a un nuovo ruolo IAM per DataBrew

1. Nel riquadro di navigazione, seleziona Ruoli, quindi Crea nuovo ruolo.
2. Per Seleziona il tipo di entità affidabile, scegli il AWS servizio con etichetta.
3. Scegli DataBrewdall'elenco, quindi scegli Avanti: Autorizzazioni.
4. Inserisci **`AwsGlueDataBrewDataResourcePolicy`** nella casella di ricerca (la policy IAM che hai creato in un passaggio precedente). Seleziona la policy e scegli Avanti: Tag.
5. Scegli Prossimo: Rivedi.
6. In Nome ruolo immetti **`AwsGlueDataBrewDataAccessRole`** e quindi seleziona Crea ruolo.

Configurazione Centro identità AWS IAM(IAM Identity Center)

Utilizzando Centro identità AWS IAM(IAM Identity Center), i tuoi utenti possono accedere DataBrew con un semplice URL, senza accedere Console di gestione AWS e senza bisogno di un AWS account.

Per configurare IAM Identity Center

1. Apri la AWS Organizations console e crea un'organizzazione se non ne hai già una. Tutte le funzionalità sono abilitate per impostazione predefinita per questa organizzazione.

   Per ulteriori informazioni, vedere Centro identità AWS IAM Prerequisiti e Creazione e gestione di un'organizzazione.
2. Apri la console Centro identità AWS IAM
3. Scegli la fonte della tua identità.

   Per impostazione predefinita, hai a disposizione uno store IAM Identity Center per una gestione degli utenti semplice e veloce. Facoltativamente, puoi invece connettere un provider di identità esterno o connettere una AWS Managed Microsoft AD directory con il tuo Active Directory locale. In questa guida, utilizziamo lo store IAM Identity Center predefinito.

   Per ulteriori informazioni, consulta Scegli la tua fonte di identità nella Guida Centro identità AWS IAM per l'utente.
4. Crea un set di autorizzazioni per DataBrew l'accesso:

a. Nel riquadro di navigazione di IAM Identity Center, scegli AWS account, quindi scegli Set di autorizzazioni.

b. Nella pagina Crea set di autorizzazioni, scegli Crea un set di autorizzazioni personalizzato.

c. Per Relay state, inserisci`https://console.aws.amazon.com/databrew/home?region=us-east-1#landing`.

   L'immissione di questo campo consente agli utenti di accedere direttamente a DataBrew.

d. Scegli Allega politiche AWS gestite DataBrew, cerca e scegli AwsGlueDataBrewFullAccessPolicy. Questa opzione offre agli utenti tutte le autorizzazioni di cui hanno bisogno. DataBrew Puoi trovare maggiori dettagli in Aggiungere una policy IAM per un utente della console.

e. (Facoltativo) Scegli Crea una politica di autorizzazioni personalizzata e personalizza le autorizzazioni per i tuoi utenti.

5. Nel riquadro di navigazione di IAM Identity Center, scegli Gruppi e scegli Crea gruppo. Inserisci il nome del gruppo e scegli Crea.

6. Aggiungi un utente all'archivio IAM Identity Center:

   a. Nel riquadro di navigazione di IAM Identity Center, scegli Utenti.

   b. Nella schermata Aggiungi utente, inserisci le informazioni richieste e scegli Invia un'e-mail all'utente con le istruzioni per la configurazione della password. L'utente dovrebbe ricevere un'e-mail con i passaggi di configurazione successivi.

   c. Scegli Avanti: Gruppi, scegli il gruppo che desideri e scegli Aggiungi utente.

      Gli utenti dovrebbero ricevere un'e-mail che li invita a utilizzare l'SSO. In questa e-mail, devono scegliere Accetta invito e impostare la password. Possono anche trovare l'URL del portale nell'e-mail. Possono utilizzare questo URL per accedere DataBrew.

7. Assegna ogni utente a un account:

   a. Apri la console IAM Identity Center e, nel riquadro di navigazione, scegli AWS account.

   b. Scegli AWS l'organizzazione e scegli un AWS account.

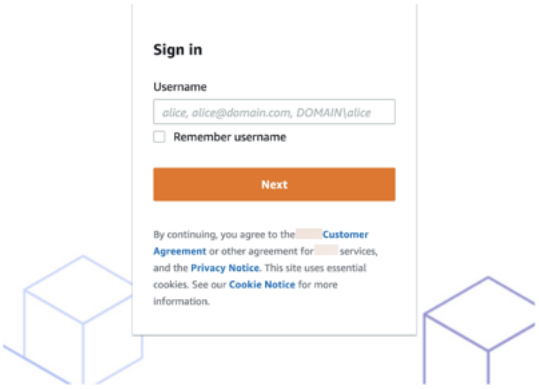
   c. Nella schermata Assegna utenti, scegli la scheda Gruppi e scegli il gruppo che desideri.

   d. Scegliere Next: Permissions sets (Successivo: set di autorizzazioni).

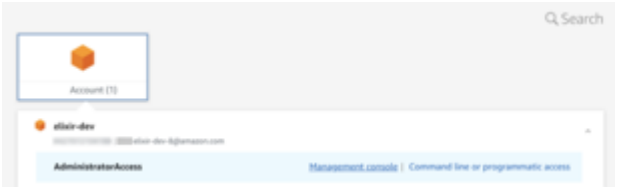
   e. Scegli il set di autorizzazioni per DataBrew e scegli Fine.

Passaggi di accesso per un Center-enabled utente IAM Identity

1. Accedi AWS utilizzando un Center-enabled account IAM Identity.



2. Fai clic su Identità AWS dell'account



3. Fai clic su Console di gestione per il reindirizzamento alla console con un clic. DataBrew

Utilizzo DataBrew come estensione in JupyterLab

> **Warning**
>
> AWS Glue DataBrew JupyterLab il supporto per le estensioni terminerà il 31 dicembre 2024, poiché JupyterLab 3 raggiungerà la fine del supporto. Per ulteriori informazioni, consulta la sezione JupyterLab 3: fine della manutenzione.

Se preferisci preparare i dati in un ambiente Jupyter Notebook, puoi utilizzare tutte le funzionalità di in.AWS Glue DataBrew JupyterLab

JupyterLab è un ambiente di sviluppo interattivo basato sul web per Jupyter Notebook. Nella JupyterLab pagina web locale, puoi aggiungere sezioni per un terminale, una sessione SQL, Python e altro. Dopo aver installato l'AWS Glue DataBrew estensione, puoi aggiungere una sezione per la DataBrew console. Funziona con tutti i notebook o altre estensioni esistenti che già possiedi, direttamente dall'ambiente. JupyterLab

Argomenti

Prerequisiti

Prima di iniziare, configura i seguenti elementi:

- Un AWS account: se non ne hai ancora uno, inizia conConfigurare un nuovo AWS account.
- Un utente AWS Identity and Access Management(IAM) con accesso alle autorizzazioni necessarie per DataBrew : per ulteriori informazioni, consultaAggiungere utenti o gruppi con DataBrew autorizzazioni.
- Un ruolo IAM da utilizzare nelle DataBrew operazioni: puoi utilizzare quello predefinito, se `AwsGlueDataBrewDataAccessRole` configurato. Per configurare ruoli IAM aggiuntivi, consultaAggiungere un ruolo IAM con autorizzazioni per le risorse di dati.
- Un' JupyterLab installazione (versione 2.2.6 o successiva) — Per ulteriori informazioni, consulta i seguenti argomenti nella JupyterLabdocumentazione:
  - JupyterLab prerequisiti
  - JupyterLab installazione: si consiglia di utilizzare`pip install jupyterlab`.
- Un' Node.js installazione (versione 12.0 o successiva).
- Un'installazione AWS Command Line Interface(AWS CLI) — Per ulteriori informazioni, vedereConfigurazione del AWS CLI.
- Un'installazione del proxy AWS Jupyter (`pip install aws-jupyter-proxy`): questa estensione viene utilizzata con un endpoint di AWS servizio per passare in modo sicuro le credenziali.AWSPer ulteriori informazioni, consulta aws-jupyter-proxy on. GitHub

Per verificare di avere i prerequisiti installati, puoi eseguire un test simile al seguente nella riga di comando, come mostrato nell'esempio seguente.

```
echo "
AWS CLI:"
which aws
aws --version
aws configure list
```

```
aws sts get-caller-identity

echo "
Python (current environment):"
which python
python --version

echo "
Node.JS:"
which node
node --version

echo "
Jupyter:"
where jupyter
jupyter --version
jupyter serverextension list
pip3 freeze | grep jupyter
```

L'output dovrebbe essere simile al seguente. Le directory variano in base al sistema operativo e alla configurazione.

```
AWS CLI:
/usr/local/bin/aws
aws-cli/2.1.2 Python/3.7.4 Darwin/19.6.0 exe/x86_64
      Name                    Value             Type    Location
      ----                    -----             ----    --------
   profile                <not set>             None    None
access_key     ****************VXW4 shared-credentials-file
secret_key     ****************MRJN shared-credentials-file
    region                us-east-1      config-file    ~/.aws/config
{
    "UserId": "",
    "Account": "111122223333",
    "Arn": "arn:aws:iam::111122223333:user/user2"
}

Python (current environment):
/usr/local/opt/python /libexec/bin/python
Python 3.8.5

Node.JS:
/usr/local/bin/node
```

```
v15.0.1

Jupyter:
/usr/local/bin/jupyter
jupyter core     : 4.6.3
jupyter-notebook : 6.0.3
qtconsole        : 4.7.5
ipython          : 7.16.1
ipykernel        : 5.3.2
jupyter client   : 6.1.6
jupyter lab      : 2.2.9
nbconvert        : 5.6.1
ipywidgets       : 7.5.1
nbformat         : 5.0.7
traitlets        : 4.3.3

config dir: /usr/local/etc/jupyter
    aws_jupyter_proxy  enabled
    - Validating...
      aws_jupyter_proxy  OK
    jupyterlab  enabled
    - Validating...
      jupyterlab 2.2.9 OK

aws-jupyter-proxy==0.1.0
jupyter-client==6.1.7
jupyter-core==4.7.0
jupyterlab==2.2.9
jupyterlab-pygments==0.1.2
jupyterlab-server==1.2.0
```

Configurazione JupyterLab per l'utilizzo dell'estensione

Dopo l'installazione JupyterLab, è necessario configurarla per proteggere l'accesso ai dati e abilitare le estensioni del server.

Per configurare una password e una crittografia

1. Imposta una password per proteggere i dati che intendi aggiungere nell'estensione. Jupyter fornisce una utility per le password. Esegui il comando seguente e inserisci la tua password preferita al prompt.

```
jupyter notebook password
```

L'output è simile al seguente.

```
Enter password:
Verify password:
[NotebookPasswordApp] Wrote hashed password to /home/ubuntu/.jupyter/
jupyter_notebook_config.json
```

2. Abilita la crittografia sul server Jupyter. Se installi Jupyter sul tuo computer locale e nessuno può accedervi tramite la rete, puoi saltare questo passaggio.

   Per configurare la crittografia con Transport Layer Security (TLS), crea un certificato personalizzato per il tuo ambiente. Per ulteriori informazioni, vedere Utilizzo di Let's Encrypt per proteggere un server nella documentazione di Jupyter.

3. Per iniziare JupyterLab, esegui il comando seguente al prompt dei comandi.

   ```
   jupyter lab
   ```

   Per ulteriori informazioni, consulta Avvio JupyterLab nella JupyterLab documentazione.

4. Mentre JupyterLab è in esecuzione, puoi accedervi da un URL simile al seguente: http://localhost:8888/lab. Se configuri la crittografia, usa `https` invece di`http`. Se hai personalizzato la porta, sostituisci il tuo numero di porta invece di`8888`.

Utilizza la procedura seguente per abilitare le estensioni di terze parti.

Per abilitare le estensioni di terze parti in JupyterLab

1. Nella JupyterLab pagina web, scegli l'icona Extension Manager nel menu a sinistra.
2. Leggi l'avviso sui rischi dell'esecuzione di estensioni di terze parti. Installa solo estensioni di sviluppatori di cui ti fidi.
3. Per abilitare le estensioni di terze parti JupyterLab, scegli Abilita.
4. Segui le istruzioni per ricostruire e ricaricare. JupyterLab

Attivazione dell'estensione per DataBrew JupyterLab

Dopo aver eseguito l'installazione sicura JupyterLab con le estensioni abilitate, installate l' DataBrew estensione DataBrew in modo da poterla eseguire sul vostro notebook.

Per installare le estensioni per DataBrew (console)

1. Per iniziare JupyterLab, esegui il comando seguente al prompt dei comandi.

   ```
   jupyter lab
   ```

2. Nella JupyterLab pagina web, scegli l'icona Extension Manager nel menu a sinistra.
3. Cerca l' DataBrew estensione inserendo "**`brew`**" per Cerca in alto a sinistra.
4. Individua aws_glue_databrew_jupyter nell'elenco, ma non fare clic su di esso. Se fai clic sul nome evidenziato dell'estensione, si apre una nuova finestra del browser con la pagina aws_glue_databrew_jupyter attiva. GitHub
5. Per installare DataBrew l'estensione, scegli una delle seguenti opzioni:

   - Nella riga di comando, esegui`jupyter labextension install aws_glue_databrew_jupyter`.
   - Scegli Installa nella parte inferiore della scheda di estensione, sotto "aws_glue_databrew_jupyter" in caratteri grigi.

   DataBrew l' JupyterLab estensione è compatibile con le versioni 1.2 e 2.x.
6. Per verificare che sia installata, `jupyter labextension list` esegui. L'output dovrebbe essere simile al seguente.

   ```
   JupyterLab v2.2.9
   Known labextensions:
      app dir: /usr/local/share/jupyter/lab  # varies by OS
           aws_glue_databrew_jupyter v1.0.1  enabled  OK
   ```

7. Ricostruisci JupyterLab utilizzando uno dei seguenti metodi:

   - Al prompt dei comandi, esegui. `jupyter lab build`
   - Nella pagina web, scegli Ricostruisci in alto a sinistra.
8. Una volta completata la compilazione, esegui una delle seguenti operazioni:

- Al prompt dei comandi, `jupyter lab` esegui.
- Nella pagina web, scegli Ricarica nel messaggio Build Complete.

9. Nella JupyterLab pagina web, chiudi Extension Manager selezionando la relativa icona nel menu a sinistra.

   Per aprire l'estensione, scegli Launch AWS Glue DataBrew dalla sezione Altro della scheda Launcher. L'estensione utilizza la AWS CLI configurazione corrente per le chiavi di accesso e le impostazioni AWS regionali.

Dopo aver completato la configurazione, puoi utilizzare la AWS Glue DataBrewscheda per interagire DataBrew dall'interno JupyterLab.

Nozioni di base su AWS Glue DataBrew

Puoi usare il seguente tutorial per guidarti nella creazione del tuo primo DataBrew progetto. Carichi un set di dati di esempio, esegui trasformazioni su quel set di dati, crei una ricetta per acquisire tali trasformazioni ed esegui un processo per scrivere i dati trasformati su Amazon S3.

Argomenti

Prerequisiti

Prima di procedere, segui le istruzioni applicabili riportate in. Configurazione AWS Glue DataBrew Quindi continua conFase 1: creazione di un progetto.

Fase 1: creazione di un progetto

In questo passaggio, si utilizza la DataBrew console per iniziare rapidamente con un progetto di esempio.

Come creare un progetto

1. Accedi a Console di gestione AWS e apri la DataBrew console all'indirizzo https://console.aws.amazon.com/databrew/.
2. Assicurati che la tua AWS regione sia selezionata in alto a destra sulla console. DataBrew Per un elenco delle AWS regioni supportate da DataBrew, consulta DataBrew endpoint e quote in.Riferimenti generali di AWS

3. Nel riquadro di navigazione, scegli Progetti, quindi scegli Crea progetto.
4. Nel riquadro dei dettagli del progetto, procedi come segue:
   - Per Nome del progetto, immetterechess-project.
   - Per Ricetta allegata, crea una nuova ricetta. Viene fornito un nome suggerito per la ricetta (chess-project-recipe).
5. Nel riquadro Seleziona un set di dati, scegli File di esempio.
6. Nel riquadro File di esempio, scegli Mosse famose di una partita di scacchi. Questo set di dati contiene informazioni dettagliate su più di 20.000 partite di scacchi.

   Per Dataset name viene fornito un nome suggerito per il set di dati (). chess-games
7. Nel riquadro Autorizzazioni di accesso, scegli. AwsGlueDataBrewDataAccessRole Si tratta di un ruolo collegato al servizio che consente di DataBrew accedere ai bucket Amazon S3 per tuo conto.
8. Scegli Crea progetto e attendi fino al DataBrew termine della preparazione del progetto. La finestra è simile alla seguente.

   I dati visualizzati rappresentano un campione del chess-games set di dati. Per impostazione predefinita, l'esempio è costituito dalle prime 500 righe del set di dati. È possibile modificare l'impostazione di questo progetto in un secondo momento.

   La barra degli strumenti fornisce l'accesso a centinaia di trasformazioni di dati che puoi applicare ai dati.

   Il riquadro delle ricette a destra nella DataBrew console tiene traccia delle trasformazioni applicate finora.

Passaggio 2: riepilogare i dati

In questo passaggio, crei una DataBrew ricetta, un insieme di trasformazioni che possono essere applicate a questo set di dati e ad altri simili. Quando la ricetta è completa, la pubblichi in modo che sia disponibile per l'uso.

Nel gioco degli scacchi, i giocatori possono essere valutati in base alle loro prestazioni contro altri giocatori. (Per ulteriori informazioni, consulta https://en.wikipedia.org/wiki/Chess_rating_system). In questo tutorial, ti concentri solo sulle partite in cui entrambi i giocatori erano di classe A, il che significa che i loro punteggi erano 1800 o più.

Per riepilogare i dati

1. Sulla barra degli strumenti di trasformazione, scegli Filtro, Per condizione, Maggiore o uguale a.
2. Imposta queste opzioni come segue:

   - Colonna sorgente - `white_rating`
   - Condizione del filtro: maggiore o uguale a 1800

   Per vedere come funziona la trasformazione, scegli Anteprima modifiche. Quindi, scegliere Apply (Applica).
3. Ripeti il passaggio precedente, ma questa volta imposta la colonna Sorgente su`black_rating`. Dopo aver applicato le modifiche, i dati di esempio contengono solo le partite in cui i giocatori di ogni fazione (bianchi e neri) erano di classe A o superiore.
4. Riassumi i dati per determinare quante partite sono state vinte da ciascuna squadra. Per fare ciò, nella barra degli strumenti di trasformazione, scegli Raggruppa.
5. Per le proprietà del gruppo, effettuate le seguenti operazioni:

   a. Nella prima riga, scegliete `winner` Nome colonna. Lascia Aggregato impostato su Raggruppa per.
   b. Nella seconda riga, scegli `victory_status` il nome della colonna. Lascia Aggregato impostato su Raggruppa per.
   c. Scegli Aggiungi un'altra colonna.
   d. Nella terza riga, scegli `winner` Nome colonna. Imposta Aggregate to Count.
   e. Per Tipo di gruppo, scegli Raggruppa come nuova tabella. Il riquadro di anteprima mostra come sarà il risultato.
   f. Scegli Fine.
6. Scegli Pubblica per salvare il lavoro, a destra nel riquadro delle ricette.
7. Per Descrizione della versione, inserisci Prima versione della mia ricetta. Quindi scegli Pubblica.

Fase 3: Aggiungere altre trasformazioni

In questo passaggio, aggiungi altre trasformazioni alla tua ricetta e ne pubblichi un'altra versione. Per perfezionare il nostro esempio, utilizziamo l'informazione che non tutte le partite di scacchi portano a un chiaro vincitore; alcune partite vengono giocate in parità.

Per aggiungere altre trasformazioni delle ricette e ripubblicarle

1. Dalla barra degli strumenti di trasformazione, scegli Filtro, Per condizione, Non è per rimuovere le partite che sono state giocate in pareggio.
2. Imposta queste opzioni come segue:

   - Colonna sorgente - `victory_status`
   - Condizione del filtro: non lo è `draw`

   Per aggiungere questa trasformazione alla tua ricetta, scegli Applica.
3. Modifica i dati in `victory_status` modo che siano più significativi. Per fare ciò, dalla barra degli strumenti di trasformazione scegli Pulisci, Sostituisci, Sostituisci valore o modello.
4. Imposta queste opzioni come segue:

   - Colonna sorgente - `victory_status`
   - Specificare i valori da sostituire: valore o modello
   - Valore da sostituire - `mate`
   - Sostituisci con valore - `checkmate`

   Per aggiungere questa trasformazione alla tua ricetta, scegli Applica.
5. Ripeti il passaggio precedente, ma passa `resign` a`other player resigned`.
6. Ripeti il passaggio precedente, ma passa `outoftime` a`time ran out`.
7. Scegli Pubblica per salvare il lavoro, a destra nel riquadro delle ricette.

Passaggio 4: Rivedi DataBrew le tue risorse

Ora che hai lavorato a un progetto di esempio, esamina le DataBrew risorse che hai creato finora.

Per esaminare le tue DataBrew risorse

1. Nel riquadro di navigazione, scegli Datasets.

   Quando hai creato il progetto di esempio, hai DataBrew creato un set di dati per te ()`chess-games`. Il file di dati di origine è archiviato in Amazon S3 ed è in formato Microsoft Excel

()`chess-games.xlsx`. Il file contiene metadati di oltre 20.000 partite di scacchi. Il `chess-games` set di dati fornisce le informazioni DataBrew necessarie per leggere i dati in quel file.

2. Nel riquadro di navigazione, scegli Progetti.

   Dovresti vedere il progetto con cui hai lavorato nei passaggi precedenti (`chess-project`). Ogni progetto richiede un set di dati, in questo caso`chess-games`. Ogni progetto richiede anche una ricetta, in modo da poter aggiungere fasi di trasformazione dei dati man mano che si procede. Quando hai creato questo progetto di esempio, hai DataBrew creato una nuova ricetta (vuota) per te e l'hai allegata al progetto.

3. Nel pannello di navigazione, scegli Recipes e nella colonna Recipe name, scegli chess-project-recipe. Questo ti mostra la ricetta che hai DataBrew creato per il tuo progetto e che hai perfezionato aggiungendo fasi di trasformazione.

4. A sinistra, visualizza le versioni delle ricette che sono state pubblicate. Scegli una di queste per visualizzare la scheda Fasi della ricetta, che mostra i dettagli e i passaggi della ricetta per quella versione.

5. Visualizza la scheda Data lineage, che mostra da dove provengono i dati e come vengono utilizzati. Per ulteriori dettagli, scegli una delle icone nel diagramma.

Fase 5: Creare un profilo dati

Quando lavori su un progetto, DataBrew visualizza statistiche come il numero di righe nell'esempio e la distribuzione di valori univoci in ogni colonna. Queste statistiche, e molte altre, rappresentano un profilo del campione.

Per richiedere un profilo dati, crea ed esegui un job di profilo.

Per profilare un set di dati

1. Nel riquadro di navigazione, scegli Jobs.
2. Nella scheda Profile jobs, scegli Crea job.
3. Per Job name, immetterechess-data-profile.
4. Per Tipo di lavoro, scegli Crea un profilo di lavoro.
5. Nel riquadro di immissione Job, effettuate le seguenti operazioni:

   - Per Run on, scegli Dataset.

- Scegli Seleziona un set di dati per visualizzare un elenco di set di dati disponibili e scegli. `chess-games`

6. Nel riquadro Impostazioni di output Job, effettuate le seguenti operazioni:

   - Per Tipo di file, scegliete JSON (JavaScript Object Notation).
   - Scegli la posizione S3 per visualizzare un elenco di bucket Amazon S3 disponibili e scegli il bucket da utilizzare. Quindi scegli Sfoglia. Nell'elenco delle cartelle`databrew-output`, scegliete e scegliete Seleziona.

7. Nel riquadro Autorizzazioni di accesso, scegli`AwsGlueDataBrewDataAccessRole`. Si tratta di un ruolo collegato al servizio che consente di DataBrew accedere ai bucket Amazon S3 per tuo conto.
8. Scegli Crea ed esegui un processo. DataBrew crea un lavoro con le tue impostazioni, quindi lo esegue.
9. Nel riquadro Cronologia dell'esecuzione del processo, attendi che lo stato del processo cambi da `Running` a`Succeeded`.
10. Per visualizzare il profilo, scegli VISUALIZZA PROFILO:

    Viene visualizzata la finestra DATASETS. Prenditi del tempo per esplorare le seguenti schede:

    - Anteprima del set di dati
    - Panoramica del profilo
    - Statistiche delle colonne
    - Statistiche sulla derivazione dei dati

Fase 6: Trasforma il set di dati

Fino ad ora, hai testato la tua ricetta solo su un campione del set di dati. Ora è il momento di trasformare l'intero set di dati creando un processo di elaborazione delle DataBrew ricette.

Quando il processo viene eseguito, DataBrew applica la ricetta a tutti i dati del set di dati e scrive i dati trasformati in un bucket Amazon S3. I dati trasformati sono separati dal set di dati originale. DataBrew non altera i dati di origine.

Prima di procedere, assicurati di avere un bucket Amazon S3 nel tuo account su cui scrivere. In quel bucket, crea una cartella da cui acquisire l'output del lavoro. DataBrew Per eseguire questi passaggi, utilizzare la procedura seguente.

Per creare un bucket e una cartella S3 per acquisire l'output del lavoro

1. Accedi a Console di gestione AWS e apri la console Amazon S3 all'indirizzo. https://console.aws.amazon.com/databrew/

   Se disponi già di un bucket Amazon S3 e disponi delle autorizzazioni di scrittura per esso, salta il passaggio successivo.
2. Se non disponi di un bucket Amazon S3, scegli Crea bucket. Per Bucket name, inserisci un nome univoco per il tuo nuovo bucket. Seleziona Crea bucket.
3. Dall'elenco dei bucket, scegli quello che desideri utilizzare.
4. Scegliere Create folder (Crea cartella).
5. Per Nome cartella`databrew-output`, immettete e scegliete Crea cartella.

Dopo aver creato un bucket e una cartella Amazon S3 per contenere il lavoro, esegui il processo utilizzando la procedura seguente.

Per creare ed eseguire un processo di creazione di ricette

1. Nel riquadro di navigazione, scegli Jobs.
2. Nella scheda Recipe jobs, scegli Crea lavoro.
3. Per Job name, immetterechess-winner-summary.
4. Per Tipo di lavoro, scegli Crea un lavoro di ricetta.
5. Nel riquadro di immissione Job, effettuate le seguenti operazioni:

   - Per Run on, scegli Dataset.
   - Scegli Seleziona un set di dati per visualizzare un elenco di set di dati disponibili e scegli. `chess-games`
   - Scegliete Seleziona una ricetta per visualizzare un elenco di ricette disponibili e scegliete. `chess-project-recipe`
6. Nel riquadro Impostazioni di output Job, effettuate le seguenti operazioni:

   - Tipo di file: scegli CSV (valori separati da virgole).

- Posizione S3: scegli questo campo per visualizzare un elenco di bucket Amazon S3 disponibili e scegli il bucket da utilizzare. Quindi scegli Sfoglia. Nell'elenco delle cartelle`databrew-output`, scegliete e scegliete Seleziona.

7. Nel riquadro Autorizzazioni di accesso, scegli`AwsGlueDataBrewDataAccessRole`. Questo ruolo collegato al servizio consente di DataBrew accedere ai bucket Amazon S3 per tuo conto.
8. Scegli Crea ed esegui un processo. DataBrew crea un lavoro con le tue impostazioni, quindi lo esegue.
9. Nel riquadro Cronologia dell'esecuzione del processo, attendi che lo stato del processo cambi da `Running` a`Succeeded`.
10. Scegli Output per accedere alla console Amazon S3. Scegli il tuo bucket S3, quindi scegli la `databrew-output` cartella per accedere all'output del lavoro.
11. (Facoltativo) Scegli Scarica per scaricare il file e visualizzarne il contenuto.

Passaggio 7: (Facoltativo) Pulizia

La procedura dettagliata è completa. Puoi continuare a utilizzare DataBrew le risorse Amazon S3 che hai creato o eliminarle.

Per eliminare le risorse

1. Apri la DataBrew console all'indirizzo e https://console.aws.amazon.com/databrew/, nel pannello di navigazione, scegli Progetti.
2. Scegli il tuo progetto (progetto di esempio). In Actions (Azioni), scegliere Delete (Elimina).
3. Nel riquadro Elimina progetto di esempio, scegli Elimina ricetta allegata. Scegli Elimina. Il progetto, insieme alla ricetta e ai lavori, verrà eliminato.
4. Nel riquadro di navigazione, scegli Datasets.
5. Scegli il tuo set di dati (**`chess-games`**) e per Azioni, scegli Elimina.
6. Apri la console Amazon S3 all'indirizzo. https://console.aws.amazon.com/s3/ Elimina la `databrew-output` cartella e il suo contenuto.

   (Facoltativo) Se sei sicuro di non aver più bisogno del tuo bucket Amazon S3, puoi eliminarlo.

Connessione ai dati con AWS Glue DataBrew

In AWS Glue DataBrew, un set di dati rappresenta dati caricati da un file o archiviati altrove. Ad esempio, i dati possono essere archiviati in Amazon S3, in un'origine dati JDBC supportata o in un catalogo dati.AWS Glue Se non stai caricando un file direttamente su DataBrew, il set di dati contiene anche dettagli su come DataBrew connetterti ai dati.

Quando crei il set di dati (ad esempio,`inventory-dataset`), inserisci i dettagli di connessione solo una volta. Da quel momento, DataBrew puoi accedere ai dati sottostanti per te. Con questo approccio, puoi creare progetti e sviluppare trasformazioni per i tuoi dati, senza doverti preoccupare dei dettagli di connessione o dei formati dei file.

Argomenti

Tipi di file supportati per le fonti di dati

I seguenti requisiti di file si applicano ai file archiviati in Amazon S3 e ai file caricati da un'unità locale. DataBrew supporta i seguenti formati di file: valori separati da virgole (CSV), Microsoft Excel, JSON, ORC e Parquet. È possibile utilizzare file con un'estensione non standard o senza estensione se il file è di uno dei tipi supportati.

Se DataBrew non riesci a dedurre il tipo di file, assicurati di selezionare tu stesso il tipo di file corretto (CSV, Excel, JSON, ORC o Parquet). I file compressi CSV, JSON, ORC e Parquet sono supportati, ma i file CSV e JSON devono includere il codec di compressione come estensione del file. Se si importa una cartella, tutti i file in essa contenuti devono essere dello stesso tipo.

I formati di file e gli algoritmi di compressione supportati sono illustrati nella tabella seguente.

> **Note**
>
> I file CSV, Excel e JSON devono essere codificati con Unicode (). UTF-8

Format (Formato)	Estensione del file (opzionale)	Estensioni per file compressi (obbligatorie)

Comma-separated valori	`.csv`	`.gz`
		`.snappy`
		`.lz4`
		`.bz2`
		`.deflate`

Cartella di lavoro Microsoft Excel	`.xlsx`	Nessun supporto per la compressione

JSON (documento JSON e righe JSON)	`.json, .jsonl`	`.gz`
		`.snappy`
		`.lz4`
		`.bz2`
		`.deflate`

Apache ORC	`.orc`	`.zlib` `.snappy`

Apache Parquet	`.parquet`	`.gz`
		`.snappy`
		`.lz4`

Connessioni supportate per sorgenti e uscite di dati

Puoi connetterti alle seguenti fonti di dati per i lavori di elaborazione delle DataBrew ricette. Queste includono qualsiasi fonte di dati che non sia un file su cui stai caricando direttamente. DataBrew L'origine dati che stai utilizzando potrebbe essere chiamata database, data warehouse o qualcos'altro. Ci riferiamo a tutti i fornitori di dati come fonti di dati o connessioni.

È possibile creare un set di dati utilizzando una delle seguenti fonti di dati.

Puoi anche utilizzare database Amazon S3 o JDBC supportati da Amazon RDS per l'output dei processi di elaborazione delle ricette.AWS Glue Data Catalog DataBrew Amazon AppFlow e AWS Data Exchange non sono supportati gli archivi dati per l'output dei processi di DataBrew elaborazione delle ricette.

- Amazon S3

  Puoi usare S3 per archiviare e proteggere qualsiasi quantità di dati. Per creare un set di dati, specifichi un URL S3 da cui DataBrew accedere a un file di dati, ad esempio: `s3://your-bucket-name/inventory-data.csv`

  DataBrew può anche leggere tutti i file in una cartella S3, il che significa che puoi creare un set di dati che si estende su più file. Per fare ciò, specifica un URL S3 in questo modulo:. `s3://your-bucket-name/your-folder-name/`

  DataBrew supporta solo le seguenti classi di storage Amazon S3: Standard, Reduced Redundancy e S3 One Standard-IA. Zone-IA DataBrew ignora i file con altre classi di storage. DataBrew ignora anche i file vuoti (file contenenti 0 byte). Per ulteriori informazioni sulle classi di storage Amazon S3, consulta Using Amazon S3 Storage Classes nella Amazon S3 Console User Guide.

- AWS Glue Data Catalog

  Puoi utilizzare il Data Catalog per definire riferimenti ai dati archiviati nel cloud.AWS Con il Data Catalog, puoi creare connessioni a singole tabelle nei seguenti servizi:

  - Catalogo dati Amazon S3
  - Catalogo dati Amazon Redshift
  - Catalogo dati Amazon RDS
  - AWS Glue

DataBrew può anche leggere tutti i file in una cartella Amazon S3, il che significa che puoi creare un set di dati che si estende su più file. A tale scopo, specifica un URL Amazon S3 in questo modulo: `s3://your-bucket-name/your-folder-name/`

Per essere utilizzate con DataBrew, le tabelle Amazon S3 definite in AWS Glue Data Catalog, devono avere una proprietà di tabella aggiunta chiamata `aclassification`, che identifica il formato dei dati come `csv` `jsonparquet`, o e as. `typeOfData` `file` Se la proprietà della tabella non è stata aggiunta al momento della creazione della tabella, puoi aggiungerla utilizzando la AWS Glue console.

DataBrew supporta solo le classi di storage Amazon S3 Standard, Reduced Redundancy e S3 One Standard-IA. Zone-IA DataBrew ignora i file con altre classi di storage. DataBrew ignora anche i file vuoti (file contenenti 0 byte). Per ulteriori informazioni sulle classi di storage Amazon S3, consulta Using Amazon S3 Storage Classes nella Amazon S3 Console User Guide.

DataBrew può accedere alle tabelle AWS Glue Data Catalog S3 anche da altri account se viene creata una politica delle risorse appropriata. Puoi creare una policy nella AWS Glue console nella scheda Impostazioni in Data Catalog. Di seguito è riportato un esempio di politica specifica per un singolo Regione AWS.

> **Warning**
>
> Questa è una politica di risorse altamente permissiva che garantisce l'accesso `*$ACCOUNT_TO*` illimitato al Data Catalog di. `*$ACCOUNT_FROM*` Nella maggior parte dei casi, si consiglia di limitare la politica delle risorse a cataloghi o tabelle specifici. Per ulteriori informazioni, consulta le politiche relative alle AWS Glue risorse per il controllo degli accessi nella Guida per gli AWS Glue sviluppatori.

In alcuni casi, potresti voler creare un progetto o eseguire un job utilizzando una tabella AWS Glue Data Catalog S3 `*$ACCOUNT_FROM*` che punti a una posizione S3 anch'essa presente in S3.AWS Glue DataBrew`*$ACCOUNT_TO*` `*$ACCOUNT_FROM*` In questi casi, il ruolo IAM utilizzato durante la creazione del progetto e del job in `*$ACCOUNT_TO*` deve avere l'autorizzazione a elencare e ottenere oggetti in quella posizione S3 da. `*$ACCOUNT_FROM*` Per ulteriori informazioni, consulta Garantire l'accesso su più account nella Developer Guide.AWS Glue

- Dati connessi tramite driver JDBC

È possibile creare un set di dati connettendosi ai dati con un driver JDBC supportato. Per ulteriori informazioni, consulta Utilizzo dei driver con AWS Glue DataBrew.

DataBrew supporta ufficialmente le seguenti fonti di dati utilizzando Java Database Connectivity (JDBC):

- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL
- Amazon Redshift
- Connettore Snowflake per Spark

Le fonti di dati possono essere posizionate ovunque ci si possa connettere ad esse. DataBrew Questo elenco include solo le connessioni JDBC che abbiamo testato e che quindi possiamo supportare.

Le origini dati Amazon Redshift e Snowflake Connector for Spark possono essere collegate in uno dei seguenti modi:

- Con un nome di tabella.
- Con una query SQL che si estende su più tabelle e operazioni.

Le query SQL vengono eseguite all'avvio di un progetto o all'esecuzione di un processo.

Per connetterti a dati che richiedono un driver JDBC non in elenco, assicurati che il driver sia compatibile con JDK 8. Per utilizzare il driver, archivialo in S3 in un bucket a cui puoi accedervi con il tuo ruolo IAM per. DataBrew Quindi indirizza il set di dati verso il file del driver. Per ulteriori informazioni, consulta Utilizzo dei driver con AWS Glue DataBrew.

Query di esempio per un SQL-based set di dati:

```
SELECT
    *
FROM
    public.customer as c
JOIN
    public.customer_address as ca on c.current_address=ca.current_address
WHERE
    ca.address_id>0 AND ca.address_id<10001 ORDER BY ca.address_id
```

Limitazioni di Custom SQL

Se utilizzi una connessione JDBC per accedere ai dati di un DataBrew set di dati, tieni presente quanto segue:

- AWS Glue DataBrew non convalida l'SQL personalizzato fornito come parte della creazione del set di dati. La query SQL verrà eseguita all'avvio di un progetto o dell'esecuzione di un processo. DataBrew prende la query fornita e la passa al motore di database utilizzando i driver JDBC predefiniti o forniti.
- Un set di dati creato con una query non valida avrà esito negativo quando viene utilizzato in un progetto o in un lavoro. Convalida la query prima di creare il set di dati.
- La funzionalità Validate SQL è disponibile solo per le fonti di Redshift-based dati Amazon.
- Se desideri utilizzare un set di dati in un progetto, limita il tempo di esecuzione delle query SQL a meno di tre minuti per evitare un timeout durante il caricamento del progetto. Controllate il runtime della query prima di creare un progetto.

- Amazon AppFlow

  Con Amazon AppFlow, puoi trasferire dati in Amazon S3 da applicazioni di terze parti (Software-as-a-Service SaaS) come Salesforce, Zendesk, Slack e. ServiceNow È quindi possibile utilizzare i dati per creare un set di dati. DataBrew

  In Amazon AppFlow, crei una connessione e un flusso per trasferire dati tra la tua applicazione di terze parti e un'applicazione di destinazione. Quando usi Amazon AppFlow con DataBrew, assicurati che l'applicazione di AppFlow destinazione Amazon sia Amazon S3. Le applicazioni di AppFlow destinazione Amazon diverse da Amazon S3 non vengono visualizzate nella DataBrew console. Per ulteriori informazioni sul trasferimento di dati da un'applicazione di terze parti e sulla creazione di AppFlow connessioni e flussi Amazon, consulta la AppFlow documentazione di Amazon.

  Quando scegli Connect new dataset nella scheda Datasets di DataBrew e fai clic su Amazon AppFlow, vedi tutti i flussi in Amazon AppFlow configurati con Amazon S3 come applicazione di destinazione. Per utilizzare i dati di un flusso per il tuo set di dati, scegli quel flusso.

  Scegliendo Crea flusso, Gestisci flussi e Visualizza dettagli per Amazon AppFlow nella DataBrew console, si apre la AppFlow console Amazon in modo da poter eseguire tali attività.

Dopo aver creato un set di dati da Amazon AppFlow, puoi eseguire il flusso e visualizzare i dettagli dell'ultima esecuzione del flusso quando visualizzi i dettagli del set di dati o i dettagli del processo. Quando esegui il flusso DataBrew, il set di dati viene aggiornato in S3 ed è pronto per essere utilizzato in. DataBrew

Quando selezioni un AppFlow flusso Amazon nella DataBrew console per creare un set di dati, possono verificarsi le seguenti situazioni:

- I dati non sono stati aggregati: se il trigger del flusso è Eseguito su richiesta o viene eseguito in base alla pianificazione con trasferimento completo dei dati, assicurati di aggregare i dati per il flusso prima di utilizzarlo per creare un set di dati. DataBrew L'aggregazione del flusso combina tutti i record del flusso in un unico file. I flussi con il tipo di trigger Run on schedule con trasferimento incrementale di dati o Run on event non richiedono l'aggregazione. Per aggregare i dati in Amazon AppFlow, scegli Modifica configurazione del flusso > Dettagli di destinazione > Impostazioni aggiuntive > Preferenza trasferimento dati.
- Il flusso non è stato eseguito: se lo stato di esecuzione di un flusso è vuoto, significa che si tratta di una delle seguenti situazioni:
  - Se il trigger per l'esecuzione del flusso è Esegui su richiesta, il flusso non è ancora stato eseguito.
  - Se il trigger per l'esecuzione del flusso è Run on event, l'evento di attivazione non si è ancora verificato.
  - Se il fattore scatenante per l'esecuzione del flusso è Esegui in base alla pianificazione, non si è ancora verificata un'esecuzione pianificata.

  Prima di creare un set di dati con un flusso, scegli Esegui flusso per quel flusso.

  Per ulteriori informazioni, consulta Amazon AppFlow flows nella Amazon AppFlow User Guide.
- AWS Data Exchange

  Puoi scegliere tra centinaia di fonti di dati di terze parti disponibili in AWS Data Exchange. Abbonandoti a queste fonti di dati, ottieni la versione più aggiornata dei dati.

  Per creare un set di dati, specifichi il nome di un prodotto di AWS Data Exchange dati a cui sei abbonato e che hai il diritto di utilizzare.

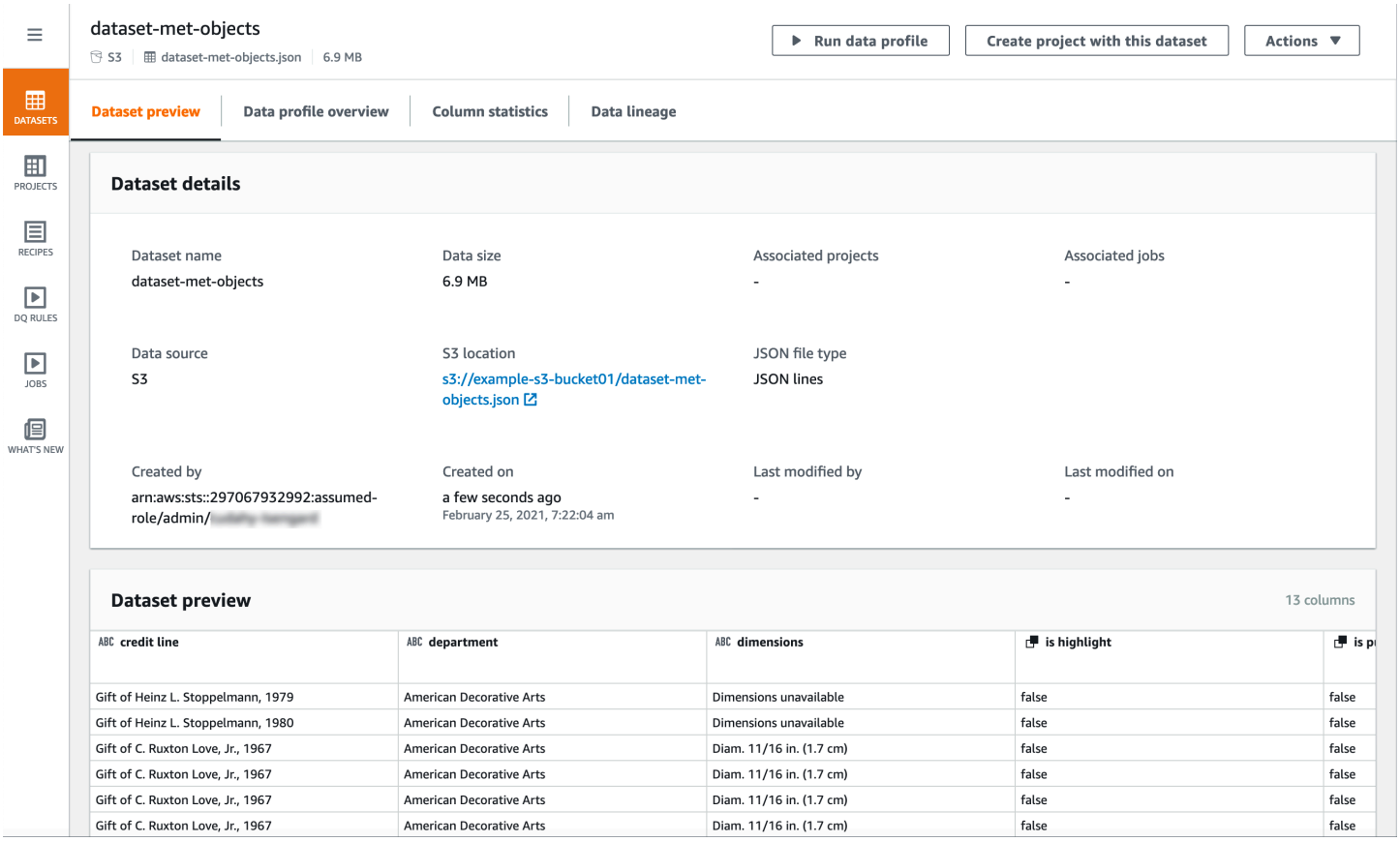
Utilizzo dei set di dati in AWS Glue DataBrew

Per visualizzare un elenco dei tuoi set di dati nella DataBrew console, scegli DATASET a sinistra. Nella pagina dei set di dati, puoi visualizzare informazioni dettagliate per ogni set di dati facendo clic sul suo nome o scegliendo Azioni, Modifica dal relativo menu contestuale.

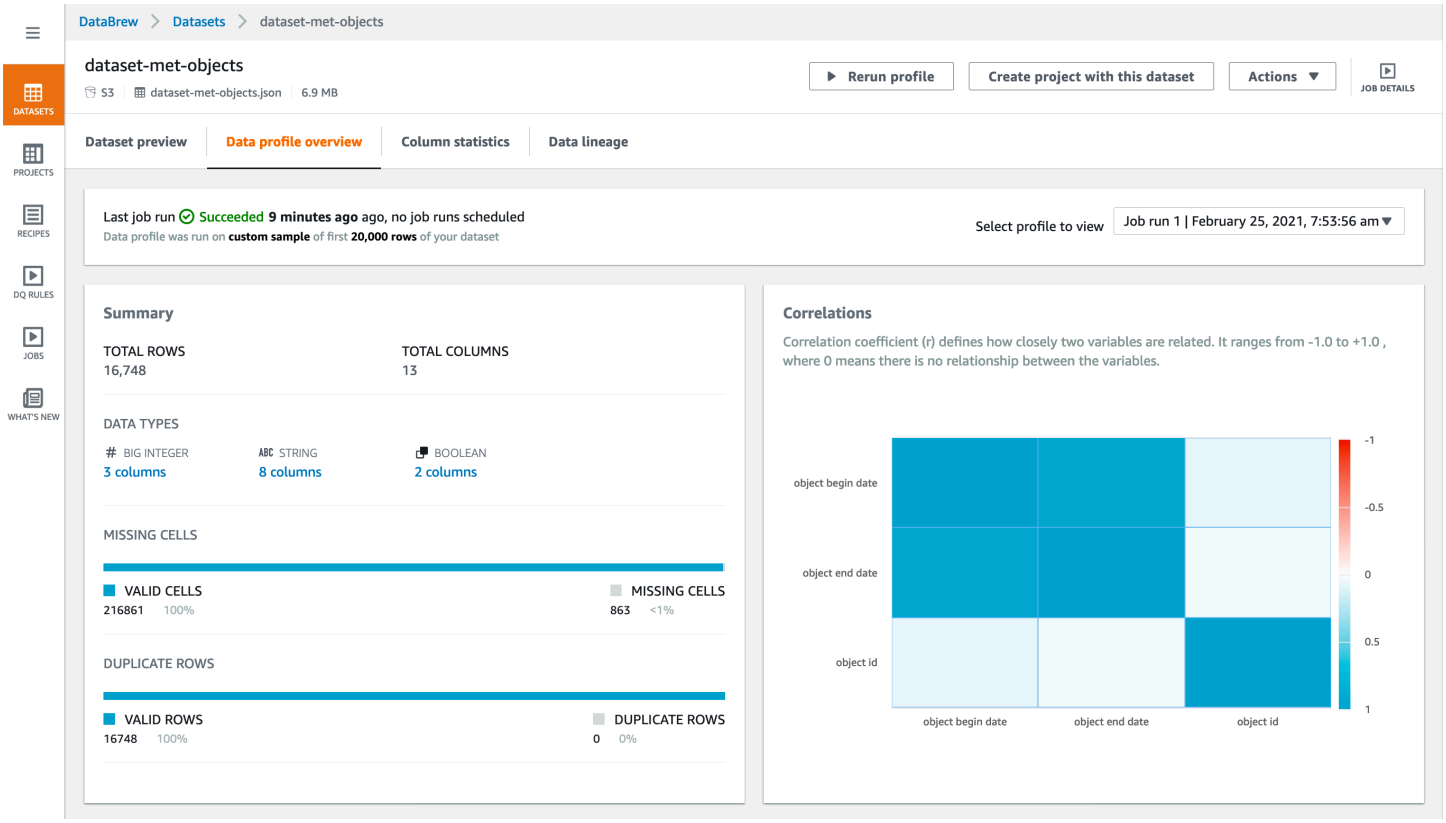
Per creare un nuovo set di dati, scegli DATASET, Connect new dataset. Diverse fonti di dati hanno parametri di connessione diversi e tu li inserisci in modo che DataBrew possa connetterti. Quando salvi la connessione e scegli Crea set di dati, DataBrew si connette ai dati e inizia a caricare i dati. Per ulteriori informazioni, consulta Connessione ai tuoi dati.

La pagina del set di dati contiene i seguenti elementi per aiutarti a esplorare i tuoi dati.

Anteprima del set di dati: in questa scheda, puoi trovare le informazioni di connessione per il set di dati e una panoramica della struttura generale del set di dati, come illustrato di seguito.

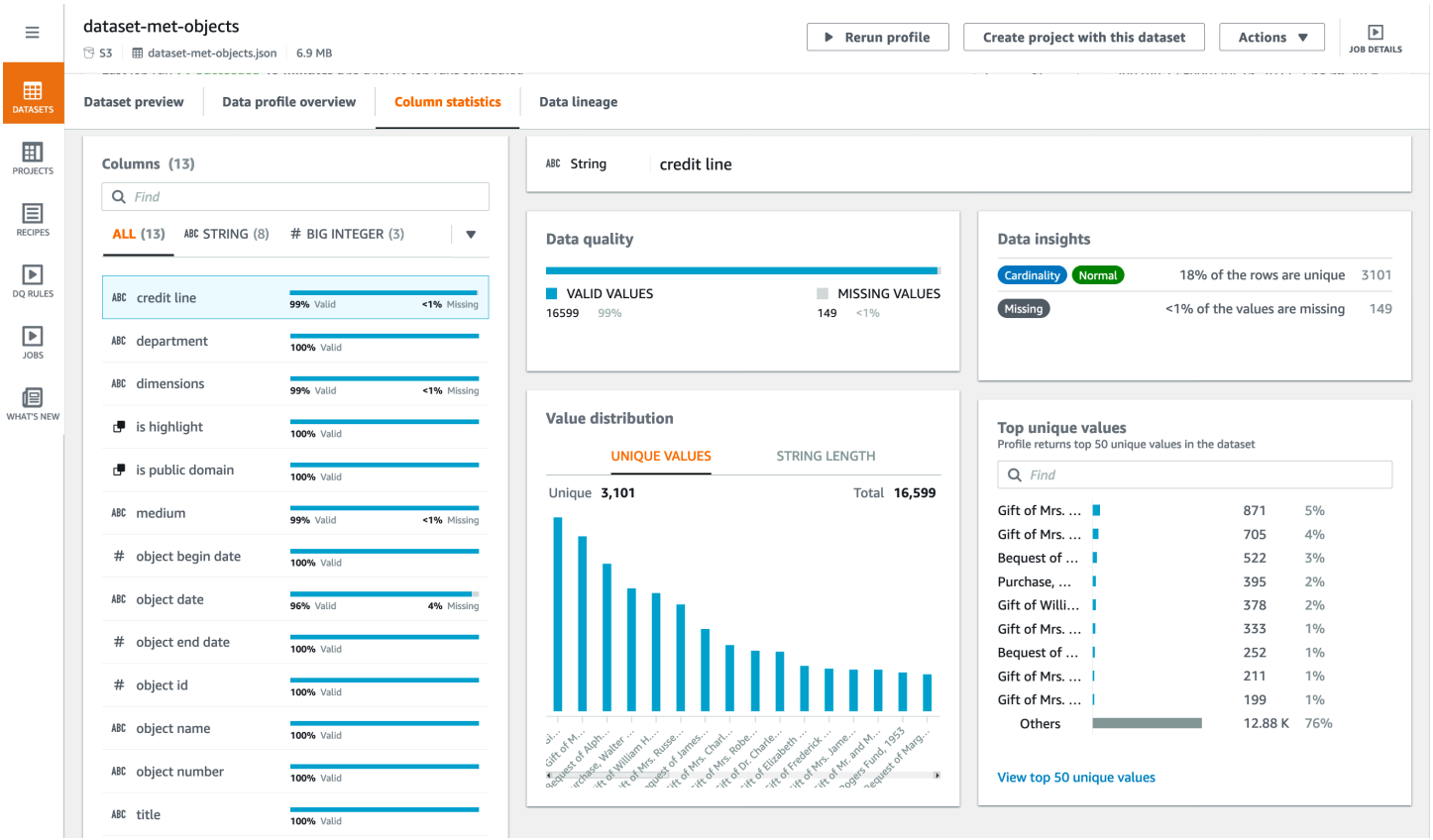


Panoramica del profilo di dati: in questa scheda, puoi trovare un profilo grafico dei dati statistici e volumetrici per il tuo set di dati, come mostrato di seguito.

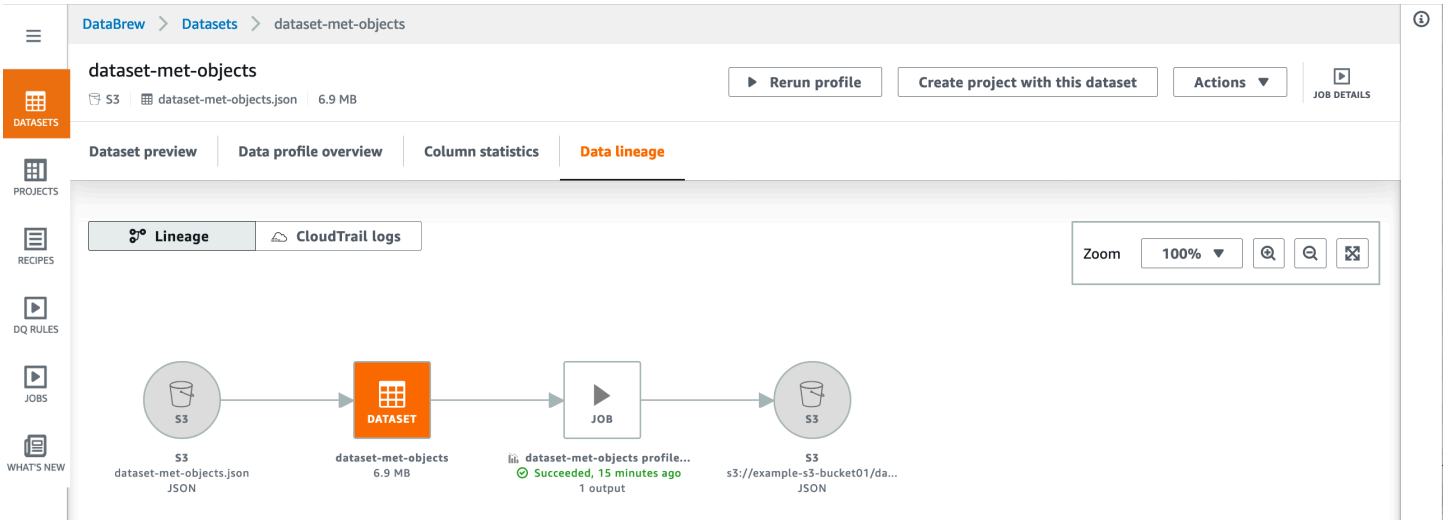


> **Note**
>
> Per creare un profilo di dati, esegui un processo di DataBrew profilazione sul tuo set di dati. Per informazioni su come eseguire questa attività, consultare Fase 5: Creare un profilo dati.

Statistiche delle colonne: in questa scheda, puoi trovare statistiche dettagliate su ogni colonna del tuo set di dati, come mostrato di seguito.



Linea dei dati: questa scheda mostra una rappresentazione grafica di come il set di dati è stato creato e di come viene utilizzato DataBrew, come illustrato di seguito.



Argomenti

- Eliminazione di un set di dati

Eliminazione di un set di dati

Se non hai più bisogno di un set di dati, puoi eliminarlo. L'eliminazione di un set di dati non influisce in alcun modo sulla fonte di dati sottostante. Rimuove semplicemente le informazioni DataBrew utilizzate per accedere alla fonte di dati.

Non puoi eliminare un set di dati se altre DataBrew risorse si basano su di esso. Ad esempio, se attualmente disponi di un DataBrew progetto che utilizza il set di dati, elimina il progetto prima di eliminare il set di dati.

Per eliminare un set di dati, scegli Dataset dal riquadro di navigazione. Scegli il set di dati che desideri eliminare, quindi per Azioni, scegli Elimina.

Connessione ai tuoi dati

Per ulteriori informazioni sulla connessione alle seguenti fonti di dati, scegli la sezione che ti riguarda.

- AWS Glue Data Catalog— È possibile utilizzare il Data Catalog per definire riferimenti agli oggetti di dati archiviati nel AWS Cloud, inclusi i seguenti servizi:
  - Amazon Redshift
  - Aurora MySQL
  - Aurora PostgreSQL
  - Amazon RDS per MySQL
  - Amazon RDS per PostgreSQL

  DataBrew riconosce tutte le autorizzazioni di Lake Formation che sono state applicate alle risorse del Data Catalog, in modo che DataBrew gli utenti possano accedere a tali risorse solo se autorizzati.

  Per creare un set di dati, specificate un nome di database Data Catalog e un nome di tabella. DataBrew si occupa degli altri dettagli di connessione.
- AWS Scambio di dati: puoi scegliere tra centinaia di fonti di dati di terze parti disponibili in AWS Data Exchange. Abbonandoti a queste fonti di dati, avrai sempre la versione più aggiornata dei dati.

  Per creare un set di dati, specifichi il nome di un prodotto di dati Data Exchange a cui sei abbonato o che hai il diritto di utilizzare.

- Connessioni con driver JDBC: puoi creare un set di dati connettendoti DataBrew a una fonte di dati. JDBC-compatible DataBrew supporta la connessione alle seguenti fonti tramite JDBC:
  - Amazon Redshift
  - Microsoft SQL Server
  - MySQL
  - Oracle
  - PostgreSQL
  - Snowflake

Argomenti

Utilizzo dei driver con AWS Glue DataBrew

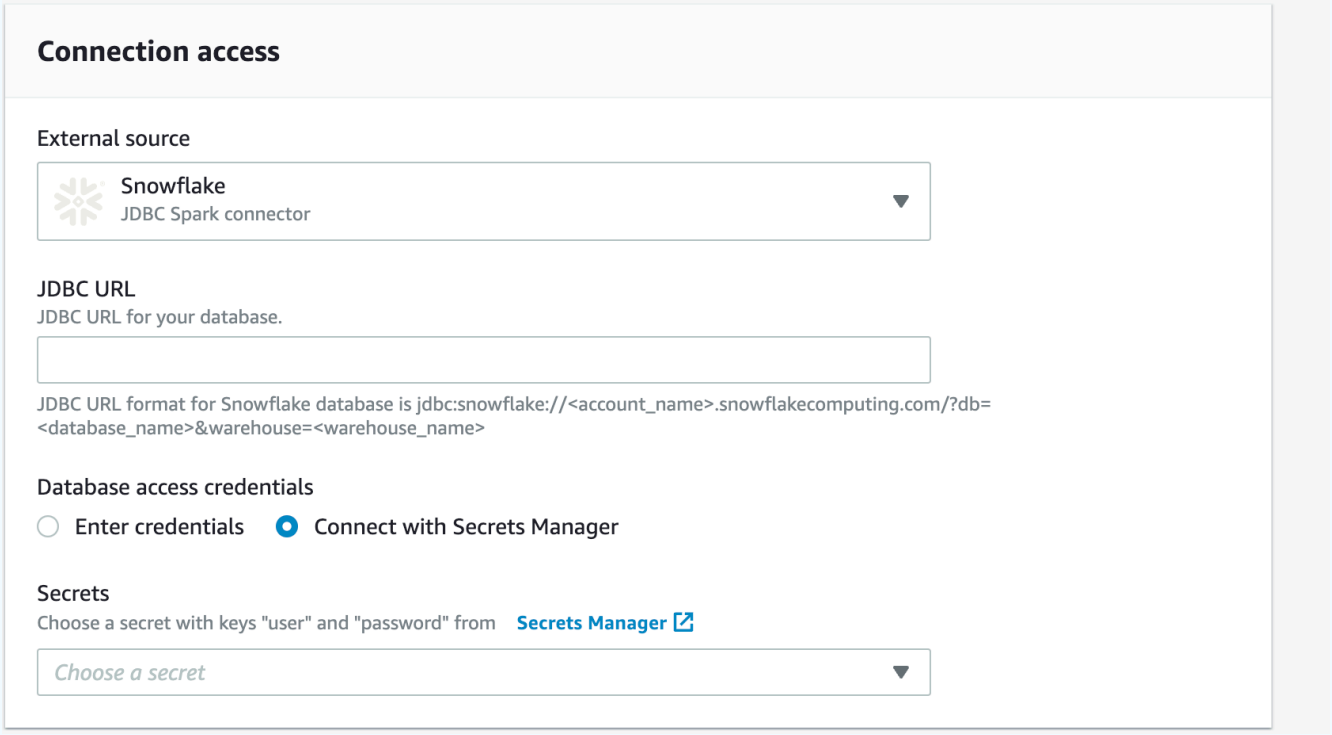
Un driver di database è un file o un URL che implementa un protocollo di connessione al database, ad esempio Java Database Connectivity (JDBC). Il driver funge da adattatore o traduttore tra uno specifico sistema di gestione di database (DBMS) e un altro sistema.

In questo caso, consente di connettersi AWS Glue DataBrew ai dati. Quindi puoi accedere a un oggetto del database, come una tabella o una vista, da un'origine dati supportata. L'origine dati che stai utilizzando potrebbe essere chiamata database, data warehouse o qualcos'altro. Tuttavia, ai fini di questa documentazione, ci riferiamo a tutti i fornitori di dati come fonti di dati o connessioni.

Per utilizzare un driver JDBC o un file jar, scarica il file o i file necessari e inseriscili in un bucket S3. Il ruolo IAM che usi per accedere ai dati deve disporre delle autorizzazioni di lettura per entrambi i file del driver.

> **Note**
>
> With AWS Glue4.0, la connessione a Snowflake come fonte di dati è supportata in modo nativo. Non è necessario fornire file personalizzati. `jar` In AWS Glue DataBrew, scegli Snowflake come connessione di origine esterna e fornisci l'URL della tua istanza Snowflake. L'URL utilizzerà un nome host nel modulo `https://account_identifier.snowflakecomputing.com`.

Fornisci le credenziali di accesso ai dati, il nome del database Snowflake e il nome dello schema Snowflake. Inoltre, se l'utente Snowflake non dispone di un set di warehouse predefinito, sarà necessario fornire un nome di warehouse.
Le connessioni Snowflake utilizzano un AWS Secrets Manager segreto per fornire informazioni sulle credenziali. Il tuo progetto e i tuoi ruoli lavorativi devono essere autorizzati a leggere questo segreto.



Per utilizzare i driver con DataBrew

1. Scopri in quale versione della fonte di dati ti trovi utilizzando il metodo fornito dal prodotto.
2. Trova la versione più recente dei connettori e dei driver richiesti. È possibile trovare queste informazioni sul sito Web dei fornitori di dati.
3. Scarica la versione richiesta dei file JDBC. Questi sono normalmente archiviati come file Java Archives (.JAR).
4. Carica i driver dalla console nel tuo bucket S3 o fornisci il percorso S3 ai tuoi file.JAR.
5. Inserisci i dettagli di connessione di base, ad esempio classe, istanza e così via.
6. Inserisci tutte le informazioni di configurazione aggiuntive necessarie all'origine dati, ad esempio informazioni sul cloud privato virtuale (VPC).

Driver JDBC supportati

Prodotto	Versione di supportata	Istruzioni e download dei driver	Query SQL supportate
Microsoft SQL Server	v6.x o versione successiva	Driver Microsoft JDBC per SQL Server	Non supportata
MySQL	v5.1 o versione successiva	Connettori MySQL	Non supportata
Oracle	v11.2 o versioni successive	Download Oracle JDBC	Non supportata
PostgreSQL	v4.2.x o versione successiva	Driver JDBC PostgreSQL	Non supportata
Amazon Redshift	v4.1 o versioni successive	Connessione ad Amazon Redshift con JDBC	Supportata
Snowflake	Per vedere la tua versione di	Per connetterti a Snowflake hai bisogno di entrambi i seguenti elementi: • Driver JDBC Snowflake • Connettore Snowflake per Spark	Supportata

Prodotto	Versione di supportata	Istruzioni e download dei driver	Query SQL supportate
	Snowflake, usa CURRENT ERSION come descritto nella documentazione di Snowflake.		

Per connetterti a database o data warehouse che richiedono una versione del driver diversa da quella supportata DataBrew nativamente, puoi fornire un driver JDBC a tua scelta. Il driver deve essere compatibile con JDK 8 o Java 8. Per istruzioni su come trovare la versione più recente del driver per il database, consulta. Utilizzo dei driver con AWS Glue DataBrew

Connessione ai dati in un file di testo con DataBrew

È possibile configurare le seguenti opzioni di formato per i file di input DataBrew supportati:

- Comma-separated file di valore (CSV)
  - delimitatori

    Il delimitatore predefinito è una virgola per i file.csv. Se il file utilizza un delimitatore diverso, scegli il delimitatore per il delimitatore CSV nella sezione Configurazioni aggiuntive quando crei il set di dati. I seguenti delimitatori sono supportati per i file.csv:
    - Virgola (,)
    - Due punti (:)
    - Semi-colon (;)
    - Barra verticale (|)

    - Tabulazione (\t)
    - Cursore (^)
    - Barra rovesciata (\)
    - Spazio
  - Valori dell'intestazione delle colonne

    Il file CSV può includere una riga di intestazione come prima riga del file. In caso contrario, DataBrew crea automaticamente una riga di intestazione.

    - Se il file CSV include una riga di intestazione, scegli Tratta la prima riga come intestazione. In tal caso, la prima riga del file CSV viene considerata come contenente i valori dell'intestazione della colonna.
    - Se il file CSV non include una riga di intestazione, scegli Aggiungi intestazione predefinita. In tal caso, DataBrew crea una riga di intestazione per il file e non considera la prima riga di dati come contenente valori di intestazione. Le intestazioni DataBrew create sono costituite da un carattere di sottolineatura e da un numero per ogni colonna del file, nel formato `Column_1` e così `Column_2` via`Column_3`.
- File JSON

  DataBrew supporta due formati per i file JSON, JSON Lines e il documento JSON. I file JSON Lines contengono una riga per riga. Nei file di documento JSON, tutte le righe sono contenute in un'unica struttura JSON o in un array. Puoi specificare il tipo di file JSON nella sezione Configurazioni aggiuntive quando crei un set di dati JSON. Il formato predefinito è JSON Lines.
- File Excel

  Quanto segue si applica ai fogli Excel in DataBrew:

  - Caricamento di fogli Excel

    Per impostazione predefinita, DataBrew carica il primo foglio del file Excel. Tuttavia, è possibile specificare un numero di foglio o un nome di foglio diverso nella sezione Configurazioni aggiuntive quando si crea un set di dati Excel.
  - Valori di intestazione delle colonne

    I tuoi fogli Excel possono includere una riga di intestazione come prima riga del file, ma in caso contrario, DataBrew creerà automaticamente una riga di intestazione.

- Se i tuoi fogli Excel includono una riga di intestazione, scegli Considera la prima riga come intestazione. In tal caso, la prima riga dei fogli Excel viene considerata come contenente i valori dell'intestazione della colonna.
- Se il file Excel non include una riga di intestazione, scegli Aggiungi intestazione predefinita. In questo modo, specifichi che DataBrew deve creare una riga di intestazione per il file e non trattare la prima riga di dati come contenente valori di intestazione. Le intestazioni DataBrew create sono costituite da un carattere di sottolineatura e da un numero per ogni colonna del file, nel formato `Column_1` e così `Column_2` via`Column_3`.

Connessione di dati in più file in Amazon S3

Con la DataBrew console, puoi navigare tra i bucket e le cartelle di Amazon S3 e scegliere un file per il tuo set di dati. Tuttavia, non è necessario che un set di dati sia limitato a un solo file.

Supponiamo di avere un bucket S3 denominato `my-databrew-bucket` che contiene una cartella denominata. `databrew-input` Supponiamo che in quella cartella siano presenti diversi file JSON, tutti con lo stesso formato di file e la stessa estensione. `.json` Sulla console, è possibile specificare un URL di origine di. `s3://my-databrew-bucket/databrew-input/` Sulla DataBrew console, puoi quindi scegliere questa cartella. Il set di dati è composto da tutti i file JSON in quella cartella.

DataBrew può elaborare tutti i file in una cartella S3, ma solo se sono soddisfatte le seguenti condizioni:

- Tutti i file nella cartella hanno lo stesso formato.
- Tutti i file della cartella hanno la stessa estensione.

Per ulteriori informazioni sui formati di file e sulle estensioni supportati, consultaDataBrew input formats.

Schemi quando si utilizzano più file come set di dati

Quando si utilizzano più file come DataBrew set di dati, gli schemi devono essere gli stessi in tutti i file. In caso contrario, Project Workspace tenta automaticamente di scegliere uno degli schemi tra più file e cerca di conformare il resto dei file del set di dati a quello schema. Questo comportamento fa sì che la vista mostrata durante Project Workspace sia irregolare e, di conseguenza, anche l'output del lavoro sarà irregolare.

Se i file devono avere schemi diversi, è necessario creare più set di dati e profilarli separatamente.

Utilizzo di percorsi parametrizzati per Amazon S3

In alcuni casi, potresti voler creare un set di dati con file che seguono una determinata convenzione di denominazione o un set di dati che può estendersi su più cartelle Amazon S3. Oppure potresti voler riutilizzare lo stesso set di dati per dati strutturati in modo identico che vengono generati periodicamente in una posizione S3 con un percorso che dipende da determinati parametri. Un esempio è un percorso denominato in base alla data di produzione dei dati.

DataBrew supporta questo approccio con percorsi S3 parametrizzati. Un percorso con parametri è un URL Amazon S3 contenente espressioni regolari o parametri di percorso personalizzati o entrambi.

Definizione di un set di dati con un percorso S3 utilizzando espressioni regolari

Le espressioni regolari nel percorso possono essere utili per abbinare più file da una o più cartelle e allo stesso tempo filtrare i file non correlati in tali cartelle.

Ecco un paio di esempi:

- Definisci un set di dati che include tutti i file JSON da una cartella il cui nome inizia con. `invoice`
- Definisci un set di dati che includa tutti i file nelle cartelle con `2020` i loro nomi.

È possibile implementare questo tipo di approccio utilizzando espressioni regolari in un percorso S3 del set di dati. Queste espressioni regolari possono sostituire qualsiasi sottostringa nella chiave dell'URL S3 (ma non il nome del bucket).

Come esempio di chiave in un URL S3, vedi quanto segue. Qui `my-bucket` c'è il nome del bucket, US East (Ohio) è la AWS regione e il nome `puppy.png` chiave.

`https://my-bucket.s3.us-west-2.amazonaws.com/puppy.png`

In un percorso S3 parametrizzato, tutti i caratteri compresi tra due parentesi angolari (<e>) vengono trattati come espressioni regolari. Due esempi sono i seguenti:

- `s3://my-databrew-bucket/databrew-input/invoice<.*>/data.json`corrisponde a tutti i file denominati`data.json`, all'interno di tutte le sottocartelle `databrew-input` i cui nomi iniziano con`invoice`.
- `s3://my-databrew-bucket/databrew-input/<.*>2020<.*>/`corrisponde a tutti i file contenuti nelle cartelle il `2020` cui nome è presente.

In questi esempi, `.*` corrisponde a zero o più caratteri.

> **Note**
>
> Puoi usare espressioni regolari solo nella parte fondamentale del percorso S3, la parte che segue il nome del bucket. Quindi `s3://my-databrew-bucket/<.*>-input/` è valido, ma non lo è. `s3://my-<.*>-bucket/<.*>-input/`

Ti consigliamo di testare le espressioni regolari per assicurarti che corrispondano solo agli URL S3 che desideri e non a quelli che non desideri.

Ecco alcuni altri esempi di espressioni regolari:

- `<\d{2}>`corrisponde a una stringa composta esattamente da due cifre consecutive, ad esempio `07` o`03`, ma non`1a2`.
- `<[a-z]+.*>`corrisponde a una stringa che inizia con una o più lettere latine minuscole e contiene zero o più altri caratteri dopo di essa. Un esempio è`a3`, o `abc/defa-z`, ma no. `A2`
- `<[^/]+>`corrisponde a una stringa che contiene qualsiasi carattere tranne una barra (/). In un URL S3, le barre vengono utilizzate per separare le cartelle nel percorso.
- `<.*=.*>`corrisponde a una stringa che contiene un segno di uguale (=), ad esempio`month=02`, o, ma no. `abc/day=2` `=10` `test`
- `<\d.*\d>`corrisponde a una stringa che inizia e finisce con una cifra e può contenere qualsiasi altro carattere tra le cifre, ad esempio, o`01-02-03`, `1abc2` ma no. `2020/Jul/21` `123a`

Definizione di un set di dati con un percorso S3 utilizzando parametri personalizzati

La definizione di un set di dati parametrizzato utilizzando parametri personalizzati offre vantaggi rispetto all'utilizzo di espressioni regolari quando potresti voler fornire parametri per una posizione S3:

- È possibile ottenere gli stessi risultati ottenuti con un'espressione regolare, senza dover conoscere la sintassi delle espressioni regolari. È possibile definire i parametri utilizzando termini familiari come «inizia con» e «contiene».
- Quando definisci un set di dati dinamico utilizzando i parametri nel percorso, puoi includere un intervallo di tempo nella definizione, ad esempio «ultimo mese» o «ultime 24 ore». In questo modo, la definizione del set di dati verrà utilizzata in seguito con i nuovi dati in entrata.

Ecco alcuni esempi di quando potresti voler utilizzare set di dati dinamici:

- Per connettere più file partizionati in base alla data dell'ultimo aggiornamento o ad altri attributi significativi in un unico set di dati. È quindi possibile acquisire questi attributi di partizione come colonne aggiuntive in un set di dati.
- Per limitare i file in un set di dati alle posizioni S3 che soddisfano determinate condizioni. Ad esempio, supponiamo che il percorso S3 contenga cartelle basate sulla data come. `folder/2021/04/01/` In questo caso, puoi parametrizzare la data e limitarla a un determinato intervallo, ad esempio «tra il 01 marzo 2021 e il 01 aprile 2021" o «Settimana scorsa».

Per definire un percorso utilizzando i parametri, definisci i parametri e aggiungili al percorso utilizzando il seguente formato:

```
s3://my-databrew-bucket/some-folder/{parameter1}/file-{parameter2}.json
```

> **Note**
>
> Come per le espressioni regolari in un percorso S3, puoi usare i parametri solo nella parte chiave del percorso, la parte che segue il nome del bucket.

Nella definizione di un parametro sono necessari due campi, nome e tipo. Il tipo può essere String, Number o Date. I parametri di tipo Date devono avere una definizione del formato della data in modo da DataBrew poter interpretare e confrontare correttamente i valori della data. Facoltativamente, è possibile definire le condizioni di corrispondenza per un parametro. Puoi anche scegliere di aggiungere i valori corrispondenti di un parametro come colonna al tuo set di dati quando viene caricato da un DataBrew processo o da una sessione interattiva.

Esempio

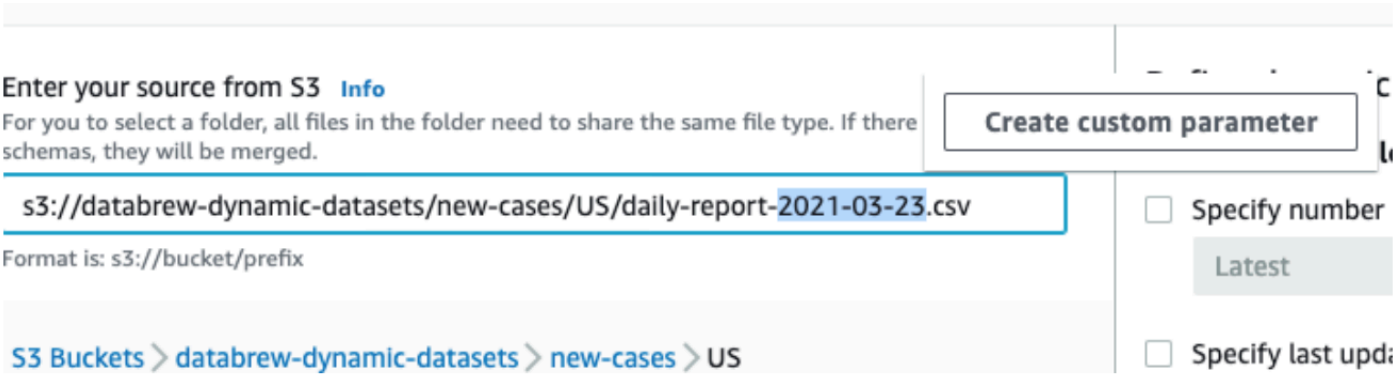
Consideriamo un esempio di definizione di un set di dati dinamico utilizzando i parametri nella DataBrew console. In questo esempio, supponiamo che i dati di input vengano scritti regolarmente in un bucket S3 utilizzando posizioni come queste:

- `s3://databrew-dynamic-datasets/new-cases/UR/daily-report-2021-03-30.csv`
- `s3://databrew-dynamic-datasets/new-cases/UR/daily-report-2021-03-31.csv`
- `s3://databrew-dynamic-datasets/new-cases/US/daily-report-2021-03-30.csv`

- `s3://databrew-dynamic-datasets/new-cases/US/daily-report-2021-03-31.csv`

Qui ci sono due parti dinamiche: un prefisso internazionale, ad esempio US, e una data nel nome del file, ad esempio 30-03-2021. Qui puoi applicare la stessa ricetta di pulizia per tutti i file. Supponiamo che tu voglia eseguire il tuo lavoro di pulizia ogni giorno. Di seguito è riportato come definire un percorso parametrizzato per questo scenario:

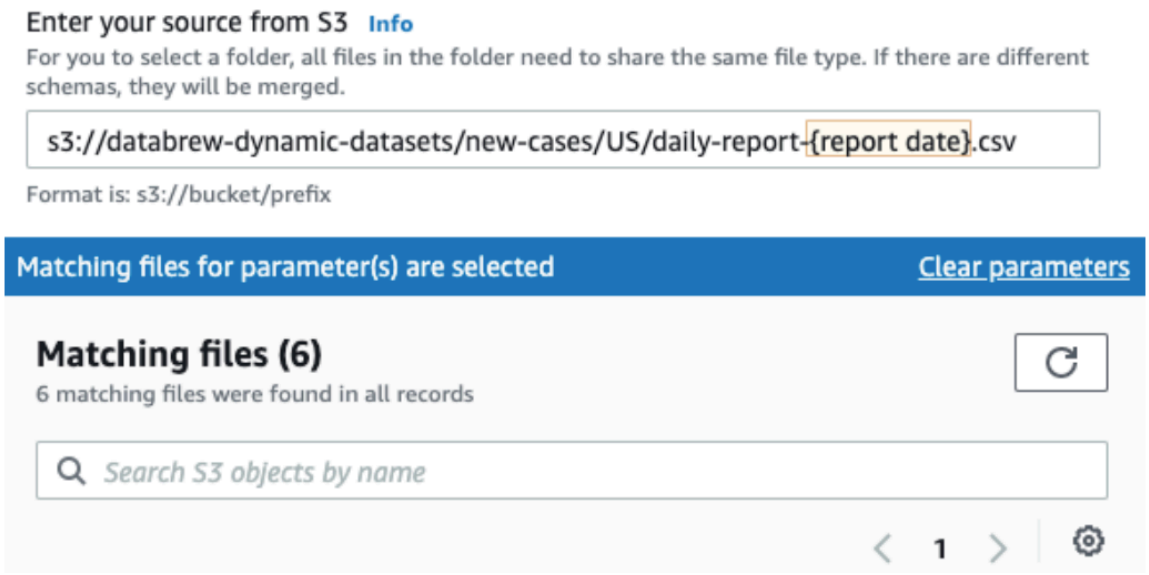
1. Accedere a un file specifico.
2. Quindi seleziona una parte variabile, ad esempio una data, e sostituiscila con un parametro. In questo caso, sostituisci una data.



3. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per Crea parametro personalizzato e imposta le relative proprietà:
   - Nome: data del rapporto
   - Tipo: data
   - Formato della data: yyyy- MM-dd (selezionato tra i formati predefiniti)
   - Condizioni (intervallo di tempo): Ultime 24 ore
   - Aggiungi come colonna: true (selezionato)

   Mantieni gli altri campi con i valori predefiniti.
4. Scegli Create (Crea).

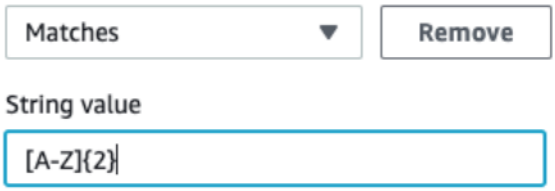
Dopo averlo fatto, viene visualizzato il percorso aggiornato, come nella schermata seguente.



Ora puoi fare lo stesso per il prefisso internazionale e parametrizzarlo come segue:

- Nome: prefisso internazionale
- Tipo: String
- Aggiungi come colonna: true (selezionato)

Non è necessario specificare condizioni se tutti i valori sono pertinenti. Nella `new-cases` cartella, ad esempio, abbiamo solo sottocartelle con codici nazionali, quindi non sono necessarie condizioni. Se hai altre cartelle da escludere, potresti utilizzare la seguente condizione.



Questo approccio limita le sottocartelle dei nuovi casi a contenere due caratteri latini maiuscoli.

Dopo questa parametrizzazione, hai solo i file corrispondenti nel nostro set di dati e puoi scegliere Crea set di dati.

> **Note**
>
> Quando si utilizzano intervalli di tempo relativi in condizioni, gli intervalli di tempo vengono valutati al momento del caricamento del set di dati. Questo vale sia che si tratti di intervalli di tempo predefiniti come «Ultime 24 ore» o intervalli di tempo personalizzati come «5 giorni fa».

Questo approccio di valutazione si applica indipendentemente dal fatto che il set di dati venga caricato durante l'inizializzazione di una sessione interattiva o durante l'avvio di un processo.

Dopo aver scelto Crea set di dati, il set di dati dinamico è pronto per l'uso. Ad esempio, è possibile utilizzarlo prima per creare un progetto e definire una ricetta di pulizia utilizzando una sessione interattiva. DataBrew È quindi possibile creare un processo programmato per l'esecuzione giornaliera. Questo processo potrebbe applicare la ricetta di pulizia ai file del set di dati che soddisfano le condizioni dei parametri al momento dell'avvio del lavoro.

Condizioni supportate per set di dati dinamici

Puoi utilizzare le condizioni per filtrare i file S3 corrispondenti utilizzando i parametri o l'attributo della data dell'ultima modifica.

Di seguito, puoi trovare elenchi di condizioni supportate per ogni tipo di parametro.

Condizioni utilizzate con i parametri String

Nome in DataBrew SDK	Sinonimi SDK	Nome nella console DataBrew	Description
è	eq, ==	È esattamente	Il valore del parametro è uguale al valore fornito nella condizione.
non è	non eq,! =	Is not (Non è)	Il valore del parametro non è lo stesso del valore fornito nella condizione.
contiene		Contiene	Il valore di stringa del parametro contiene il valore fornito nella condizione.
non contiene		Non contiene	Il valore di stringa del parametro non contiene il valore

Nome in DataBrew SDK	Sinonimi SDK	Nome nella console DataBrew	Description
			fornito nella condizione.
inizia_con		Inizia con	Il valore di stringa del parametro inizia con il valore fornito nella condizione.
not starts_with		Non inizia con	Il valore di stringa del parametro non inizia con il valore fornito nella condizione.
fine_con		Ends with	Il valore di stringa del parametro termina con il valore fornito nella condizione.
non finisce con		Non termina con	Il valore di stringa del parametro non termina con il valore fornito nella condizione.
fiammiferi		Corrispondenze	Il valore del parametro corrisponde all'espressione regolare fornita nella condizione.
non corrisponde		Non corrisponde	Il valore del parametro non corrisponde all'espressione regolare fornita nella condizione.

> **Note**
>
> Tutte le condizioni per i parametri String utilizzano il confronto con distinzione tra maiuscole e minuscole. Se non sei sicuro del tipo di maiuscole/minuscole utilizzato in un percorso S3, puoi utilizzare la condizione «matches» con un valore di espressione regolare che inizia con. `(?i)` In questo modo si ottiene un confronto senza distinzione tra maiuscole e minuscole. Ad esempio, supponete di volere che il parametro di stringa inizi con`abc`, ma che `Abc` sia `ABC` anche possibile. In questo caso, puoi utilizzare la condizione «matches» con `(?i)^abc` come valore della condizione.

Condizioni utilizzate con i parametri Number

Nome nell' DataBrew SDK	Sinonimi SDK	Nome nella console DataBrew	Description
è	eq, ==	È esattamente	Il valore del parametro è uguale al valore fornito nella condizione.
non è	non eq,! =	Is not (Non è)	Il valore del parametro non è lo stesso del valore fornito nella condizione.
minore_di	lt, <	Less than	Il valore numerico del parametro è inferiore al valore fornito nella condizione.
meno di uguale	litigio, <=	Minore o uguale a	Il valore numerico del parametro è minore o uguale al valore fornito nella condizione.

Nome nell' DataBrew SDK	Sinonimi SDK	Nome nella console DataBrew	Description
maggiore_di	gt, >	Greater than	Il valore numerico del parametro è maggiore del valore fornito nella condizione.
greater_than_equal	arrivare, =>	Maggiore o uguale a	Il valore numerico del parametro è maggiore o uguale al valore fornito nella condizion e.

Condizioni utilizzate con i parametri Date

Nome nell' DataBrew SDK	Nome nella console DataBrew	Formato del valore delle condizioni (SDK)	Description
after	Start (Avvio)	Formato di data ISO 8601 come o `2021-03-3 0T01:00:0 0Z 2021-03-3 0T01:00-07:00`	Il valore del parametro date è successivo alla data fornita nella condizione.
before	End	Formato di data ISO 8601 come o `2021-03-3 0T01:00:0 0Z 2021-03-3 0T01:00-07:00`	Il valore del parametro date è precedente alla data fornita nella condizione.
relativo_dopo	Inizio (relativo)	Numero positivo o negativo di unità di tempo, ad esempio `-48h` o`+7d`.	Il valore del parametro date è successivo alla data relativa fornita nella condizione.

Nome nell' DataBrew SDK	Nome nella console DataBrew	Formato del valore delle condizioni (SDK)	Description
			Le date relative vengono valutate quando il set di dati viene caricato, quando viene inizializ zata una sessione interattiva o quando viene avviato un processo associato. Questo è il momento che negli esempi viene chiamato «adesso».
relative_before	Fine (relativa)	Numero positivo o negativo di unità di tempo, ad esempio `-48h` o`+7d`.	Il valore del parametro date è precedente alla data relativa fornita nella condizione. Le date relative vengono valutate quando il set di dati viene caricato, quando viene inizializ zata una sessione interattiva o quando viene avviato un processo associato. Questo è il momento che negli esempi viene chiamato «adesso».

Se utilizzi l'SDK, fornisci le date relative nel seguente formato:`±{number_of_time_units}{time_unit}`. Puoi usare queste unità di tempo:

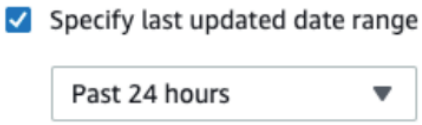
- -1h (1 ora fa)
- +2d (tra 2 giorni)
- -120m (120 minuti fa)
- 5000 secondi (tra 5.000 secondi)
- -3w (3 settimane fa)
- +4M (tra 4 mesi)
- -1y (1 anno fa)

Le date relative vengono valutate quando il set di dati viene caricato, quando viene inizializzata una sessione interattiva o quando viene avviato un processo associato. Questo è il momento chiamato «adesso» negli esempi precedenti.

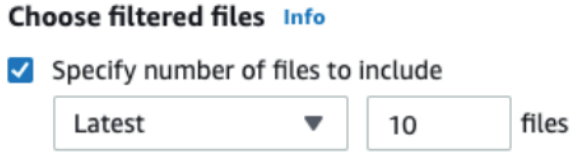
Configurazione delle impostazioni per set di dati dinamici

Oltre a fornire un percorso S3 parametrizzato, puoi configurare altre impostazioni per set di dati con più file. Queste impostazioni filtrano i file S3 in base alla data dell'ultima modifica e limitano il numero di file.

Analogamente all'impostazione di un parametro di data in un percorso, puoi definire un intervallo di tempo in cui i file corrispondenti sono stati aggiornati e includere solo quei file nel tuo set di dati. Puoi definire questi intervalli utilizzando date assolute come «30 marzo 2021" o intervalli relativi come «Ultime 24 ore».



Per limitare il numero di file corrispondenti, seleziona un numero di file maggiore di 0 e scegli se desideri i file corrispondenti più recenti o quelli più vecchi.



Tipi di dati

I dati per ogni colonna del set di dati vengono convertiti in uno dei seguenti tipi di dati:

- byte: numeri interi con segno a 1 byte. L'intervallo di numeri è compreso tra -128 e 127.
- breve: numeri interi con segno a 2 byte. L'intervallo di numeri è compreso tra -32768 e 32767.
- intero: numeri interi con segno a 4 byte. L'intervallo di numeri è compreso tra -2147483648 e 2147483647.
- long: numeri interi con segno a 8 byte. L'intervallo di numeri va da -9223372036854775808 a 9223372036854775807.
- float — numeri in virgola mobile a precisione singola a 4 byte.
- double — numeri in virgola mobile a doppia precisione da 8 byte.
- decimale: numeri decimali firmati con un massimo di 38 cifre in totale e 18 cifre dopo la virgola decimale.
- string — Valori delle stringhe di caratteri.
- booleano — Il tipo booleano ha uno dei due valori possibili: `true` e `false` o `yes` e `no`.
- timestamp — Valori che comprendono i campi anno, mese, giorno, ora, minuto e secondo.
- data: valori che comprendono i campi anno, mese e giorno.

Tipi di dati avanzati

I tipi di dati avanzati sono tipi di dati DataBrew rilevati all'interno di una colonna di stringhe in un progetto e pertanto non fanno parte di un set di dati. Per informazioni sui tipi di dati avanzati, consulta Tipi di dati avanzati.

Tipi di dati avanzati

I tipi di dati avanzati sono tipi di dati che vengono DataBrew rilevati all'interno di una colonna di stringhe in un progetto mediante la corrispondenza di modelli. Quando si fa clic su una colonna di stringhe, la colonna viene contrassegnata come tipo di dati avanzato corrispondente se il 50% o più dei valori nella colonna soddisfano i criteri per quel tipo di dati.

I tipi di dati che è DataBrew possibile rilevare sono:

- Date/timestamp

- SSN
- Numero di telefono
- Email
- carta di credito
- Gender
- IP address (Indirizzo IP)
- URL
- Codice postale
- Paese
- Valuta
- Stato
- Città

Puoi utilizzare le seguenti trasformazioni per lavorare con tipi di dati avanzati:

- GET_ADVANCED_DATATYPE: data una colonna di stringhe, identifica l'eventuale tipo di dati avanzato della colonna.
- EXTRACT_ADVANCED_DATATYPE_DETAILS: estrae i dettagli per un tipo di dati avanzato.
- ADVANCED_DATATYPE_FILTER: filtra una colonna di origine corrente in base al rilevamento avanzato del tipo di dati.
- ADVANCED_DATATYPE_FLAG: crea una nuova colonna di bandiera in base ai valori della colonna di origine corrente.

Convalida della qualità dei dati in AWS Glue DataBrew

Per garantire la qualità dei set di dati, è possibile definire un elenco di regole di qualità dei dati in un set di regole. Un set di regole è un insieme di regole che confrontano diverse metriche dei dati con i valori previsti. Se uno qualsiasi dei criteri di una regola non viene soddisfatto, l'intero set di regole non viene convalidato. È quindi possibile esaminare i singoli risultati per ogni regola. Per qualsiasi regola che causa un errore di convalida, è possibile apportare le correzioni necessarie e riconvalidare.

Di seguito sono riportati alcuni esempi di regole:

- Il valore nella colonna `"APY"` è compreso tra 0 e 100
- Il numero di valori mancanti nella colonna `group_name` non supera il 5%

Puoi definire ogni regola per una singola colonna o applicarla indipendentemente a più colonne selezionate, ad esempio:

- Il valore massimo non supera 100 per le colonne `"rate""pay","increase"`.

Una regola può essere costituita da più controlli semplici. È possibile definire se devono essere tutti veri o uno qualsiasi, ad esempio:

- Il valore nella colonna `"ProductId"` deve iniziare con `"asin-"` AND la lunghezza del valore nella colonna `"ProductId"` è 32.

È possibile verificare le regole in base a valori aggregati come `maxmin`, o `number of duplicate values` in cui viene confrontato un solo valore, oppure a valori non aggregati in ogni riga di una colonna. In quest'ultimo caso, puoi anche definire una soglia di «superamento» come. `value in columnA > value in columnB for at least 95% of rows`

Come per le informazioni sul profilo, è possibile definire regole di qualità dei dati a livello di colonna solo per colonne di tipo semplice, come stringhe e numeri. Non è possibile definire regole di qualità dei dati per colonne di tipi complessi, come matrici o strutture. Per ulteriori dettagli sull'utilizzo delle informazioni del profilo, vedereCreare e lavorare con AWS Glue DataBrew lavori di profilo.

Convalida delle regole di qualità dei dati

Dopo aver definito un set di regole, puoi aggiungerlo a un job di profilo per la convalida. È possibile definire più di un set di regole per un set di dati.

Ad esempio, un set di regole potrebbe contenere regole con criteri minimamente accettabili. Un errore di convalida per quel set di regole potrebbe significare che i dati non sono accettabili per un ulteriore utilizzo. Un esempio sono i valori mancanti nelle colonne chiave di un set di dati utilizzato per la formazione sull'apprendimento automatico. È possibile utilizzare un secondo set di regole con regole più rigorose per verificare se la qualità del set di dati è tale da non richiedere alcuna pulizia.

È possibile applicare uno o più set di regole definiti per un determinato set di dati in una configurazione del processo di profilo. Quando il job del profilo viene eseguito, produce un rapporto di convalida oltre al profilo di dati. Il rapporto di convalida è disponibile nella stessa posizione dei dati del profilo. Come per le informazioni sul profilo, puoi esplorare i risultati nella DataBrew console. Nella visualizzazione dei dettagli del set di dati, scegli la scheda Qualità dei dati per visualizzare i risultati. Per maggiori dettagli sull'utilizzo delle informazioni del profilo, consultaCreare e lavorare con AWS Glue DataBrew lavori di profilo.

Agire in base ai risultati della convalida

Quando un processo DataBrew di profilo viene completato, DataBrew invia un CloudWatch evento Amazon con i dettagli dell'esecuzione di tale processo. Se hai anche configurato il job per convalidare le regole di qualità dei dati, DataBrew invia un evento per ogni set di regole convalidato. L'evento contiene il risultato (`SUCCEEDED`,`FAILED`, o`ERROR`) e un collegamento al rapporto dettagliato di convalida della qualità dei dati. È quindi possibile automatizzare ulteriori azioni richiamando l'azione successiva in base allo stato della convalida. Per ulteriori informazioni sul collegamento degli eventi alle azioni mirate, come notifiche Amazon SNS, invocazioni di AWS Lambda funzioni e altro, consulta la pagina Guida introduttiva ad Amazon. EventBridge

Di seguito è riportato un esempio di evento DataBrew Validation Result:

```
{
  "version": "0",
  "id": "fb27348b-112d-e7c2-560d-85e7c2c09964",
  "detail-type": "DataBrew Ruleset Validation Result",
  "source": "aws.databrew",
  "account": "123456789012",
```

```
  "time": "2021-11-18T13:15:46Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "datasetName": "MyDataset",
    "jobName": "MyProfileJob",
    "jobRunId": "db_f07954d20d083de0c1fc1eee11498d8635ee5be4ca416af27d33933e91ff4e6e",
    "rulesetName": "MyRuleset",
    "validationState": "FAILED",
    "validationReportLocation": "s3://MyBucket/MyKey/
MyDataset_f07954d20d083de0c1fc1eee11498d8635ee5be4ca416af27d33933e91ff4e6e_dq-
validation-report.json"
  }
}
```

Puoi utilizzare attributi di eventi come `detail-type` `source` e proprietà annidate dell'`detail`attributo per creare modelli di eventi in Amazon Eventbridge. Ad esempio, uno schema di eventi che corrisponda a tutte le convalide non riuscite di qualsiasi DataBrew processo sarebbe simile al seguente:

```
{
  "source": ["aws.databrew"],
  "detail-type": ["DataBrew Ruleset Validation Result"],
  "detail": {
   "validationState": ["FAILED"]
  }
}
```

Per un esempio di creazione di un set di regole e di convalida delle relative regole, vedi. Creazione di un set di regole con regole di qualità dei dati Per ulteriori informazioni sull'utilizzo CloudWatch degli eventi in, vedere DataBrew Automazione DataBrew con eventi CloudWatch

Creazione di un set di regole con regole di qualità dei dati

Nella procedura seguente, è possibile trovare un esempio di creazione di un set di regole e di applicazione a un set di dati. Un set di regole è un insieme di regole che confrontano diverse metriche dei dati con i valori previsti. È quindi possibile utilizzare questo set di regole in un processo di profilatura per convalidare le regole di qualità dei dati che include.

Per creare un set di regole di esempio con regole di qualità dei dati

1. Accedi a Console di gestione AWS e apri la DataBrew console all'indirizzo. https://console.aws.amazon.com/databrew/
2. Scegli DQ RULES dal riquadro di navigazione, quindi scegli Crea set di regole per la qualità dei dati.
3. Inserisci un nome per il tuo set di regole. Facoltativamente, inserisci una descrizione per il tuo set di regole.
4. In Set di dati associato, scegli un set di dati da associare al set di regole.

   Dopo aver selezionato un set di dati, puoi visualizzare il riquadro di anteprima del set di dati a destra.
5. Utilizza l'anteprima nel riquadro di anteprima del set di dati per esplorare i valori e lo schema del set di dati mentre determini le regole di qualità dei dati da creare. L'anteprima può darti informazioni sui potenziali problemi che potresti avere con i dati.

   Alcune fonti di dati, come i database, non supportano l'anteprima dei dati. In tal caso, puoi eseguire un processo di profilazione senza prima convalidare le regole di qualità dei dati. È quindi possibile ottenere informazioni sullo schema dei dati e sulla distribuzione dei valori utilizzando il profilo dati.
6. Controlla la scheda Consigli, che elenca alcuni suggerimenti di regole che puoi utilizzare per creare il tuo set di regole. Puoi selezionare tutti, alcuni o nessuno dei consigli.

   Dopo aver selezionato i consigli pertinenti, scegli Aggiungi al set di regole.

   Questo aggiungerà delle regole al tuo set di regole. Ispeziona e modifica i parametri se necessario. Nota che nelle regole di qualità dei dati possono essere utilizzate solo colonne di tipi semplici come stringhe, numeri e booleani.
7. Scegli Aggiungi un'altra regola per aggiungere una regola non coperta dai consigli. Puoi modificare i nomi delle regole per semplificare l'interpretazione dei risultati di convalida in un secondo momento.
8. Utilizza l'ambito del controllo della qualità dei dati per scegliere se selezionare singole colonne per ogni controllo in questa regola o se devono essere applicate a un gruppo di colonne selezionato. Ad esempio, se il set di dati ha diverse colonne numeriche che devono avere valori compresi tra 0 e 100, puoi definire la regola una volta e selezionare tutte queste colonne da controllare con questa regola.

9. Se la regola prevede più di un controllo, nel menu a discesa Criteri di successo della regola, scegli se tutti i controlli devono essere soddisfatti o quali soddisfano i criteri.
10. Seleziona un controllo che verrà eseguito per verificare questa regola nel menu a discesa Controllo della qualità dei dati. Per ulteriori informazioni sui controlli disponibili, consultaControlli disponibili.
11. Se hai scelto Controllo individuale per ogni colonna nell'ambito del controllo della qualità dei dati, scegli una colonna. Seleziona o digita il nome della colonna per questo controllo.
12. Seleziona i parametri in base al controllo. Alcune condizioni accettano solo i valori personalizzati forniti e altre supportano anche il riferimento a un'altra colonna.
13. Se scegli i controlli per i valori delle colonne come la condizione Contiene la condizione per i valori di stringa, puoi specificare la soglia di «superamento». Ad esempio, se desideri che almeno il 95 percento dei valori soddisfi la condizione, devi scegliere Maggiore di uguale come Condizione della soglia, inserire 95 come Soglia e lasciare «% (percentuale) righe» nel menu a discesa successivo della sezione Soglia. Oppure, se non desideri più di 10 righe in cui manca il valore, la condizione è vera, puoi selezionare Meno di uguale come condizione, inserire 10 per Soglia e scegliere le righe nel menu a discesa successivo. Tieni presente che potresti ottenere risultati diversi se utilizzi campioni di dimensioni diverse durante la convalida.
14. Aggiungi altre regole se necessario.
15. Scegli Crea set di regole.

Creazione di un profilo (job) utilizzando un set di regole

Dopo aver creato un set di regole come descritto in precedenza, verrai indirizzato alla pagina delle regole sulla qualità dei dati, che mostra tutti i set di regole del tuo account.

Per creare un profilo, un lavoro che includa un set di regole

1. Scegli il nome del set di regole che hai creato in precedenza per visualizzarne i dettagli.
2. Scegli Crea profilo, lavoro con set di regole.

   Il nome del Job viene inserito automaticamente, ma è possibile modificarlo in base alle esigenze.
3. Per Job run sample, puoi scegliere di eseguire l'intero set di dati o un numero limitato di righe.

   Se scegliete di eseguire un campione di dimensioni limitate, tenete presente che per alcune regole, i risultati potrebbero differire rispetto all'intero set di dati.

4. Per le impostazioni dell'output del lavoro, scegli una posizione S3 per l'output del lavoro. Scegli una cartella in un bucket denominato Amazon S3 a cui hai accesso. Se inserisci un nome di cartella per questo bucket che non esiste, questa cartella viene creata.

   Una volta completato con successo il processo di profilazione, questa cartella conterrà i profili dei dati e del rapporto di convalida delle regole di qualità dei dati in formato JSON.
5. In Regole di qualità dei dati, tieni presente che il tuo set di regole è elencato in Nome del set di regole sulla qualità dei dati.
6. In Autorizzazioni, seleziona o crea un ruolo per concedere DataBrew l'accesso alla lettura dalla posizione di input di Amazon S3 e alla scrittura nella posizione di output del lavoro. Se non hai un ruolo pronto, seleziona Crea nuovo ruolo IAM.
7. Modifica eventuali altre impostazioni opzionali come descritto in[Creare e lavorare con AWS Glue DataBrew lavori di profilo](), se necessario.
8. Scegli Crea ed esegui il processo.

Ispezione dei risultati di convalida e aggiornamento delle regole di qualità dei dati

Una volta completato il processo di creazione del profilo, puoi visualizzare i risultati di convalida delle regole di qualità dei dati e, se necessario, aggiornare le regole.

Per visualizzare i dati di convalida per le tue regole di qualità dei dati

1. Sulla DataBrew console, scegli Visualizza profilo dati. In questo modo viene visualizzata la scheda Panoramica del profilo di dati per il set di dati.
2. Scegli la scheda Regole di qualità dei dati. In questa scheda puoi visualizzare i risultati di tutte le regole di qualità dei dati.
3. Seleziona una singola regola per maggiori dettagli su quella regola.

Per ogni regola che non è riuscita a convalidare, puoi apportare le correzioni necessarie.

Per aggiornare le regole sulla qualità dei dati

1. Nel riquadro di navigazione, scegli DQ RULES.
2. In Nome del set di regole sulla qualità dei dati, scegli il set di dati che contiene le regole che intendi modificare.

3. Scegli la regola che desideri modificare, quindi scegli Modifica.
4. Apporta le correzioni necessarie, quindi scegli Aggiorna set di regole.
5. Eseguite nuovamente il processo. Ripetere questo processo fino al termine di tutte le convalide.

Controlli disponibili

La tabella seguente elenca i riferimenti per tutte le condizioni disponibili che possono essere utilizzate nelle regole. Tieni presente che le condizioni aggregate non possono essere combinate con condizioni non aggregate nella stessa regola.

> **Note**
>
> Per gli utenti SDK, per applicare la stessa regola a più colonne, utilizza l'ColumnSelectorsattributo di una regola e specifica le colonne convalidate utilizzando i loro nomi o un'espressione regolare. In questo caso, dovresti usare l'implicito. CheckExpression Ad esempio, `“> :val”` per confrontare i valori in ciascuna delle colonne selezionate con il valore fornito. DataBrew utilizza la sintassi implicita per la definizione FilterExpressionin set di dati dinamici. Se desideri specificare una o più colonne per ogni controllo singolarmente, non impostare l'attributo. ColumnSelectors Fornite invece un'espressione esplicita. Ad esempio, `“:col > :val”` come previsto da CheckExpressionuna regola.

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK
Condizioni aggregate del set di dati	Numero di righe		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(ROWS_COUNT) > :val", "SubstitutionMap": {":val", "10000"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK
	Numero di colonne		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(COLUMNS_COUNT) == :val", "SubstitutionMap": {":val", "20"}`

	Righe duplicate		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(DUPLICATE_ROWS_COUNT) < :val", "SubstitutionMap": {":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(DUPLICATE_ROWS_PERCENTAGE) < :val", "SubstitutionMap": {":val", "5"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

Condizioni statistiche aggregate delle colonne	Valori mancanti		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(MISSING_VALUES_COUNT) < :val", "SubstitutionMap": {":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(MISSING_VALUES_PERCENTAGE) < :val", "SubstitutionMap": {":val", "5"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori duplicati		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(DUPLICATE_VALUES_COUNT) < :val", "SubstitutionMap": {":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(DUPLICATE_VALUES_PERCENTAGE) < :val", "SubstitutionMap": {":val", "5"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori validi		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(VALID_VALUES_COUNT) > :val", "SubstitutionMap": {":val", "10000"}`
				
				or
				
				`"CheckExpression": "AGG(VALID_VALUES_PERCENTAGE) > :val", "SubstitutionMap": {":val", "95"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori distinti		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(DISTINCT_VALUES_COUNT) > :val", "SubstitutionMap": {":val", "1000"}`
				
				or
				
				`"CheckExpression": "AGG(DISTINCT_VALUES_PERCENTAGE) >= :val", "SubstitutionMap": {":val", "50"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori unici		Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(UNIQUE_VALUES_COUNT) > :val", "SubstitutionMap": {":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(UNIQUE_VALUES_PERCENTAGE) > :val", "SubstitutionMap": {":val", "20"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori anomali	Z-score soglia	Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(Z_SCORE_OUTLIERS_COUNT, :zscore_dev) < :val", "SubstitutionMap": {":zscore_dev": "4", ":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(Z_SCORE_OUTLIERS_PERCENTAGE) < :val", "SubstitutionMap": {":val", "5"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Statistiche sulla distribuzione del valore	Nome delle statistiche (vedi tabella successiva)	Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(<STAT_NAME>) < :val", "SubstitutionMap": {":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(<STAT_NAME>, :param) < :val", "SubstitutionMap": {":param": "0.25", :val", "5"}`
				
				Note
				Vedi la tabella successiva per i valori possibili `STAT_NAME`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Statistiche numeriche	Nome delle statistiche (vedi tabella successiva)	Confronto numerico con un valore personalizzato	`"CheckExpression": "AGG(<STAT_NAME>) < :val", "SubstitutionMap": {":val", "100"}`
				
				or
				
				`"CheckExpression": "AGG(<STAT_NAME>, :param) < :val", "SubstitutionMap": {":param": "0.25", :val", "5"}`
				
				Note
				Vedi la tabella successiva per i valori possibili `STAT_NAME`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK
Non aggregato (accetta la soglia)	Il valore è esattamente		Confronto esatto con un elenco di valori	`"CheckExpression": ":col IN :list", "SubstitutionMap": {":col": "`size`", ":list": "[\"S\",\"M\",\"L\",\"XL\"]"}`
	Il valore non è esattamente		Il valore non deve corrispondere esattamente a nessun valore di un elenco	`"CheckExpression": ":col NOT IN :list", "SubstitutionMap": {":col": "`domain`", ":list": "[\"GOV\",\"ORG\"]"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori di stringa		Confronto di stringhe con un valore personalizzato o un'altra colonna di stringhe	`"CheckExpression": ":col STARTS_WITH :val", "SubstitutionMap": {":col": "`url`", ":val": "http"}`
				
				or
				
				`"CheckExpression": ":col1 contains :col2", "SubstitutionMap": {":col1": "`url`", ":col2": "`company_name`"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Valori numerici		Confronto numerico con un valore personalizzato o un'altra colonna numerica	`"CheckExpression": ":col IS_BETWEEN :val1 and :val2", "SubstitutionMap": {":col": "`APY`", ":val1": "0", ":val2": "10"}`
				
				or
				
				`"CheckExpression": ":col1 <= :col2", "SubstitutionMap": {":col1": "`bank_rate`", ":col2": "`fed_rate`"}`

Tipo di condizioni	Controllo della qualità dei dati	Parametri aggiuntivi	Tipo di confronto	Esempio di sintassi SDK

	Lunghezza della stringa di valori		Confronto numerico con un valore personalizzato o un'altra colonna numerica	`"CheckExpression": "length(:col) IS_BETWEEN :val1 and :val2", "SubstitutionMap": {":col": "`identifier`", ":val1": "8", ":val2": "12"}`
				
				or
				
				`"CheckExpression": "length(:col1) <= :col2", "SubstitutionMap": {":col1": "`name`", ":col2": "`max_name_len`"}`

Confronti numerici

DataBrew supporta le seguenti operazioni per il confronto numerico: Is equals (= =), Is not equals (! =), Minore di (<), Minore di uguale (< =), Maggiore di (>), Maggiore di (> =) e È compreso tra (is_between:val1 e:val2).

Confronti tra stringhe

Sono supportati i seguenti confronti tra stringhe: Inizia con, Non inizia con, Termina con, Non finisce con, Contiene, Non contiene, È uguale, Non è uguale, Corrisponde, Non corrisponde.

La tabella seguente mostra le statistiche disponibili che è possibile utilizzare per le statistiche sulla distribuzione dei valori e le statistiche numeriche:

Controllo della qualità dei dati	Nome delle statistiche	Parametri aggiuntivi	Sintassi SDK
Statistiche sulla distribuzione del valore	Min		`"CheckExpression": "AGG(MAX) < :val", "SubstitutionMap": {":val", "100"}`
	Max		`"CheckExpression": "AGG(MIN) > :val", "SubstitutionMap": {":val", "0"}`
	Mediana		`"CheckExpression": "AGG(MEDIAN) >= :val", "SubstitutionMap": {":val", "50"}`

Controllo della qualità dei dati	Nome delle statistiche	Parametri aggiuntivi	Sintassi SDK
	Media		`"CheckExpression": "AGG(MEAN) <= :val", "SubstitutionMap": {":val", "10"}`
	Modalità		`"CheckExpression": "AGG(MODE) > :val", "SubstitutionMap": {":val", "0"}`
	Deviazione standard		`"CheckExpression": "AGG(STANDARD_DEVIATION) > :val", "SubstitutionMap": {":val", "0"}`
	Entropia		`"CheckExpression": "AGG(ENTROPY) > :val", "SubstitutionMap": {":val", "0"}`

Controllo della qualità dei dati	Nome delle statistiche	Parametri aggiuntivi	Sintassi SDK
Statistiche numeriche	Somma		`"CheckExpression": "AGG(SUM) > :val", "SubstitutionMap": {":val", "0"}`
	Curtosi		`"CheckExpression": "AGG(KURTOSIS) > :val", "SubstitutionMap": {":val", "0"}`
	Asimmetria		`"CheckExpression": "AGG(SKEWNESS) > :val", "SubstitutionMap": {":val", "0"}`
	Varianza		`"CheckExpression": "AGG(VARIANCE) > :val", "SubstitutionMap": {":val", "0"}`

Controllo della qualità dei dati	Nome delle statistiche	Parametri aggiuntivi	Sintassi SDK
	Deviazione assoluta		`"CheckExpression": "AGG(MEDIAN_ABSOLUTE_DEVIATION) > :val", "SubstitutionMap": {":val", "0"}`
	Quantile	Quantile: uno tra «0,25», «0,5», «0,75»	`"CheckExpression": "AGG(QUANTILE, :pct) > :val", "SubstitutionMap": {":pct": "0.25", ":val", "0"}`

Creazione e utilizzo AWS Glue DataBrew progetti

In AWS Glue DataBrew, un progetto è il fulcro delle tue attività di analisi e trasformazione dei dati.

Quando crei un progetto, unisci due componenti fondamentali:

- Un set di dati, per fornire accesso in sola lettura ai dati di origine. Per ulteriori informazioni, consulta Connessione ai dati con AWS Glue DataBrew.
- Una ricetta per applicare le trasformazioni DataBrew dei dati al set di dati. Per ulteriori informazioni, consulta Creazione e utilizzo AWS Glue DataBrew recipes.

La DataBrew console presenta il progetto in un'interfaccia utente altamente interattiva e intuitiva. Ti incoraggia a sperimentare centinaia di trasformazioni dei dati, così puoi imparare come funzionano e quale effetto hanno sui tuoi dati.

I dati che vedi nella visualizzazione del progetto sono un esempio del tuo set di dati. Poiché i set di dati possono essere molto grandi, con migliaia o addirittura milioni di righe, l'utilizzo di un campione aiuta a garantire che la DataBrew console rimanga reattiva mentre si trasformano i dati di esempio in vari modi. Per impostazione predefinita, l'esempio è costituito dalle prime 500 righe di dati del set di dati. È possibile scegliere diverse impostazioni per la dimensione del campione e le righe da scegliere.

Man mano che trasformate i dati di esempio, vi DataBrew aiuta a creare e perfezionare la ricetta del progetto, una serie dettagliata delle trasformazioni applicate finora. La ricetta in corso di lavorazione viene salvata automaticamente, quindi puoi abbandonare la visualizzazione del progetto in qualsiasi momento, riprenderla in un secondo momento e riprendere da dove avevi interrotto.

Quando la ricetta è pronta per l'uso, puoi pubblicarla. La pubblicazione di una ricetta la rende disponibile nel sottosistema DataBrew job, dove è possibile applicarla all'intero set di dati o creare un profilo di dati completo che consenta di comprendere la struttura, il contenuto e le caratteristiche statistiche dei dati.

Argomenti

Creare un progetto

Per creare un progetto, utilizzare la procedura seguente.

Come creare un progetto

1. Accedere a Console di gestione AWS e aprire la DataBrew console.
2. Nel riquadro di navigazione, scegli PROGETTI. Quindi scegli Crea progetto.
3. Inserire un nome per il progetto. Quindi scegli una ricetta da allegare al tuo progetto:

   - Scegli Crea nuova ricetta se parti dall'inizio. In questo modo si crea una nuova ricetta vuota e la si allega al progetto.
   - Scegliete Modifica ricetta esistente se avete una ricetta pubblicata in precedenza che desiderate utilizzare per questo progetto. Se la ricetta è attualmente associata a un altro progetto o ha dei lavori definiti per essa, non puoi usarla nel tuo nuovo progetto. Scegli Sfoglia le ricette per vedere quali ricette sono disponibili.
   - Scegli Importa passaggi dalla ricetta se hai una ricetta esistente che è stata pubblicata in precedenza e desideri importarne i passaggi, quindi procedi come segue:
     1. Scegli Sfoglia le ricette per vedere quali ricette sono disponibili.
     2. Scegli la versione pubblicata della ricetta che desideri utilizzare. Una ricetta può avere più versioni, a seconda della frequenza con cui l'hai pubblicata mentre lavoravi nella visualizzazione del progetto.
     3. Scegliete Visualizza i passaggi della ricetta per esaminare le trasformazioni dei dati nella ricetta.
4. Dopo aver creato una ricetta, scegli il set di dati con cui vuoi lavorare nel riquadro Seleziona un set di dati:

   - I miei set di dati: scegli un set di dati che hai creato in precedenza. Per ulteriori informazioni, consulta Creare un progetto.)
   - File di esempio: crea un nuovo set di dati basato su dati di esempio gestiti da.AWS Questi dati di esempio sono un ottimo modo per scoprire cosa è DataBrew possibile fare, senza dover fornire i propri dati. Assicurati di inserire un nome per il tuo set di dati.
   - Nuovo set di dati: crea un nuovo set di dati. Per ulteriori informazioni, consulta Creare un progetto.
5. Per le autorizzazioni di accesso, scegli un ruolo AWS Identity and Access Management(IAM) che DataBrew consenta la lettura dalla posizione di input di Amazon S3. Per una

sede S3 di proprietà del tuo AWS account, puoi scegliere il ruolo gestito dal servizio. `AwsGlueDataBrewDataAccessRole` In questo modo puoi accedere DataBrew alle risorse S3 di tua proprietà.

6. Nel riquadro Campionamento, puoi trovare le opzioni per DataBrew creare un campione di dati dal tuo set di dati.

   Per Tipo, scegli come DataBrew ottenere le righe dal tuo set di dati:

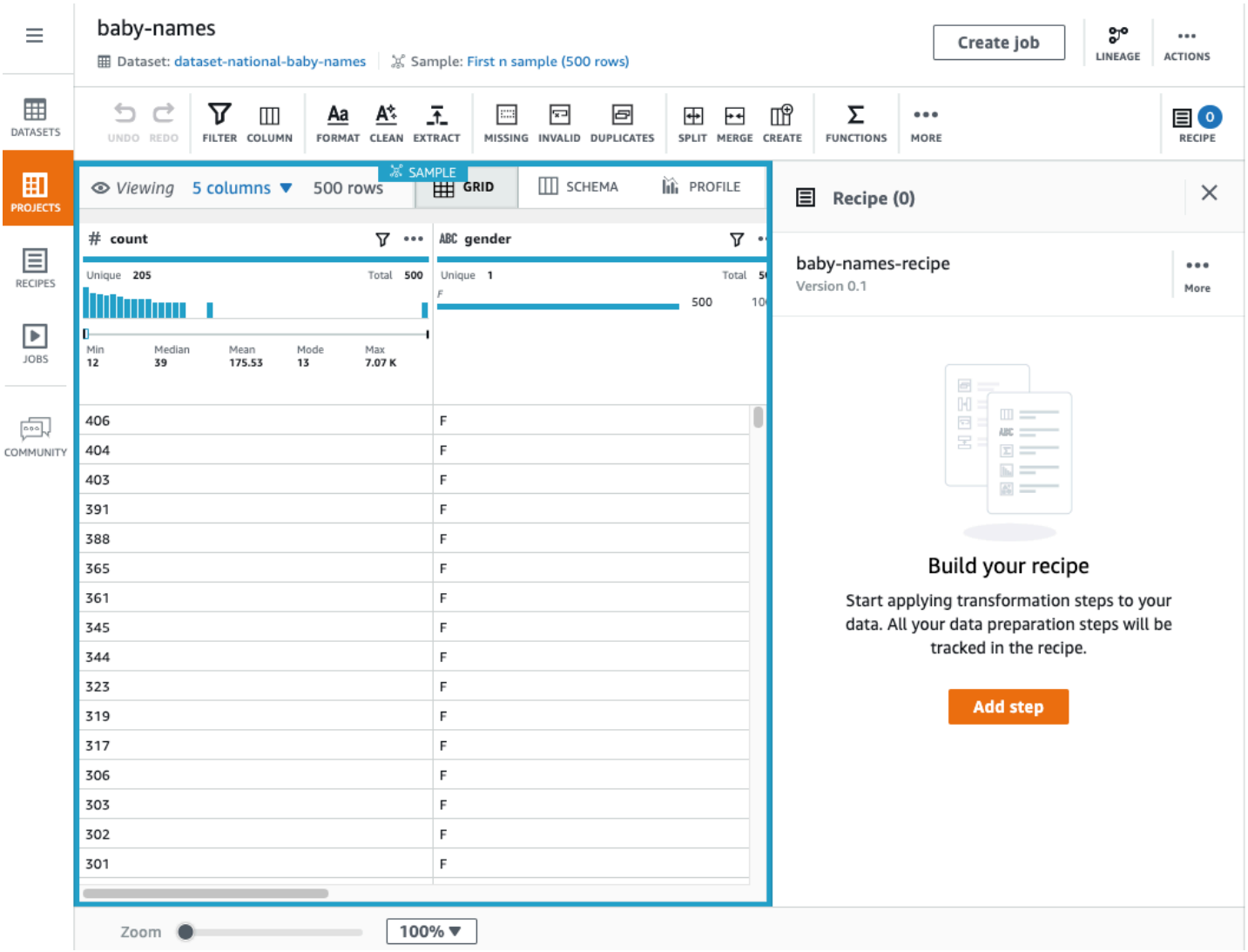
   - Usa First n rows per creare un campione basato sulle prime righe del set di dati.
   - Usa Righe casuali per creare un campione basato su una selezione casuale di righe nel set di dati.
   - Scegli il numero di righe da visualizzare nell'esempio: 500, 1.000, 2.500 o una dimensione del campione personalizzata, fino a un massimo di 5.000 righe. Una dimensione del campione più piccola consente di DataBrew eseguire le trasformazioni più velocemente, risparmiando tempo durante lo sviluppo della ricetta. Una dimensione del campione più ampia riflette in modo più accurato la composizione dei dati di origine sottostanti. Tuttavia, l'inizializzazione della sessione di progetto e le trasformazioni interattive sono più lente.
7. (Facoltativo) Scegliete Tag per allegare tag al set di dati.

   I tag sono semplici etichette costituite da una chiave definita dall'utente e da un valore opzionale che possono semplificare la gestione, la ricerca e il filtraggio DataBrew dei progetti per scopo, proprietario, ambiente o altri criteri.
8. Quando le impostazioni sono quelle desiderate, scegliete Crea lavoro.

DataBrew crea un nuovo set di dati se necessario, crea una nuova ricetta se necessario, crea l'esempio di dati e crea una sessione di progetto interattiva. Il completamento di questo processo può richiedere un paio di minuti. Quando il progetto è pronto per l'uso, puoi iniziare a lavorare con il campione di dati.

Panoramica di una sessione di DataBrew progetto

In una sessione di DataBrew progetto, lavori all'interno di uno spazio di lavoro interattivo.



Il riquadro a sinistra mostra la visualizzazione corrente dei dati. Il riquadro di destra mostra la ricetta di trasformazione del progetto, che al momento è vuota.

Nell'angolo in alto a destra della griglia di dati, ci sono tre schede:`GRID`,, e. `SCHEMA` `PROFILE` Scegliendo una di queste schede viene visualizzata una vista corrispondente nell'area di lavoro; queste viste sono descritte di seguito.

Visualizzazione a griglia

La visualizzazione a griglia è la visualizzazione predefinita, in cui l'esempio viene mostrato in formato tabulare. Utilizzate la procedura seguente per una breve panoramica della visualizzazione a griglia.

Per eseguire una panoramica della visualizzazione a griglia

1. Inizia visualizzando l'intero spazio:

a. Scorri verso sinistra e destra per visualizzare tutte le colonne.

b. Scorri verso l'alto e verso il basso per visualizzare tutti i valori dei dati.

c. Usa il controllo dello zoom nella parte inferiore dell'area di lavoro per regolare il livello di ingrandimento della griglia.

2. In alto a destra, visualizza quante colonne dell'esempio sono visualizzate e il numero corrente di righe nell'esempio.

   Per modificare le colonne visualizzate, scegli il link N colonne (dove N è il numero di colonne attualmente visualizzate). Scegliete le colonne desiderate e scegliete Mostra colonne selezionate.

3. Ora puoi iniziare a sperimentare con le DataBrew trasformazioni. Esegui quanto segue:

   a. Dalla barra degli strumenti di trasformazione, scegli Scegli formato, Cambia in maiuscolo.

   b. Per la colonna Sorgente, scegli una colonna che contenga i dati dei caratteri.

   c. Lascia le altre impostazioni ai valori predefiniti.

   d. Per vedere come appariranno i dati trasformati, scegli Anteprima modifiche. Quindi, per aggiungere questa trasformazione alla tua ricetta, scegli Applica.

   Ogni volta che applichi una trasformazione dei dati, la DataBrew aggiunge alla copia di lavoro della ricetta. Viene visualizzato sul lato destro dell'area di lavoro.

4. Esegui quanto segue:

   a. Dalla barra degli strumenti di trasformazione, scegli Crea, in base a una funzione.

   b. Per Seleziona una funzione, scegli`SQUARE ROOT`.

   c. Per la colonna Sorgente, scegliete una colonna che contenga dati numerici.

   d. Lascia le altre impostazioni ai valori predefiniti,.

   e. Scegli Anteprima delle modifiche per vedere come appaiono i dati trasformati. Quindi, per aggiungere questa trasformazione alla tua ricetta, scegli Applica.

5. Comprimi il riquadro delle ricette in alto a destra scegliendo RICETTA. Per espandere il riquadro delle ricette, scegli nuovamente RICETTA.

Pubblicazione di una nuova versione della ricetta

Man mano che continui ad applicare le trasformazioni, il numero di passaggi della ricetta aumenta. In qualsiasi momento, puoi pubblicare una nuova versione della tua ricetta. La pubblicazione di una ricetta la rende disponibile altrove DataBrew. In questo modo, è possibile eseguire un processo di elaborazione delle ricette per trasformare l'intero set di dati, anziché trasformare solo l'esempio di dati del progetto.

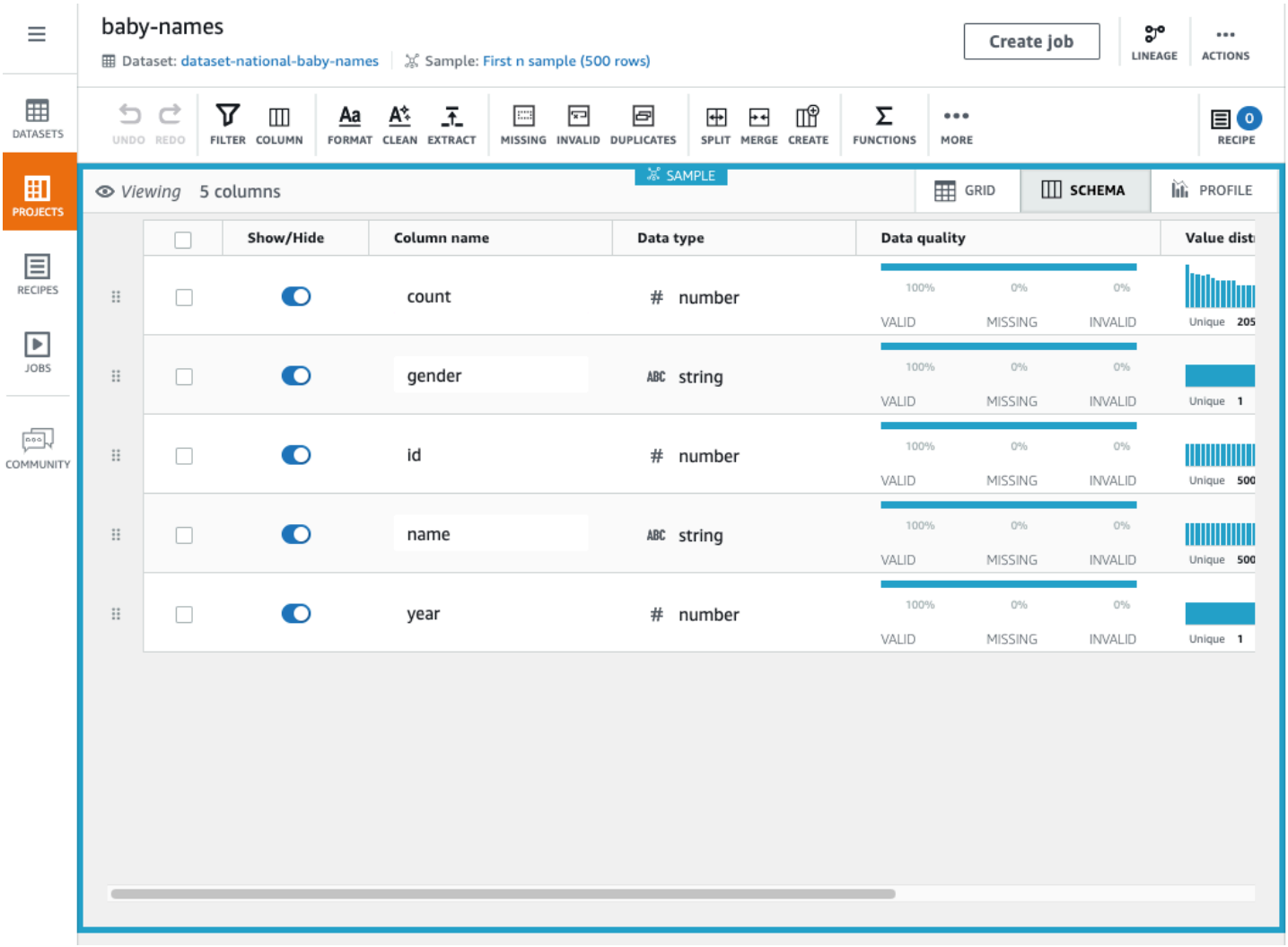
La pubblicazione delle ricette incoraggia anche un approccio incrementale e iterativo allo sviluppo delle ricette: puoi pubblicare nuove versioni della ricetta man mano che procedi, in modo da poter ricorrere a una versione della ricetta «ultimo prodotto conosciuto», se necessario.

Per pubblicare una nuova versione di una ricetta

- Nel riquadro della ricetta, scegliete Pubblica. Inserisci una descrizione per questa versione della ricetta e scegli Pubblica.

Visualizzazione dello schema

Se si sceglie la scheda SCHEMA, la visualizzazione cambia, come mostrato nella schermata seguente.



Nella visualizzazione schema, puoi visualizzare le statistiche sui valori dei dati in ogni colonna.

Nella colonna all'estrema sinistra, accanto a Show/Hide, scegli una delle colonne di dati. Il riquadro dei dettagli della colonna viene visualizzato a destra. Questo riquadro mostra un riepilogo delle statistiche per i valori delle colonne.

È possibile rinominare una colonna inserendo un nuovo nome per Nome colonna.

È possibile riorganizzare l'ordine delle colonne trascinandole.

Visualizzazione del profilo

Se scegli la scheda PROFILO, puoi visualizzare informazioni volumetriche dettagliate sul tuo progetto. Prima di farlo, esegui un DataBrew lavoro per creare il profilo.

Per una panoramica della visualizzazione del profilo

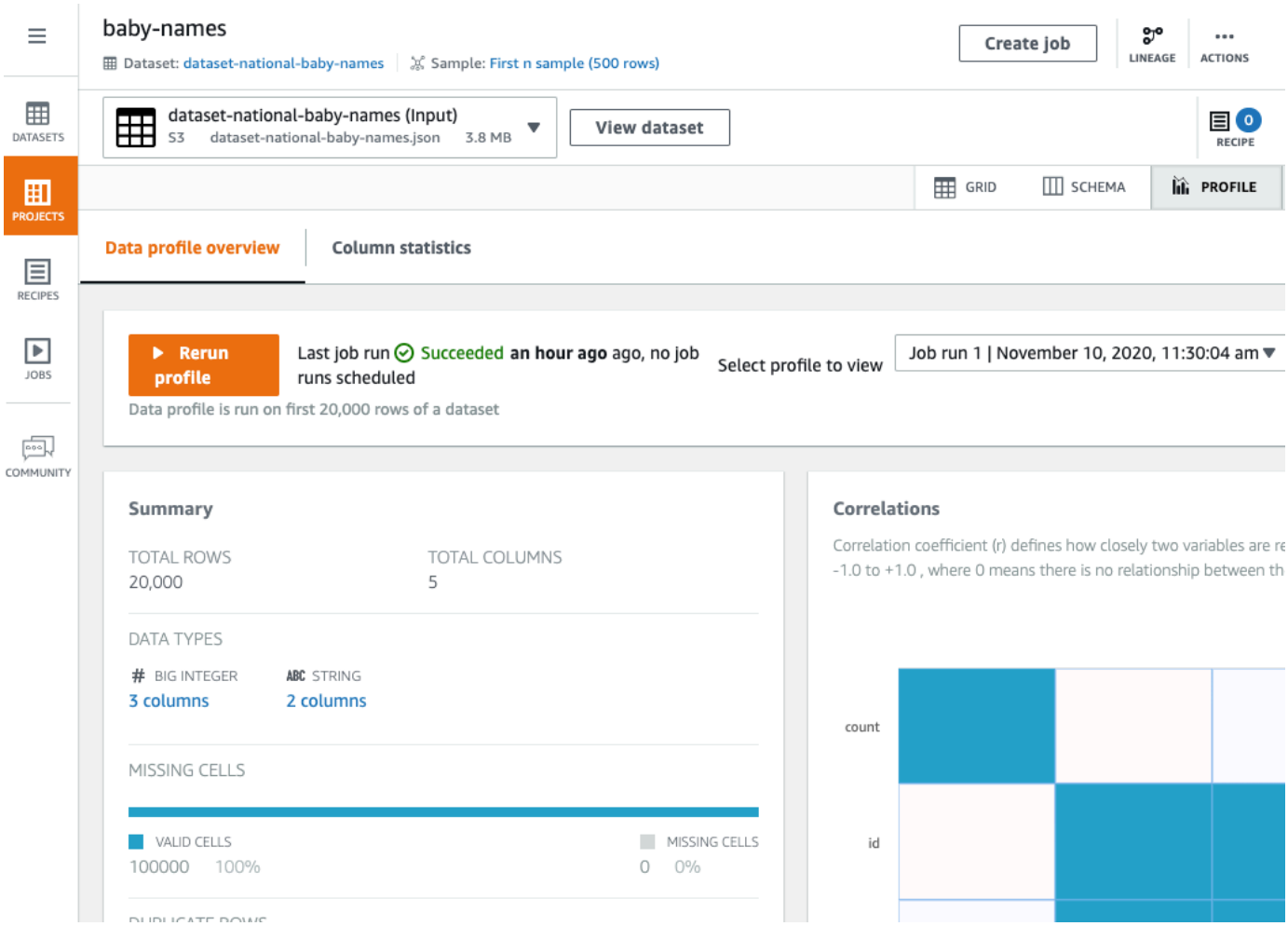
1. Scegli Crea lavoro e inserisci un nome per il tuo lavoro.
2. Per Job output, scegliete CSV per il tipo di file.
3. Trova o crea un bucket e una cartella Amazon S3 nel tuo AWS account in cui desideri che venga scritto l'output del DataBrew lavoro:

   - Se hai già questo bucket e questa cartella Amazon S3, scegli Sfoglia e individuali. Assicurati di disporre delle autorizzazioni di scrittura per entrambi.
   - Se non hai questo bucket e questa cartella Amazon S3, creali:

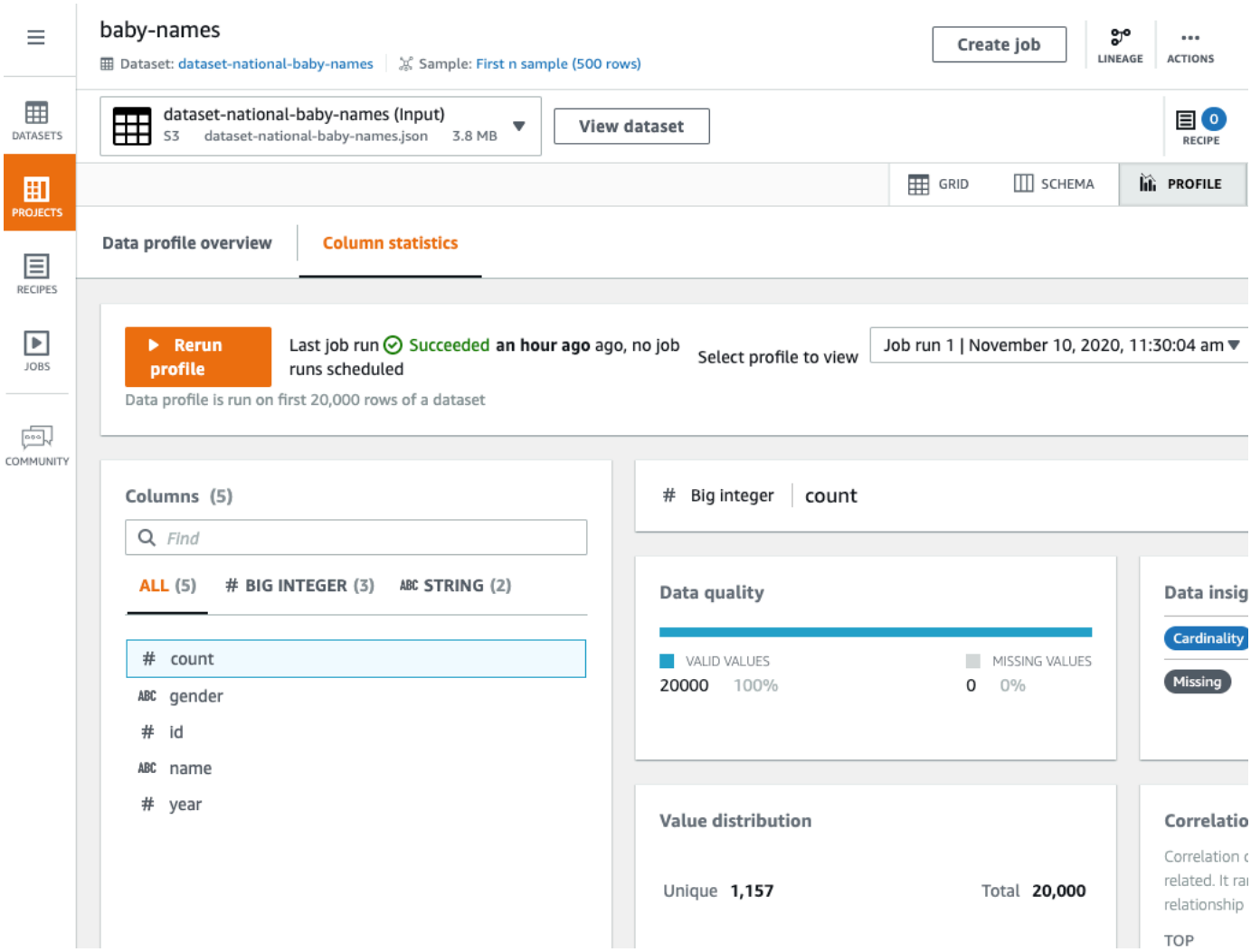
     1. Apri la console Amazon S3 all'indirizzo. https://console.aws.amazon.com/s3/
     2. Se non disponi di un bucket Amazon S3, scegli Crea bucket. Per Bucket name, inserisci un nome univoco per il tuo nuovo bucket. Seleziona Crea bucket.
     3. Dall'elenco dei bucket, scegli quello che desideri utilizzare.
     4. Scegliere Create folder (Crea cartella). Per Nome cartella`databrew-output`, immettete e scegliete Crea cartella.
4. Per le autorizzazioni di accesso, scegli un ruolo IAM che DataBrew consenta di scrivere nella tua posizione di output Amazon S3.

   Per una sede S3 di proprietà del tuo AWS account, puoi scegliere il ruolo di gestione del `AwsGlueDataBrewDataAccessRole` servizio. In questo modo puoi accedere DataBrew alle risorse S3 di tua proprietà.
5. Lascia le altre impostazioni ai valori predefiniti e scegli Crea ed esegui il processo.
6. Una volta completato il processo, l'area di lavoro visualizza un riepilogo grafico del profilo dei dati.

   La scheda Panoramica del profilo di dati mostra un riepilogo di alto livello delle caratteristiche dei dati, come mostrato nella schermata seguente.



La scheda Statistiche delle colonne mostra una suddivisione colonna per colonna dei valori dei dati:



Eliminazione di un progetto

Se non hai più bisogno di un progetto, puoi eliminarlo.

Come eliminare un progetto

1. Nel riquadro di navigazione, scegli PROGETTI.
2. Scegli il progetto che desideri eliminare, quindi per Azioni, scegli Elimina. .

Creazione e utilizzo AWS Glue DataBrew recipes

In DataBrew, una ricetta è un insieme di passaggi di trasformazione dei dati. Puoi applicare questi passaggi a un campione di dati o applicare la stessa ricetta a un set di dati.

Il modo più semplice per sviluppare una ricetta è creare un DataBrew progetto, in cui puoi lavorare in modo interattivo con un campione dei tuoi dati. Per ulteriori informazioni, consulta. Creazione e utilizzo AWS Glue DataBrew progetti Come parte del flusso di lavoro per la creazione del progetto, viene creata e allegata al progetto una nuova ricetta (vuota). Puoi quindi iniziare a creare la tua ricetta aggiungendo trasformazioni di dati.

> **Note**
>
> Puoi includere fino a 100 trasformazioni di dati in una singola DataBrew ricetta.

Man mano che procedi con lo sviluppo della ricetta, puoi salvare il lavoro pubblicando la ricetta. DataBrew mantiene un elenco di versioni pubblicate della ricetta. Puoi utilizzare qualsiasi versione pubblicata in un processo di ricetta, per eseguire la ricetta (in un processo di ricetta) per trasformare il tuo set di dati. Puoi anche scaricare una copia dei passaggi della ricetta, in modo da riutilizzare la ricetta in altri progetti o altre trasformazioni di set di dati.

Puoi anche sviluppare DataBrew ricette a livello di codice, utilizzando AWS Command Line Interface(AWS CLI) o uno degli SDK.AWSNell' DataBrew API, le trasformazioni sono note come azioni di ricetta.

> **Note**
>
> In una sessione di DataBrew progetto interattiva, ogni trasformazione di dati applicata comporta una chiamata all' DataBrew API. Queste chiamate API avvengono automaticamente, senza che tu debba conoscere i dettagli dietro le quinte.

Anche se non sei un programmatore, è utile comprendere la struttura di una ricetta e come DataBrew organizza le azioni della ricetta.

Argomenti

Pubblicazione di una nuova versione della ricetta

Pubblichi nuove versioni di una ricetta in una sessione interattiva DataBrew del progetto.

Per pubblicare una nuova versione della ricetta

1. Nel riquadro della ricetta, scegliete Pubblica.
2. Inserisci una descrizione per questa versione della ricetta e scegli Pubblica.

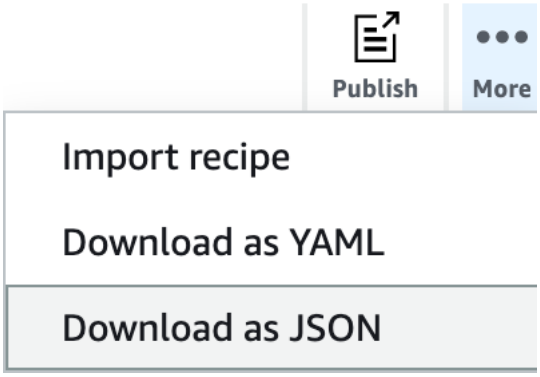
Puoi visualizzare tutte le ricette pubblicate e le relative versioni scegliendo PROGETTI dal pannello di navigazione.

Definizione della struttura di una ricetta

Quando crei per la prima volta un progetto utilizzando la DataBrew console, definisci una ricetta da associare a quel progetto. Se non hai una ricetta esistente, la console ne crea una per te.

Mentre lavori con il tuo progetto nella console, usi la barra degli strumenti di trasformazione per applicare azioni ai dati di esempio del tuo set di dati. La console mostra i passaggi della ricetta e l'ordine di tali passaggi, man mano che continui a creare la ricetta. Puoi iterare e rifinire la ricetta finché non sei soddisfatto dei passaggi.

NelNozioni di base su AWS Glue DataBrew, crei una ricetta per trasformare un set di dati di famose partite di scacchi. Puoi scaricare una copia dei passaggi della ricetta, scegliendo Scarica come JSON o Scarica come YAML, come mostrato nella schermata seguente.



Il file JSON scaricato contiene le azioni relative alla ricetta corrispondenti alle trasformazioni che hai aggiunto alla ricetta.

Una nuova ricetta non prevede passaggi. Puoi rappresentare una nuova ricetta come un elenco JSON vuoto, come mostrato di seguito.

```
[ ]
```

Di seguito è riportato un esempio di tale file, per`chess-project-recipe`. L'elenco JSON contiene diversi oggetti che descrivono i passaggi della ricetta. Ogni oggetto nell'elenco JSON è racchiuso tra parentesi graffe (). `{ }` Le righe JSON sono delimitate da virgole.

```
[
    {
        "Action": {
            "Operation": "REMOVE_VALUES",
            "Parameters": {
                "sourceColumn": "black_rating"
            }
        },
        "ConditionExpressions": [
            {
                "Condition": "LESS_THAN",
                "Value": "1800",
                "TargetColumn": "black_rating"
            }
        ]
    },
    {
        "Action": {
            "Operation": "REMOVE_VALUES",
            "Parameters": {
                "sourceColumn": "white_rating"
            }
        },
        "ConditionExpressions": [
            {
                "Condition": "LESS_THAN",
                "Value": "1800",
                "TargetColumn": "white_rating"
            }
        ]
```

```
    },
    {
        "Action": {
            "Operation": "GROUP_BY",
            "Parameters": {
                "groupByAggFunctionOptions": "[{\"sourceColumnName\":\"winner\",
\"targetColumnName\":\"winner_count\",\"targetColumnDataType\":\"int\",\"functionName
\":\"COUNT\"}]",
                "sourceColumns": "[\"winner\",\"victory_status\"]",
                "useNewDataFrame": "true"
            }
        }
    },
    {
        "Action": {
            "Operation": "REMOVE_VALUES",
            "Parameters": {
                "sourceColumn": "winner"
            }
        },
        "ConditionExpressions": [
            {
                "Condition": "IS",
                "Value": "[\"draw\"]",
                "TargetColumn": "winner"
            }
        ]
    },
    {
        "Action": {
            "Operation": "REPLACE_TEXT",
            "Parameters": {
                "pattern": "mate",
                "sourceColumn": "victory_status",
                "value": "checkmate"
            }
        }
    },
    {
        "Action": {
            "Operation": "REPLACE_TEXT",
            "Parameters": {
                "pattern": "resign",
                "sourceColumn": "victory_status",
```

```
                "value": "other player resigned"
            }
        }
    },
    {
        "Action": {
            "Operation": "REPLACE_TEXT",
            "Parameters": {
                "pattern": "outoftime",
                "sourceColumn": "victory_status",
                "value": "ran out of time"
            }
        }
    }
]
```

È più facile vedere che ogni azione è una riga individuale se aggiungiamo solo nuove righe per nuove azioni, come mostrato di seguito.

```
[
 { "Action": { "Operation": "REMOVE_VALUES", "Parameters": { "sourceColumn":
 "black_rating" } }, "ConditionExpressions": [ { "Condition": "LESS_THAN", "Value":
 "1800", "TargetColumn": "black_rating" } ] },
 { "Action": { "Operation": "REMOVE_VALUES", "Parameters": { "sourceColumn":
 "white_rating" } }, "ConditionExpressions": [ { "Condition": "LESS_THAN", "Value":
 "1800", "TargetColumn": "white_rating" } ] },
 { "Action": { "Operation": "GROUP_BY", "Parameters": { "groupByAggFunctionOptions":
 "[{\"sourceColumnName\":\"winner\",\"targetColumnName\":\"winner_count\",
\"targetColumnDataType\":\"int\",\"functionName\":\"COUNT\"}]", "sourceColumns":
 "[\"winner\",\"victory_status\"]", "useNewDataFrame": "true" } } },
 { "Action": { "Operation": "REMOVE_VALUES", "Parameters": { "sourceColumn":
 "winner" } }, "ConditionExpressions": [ { "Condition": "IS", "Value": "[\"draw\"]",
 "TargetColumn": "winner" } ] },
 { "Action": { "Operation": "REPLACE_TEXT", "Parameters": { "pattern": "mate",
 "sourceColumn": "victory_status", "value": "checkmate" } } },
 { "Action": { "Operation": "REPLACE_TEXT", "Parameters": { "pattern": "resign",
 "sourceColumn": "victory_status", "value": "other player resigned" } } },
 { "Action": { "Operation": "REPLACE_TEXT", "Parameters": { "pattern": "outoftime",
 "sourceColumn": "victory_status", "value": "ran out of time" } } }
]
```

Le azioni vengono eseguite in sequenza, nello stesso ordine del file:

- `REMOVE_VALUES`— Per filtrare tutte le partite in cui il punteggio di un giocatore è inferiore a 1.800, il punteggio minimo richiesto per essere un giocatore di scacchi di classe A. Questa azione si ripete in due occasioni: una per rimuovere i giocatori dal lato nero che non appartengono almeno alla classe A, e un'altra per rimuovere i giocatori dalla parte bianca che non sono a questo livello.
- `GROUP_BY`— Per riassumere i dati. In questo caso, GROUP_BY ordina le righe in gruppi in base ai valori di `winner` (e). `black` `white` Ciascuno di questi gruppi viene quindi ulteriormente suddiviso, ordinando le righe in sottogruppi in base ai valori di `victory_status` (`mate`,, `resign` e). `outoftime` `draw` Infine, viene contato il numero di occorrenze per ogni sottogruppo. Il riepilogo risultante sostituisce quindi il campione di dati originale.
- `REMOVE_VALUES`— Per eliminare i risultati delle partite terminate con`draw`.
- `REPLACE_TEXT`— Per modificare i valori di`victory_status`. Esistono tre occorrenze di questa azione, una per`mate`, e. `resign` `outoftime`

In una sessione interattiva DataBrew del progetto, ciascuna `RecipeAction` corrisponde a una trasformazione dei dati che si applica a un campione di dati.

DataBrew fornisce oltre 200 azioni relative alle ricette. Per ulteriori informazioni, consulta Fase della ricetta e riferimento alla funzione.

Utilizzo delle condizioni

È possibile utilizzare le condizioni per restringere l'ambito di un'azione relativa alla ricetta. Le condizioni vengono utilizzate nelle trasformazioni che filtrano i dati, ad esempio rimuovendo le righe indesiderate in base a un particolare valore di colonna.

Diamo un'occhiata più da vicino alle azioni di una ricetta. `chess-project-recipe`

```
{
  "Action": {
    "Operation": "REMOVE_VALUES",
    "Parameters": {
      "sourceColumn": "black_rating"
    }
  },
  "ConditionExpressions": [
    {
      "Condition": "LESS_THAN",
      "Value": "1800",
      "TargetColumn": "black_rating"
```

```
      }
    ]
  }
```

Questa trasformazione legge i valori nella `black_rating` colonna. L'`ConditionExpressions`elenco determina i criteri di filtro: qualsiasi riga con un `black_rating` valore inferiore a 1.800 viene rimossa dal set di dati.

Una successiva trasformazione nella ricetta fa la stessa cosa, per. `white_rating` In questo modo, i dati sono limitati ai giochi in cui ogni giocatore (bianco o nero) è classificato nella classe A o superiore.

Ecco un altro esempio di condizione, applicata a una colonna di dati relativi a un personaggio.

```
{
  "Action": {
    "Operation": "REMOVE_VALUES",
    "Parameters": {
      "sourceColumn": "winner"
    }
  },
  "ConditionExpressions": [
    {
      "Condition": "IS",
      "Value": "[\"draw\"]",
      "TargetColumn": "winner"
    }
  ]
}
```

Questa trasformazione legge i valori nella `winner` colonna, cerca il valore `draw` e rimuove quelle righe. In questo modo, i dati sono limitati solo ai giochi in cui c'è stato un chiaro vincitore.

DataBrew supporta le seguenti condizioni:

- `IS`— Il valore nella colonna è uguale al valore fornito nella condizione.
- `IS_NOT`— Il valore nella colonna non è lo stesso del valore fornito nella condizione.
- `IS_BETWEEN`— Il valore nella colonna è compreso tra i `LESS_THAN_EQUAL` parametri `GREATER_THAN_EQUAL` e.
- `CONTAINS`— Il valore della stringa nella colonna contiene il valore fornito nella condizione.

- `NOT_CONTAINS`— Il valore nella colonna non contiene la stringa di caratteri fornita nella condizione.
- `STARTS_WITH`— Il valore nella colonna inizia con la stringa di caratteri fornita nella condizione.
- `NOT_STARTS_WITH`— Il valore nella colonna non inizia con la stringa di caratteri fornita nella condizione.
- `ENDS_WITH`— Il valore nella colonna termina con la stringa di caratteri fornita nella condizione.
- `NOT_ENDS_WITH`— Il valore nella colonna non termina con la stringa di caratteri fornita nella condizione.
- `LESS_THAN`— Il valore nella colonna è inferiore al valore fornito nella condizione.
- `LESS_THAN_EQUAL`— Il valore nella colonna è inferiore o uguale al valore fornito nella condizione.
- `GREATER_THAN`— Il valore nella colonna è maggiore del valore fornito nella condizione.
- `GREATER_THAN_EQUAL`— Il valore nella colonna è maggiore o uguale al valore fornito nella condizione.
- `IS_INVALID`— Il valore nella colonna ha un tipo di dati errato.
- `IS_MISSING`— Non vi è alcun valore nella colonna.

Creazione, esecuzione e pianificazione AWS Glue DataBrew jobs

AWS Glue DataBrew dispone di un sottosistema di lavoro che serve a due scopi:

1. Applicazione di una ricetta di trasformazione dei dati a un DataBrew set di dati. Lo fai con un lavoro di DataBrew ricetta.
2. Analisi di un set di dati per creare un profilo completo dei dati. Lo fai con un lavoro di DataBrew profilo.

Argomenti

Creare e lavorare con AWS Glue DataBrew lavori di ricette

Usa un processo di DataBrew ricetta per pulire e normalizzare i dati in un DataBrew set di dati e scrivi il risultato in una posizione di output a tua scelta. L'esecuzione di un processo di ricetta non influisce sul set di dati o sui dati di origine sottostanti. Quando un processo viene eseguito, si connette ai dati di origine in modalità di sola lettura. L'output del lavoro viene scritto in una posizione di output definita in Amazon S3 AWS Glue Data Catalog, o in un database JDBC supportato.

Utilizza la seguente procedura per creare un DataBrew processo di elaborazione delle ricette.

Per creare un processo di creazione di ricette

1. Accedi a Console di gestione AWS e apri la DataBrew console all'indirizzo https://console.aws.amazon.com/databrew/.
2. Scegli JOBS dal pannello di navigazione, scegli la scheda Recipe jobs, quindi scegli Crea lavoro.
3. Inserisci un nome per il tuo lavoro, quindi scegli Crea un lavoro di ricetta.
4. Per Job input, inserisci i dettagli sul lavoro che desideri creare: il nome del set di dati da elaborare e la ricetta da utilizzare.

   Un processo di ricetta utilizza una DataBrew ricetta per trasformare un set di dati. Per utilizzare una ricetta, assicurati di pubblicarla prima.

5. Configura le impostazioni di output del lavoro.

   Fornisci una destinazione per la tua produzione lavorativa. Se non disponi di una DataBrew connessione configurata per la destinazione di output, configurala prima nella scheda DATASETS come descritto inConnessioni supportate per sorgenti e uscite di dati. Scegliete una delle seguenti destinazioni di output:

   - Amazon S3, con o senza supporto AWS Glue Data Catalog
   - Amazon Redshift, con o senza supporto AWS Glue Data Catalog
   - JDBC
   - Tavoli Snowflake
   - Tabelle di database Amazon RDS con AWS Glue Data Catalog supporto. Le tabelle di database Amazon RDS supportano i seguenti motori di database:
     - Amazon Aurora
     - MySQL
     - Oracle
     - PostgreSQL
     - Microsoft SQL Server
   - Amazon S3 con AWS Glue Data Catalog supporto.

   Per l'AWS Glue Data Catalog output basato su AWS Lake Formation, DataBrew supporta solo la sostituzione di file esistenti. In questo approccio, i file vengono sostituiti per mantenere intatte le autorizzazioni esistenti di Lake Formation per il tuo ruolo di accesso ai dati. Inoltre, DataBrew dà la precedenza alla posizione di Amazon S3 nella tabella.AWS Glue Data Catalog Pertanto, non puoi sovrascrivere la posizione di Amazon S3 durante la creazione di un processo di creazione di ricette.

   In alcuni casi, la posizione di Amazon S3 nell'output del lavoro è diversa dalla posizione Amazon S3 nella tabella Data Catalog. In questi casi, DataBrew aggiorna automaticamente la definizione del processo con la posizione Amazon S3 dalla tabella del catalogo. Lo fa quando aggiorni o avvii i lavori esistenti.

6. Solo per le destinazioni di output di Amazon S3, hai altre scelte:

   a. Scegli uno dei formati di output dei dati disponibili per Amazon S3, la compressione opzionale e un delimitatore personalizzato opzionale. I delimitatori supportati per i file di

output sono gli stessi di quelli di input: virgola, virgola, punto e virgola, pipe, tab, caret, backslash e space. Per i dettagli sulla formattazione, consultate la tabella seguente.

Format (Formato)	Estensione del file (non compressa)	Estensioni di file (compresse)
Comma-separated valori	`.csv`	`.csv.snappy` , `.csv.gz`, `.csv.lz4`, `csv.bz2`, `.csv.deflate` , `csv.br`
Tab-separated valori	`.csv`	`.tsv.snappy` , `.tsv.gz`, `.tsv.lz4`, `tsv.bz2`, `.tsv.deflate` , `tsv.br`
Apache Parquet	`.parquet`	`.parquet.snappy` , `.parquet.gz` , `.parquet.lz4` , `.parquet.lzo` , `.parquet.br`
AWS Glue parquet	Non supportata	`.glue.parquet.snappy`
Apache Avro	`.avro`	`.avro.snappy` , `.avro.gz`, `.avro.lz4` , `.avro.bz2` , `.avro.deflate` , `.avro.br`
Apache ORC	`.orc`	`.orc.snappy` , `.orc.lzo`, `.orc.zlib`
XML	`.xml`	`.xml.snappy` , `.xml.gz`, `.xml.lz4`, `.xml.bz2`, `.xml.deflate` , `.xml.br`

Format (Formato)	Estensione del file (non compressa)	Estensioni di file (compresse)
JSON (solo formato JSON Lines)	`.json`	`.json.snappy`, `.json.gz`, `.json.lz4`, `.json.bz2`, `.json.deflate`, `.json.br`
Tableau Hyper	Non supportata	Non applicabile

b. Scegli se stampare un singolo file o più file. Esistono tre opzioni per l'output di file con Amazon S3:

- Generazione automatica di file (scelta consigliata): consente di DataBrew determinare il numero ottimale di file di output.
- Output a file singolo: consente la generazione di un singolo file di output. Questa opzione può comportare un ulteriore tempo di esecuzione del lavoro poiché è necessaria la post-elaborazione.
- Output di più file: consente di specificare il numero di file per l'output del lavoro. I valori validi sono 2—999. È possibile che venga emesso un numero inferiore di file di quelli specificati se si utilizza il partizionamento delle colonne o se il numero di righe nell'output è inferiore al numero di file specificato.

c. (Facoltativo) Scegliete il partizionamento delle colonne per l'output del processo di creazione delle ricette.

Il partizionamento delle colonne offre un altro modo per suddividere l'output del processo di preparazione della ricetta in più file. Il partizionamento delle colonne può essere utilizzato con un output Amazon S3 nuovo o esistente o con un nuovo output Amazon S3 di Data Catalog. Non può essere utilizzato con le tabelle Amazon S3 di Data Catalog esistenti. I file di output si basano sui valori dei nomi delle colonne specificati. Se i nomi delle colonne specificati sono univoci, i percorsi delle cartelle Amazon S3 risultanti si basano sull'ordine dei nomi delle colonne.

Per un esempio di partizionamento delle colonne, consulta Esempio di partizionamento delle colonne quanto segue.

7. (Facoltativo) Scegliete Abilita la crittografia per l'output del lavoro per crittografare l'output del lavoro che DataBrew scrive nella posizione di output, quindi scegliete il metodo di crittografia:

   - Usa la SSE-S3 crittografia: l'output viene crittografato utilizzando la crittografia lato server con chiavi di crittografia gestite da Amazon S3.
   - Usa AWS Key Management Service(AWS KMS): l'output viene crittografato utilizzando.AWS KMS Per utilizzare questa opzione, scegli l'Amazon Resource Name (ARN) della AWS KMS chiave che desideri utilizzare. Se non disponi di una AWS KMS chiave, puoi crearne una scegliendo Crea una AWS KMS chiave.

8. Per le autorizzazioni di accesso, scegli un ruolo AWS Identity and Access Management(IAM) che DataBrew consenta di scrivere nella tua posizione di output. Per una sede di proprietà del tuo AWS account, puoi scegliere il ruolo gestito dal `AwsGlueDataBrewDataAccessRole` servizio. In questo modo puoi accedere DataBrew alle AWS risorse di tua proprietà.

9. Nel riquadro Impostazioni avanzate del processo, puoi scegliere altre opzioni per l'esecuzione del lavoro:

   - Numero massimo di unità: DataBrew elabora i lavori utilizzando più nodi di elaborazione, in esecuzione in parallelo. Il numero predefinito di nodi è 5. Il numero massimo di nodi è 149.
   - Job timeout: se l'esecuzione di un processo richiede più del numero di minuti impostato qui, fallisce con un errore di timeout. Il valore predefinito è 2.880 minuti o 48 ore.
   - Numero di tentativi: se un processo fallisce durante l'esecuzione, DataBrew puoi provare a eseguirlo di nuovo. Per impostazione predefinita, il processo non viene ritentato.
   - Abilita Amazon CloudWatch Logs for job: consente di DataBrew pubblicare informazioni diagnostiche su CloudWatch Logs. Questi log possono essere utili per la risoluzione dei problemi o per maggiori dettagli su come viene elaborato il lavoro.

10. Per Schedule jobs, è possibile applicare una DataBrew pianificazione dei lavori in modo che il lavoro venga eseguito in un determinato momento o su base ricorrente. Per ulteriori informazioni, consulta Automatizzazione delle esecuzioni dei lavori con una pianificazione.

11. Quando le impostazioni sono quelle che desideri, scegli Crea lavoro. Oppure, se desideri eseguire il lavoro immediatamente, scegli Crea ed esegui processo.

Puoi monitorare l'avanzamento del lavoro controllandone lo stato mentre il lavoro è in esecuzione. Una volta completata l'esecuzione del processo, lo stato passa a Riuscito. L'output del lavoro è ora disponibile nella posizione di output prescelta.

DataBrew salva la definizione del processo, in modo da poter eseguire lo stesso lavoro in un secondo momento. Per eseguire nuovamente un lavoro, scegli Jobs dal pannello di navigazione. Scegli il lavoro con cui vuoi lavorare, quindi scegli Esegui lavoro.

Esempio di partizionamento delle colonne

Come esempio di partizionamento delle colonne, si supponga di specificare tre colonne, ciascuna delle quali contiene uno dei due valori possibili. La `Dept` colonna può avere il valore `Admin` o. `Eng` La `Staff-type` colonna può avere il valore `Part-time` o`Full-time`. La `Location` colonna può avere il valore `Office1` o`Office2`. I bucket Amazon S3 per il tuo output di lavoro hanno un aspetto simile al seguente.

```
s3://bucket/output-folder/Dept=Admin/Staff-type=Part-time/Area=Office1/
jobId_timestamp_part0001.csv
s3://bucket/output-folder/Dept=Admin/Staff-type=Part-time/Location=Office2/
jobId_timestamp_part0002.csv
s3://bucket/output-folder/Dept=Admin/Staff-type=Full-time/Location=Office1/
jobId_timestamp_part0003.csv
s3://bucket/output-folder/Dept=Admin/Staff-type=Full-time/Location=Office2/
jobId_timestamp_part0004.csv
s3://bucket/output-folder/Dept=Eng/Staff-type=Part-time/Location=Office1/
jobId_timestamp_part0005.csv
s3://bucket/output-folder/Dept=Eng/Staff-type=Part-time/Location=Office2/
jobId_timestamp_part0006.csv
s3://bucket/output-folder/Dept=Eng/Staff-type=Full-time/Location=Office1/
jobId_timestamp_part0007.csv
s3://bucket/output-folder/Dept=Eng/Staff-type=Full-time/Location=Office2/
jobId_timestamp_part0008.csv
```

Automatizzazione delle esecuzioni dei lavori con una pianificazione

È possibile eseguire nuovamente i DataBrew lavori in qualsiasi momento e anche automatizzare le esecuzioni dei lavori con una DataBrew pianificazione.

Per eseguire nuovamente un processo DataBrew

1. Accedi a Console di gestione AWS e apri la DataBrew console all'indirizzo. https://console.aws.amazon.com/databrew/
2. Nel riquadro di navigazione, scegli Jobs. Scegli il lavoro che desideri eseguire, quindi scegli Esegui processo.

Per eseguire un DataBrew lavoro in un momento particolare o su base ricorrente, crea una pianificazione dei DataBrew lavori. È quindi possibile impostare il lavoro in modo che venga eseguito in base alla pianificazione.

Per creare una pianificazione dei DataBrew lavori

1. Nel riquadro di navigazione della DataBrew console, scegli Jobs. Scegli la scheda Pianificazioni e scegli Aggiungi pianificazione.
2. Inserisci un nome per la tua pianificazione, quindi scegli un valore per Frequenza di esecuzione:
   - Periodico: scegli la frequenza con cui desideri che il processo venga eseguito (ad esempio, ogni 12 ore). Quindi scegli il giorno o i giorni in cui eseguire il lavoro. Facoltativamente, è possibile inserire l'ora del giorno in cui viene eseguito il lavoro.
   - A un'ora particolare: immettere l'ora del giorno in cui si desidera che il lavoro venga eseguito. Quindi scegli il giorno o i giorni in cui eseguire il lavoro.
   - Inserisci CRON: definisci la pianificazione dei lavori inserendo un'espressione cron valida. Per ulteriori informazioni, consulta Lavorare con le espressioni cron per i lavori di creazione di ricette.
3. Dopo aver selezionato le impostazioni desiderate, scegli Save (Salva).

Per associare un lavoro a una pianificazione

1. Nel riquadro di navigazione, scegli Jobs.
2. Scegli il lavoro su cui vuoi lavorare, quindi per Azioni, scegli Modifica. .
3. Nel riquadro Pianifica lavori, scegli Associa pianificazione. Scegli il nome della pianificazione che desideri utilizzare.
4. Dopo aver selezionato le impostazioni desiderate, scegli Save (Salva).

Lavorare con le espressioni cron per i lavori di creazione di ricette

Le espressioni Cron hanno sei campi obbligatori separati da uno spazio vuoto. La sintassi è esposta di seguito.

```
Minutes Hours Day-of-month Month Day-of-week Year
```

Nella sintassi precedente, i seguenti valori e caratteri jolly vengono utilizzati per i campi indicati.

Campi	Valori	Caratteri jolly
Minuti	0-59	, - * /
Ore	0-23	, - * /
Day-of-month	1-31	, - * ? / L W
Mese	1—12 oppure JAN-DEC	, - * /
Day-of-week	1—7 o SUN-SAT	, - * ? / L
Anno	1970–2199	, - * /

Usa questi caratteri jolly come segue:

- Il carattere jolly , (virgola) include valori aggiuntivi. Nel `Month` campo, `JAN,FEB,MAR` include gennaio, febbraio e marzo.
- Il carattere jolly - (en dash) specifica gli intervalli. Nel `Day` campo, 1—15 include i giorni da 1 a 15 del mese specificato.
- Il carattere jolly * (asterisco) include tutti i valori nel campo. Nel `Hours` campo, * include ogni ora.
- Il carattere jolly / (barra) specifica gli incrementi. Nel `Minutes` campo è possibile inserire **`1/10`** per specificare ogni decimo minuto, a partire dal primo minuto dell'ora (ad esempio, l'undicesimo, il ventunesimo e il 31° minuto).
- Il carattere jolly ? (punto interrogativo) specifica un valore. Ad esempio, supponiamo che nel **`Day-of-month`** campo si inserisca 7. Se non ti interessa in che giorno della settimana è il settimo, puoi inserire? sul `Day-of-week` campo.
- Il carattere L nel `Day-of-week` campo `Day-of-month` or specifica l'ultimo giorno del mese o della settimana.
- Il carattere jolly W nel campo `Day-of-month` specifica un giorno feriale. Nel campo `Day-of-month`, `3W` specifica il giorno più vicino al terzo giorno feriale del mese.

Questi campi e valori presentano le seguenti limitazioni:

- Non puoi specificare i campi `Day-of-month` e `Day-of-week` nella stessa espressione cron. Se specifichi un valore in uno dei campi, devi usare un carattere ? nell'altro campo.

- Le espressioni Cron che portano a tassi superiori a 5 minuti non sono supportate.

Quando crei una pianificazione puoi utilizzare le seguenti stringhe cron di esempio.

Minuti	Ore	Giorno del mese	Mese	Giorno della settimana	Anno	Significato
0	10	*	*	?	*	Esegui ogni giorno alle 10:00 (UTC)
15	12	*	*	?	*	Esegui ogni giorno alle 12.15 (UTC)
0	18	?	*	MON-FRI	*	Esegui dal lunedì al venerdì alle 18.00 (UTC)
0	8	1	*	?	*	Esegui alle 8:00 (UTC) ogni primo giorno del mese
0/15	*	*	*	?	*	Esegui ogni 15 minuti
0/10	*	?	*	MON-FRI	*	Esegui dal lunedì al venerdì

Minuti	Ore	Giorno del mese	Mese	Giorno della settimana	Anno	Significato
						ogni 10 minuti
0/5	8-17	?	*	MON-FRI	*	Esegui dal lunedì al venerdì dalle 8.00 alle 17:55 (UTC) ogni 5 minuti

Ad esempio, puoi utilizzare la seguente espressione cron per eseguire un job ogni giorno alle 12:15 UTC.

```
15 12 * * ? *
```

Eliminazione di lavori e pianificazioni di lavoro

Se non hai più bisogno di un lavoro o di una pianificazione dei lavori, puoi eliminarli.

Come eliminare un processo

1. Nel riquadro di navigazione, scegli Jobs.
2. Scegli il lavoro che desideri eliminare, quindi per Azioni, scegli Elimina. .

Per eliminare una pianificazione dei lavori

1. Nel riquadro di navigazione, scegli Lavori, quindi scegli la scheda Pianificazioni.
2. Scegli la pianificazione che desideri eliminare, quindi per Azioni, scegli Elimina. .

Creare e lavorare con AWS Glue DataBrew lavori di profilo

I job di profilo eseguono una serie di valutazioni su un set di dati e inviano i risultati ad Amazon S3. Le informazioni raccolte dalla profilazione dei dati ti aiutano a comprendere il tuo set di dati e a decidere che tipo di fasi di preparazione dei dati potresti voler eseguire nei tuoi processi di elaborazione delle ricette.

Il modo più semplice per eseguire un processo di profilatura è utilizzare le impostazioni predefinite. DataBrew È possibile configurare il job di profilo prima di eseguirlo in modo che restituisca solo le informazioni desiderate.

Utilizzare la procedura seguente per creare un job DataBrew di profilo.

Per creare un profilo, un lavoro

1. Accedi a Console di gestione AWS e apri la DataBrew console all'indirizzo https://console.aws.amazon.com/databrew/.
2. Scegli JOBS dal pannello di navigazione, scegli la scheda Profile jobs, quindi scegli Crea lavoro.
3. Inserisci un nome per il tuo lavoro, quindi scegli Crea un profilo di lavoro.
4. Per Job input, fornisci il nome del set di dati da profilare.
5. (Facoltativo) Configurate quanto segue nel riquadro Configurazioni del profilo di dati:

   - Configurazioni a livello di set di dati: configura i dettagli del lavoro del tuo profilo per tutte le colonne del set di dati.

     Facoltativamente, puoi attivare la capacità di rilevare e contare le righe duplicate nel set di dati. Puoi anche scegliere Abilita la matrice di correlazioni e selezionare le colonne per vedere quanto strettamente sono correlati i valori in più colonne. Per i dettagli sulle statistiche che puoi configurare a livello di set di dati, consulta. Statistiche configurabili a livello di set di dati Puoi configurare le statistiche sulla DataBrew console o utilizzando l' DataBrewAPI o gli AWS SDK.
   - Configurazioni a livello di colonna: utilizzando le impostazioni di configurazione del profilo predefinite, puoi selezionare le colonne da includere nel lavoro del tuo profilo. Utilizza Aggiungi modifica alla configurazione per selezionare le colonne per le quali limitare il numero di statistiche raccolte o sostituire la configurazione predefinita di determinate statistiche. Per i dettagli sulle statistiche che è possibile configurare a livello di colonna, vedere. Statistiche configurabili a livello di colonna Puoi configurare le statistiche sulla DataBrew console o utilizzando l' DataBrew API o gli AWS SDK.

Assicurati che tutte le modifiche di configurazione specificate si applichino alle colonne che hai incluso nel lavoro del tuo profilo. Se ci sono conflitti tra diverse sostituzioni configurate per una colonna, l'ultima sovrascrittura in conflitto ha la priorità.

6. (Facoltativo) È possibile creare regole di qualità dei dati e applicare set di regole aggiuntivi associati a questo set di dati o rimuovere quelli già applicati. Per ulteriori informazioni sulla convalida della qualità dei dati, vedere. Convalida della qualità dei dati in AWS Glue DataBrew
7. Nel riquadro Impostazioni avanzate del processo, puoi scegliere altre opzioni per l'esecuzione del lavoro:
   - Numero massimo di unità: DataBrew elabora i lavori utilizzando più nodi di elaborazione, in esecuzione in parallelo. Il numero predefinito di nodi è 5. Il numero massimo di nodi è 149.
   - Job timeout: se l'esecuzione di un processo richiede più del numero di minuti impostato qui, fallisce con un errore di timeout. Il valore predefinito è 2.880 minuti o 48 ore.
   - Numero di tentativi: se un processo fallisce durante l'esecuzione, DataBrew puoi provare a eseguirlo di nuovo. Per impostazione predefinita, il processo non viene ritentato.
   - Abilita Amazon CloudWatch Logs for job: consente di DataBrew pubblicare informazioni diagnostiche su CloudWatch Logs. Questi log possono essere utili per la risoluzione dei problemi o per maggiori dettagli su come viene elaborato il lavoro.
8. Per la pianificazione associata, è possibile applicare una pianificazione delle DataBrew mansioni in modo che l'operazione venga eseguita in un determinato momento o su base ricorrente. Per ulteriori informazioni, consulta Automatizzazione delle esecuzioni dei lavori con una pianificazione.
9. Quando le impostazioni sono quelle che desideri, scegli Crea lavoro. Oppure, se desideri eseguire il lavoro immediatamente, scegli Crea ed esegui processo.

Creazione di una configurazione del lavoro di profilo in modo programmatico in AWS Glue DataBrew

In questa sezione, puoi trovare le descrizioni delle fasi e delle funzioni del processo di profilazione che puoi utilizzare a livello di codice. Puoi usarli da AWS Command Line Interface(AWS CLI) o utilizzando uno degli AWS SDK.

In un job di profilazione, puoi personalizzare una configurazione per controllare come DataBrew valuta il tuo set di dati. Puoi applicare la configurazione a un set di dati o applicarla a colonne

particolari. Puoi creare la configurazione quando crei un job di profilo e poi aggiornarla in qualsiasi momento.

Una struttura di configurazione del profilo include quattro parti:

- ProfileColumns sezione
- DatasetStatisticsConfiguration sezione
- ColumnStatisticsConfigurations sezione
- EntityDetectorConfiguration sezione per la configurazione delle PII

Di seguito è riportato un esempio.

```
{
    "ProfileColumns": [
        {
            "Name": "example"
        },
        {
            "Regex": "example.*"
        }
    ],
    "DatasetStatisticsConfiguration": {
        "IncludedStatistics": [
            "CORRELATION"
        ],
        "Overrides": [
            {
                "Statistic": "CORRELATION",
                "Parameters": {
                    "columnSelectors": "[{\"name\":\"example\"}, {\"regex\":\"example.*
\"}]"
                }
            }
        ]
    },
    "ColumnStatisticsConfigurations": [
        {
            "Selectors": [
                {
                    "Name": "example"
                }
```

```
            ],
            "Statistics": {
                "IncludedStatistics": [
                    "CORRELATION",
                    "DUPLICATE_ROWS_COUNT"
                ],
                "Overrides": [
                    {
                        "Statistic": "VALUE_DISTRIBUTION",
                        "Parameters": {
                            "binNumber": "10"
                        }
                    }
                ]
            }
        }
    ]
}
```

ProfileColumns sezione

Nella `ProfileColumns` sezione della tua struttura, imposta le colonne del tuo set di dati che desideri valutare nel tuo profilo job. `ProfileColumns`è un elenco di selettori di colonna ()`Selectors`. È possibile specificare un nome di colonna o un'espressione regolare in un selettore di colonne. Di seguito è riportato un esempio.

```
"ProfileColumns": [{"Name": "example"}, {"Regex": "example.*"}]
```

Quando `ProfileColumns` viene specificato, nel job del profilo `ProfileColumns` vengono incluse solo le colonne i cui nomi corrispondono a un nome o a un'espressione regolare in. Se il job del profilo non supporta il tipo di dati di una colonna selezionata, DataBrew salta la colonna selezionata durante l'esecuzione del lavoro.

Se non ProfileColumns è definito, il job di profilo valuta tutte le colonne supportate. Le colonne supportate sono colonne contenenti dati di un tipo di dati supportato:`ByteType`,`ShortType`,`IntegerType`,`LongType`,`FloatType`, `DoubleTypeString`, o. `Boolean`

DatasetStatisticsConfiguration sezione

Nella `DatasetStatisticsConfiguration` sezione della struttura, puoi creare una configurazione per le valutazioni tra colonne. La configurazione include `IncludedStatistics` e. `Overrides` Di seguito è riportato un esempio.

```
"DatasetStatisticsConfiguration": {
    "IncludedStatistics": ["CORRELATION"],
    "Overrides": [
        {
            "Statistic": "CORRELATION",
            "Parameters": {
                "columnSelectors": "[{\"name\":\"example\"}, {\"regex\":\"example.*
\"}]"
            }
        }
    ]
}
```

È possibile selezionare le valutazioni desiderate aggiungendo i nomi delle valutazioni`IncludedStatistics`. Di seguito è riportato un esempio.

```
"IncludedStatistics": ["CORRELATION", "DUPLICATE_ROWS_COUNT"]
```

Quando si specifica`IncludedStatistics`, solo le valutazioni presenti nell'elenco vengono incluse nella mansione del profilo. Se non `IncludedStatistics` è definito, il job del profilo esegue tutte le valutazioni supportate con le impostazioni predefinite. È possibile escludere tutte le valutazioni aggiungendo NONE a`IncludedStatistics`. Di seguito è riportato un esempio.

```
"IncludedStatistics": ["NONE"]
```

Statistiche configurabili a livello di set di dati

Nella `DatasetStatisticsConfiguration` sezione della struttura, un profilo job supporta le valutazioni mostrate nella tabella seguente.

Nome statistico	Descrizione	Tipi di dati supportati	Stato predefinito	Attributi del risultato del profilo	Tipo di risultato del profilo
DUPLICATE_ROWS_COUNT	Conteggio delle righe duplicate nel set di dati	tutto	Attiva	duplicatoRowsCoun	Int
CORRELAZIONE	Coefficiente di correlazione di Pearson tra due colonne	numero	Attiva	correlazioni (in ogni colonna selezionata)	Oggetto

In`IncludedStatistics`, puoi sovrascrivere le impostazioni predefinite di ogni valutazione aggiungendo un'eccezione. Ogni override include il nome di una particolare valutazione e una mappa dei parametri.

In`DatasetStatisticsConfiguration`, un job di profilo supporta l'`CORRELATIO`Noverride. Questa sovrascrittura calcola il coefficiente di correlazione di Pearson tra due colonne da un elenco di colonne selezionate. L'impostazione predefinita prevede la selezione delle prime 10 colonne numeriche. È possibile specificare un numero di colonne o un elenco di selettori di colonne per sovrascrivere l'impostazione predefinita.

`CORRELATIO`Naccetta questi parametri:

- `columnNumber`— Il numero di colonne numeriche. Il job del profilo seleziona le prime n colonne dal set di dati. Questo valore deve essere maggiore di 1. Utilizzare `"ALL"` per selezionare tutte le colonne numeriche.
- `columnSelectors:`— Elenco di selettori di colonne. Ogni selettore può avere un nome di colonna o un'espressione regolare.

Di seguito è riportato un esempio.

```
{
    "Statistic": "CORRELATION",
    "Parameters": {
        "columnSelectors": "[{\"name\":\"example\"}, {\"regex\":\"example.*\"}]"
    }
}
```

ColumnStatisticsConfigurations sezione

Nella `ColumnStatisticsConfigurations` sezione della struttura, puoi creare configurazioni per colonne particolari. `ColumnStatisticsConfigurations`è un elenco di `ColumnStatisticsConfiguration` impostazioni. In`ColumnStatisticsConfiguration`, ci sono `Selectors` un elenco di selettori di colonne e `Statistics` per la configurazione delle statistiche. Di seguito è riportato un esempio.

```
{
    "Selectors": [{"Name": "example"}
    ],
    "Statistics": {
       "IncludedStatistics": ["CORRELATION", "DUPLICATE_ROWS_COUNT"]
        "Overrides": [
            {
                "Statistic": "VALUE_DISTRIBUTION",
                "Parameters": {
                    "binNumber": "10"
                }
            }
        ]
    }
}
```

`Selectors`è un elenco di selettori di colonna. Analogamente`ProfileColumns`, è possibile specificare un nome di colonna o un'espressione regolare in ogni selettore di colonna. Quando si specifica`Selectors`, la configurazione delle colonne viene applicata alle colonne che corrispondono a qualsiasi selettore di colonna in. `Selectors` Altrimenti, la configurazione viene applicata a tutte le colonne supportate.

In`Statistics`, puoi sovrascrivere le impostazioni delle colonne selezionate. Come con`DatasetStatisticsConfiguration`, `Statistics` ha `IncludedStatistics` e`Overrides`.

Per selezionare le valutazioni desiderate, aggiungete i nomi delle valutazioni `aIncludedStatistics`.

```
"IncludedStatistics": ["CORRELATION", "DUPLICATE_ROWS_COUNT"]
```

Quando lo specificate`IncludedStatistics`, solo le valutazioni presenti nell'elenco vengono incluse nella mansione del profilo. In caso contrario, il job di profilo esegue tutte le valutazioni supportate con le impostazioni predefinite.

È possibile escludere tutte le valutazioni aggiungendole `NONE` `aIncludedStatistics`.

```
"IncludedStatistics": ["NONE"]
```

In alcuni casi, potrebbero esserci più configurazioni diverse `ColumnStatisticsConfigurations` `IncludedStatistics` che possono essere applicate alla stessa colonna. In questi casi, il job del profilo seleziona l'ultima configurazione `ColumnStatisticsConfigurations` e la applica `IncludedStatistics` alla colonna selezionata. Una nuova configurazione sostituisce le configurazioni precedenti.

Statistiche configurabili a livello di colonna

In`ColumnStatisticsConfigurations`, un job di profilo supporta le valutazioni mostrate nella tabella seguente.

Un tipo di dati supportato `number` in questa tabella indica che il tipo di dati dell'attributo è uno dei seguenti: `ByteTypeShortType`,`IntegerType`,`LongType`,`FloatType`, o`DoubleType`.

Nome statistico	Descrizione	Tipi di dati supportati	Stato predefinito	Attributi del risultato del profilo	Tipo di risultato del profilo
–	Nome della colonna.	tutto	–	nome	stringa

Nome statistico	Descrizione	Tipi di dati supportati	Stato predefinito	Attributi del risultato del profilo	Tipo di risultato del profilo
–	Tipo di dati della colonna.	tutto	–	tipo	stringa
DISTINCT_VALUES_COUNT	Numero di valori distinti. Un valore distinto è un valore che appare almeno una volta.	number/boolean/string	Abilitato	distintoValuesCount	Int
ENTROPIA	Entropia (teoria dell'informazione).	number/boolean/stringa	Abilitato	entropia	Double
INTER_QUARTILE_RANGE	Intervallo tra il 25 percento e il 75 percento dei numeri.	numero	Abilitato	Intervallo interquartile	Double
CURTOSI	Curtosi della colonna.	numero	Abilitato	curtosi	Double
MAX	Valore massimo nella colonna.	number/string lunghezza	Abilitato	max	Int/Double
VALORI_MASSIMI	Elenco dei valori massimi nella colonna e dei relativi conteggi.	numero	Abilitato	Valori massimi	List
MEAN	Valore medio dei valori nella colonna.	number/string lunghezza	Abilitato	mean	Double
MEDIAN	Mediana dei valori nella colonna.	number/string lunghezza	Abilitato	median	Double

Nome statistico	Descrizione	Tipi di dati supportati	Stato predefinito	Attributi del risultato del profilo	Tipo di risultato del profilo
DEVIAZIONE_ASSOLUTA MEDIANA	La mediana delle differenze assolute tra ogni punto dati e la mediana di una colonna numerica.	numero	Abilitato	medianaAbsoluteDeviation	Double
MIN	Valore minimo nella colonna.	number/string lunghezza	Abilitato	min	Int/Double
VALORI MINIMI	Elenco dei valori minimi nella colonna e dei relativi conteggi.	numero	Abilitato	Valori minimi	List
CONTO_VALORI_MANCANTI	Numero di valori mancanti nella colonna. Le stringhe nulle e vuote sono considerate mancanti.	tutto	Abilitato	mancantiValuesCount	Int
MODE	Il valore più frequente nella colonna. Se vengono visualizzati più valori con tale frequenza, la modalità è uno di quei valori.	number/string lunghezza	Abilitato	mode	Int/Double

Nome statistico	Descrizione	Tipi di dati supportati	Stato predefinito	Attributi del risultato del profilo	Tipo di risultato del profilo
VALORI PIÙ COMUNI	Elenco dei valori più comuni nella colonna.	number/boolean/string	Abilitato	la maggior parte CommonValues	List
OUTLIER_DETECTION	Rileva i valori anomali nella colonna mediante l'algoritmo Z_score. Conta il numero di valori anomali ed estrai un elenco di campioni dai valori anomali rilevati.	number/string lunghezza	Abilitato	zScoreOutliersCount, zScoreOutliersSample	Int/List
PERCENTILI	Valori percentili della colonna numerica (5%, 25%, 75%, 95%).	numero	Abilitato	percentile 5, percentile 25, percentile 75, percentile 95	Double
RANGE	Intervallo di valori nella colonna.	numero	Abilitato	range	Int/Double
ASIMMETRIA	Asimmetria dei valori nella colonna.	numero	Abilitato	asimmetria	Double

Nome statistico	Descrizione	Tipi di dati supportati	Stato predefinito	Attributi del risultato del profilo	Tipo di risultato del profilo
DEVIAZIONE STANDARD	Esempio imparziale di deviazione standard dei valori nella colonna.	number/string lunghezza	Abilitato	deviazione standard	Double
SUM	Somma dei valori nella colonna.	numero	Abilitato	sum	Int/Double
UNIQUE_VALUES_COUNT	Numero di valori univoci. Un valore univoco significa che il valore viene visualizzato una sola volta.	number/boolean/string	Abilitato	unicoValuesCount	Int
DISTRIBUZIONE DEL VALORE	Misura della distribuzione dei valori nella colonna per intervallo.	number/string lunghezza	Abilitato	Distribuzione del valore	List
VARIANCE	Varianza dei valori nella colonna.	numero	Abilitato	variance	Double
Z_SCORE_DISTRIBUTION	Misura della distribuzione dei valori del punteggio z dei punti dati per intervallo.	numero	Abilitato	zScoreDistribution	List
CONTO_ZERO	Numero di zeri (0) nella colonna.	numero	Abilitato	Conteggio zeri	Int

In`IncludedStatistics`, puoi sovrascrivere i parametri predefiniti di ogni valutazione aggiungendo un override. Ogni override include il nome di una particolare valutazione e una mappa dei parametri.

Parametri per le colonne ColumnStatisticsConfigurations

In`ColumnStatisticsConfigurations`, un job di profilo supporta i seguenti parametri.

In alcuni casi, potrebbero esserci più configurazioni diverse `ColumnStatisticsConfigurations` `IncludedStatistics` che possono essere applicate alla stessa colonna. In questi casi, il job del profilo seleziona l'ultima configurazione `ColumnStatisticsConfigurations` e la applica `IncludedStatistics` alla colonna selezionata. Una nuova configurazione sostituisce le configurazioni precedenti.

VALORI_MASSIMI

Elenca i valori massimi nella colonna numerica e i relativi conteggi. La dimensione predefinita dell'elenco è 5. È possibile sovrascrivere la dimensione dell'elenco specificando un valore per. `sampleSize`

Settings (Impostazioni)

`sampleSize`— La dimensione dell'elenco che include il numero e il conteggio massimi di valori nella colonna numerica. Questo valore deve essere maggiore di 0. Usa `"ALL"` per elencare tutti i valori.

Esempio

```
{
    "Statistic": "MAXIMUM_VALUES",
    "Parameters": {
        "sampleSize": "5"
    }
}
```

VALORI_MINIMI

Elenca i valori minimi nella colonna numerica e i relativi conteggi. La dimensione predefinita dell'elenco è 5. È possibile sovrascrivere la dimensione dell'elenco specificando un valore per. `sampleSize`

Settings (Impostazioni)

`sampleSize`— La dimensione dell'elenco che include il numero e il conteggio massimi di valori nella colonna numerica. Questo valore deve essere maggiore di 0. Usa `"ALL"` per elencare tutti i valori.

Esempio

```
{
    "Statistic": "MINIMUM_VALUES",
    "Parameters": {
        "sampleSize": "5"
    }
}
```

MOST_COMMON_VALUES

Elenca i valori più comuni nella colonna e i relativi conteggi. La dimensione predefinita dell'elenco è 50. È possibile sovrascrivere la dimensione dell'elenco specificando un valore per. `sampleSize`

Settings (Impostazioni)

`sampleSize`— La dimensione dell'elenco che include il numero e il conteggio massimi di valori nella colonna numerica. Questo valore deve essere maggiore di 0. Usa `"ALL"` per elencare tutti i valori.

Esempio

```
{
    "Statistic": "MOST_COMMON_VALUES",
    "Parameters": {
        "sampleSize": "50"
    }
}
```

OUTLIER_DETECTION

Rileva i valori anomali nella colonna numerica o nella colonna di stringhe (in base alla lunghezza della stringa) mediante l'algoritmo Z_score.

Il job del tuo profilo conta il numero di valori anomali e genera un elenco di esempio di valori anomali e i relativi punteggi z. L'elenco dei campioni è ordinato in base al valore assoluto dello z-score. La dimensione predefinita dell'elenco è 50.

L'algoritmo Z_Score identifica un valore come valore anomalo quando si discosta dalla media di oltre la soglia di deviazione standard. La soglia anomala predefinita è 3.

Puoi fornire un'altra soglia, una soglia lieve, per ottenere maggiori informazioni. La soglia moderata deve essere inferiore alla soglia. Questa funzionalità è disattivata per impostazione predefinita. Quando viene specificata una soglia moderata, il lavoro del tuo profilo restituisce un altro conteggio,`zScoreMildOutliersCount`. In questo caso, `zScoreOutliersSample` è inoltre possibile includere un campione di valori anomali di soglia lieve.

Settings (Impostazioni)

- `threshold`— Il valore di soglia da utilizzare per rilevare i valori anomali. Questo valore deve essere maggiore o uguale a 0.
- `mildThreshold`— Il valore di soglia moderato da utilizzare per rilevare valori anomali. Questo valore deve essere maggiore o uguale a 0 e minore di. `threshold`
- `sampleSize`— La dimensione dell'elenco che include i valori anomali nella colonna. Si usa `"ALL"` per elencare tutti i valori.

Esempio

```
{
    "Statistic": "OUTLIER_DETECTION",
    "Parameters": {
        "threshold": "5",
        "mildThreshold": "3.5",
        "sampleSize": "20"
    }
}
```

VALUE_DISTRIBUTION

Misura la distribuzione dei valori nella colonna in base agli intervalli dei valori. Un job di profilo raggruppa i valori di una colonna numerica o di una colonna di stringhe (in base alla lunghezza della stringa) in contenitori per intervalli numerici e genera un elenco di contenitori. I contenitori sono consecutivi e il limite superiore di un bucket è il limite inferiore per il bucket successivo.

Settings (Impostazioni)

`binNumber`— Numero di contenitori. Questo valore deve essere maggiore di 0.

Esempio

```
{
    "Statistic": "VALUE_DISTRIBUTION",
    "Parameters": {
        "binNumber": "5"
    }
}
```

Z_SCORE_DISTRIBUTION

Misura la distribuzione dei punteggi z dei valori nella colonna numerica. Un job di profilo raggruppa i punteggi z di valori in contenitori per intervalli numerici e genera un elenco di contenitori. I contenitori sono consecutivi e il limite superiore di un bucket è il limite inferiore per il bucket successivo.

Settings (Impostazioni)

`binNumber`— Numero di contenitori. Questo valore deve essere maggiore di 0.

Esempio

```
{
    "Statistic": "Z_SCORE_DISTRIBUTION",
    "Parameters": {
        "binNumber": "5"
    }
}
```

EntityDetectorConfiguration sezione per la configurazione delle PII

Nella `EntityDetectorConfiguration` sezione della struttura, puoi configurare i tipi di entità nel tuo set di dati che desideri DataBrew rilevare come informazioni di identificazione personale (PII) per un lavoro di profilo.

EntityTypes

Puoi configurare i tipi di entità che desideri rilevare come PII DataBrew per il lavoro del tuo profilo. Quando non `EntityDetectorConfiguration` è definito, il rilevamento delle entità è disabilitato. I seguenti tipi di entità possono essere rilevati nel set di dati:

- USA_SSN
- EMAIL
- USA_ITIN
- USA_PASSPORT_NUMBER
- PHONE_NUMBER
- USA_DRIVING_LICENSE
- BANK_ACCOUNT
- CREDIT_CARD
- IP_ADDRESS
- MAC_ADDRESS
- USA_DEA_NUMBER
- USA_HCPCS_CODE
- USA_NATIONAL_PROVIDER_IDENTIFIER
- USA_NATIONAL_DRUG_CODE
- USA_HEALTH_INSURANCE_CLAIM_NUMBER
- USA_MEDICARE_BENEFICIARY_IDENTIFIER
- USA_CPT_CODE
- PERSON_NAME
- DATE

`USA_ALL`È supportato anche il gruppo di tipi di entità che include tutti i tipi di entità sopra indicati tranne `PERSON_NAME` e`DATE`.

Il tipo di `EntityTypes` è un array di stringhe.

AllowedStatistics

Configura le statistiche che possono essere eseguite su colonne che contengono entità rilevate. Se non `AllowedStatistics` è definito, non verrà calcolata alcuna statistica sulle colonne che

contengono le entità rilevate. Statistiche configurabili a livello di colonnaPer un elenco di valori validi per il `AllowedStatistics` parametro, vedere.

Il tipo di `AllowedStatistics` è una matrice di `AllowedStatistics` oggetti.

Sicurezza in AWS Glue DataBrew

La sicurezza del cloud AWSè la massima priorità. In qualità di AWS cliente, puoi beneficiare di data center e architetture di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra te e te.AWS Il modello di responsabilità condivisa descrive questo aspetto come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud:AWSè responsabile della protezione dell'infrastruttura che gestisce AWS i servizi nel AWS cloud.AWS ti fornisce anche servizi che puoi utilizzare in modo sicuro. Third-partyi revisori testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito dei AWS Programmi di AWS conformità dei Programmi di conformità dei di . Per informazioni sui programmi di conformità applicabili AWS Glue DataBrew, consulta i AWS servizi in Scope by Compliance Program AWS .
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. L'utente è anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della propria azienda e le leggi e normative vigenti.

Questa documentazione ti aiuta a capire come applicare il modello di responsabilità condivisa durante l'utilizzo AWS Glue DataBrew. I seguenti argomenti mostrano come eseguire la configurazione DataBrew per soddisfare gli obiettivi di sicurezza e conformità. Imparerai anche a utilizzare altri AWS servizi che ti aiutano a monitorare e proteggere DataBrew le tue risorse.

Argomenti

Protezione dei dati in AWS Glue DataBrew

DataBrew offre diverse funzionalità progettate per aiutarti a proteggere i tuoi dati.

Argomenti

Ulteriori informazioni su come il modello di responsabilità condivisaAWS si applica alla protezione dei dati in AWS Glue DataBrew. Come descritto in questo modello,AWSè responsabile della protezione dell'infrastruttura globale che gestisce tutti i Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, consulta Domande frequenti sulla privacy dei dati . Per ulteriori informazioni sulla protezione dei dati in Europa, consulta il Centro generale sulla protezione dei dati (GDPR).

Ai fini della protezione dei dati, ti consigliamo di proteggere Account AWS le credenziali e configurare singoli utenti con Centro identità AWS IAM o AWS Identity and Access Management(IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- SSL/TLS Da utilizzare per comunicare con AWS le risorse. È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con AWS CloudTrail. Per informazioni sull'utilizzo dei CloudTrail percorsi per acquisire AWS le attività, consulta Lavorare con i CloudTrail percorsi nella Guida per l'AWS CloudTrail utente.
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.

- Se hai bisogno di moduli crittografici convalidati FIPS 140-3 per accedere AWS tramite un'interfaccia a riga di comando o un'API, usa un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il Federal Information Processing Standard (FIPS) 140-3.

Ti consigliamo di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando lavori o Servizi AWS utilizzi la console, l'API DataBrew o gli SDK.AWS CLIAWS I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando si fornisce un URL a un server esterno, suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la richiesta al server.

Crittografia dei dati a riposo

DataBrew supporta la crittografia dei dati inattivi per DataBrew progetti e lavori. I progetti e i lavori possono leggere dati crittografati e i lavori possono scrivere dati crittografati chiamando AWS Key Management Service(AWS KMS) per generare chiavi e decrittografare i dati. È inoltre possibile utilizzare le chiavi KMS per crittografare i registri dei lavori generati dai lavori. DataBrew È possibile specificare le chiavi di crittografia utilizzando la DataBrew console o l'API. DataBrew

> **Important**
>
> AWS Glue DataBrew supporta solo chiavi AWS KMS simmetriche. Per ulteriori informazioni, consulta le chiavi AWS KMS nella Guida per gli sviluppatori.AWS Key Management Service

Quando crei lavori DataBrew con la crittografia abilitata, puoi utilizzare la DataBrew console per specificare le chiavi di crittografia S3-managed sul lato server (SSE-S3) o le chiavi KMS archiviate in AWS KMS(SSE-KMS) per crittografare i dati inattivi.

> **Important**
>
> Quando utilizzi un set di dati Amazon Redshift, gli oggetti scaricati nella directory temporanea fornita vengono crittografati con. SSE-S3

Crittografia dei dati scritti dai job DataBrew

DataBrew i job possono scrivere su destinazioni Amazon S3 crittografate e Amazon CloudWatch Logs crittografati.

Argomenti

- Configurazione per l'utilizzo DataBrew della crittografia
- Creazione di un percorso per AWS KMS per lavori in VPC
- Configurazione della crittografia con AWS Chiavi KMS

Configurazione per l'utilizzo DataBrew della crittografia

Segui questa procedura per configurare DataBrew l'ambiente per l'utilizzo della crittografia.

Per configurare DataBrew l'ambiente in modo che utilizzi la crittografia

1. Crea o aggiorna le tue chiavi AWS KMS per concedere AWS KMS le autorizzazioni ai ruoli AWS Identity and Access Management(IAM) che vengono trasferiti ai DataBrew job. Questi ruoli IAM vengono utilizzati per crittografare CloudWatch i log e le destinazioni Amazon S3. Per ulteriori informazioni, consulta Encrypt Log Data in CloudWatch Logs Using AWS KMS nella Amazon CloudWatch Logs User Guide.

   Nell'esempio seguente *"role1""role2"*, e *"role3"* sono i ruoli IAM che vengono trasferiti ai job. DataBrew Questa informativa descrive una politica chiave KMS che autorizza i ruoli IAM elencati a crittografare e decrittografare con questa chiave KMS.

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "logs.region.amazonaws.com",
        "AWS": [
            "role1",
            "role2",
            "role3"
        ]
    },
    "Action": [
        "kms:Encrypt*",
        "kms:Decrypt*",
```

```
            "kms:ReEncrypt*",
            "kms:GenerateDataKey*",
            "kms:Describe*"
        ],
        "Resource": "*"
    }
```

L'`Service`istruzione, mostrata come`"Service": "logs.region.amazonaws.com"`, è obbligatoria se si utilizza la chiave per crittografare i log. CloudWatch

2. Assicuratevi che la AWS KMS chiave sia impostata su `ENABLED` prima di essere utilizzata.

Per ulteriori informazioni sulla specificazione delle autorizzazioni utilizzando i criteri AWS KMS chiave, vedere Utilizzo dei criteri chiave in.AWS KMS

Creazione di un percorso per AWS KMS per lavori in VPC

Puoi connetterti direttamente a AWS KMS attraverso un endpoint privato nel cloud privato virtuale (VPC, Virtual Private Cloud) invece che tramite Internet. Quando utilizzi un endpoint VPC, la comunicazione tra il tuo VPC e il tuo VPC AWS KMS viene condotta interamente all'interno della rete.AWS

Puoi creare un endpoint AWS KMS VPC all'interno di un VPC. Senza questo passaggio, i tuoi DataBrew lavori potrebbero fallire con un. `kms timeout` Per istruzioni dettagliate, consulta Connessione a AWS KMS un endpoint VPC nella Guida per gli AWS Key Management Service sviluppatori.

Seguendo queste istruzioni, sulla console VPC, assicurati di fare quanto segue:

- Scegli Abilita nome DNS privato.
- Per il gruppo di sicurezza, scegli il gruppo di sicurezza (inclusa una regola di autoreferenziazione) che usi per il tuo DataBrew lavoro che accede a Java Database Connectivity (JDBC).

Quando esegui un DataBrew processo che accede agli archivi dati JDBC, DataBrew devi disporre di un percorso verso l'endpoint.AWS KMSÈ possibile fornire il percorso con un gateway NAT (Network Address Translation) o con un endpoint AWS KMS VPC. Per creare un gateway NAT, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

Configurazione della crittografia con AWS Chiavi KMS

Quando abiliti la crittografia su un processo, si applica sia ad Amazon S3 che a. CloudWatch Il ruolo IAM passato deve disporre delle seguenti AWS KMS autorizzazioni.

Per ulteriori informazioni, consulta i seguenti argomenti nella Guida per l'utente di Amazon Simple Storage Service:

- Per informazioni su`SSE-S3`, consulta Protezione dei dati tramite Server-Side crittografia con Amazon S3-Managed Encryption Keys (SSE-S3).
- Per informazioni su`SSE-KMS`, consulta Protezione dei dati tramite Server-Side crittografia con chiavi AWS gestite da KMS (). SSE-KMS

Crittografia dei dati in transito

AWS fornisce la crittografia Secure Sockets Layer (SSL) per i dati in trasferimento.

DataBrew viene fornito il supporto per le fonti di dati JDBC.AWS Glue Quando ci si connette a sorgenti dati JDBC, DataBrew utilizza le impostazioni della AWS Glue connessione, inclusa l'opzione Richiedi connessione SSL. Per ulteriori informazioni, consulta AWS Glue Connection Properties,AWS Glue nella Developer Guide.AWS Glue

AWS KMS fornisce sia la crittografia «bring your own key» che la crittografia lato server per l'elaborazione di DataBrew estrazioni, trasformazioni, caricamenti (ETL) e per.AWS Glue Data Catalog

Gestione delle chiavi

Puoi utilizzare IAM con DataBrew per definire utenti,AWS risorse, gruppi, ruoli e policy granulari in materia di accesso, rifiuto e altro ancora.

Puoi definire l'accesso ai metadati utilizzando politiche basate sia sulle risorse che sull'identità, a seconda delle esigenze della tua organizzazione. Resource-based le politiche elencano i principali a cui è consentito o negato l'accesso alle risorse, consentendoti di impostare politiche come l'accesso tra account. Le policy basate sull'identità sono specificamente collegate a utenti, gruppi e ruoli all'interno di IAM.

DataBrew supporta la creazione di una propria AWS KMS key crittografia «bring your own key». DataBrew fornisce anche la crittografia lato server utilizzando le chiavi KMS di for Jobs.AWS KMS DataBrew

Identificazione e gestione delle informazioni di identificazione personale (PII)

Quando si creano funzioni analitiche o modelli di apprendimento automatico, sono necessarie misure di protezione per prevenire l'esposizione dei dati di informazioni di identificazione personale (PII). Le PII sono dati personali che possono essere utilizzati per identificare un individuo, ad esempio un indirizzo, un numero di conto bancario o un numero di telefono. Ad esempio, quando analisti di dati e data scientist utilizzano set di dati per scoprire informazioni demografiche generali, non dovrebbero avere accesso alle informazioni personali di individui specifici.

DataBrew fornisce meccanismi di mascheramento dei dati per offuscare i dati PII durante il processo di preparazione dei dati. A seconda delle esigenze dell'organizzazione, sono disponibili diversi meccanismi di redazione dei dati PII. Puoi offuscare i dati PII in modo che gli utenti non possano ripristinarli, oppure puoi rendere l'offuscamento reversibile.

L'identificazione e il mascheramento dei dati PII DataBrew implica la creazione di una serie di trasformazioni che i clienti possono utilizzare per redigere i dati PII. Parte di questo processo consiste nel fornire il rilevamento e le statistiche dei dati PII nella dashboard panoramica di Data Profile sulla console. DataBrew

È possibile utilizzare le seguenti tecniche di mascheramento dei dati:

- Sostituzione: sostituisci i dati PII con altri valori dall'aspetto autentico.
- Mescola: mescola il valore della stessa colonna in righe diverse.
- Crittografia deterministica: applica algoritmi di crittografia deterministica ai valori delle colonne. La crittografia deterministica produce sempre lo stesso testo cifrato per un valore.
- Crittografia probabilistica: applica algoritmi di crittografia probabilistica ai valori delle colonne. La crittografia probabilistica produce un testo cifrato diverso ogni volta che viene applicata.
- Decrittografia: decripta le colonne in base alle chiavi di crittografia.
- Annullamento o eliminazione: sostituisci un campo particolare con un valore nullo o elimina la colonna.
- Mascheratura: usa il rimescolamento dei caratteri o maschera alcune parti delle colonne.
- Hashing: applica funzioni hash ai valori delle colonne.

Per ulteriori informazioni sull'utilizzo delle trasformazioni, consulta i passaggi della ricetta relativi alle informazioni di identificazione personale (PII). Per ulteriori informazioni sull'utilizzo dei processi

di profilo per rilevare le PII, incluso un elenco dei tipi di entità che possono essere rilevati, consulta la EntityDetectorConfiguration sezione relativa alla configurazione delle informazioni personali in Creazione di una configurazione di job di profilo in modo programmatico.

DataBrew dipendenza da altri AWS services

Per utilizzare la DataBrew console, devi disporre di un set minimo di autorizzazioni per utilizzare DataBrew le risorse del tuo AWS account. Oltre a queste DataBrew autorizzazioni, la console richiede le autorizzazioni dei seguenti servizi:

- CloudWatch Registra le autorizzazioni per visualizzare i registri.
- Autorizzazioni IAM per elencare e passare i ruoli.
- Autorizzazioni Amazon EC2 per elencare VPC, sottoreti, gruppi di sicurezza, istanze e altri oggetti. DataBrew utilizza queste autorizzazioni per configurare elementi di Amazon EC2 come i VPC durante l'esecuzione dei processi. DataBrew
- Autorizzazioni Amazon S3 per elencare bucket e oggetti.
- AWS Glue autorizzazioni per leggere oggetti AWS Glue dello schema, come database, partizioni, tabelle e connessioni.
- AWS Lake Formation autorizzazioni per lavorare con i data lake Lake Formation.

Gestione delle identità e degli accessi per AWS Glue DataBrew

AWS Identity and Access Management(IAM) è uno strumento Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse. DataBrew IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

Argomenti

- Autenticazione con identità
- Gestione dell’accesso tramite policy
- AWS Glue DataBrew and AWS Lake Formation
- In che modo AWS Glue DataBrew funziona con IAM
- Identity-based esempi di policy per AWS Glue DataBrew
- AWS politiche gestite per AWS Glue DataBrew

- Risoluzione dei problemi di identità e accesso in AWS Glue DataBrew

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi autenticarti come utente IAM o assumendo un ruolo IAM.Utente root dell'account AWS

Puoi accedere come identità federata utilizzando credenziali provenienti da una fonte di identità come Centro identità AWS IAM(IAM Identity Center), autenticazione Single Sign-On o credenziali. Google/ Facebook Per ulteriori informazioni sull'accesso, consulta Come accedere all'Account AWS nella Guida per l'utente di Accedi ad AWS.

Per l'accesso programmatico,AWS fornisce un SDK e una CLI per firmare crittograficamente le richieste. Per ulteriori informazioni, consulta AWS Signature Version 4 per le richieste API nella Guida per l'utente di IAM.

Account AWS utente root

Quando si crea un Account AWS, si inizia con un'identità di accesso denominata utente Account AWS root che ha accesso completo a tutte Servizi AWS le risorse. Consigliamo vivamente di non utilizzare l'utente root per le attività quotidiane. Per le attività che richiedono le credenziali come utente root, consulta Attività che richiedono le credenziali dell'utente root nella Guida per l'utente di IAM.

Utenti e gruppi

Un utente IAM è una identità che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ti consigliamo di utilizzare credenziali temporanee invece di utenti IAM con credenziali a lungo termine. Per ulteriori informazioni, consulta Richiedere agli utenti umani di utilizzare la federazione con un provider di identità per accedere AWS utilizzando credenziali temporanee nella Guida per l'utente IAM.

Un gruppo IAM specifica una raccolta di utenti IAM e semplifica la gestione delle autorizzazioni per gestire gruppi di utenti di grandi dimensioni. Per ulteriori informazioni, consulta Casi d'uso per utenti IAM nella Guida per l'utente di IAM.

Ruoli IAM

Un ruolo IAM è un'identità con autorizzazioni specifiche che fornisce credenziali temporanee. Puoi assumere un ruolo passando da un ruolo utente a un ruolo IAM (console) o chiamando un'operazione

AWS CLI o AWS API. Per ulteriori informazioni, consulta Metodi per assumere un ruolo nella Guida per l'utente di IAM.

I ruoli IAM sono utili per l'accesso degli utenti federati, le autorizzazioni utente IAM temporanee, l'accesso multi-account, l'accesso multi-servizio e le applicazioni in esecuzione su Amazon EC2. Per maggiori informazioni, consultare Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Gestione dell'accesso tramite policy

Puoi controllare l'accesso AWS creando policy e associandole a AWS identità o risorse. Una policy definisce le autorizzazioni quando è associata a un'identità o a una risorsa.AWS valuta queste politiche quando un preside effettua una richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per maggiori informazioni sui documenti delle policy JSON, consulta Panoramica delle policy JSON nella Guida per l'utente IAM.

Utilizzando le policy, gli amministratori specificano chi ha accesso a cosa definendo quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Un amministratore IAM crea le policy IAM e le aggiunge ai ruoli, che gli utenti possono quindi assumere. Le policy IAM definiscono le autorizzazioni indipendentemente dal metodo utilizzato per eseguirle.

Identity-based politiche

Identity-based le politiche sono documenti di policy sulle autorizzazioni JSON che alleghi a un'identità (utente, gruppo o ruolo). Tali policy controllano le operazioni autorizzate per l'identità, nonché le risorse e le condizioni in cui possono essere eseguite. Per informazioni su come creare una policy basata su identità, consultare Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente IAM.

Identity-based le politiche possono essere politiche in linea (incorporate direttamente in una singola identità) o politiche gestite (politiche autonome collegate a più identità). Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta Scegliere tra policy gestite e policy in linea nella Guida per l'utente di IAM.

Resource-based politiche

Resource-based le politiche sono documenti di policy JSON allegati a una risorsa. Gli esempi includono le policy di trust dei ruoli IAM e le policy dei bucket di Amazon S3. Nei servizi che

supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. In una policy basata sulle risorse è obbligatorio specificare un'entità principale.

Resource-based le politiche sono politiche in linea che si trovano in quel servizio. Non è possibile utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

DataBrew non supporta politiche basate sulle risorse.

Liste di controllo degli accessi (ACL)

Le liste di controllo degli accessi (ACL) controllano quali entità principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3 e Amazon VPC sono esempi di servizi che supportano gli ACL.AWS WAF Per maggiori informazioni sulle ACL, consulta Panoramica delle liste di controllo degli accessi (ACL) nella Guida per gli sviluppatori di Amazon Simple Storage Service.

DataBrew non supporta gli ACL.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi che possono impostare le autorizzazioni massime concesse da tipi di policy più comuni:

- Limiti delle autorizzazioni: imposta il numero massimo di autorizzazioni che una policy basata su identità ha la possibilità di concedere a un'entità IAM. Per ulteriori informazioni, consulta Limiti delle autorizzazioni per le entità IAM nella Guida per l'utente di IAM.
- Policy di controllo dei servizi (SCP): specifica il numero massimo di autorizzazioni per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. Per ulteriori informazioni, consultare Policy di controllo dei servizi nella Guida per l'utente di AWS Organizations.
- Policy di controllo delle risorse (RCP): imposta le autorizzazioni massime disponibili per le risorse degli account. Per ulteriori informazioni, consulta Policy di controllo delle risorse (RCP) nella Guida per l'utente di AWS Organizations.
- Policy di sessione: policy avanzate passate come parametro quando si crea una sessione temporanea per un ruolo o un utente federato. Per maggiori informazioni, consultare Policy di sessione nella Guida per l'utente IAM.

Più tipi di policy

Quando a una richiesta si applicano più tipi di policy, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire o meno una richiesta quando sono coinvolti più tipi di policy, consulta Logica di valutazione delle policy nella IAM User Guide.

AWS Glue DataBrew and AWS Lake Formation

AWS Glue DataBrew supporta AWS Lake Formation le autorizzazioni per le AWS Glue Data Catalog tabelle. Quando un set di dati utilizza una AWS Glue Data Catalog tabella registrata con Lake Formation, il ruolo IAM fornito a progetti o lavori deve avere le autorizzazioni DESCRIBE e SELECT Lake Formation sulla tabella.

AWS Glue DataBrew supporta la scrittura su AWS Glue Data Catalog tabelle basate su.AWS Lake Formation Quando un DataBrew lavoro utilizza un Data Catalog registrato con Lake Formation, il ruolo IAM fornito ai lavori deve disporre delle autorizzazioni INSERT, ALTER e DELETE di Lake Formation per le tabelle coinvolte. Il ruolo IAM deve disporre `glue:UpdateTable` delle autorizzazioni e anche delle autorizzazioni per la posizione dei dati associata alla tabella Data Catalog.

In che modo AWS Glue DataBrew funziona con IAM

Prima di utilizzare IAM per gestire l'accesso a DataBrew, è necessario comprendere con quali funzionalità IAM è disponibile l'uso DataBrew. Per avere una visione di alto livello di come DataBrew e altri AWS servizi funzionano con IAM, consulta AWS Services That Work with IAM nella IAM User Guide.

Argomenti

- DataBrew politiche basate sull'identità
- Resource-based politiche in DataBrew
- DataBrew Ruoli IAM

DataBrew politiche basate sull'identità

Con le policy basate su identità IAM puoi specificare azioni e risorse consentite o rifiutate, nonché le condizioni in base alle quali le azioni sono consentite o rifiutate. DataBrew supporta azioni, risorse e chiavi di condizione specifiche. Per informazioni su tutti gli elementi utilizzati in una policy JSON,

consulta Documentazione di riferimento degli elementi delle policy JSON IAM nella Guida per l'utente IAM.

Azioni

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, una policy AWS JSON può specificare quale principale può eseguire azioni su quali risorse e in quali condizioni.

L'elemento Azione di una policy JSON descrive le azioni a cui è possibile consentire o negare l'accesso in una policy. Le operazioni di policy hanno spesso lo stesso nome dell'operazione API AWS. Ci sono alcune eccezioni, ad esempio le operazioni di sola autorizzazione che non hanno un'operazione API corrispondente. Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includere le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Le azioni politiche DataBrew utilizzano il seguente prefisso prima dell'azione:. `databrew:` Ad esempio, per concedere a qualcuno l'autorizzazione per eseguire un'istanza Amazon EC2 con l'operazione API `RunInstances` Amazon EC2, è necessario includere l'operazione `ec2:RunInstances` nella policy. Le dichiarazioni politiche devono includere un `NotAction` elemento `Action` or. DataBrew definisce il proprio insieme di azioni che descrivono le attività che è possibile eseguire con esso.

Per specificare più operazioni in una singola istruzione, separarle con una virgola come mostrato di seguito.

```
"Action": [
      "databrew:CreateRecipeJob",
      "databrew:UpdateSchedule"
```

Puoi specificare più operazioni tramite caratteri jolly (*). Ad esempio, per specificare tutte le operazioni che iniziano con la parola `Describe`, includi la seguente operazione.

```
"Action": "databrew:Describe*"
```

Per visualizzare un elenco di DataBrew azioni, consulta Actions Defined by AWS Glue DataBrew nella IAM User Guide.

Resources

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento JSON `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Come best practice, specifica una risorsa utilizzando il suo nome della risorsa Amazon (ARN). Per le azioni che non supportano le autorizzazioni a livello di risorsa, si utilizza un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

```
"Resource": "*"
```

Le seguenti sono le DataBrew API che non supportano le autorizzazioni a livello di risorsa:

- ListDatasets
- ListJobs
- ListProjects
- ListRecipes
- ListRulesets
- ListSchedules

La risorsa del DataBrew set di dati ha il seguente Amazon Resource Name (ARN).

```
arn:${Partition}:databrew:${Region}:${Account}:dataset/${Name}
```

Per ulteriori informazioni sul formato degli ARN, consulta Amazon Resource Names (ARNs) e AWS Service Namespaces.

Ad esempio, per specificare l'`i-1234567890abcdef0`istanza nell'istruzione, utilizzare il seguente ARN.

```
"Resource": "arn:aws:databrew:us-east-1:123456789012:dataset/my-chess-dataset"
```

Per specificare tutte le istanze database che appartengono a un account specifico, utilizza il carattere jolly (*).

```
"Resource": "arn:aws:databrew:us-east-1:123456789012:dataset/*"
```

Non è possibile eseguire alcune DataBrew azioni, ad esempio quelle per la creazione di risorse, su una risorsa specifica. In questi casi, è necessario utilizzare il carattere jolly (*).

```
"Resource": "*"
```

Per visualizzare un elenco dei tipi di DataBrew risorse e dei relativi ARN, consulta Resources Defined by AWS Glue DataBrew nella IAM User Guide. Per informazioni sulle operazioni con cui è possibile specificare l'ARN di ogni risorsa, consulta Operazioni definite da AWS Glue DataBrew.

Chiavi di condizione

DataBrew non fornisce chiavi di condizione specifiche del servizio, ma supporta l'utilizzo di alcune chiavi di condizione globali. Per vedere tutte le chiavi di condizione AWS globali, consulta le chiavi di contesto delle condizioni AWS globali nella Guida per l'utente IAM.

Esempi

Per visualizzare esempi di politiche DataBrew basate sull'identità, consulta. Identity-based esempi di policy per AWS Glue DataBrew

Resource-based politiche in DataBrew

DataBrew non supporta le politiche basate sulle risorse.

DataBrew Ruoli IAM

Un ruolo IAM è un'entità all'interno del tuo AWS account che dispone di autorizzazioni specifiche.

Utilizzo di credenziali temporanee con DataBrew

È possibile utilizzare credenziali temporanee per effettuare l'accesso con la federazione, assumere un ruolo IAM o un ruolo multi-account. È possibile ottenere credenziali di sicurezza temporanee chiamando operazioni AWS STS API come AssumeRoleo. GetFederationToken

DataBrew supporta l'utilizzo di credenziali temporanee.

Service-linked ruoli

Service-linked i ruoli consentono ai AWS servizi di accedere alle risorse di altri servizi per completare un'azione per conto dell'utente. Service-linked i ruoli vengono visualizzati nel tuo account IAM e sono di proprietà del servizio. Un amministratore può visualizzare, ma non modificare le autorizzazioni dei ruoli collegati ai servizi.

Scegliere un ruolo IAM in DataBrew

Quando crei una risorsa del set di dati in DataBrew, scegli un ruolo IAM per consentire DataBrew l'accesso per tuo conto. Se in precedenza hai creato un ruolo di servizio o un ruolo collegato al servizio, ti DataBrew fornisce un elenco di ruoli tra cui scegliere. Assicurati di scegliere un ruolo che consenta l'accesso in lettura a un bucket o a una AWS Glue Data Catalog risorsa Amazon S3, a seconda dei casi.

Identity-based esempi di policy per AWS Glue DataBrew

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare risorse DataBrew . Inoltre, non possono eseguire attività utilizzando le Console di gestione AWSAWS CLI, o AWS API. Un amministratore deve creare le policy IAM che concedono a utenti e ruoli l'autorizzazione per eseguire operazioni API specifiche sulle risorse specificate di cui hanno bisogno. L'amministratore devi quindi collegare queste policy a utenti o gruppi che richiedono tali autorizzazioni.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta Creazione di policy nella scheda JSON nella Guida per l'utente IAM.

Argomenti

- Best practice delle policy
- Utilizzo della console DataBrew
- Consentire agli utenti di visualizzare le loro autorizzazioni
- Gestione delle DataBrew risorse in base ai tag

Best practice delle policy

Identity-based le politiche determinano se qualcuno può creare, accedere o eliminare DataBrew risorse nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l'Account AWS. Quando si creano o modificano policy basate sull'identità, seguire queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche AWS gestite che concedono le autorizzazioni per molti casi d'uso comuni. Sono disponibili nel tuo.Account AWS Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti

specifiche per i tuoi casi d'uso. Per maggiori informazioni, consulta Policy gestite da AWS o Policy gestite da AWS per le funzioni dei processi nella Guida per l'utente di IAM.

- Applicazione delle autorizzazioni con privilegio minimo - Quando si impostano le autorizzazioni con le policy IAM, concedere solo le autorizzazioni richieste per eseguire un'attività. È possibile farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegio minimo. Per maggiori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso - Per limitare l'accesso ad azioni e risorse è possibile aggiungere una condizione alle policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio CloudFormation. Per maggiori informazioni, consultare la sezione Elementi delle policy JSON di IAM: condizione nella Guida per l'utente di IAM.
- Utilizzo dello strumento di analisi degli accessi IAM per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali - Lo strumento di analisi degli accessi IAM convalida le policy nuove ed esistenti in modo che aderiscano al linguaggio (JSON) della policy IAM e alle best practice di IAM. Lo strumento di analisi degli accessi IAM offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per maggiori informazioni, consultare Convalida delle policy per il Sistema di analisi degli accessi IAM nella Guida per l'utente di IAM.
- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungere le condizioni MFA alle policy. Per maggiori informazioni, consultare Protezione dell'accesso API con MFA nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consulta Best practice di sicurezza in IAM nella Guida per l'utente di IAM.

Utilizzo della console DataBrew

Per accedere alla AWS Glue DataBrew console, è necessario disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle DataBrew risorse del tuo AWS account. Se crei una politica basata sull'identità che è più restrittiva delle autorizzazioni minime richieste, la console non funziona come previsto per gli utenti o i ruoli con quella politica.

Per garantire che utenti e ruoli possano utilizzare la DataBrew console, allega anche la seguente politica AWS gestita alle entità. Per ulteriori informazioni, consulta Aggiunta di autorizzazioni a un utente nella Guida per l'utente IAM.

```
AWSDataBrewConsoleAccess
```

Non è necessario consentire autorizzazioni minime per la console per gli utenti che effettuano chiamate solo verso AWS CLI o l' DataBrew API. Al contrario, è possibile accedere solo alle operazioni che soddisfano l'operazione API che stai cercando di eseguire.

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono collegate alla relativa identità utente. Questa politica include le autorizzazioni per completare questa azione sulla console o utilizzando l'API o a livello di codice.AWS CLIAWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
```

```
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Gestione delle DataBrew risorse in base ai tag

È possibile utilizzare le condizioni della politica basata sull'identità per gestire DataBrew le risorse in base ai tag, ad esempio per eliminare, aggiornare o descrivere le risorse. L'esempio seguente mostra una politica che nega l'eliminazione di un progetto. Tuttavia, l'eliminazione viene negata solo se il tag del progetto Owner ha il valore di admin. Questo criterio concede anche le autorizzazioni necessarie per negare questa azione sulla console.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Sid": "DeleteResourceInConsole",
         "Effect": "Allow",
         "Action": "databrew:DeleteProject",
         "Resource": "*"
       },
       {
         "Sid": "DenyDeleteProjectIfAdminTag",
         "Effect": "Deny",
         "Action": "databrew:DeleteProject",
         "Resource": "arn:aws:databrew:*:*:project/*",
         "Condition": {
            "StringEquals": {"aws:ResourceTag/Owner": "admin"}
         }
      }
   ]
}
```

È possibile collegare questa policy agli utenti nell'account. Se un utente di nome richard-roe tenta di eliminare un DataBrew progetto, la risorsa non deve essere contrassegnata con owner=admin o owner=admin. In caso contrario, all'utente viene negata l'autorizzazione a eliminare il progetto. La chiave del tag di condizione Owner corrisponde sia a Owner che a owner perché i nomi delle chiavi di condizione non fanno distinzione tra maiuscole e minuscole. Per ulteriori informazioni, consulta la sezione Elementi delle policy JSON di IAM: condizione nella Guida per l'utente di IAM.

> **Note**
>
> ListDatasets,, ListJobs, ListProjects ListRecipes ListRulesets, e ListSchedules non supportano il controllo degli accessi basato su tag.

AWS politiche gestite per AWS Glue DataBrew

Per aggiungere autorizzazioni a utenti, gruppi e ruoli, è più facile utilizzare le policy AWS gestite che scriverle autonomamente. La creazione di policy gestite dai clienti IAM che forniscono al team solo le autorizzazioni di cui ha bisogno richiede tempo e competenza. Per iniziare rapidamente, puoi utilizzare le nostre politiche AWS gestite. Queste politiche coprono casi d'uso comuni e sono disponibili nel tuo AWS account. Per ulteriori informazioni sulle policy AWS gestite, consulta le policy AWS gestite nella IAM User Guide.

AWS i servizi mantengono e aggiornano le politiche AWS gestite. Non è possibile modificare le autorizzazioni nelle politiche AWS gestite. I servizi aggiungono occasionalmente autorizzazioni aggiuntive a una policy AWS gestita per supportare nuove funzionalità. Questo tipo di aggiornamento interessa tutte le identità (utenti, gruppi e ruoli) a cui è collegata la policy. È più probabile che i servizi aggiornino una politica AWS gestita quando viene lanciata una nuova funzionalità o quando diventano disponibili nuove operazioni. I servizi non rimuovono le autorizzazioni da una policy AWS gestita, quindi gli aggiornamenti delle policy non comprometteranno le autorizzazioni esistenti.

Inoltre,AWS supporta politiche gestite per le funzioni lavorative che si estendono su più servizi. Ad esempio, la policy ReadOnlyAccessAWS gestita fornisce l'accesso in sola lettura a tutti i AWS servizi e le risorse. Quando un servizio lancia una nuova funzionalità,AWS aggiunge autorizzazioni di sola lettura per nuove operazioni e risorse. Per un elenco e le descrizioni delle politiche relative alle funzioni lavorative, consulta le politiche AWS gestite per le funzioni lavorative nella Guida per l'utente IAM.

DataBrew aggiornamenti a AWS policy gestite

Visualizza i dettagli sugli aggiornamenti delle politiche AWS gestite DataBrew da quando questo servizio ha iniziato a tenere traccia di queste modifiche. Per ricevere avvisi automatici sulle modifiche a questa pagina, iscriviti al feed RSS nella pagina della cronologia dei DataBrew documenti. La policy gestita è disponibile nella console AWS IAM all'indirizzo. AwsGlueDataBrewFullAccessPolicy

Modifica	Descrizione	Data
AWSGlueDataBrewServiceRole—AWS GlueÈ stata aggiunta l'autorizzazione di lettura per.	Questo aggiornamento aggiunge`glue:GetCustomEntityType` . Questa autorizzazione è necessaria per eseguire i lavori di AWS Glue DataBrew profilo con PII-identification enabled.	20 marzo 2024
AWSGlueDataBrewServiceRole-AWS GlueÈ stata aggiunta l'autorizzazione di lettura per.	Questo aggiornamento aggiunge`glue:BatchGetCustomEntityTypes` . Questa autorizzazione è necessaria per eseguire i lavori di AWS Glue DataBrew profilo con PII-identification enabled.	09 maggio 2022
AwsGlueDataBrewFullAccessPolicy- Sono state aggiunte le autorizzazioni di lettura per Amazon Redshift-Data DescribeStatements e Amazon GetLifecycleConfiguration S3.	Questo aggiornamento aggiunge il supporto `redshift-data:DescribeStatement` per la convalida di SQL durante la creazione di un Redshift-based set di dati Amazon. Inoltre, consente `s3:GetLifecycleConfiguration` di valutare se il ciclo di	4 febbraio 2022

Modifica	Descrizione	Data
	vita del prefisso del bucket Amazon S3 fornito come directory temporanea ha o meno configurato il ciclo di vita. Inoltre, questa modifica sostituisce le autorizzazioni «databrew: *» con un elenco esplicito di autorizzazioni che include tutte le API. DataBrew	
AwsGlueDataBrewFullAccessPolicy- sono state aggiunte Read/write le autorizzazioni per AWS Secrets Manager.	Questo aggiornamento aggiunge `secretsmanager:CreateSecret`, `secretsmanager:GetSecretValue` per un segreto denominat `odatabrew!default` , un segreto predefinito da utilizzare con le DataBrew trasformazioni. Inoltre, aggiunge le autorizzazioni CreateSecret per i segreti con il prefisso `AwsGlueDataBrew-` per la creazione di segreti dalla console. DataBrew GenerateRandom, descritto nell'AWS Key Management Service API Reference, viene utilizzato per generare una stringa di byte casuale crittograficamente sicura.	18 novembre 2021

Modifica	Descrizione	Data
AWSGlueDataBrewServiceRole- sono state aggiunte Read/write le autorizzazioni per AWS Secrets Manager.	Questo aggiornamento aggiunge `secretsmanager:GetSecretValue` un segreto denominat `odatabrew!default` , un segreto predefinito da utilizzare con le DataBrew trasformazioni.	18 novembre 2021
AwsGlueDataBrewFullAccessPolicy- sono state aggiunte Read/write le autorizzazioni per AWS Secrets Manager.	Questo aggiornamento aggiunge `secretsmanager:CreateSecret`, `secretsmanager:GetSecretValue` per un segreto denominat `odatabrew!default` , un segreto predefinito da utilizzare con le DataBrew trasformazioni. Inoltre, aggiunge le autorizzazioni CreateSecret per i segreti con il prefisso `AwsGlueDataBrew-` per la creazione di segreti dalla console. DataBrew `kms:GenerateRandom` (`https://docs.aws.amazon.com/kms/latest/APIReference/API_GenerateRandom.html`) viene utilizzato per generare una stringa di byte casuale crittograficamente sicura.	18 novembre 2021

Modifica	Descrizione	Data
AWSGlueDataBrewServiceRole- sono state aggiunte Read/write le autorizzazioni per AWS Secrets Manager.	Questo aggiornamento aggiunge `secretsmanager:GetSecretValue` un segreto denominat `odatabrew!default` , un segreto predefinito da utilizzare con le DataBrew trasformazioni.	18 novembre 2021
AwsGlueDataBrewFullAccessPolicy- Sono state aggiunte le autorizzazioni di lettura per i database AWS Glue del catalogo e le autorizzazioni di creazione per la tabella AWS Glue del catalogo.	Questo aggiornamento aggiunge le autorizzazioni per elencare i database AWS Glue del catalogo e creare nuove tabelle di catalogo in un database esistente come parte della configurazione dell'output per i lavori. DataBrew	30 giugno 2021
AwsGlueDataBrewFullAccessPolicy- sono state Read/write aggiunte le autorizzazioni per la funzionalità del AppFlow set di dati Amazon.	Questo aggiornamento aggiunge le autorizzazioni per leggere i flussi e le esecuzioni di AppFlow flussi Amazon esistenti e per creare esecuzioni di flussi.	28 Aprile 2021

Modifica	Descrizione	Data
AwsGlueDataBrewFullAccessPolicy- Sono state aggiunte le autorizzazioni di lettura per i set di dati del database.	Questo aggiornamento aggiunge le autorizzazioni per leggere le AWS Glue connessioni esistenti e creare nuove AWS Glue connessioni da utilizzare con. DataBrew Inoltre, per semplificare l'esperienza di creazione di nuove connessioni da console, consente di elencare le risorse Amazon VPC e i cluster Amazon Redshift. Permette inoltre di elencare, ma non leggere, i segreti.AWS Secrets Manager	30 marzo 2021
DataBrew ha iniziato a tenere traccia delle modifiche	DataBrew ha iniziato a tenere traccia delle modifiche per le sue politiche AWS gestite.	30 marzo 2021

Risoluzione dei problemi di identità e accesso in AWS Glue DataBrew

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con DataBrew IAM.

Argomenti

- Non sono autorizzato a eseguire alcuna azione in DataBrew
- Non sono autorizzato a eseguire iam: PassRole
- Voglio consentire l'accesso a persone esterne al mio AWS account per accedere alle mie DataBrew risorse

Non sono autorizzato a eseguire alcuna azione in DataBrew

Se ti Console di gestione AWS dice che non sei autorizzato a eseguire un'azione, contatta l'amministratore per ricevere assistenza. L'amministratore è colui che ti ha fornito le credenziali di accesso.

L'errore di esempio seguente si verifica quando l'utente `mateojackson` cerca di utilizzare la console per visualizzare i dettagli relativi a un progetto ma non dispone di autorizzazioni `databrew:DescribeProject`.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 databrew:DescribeProject on resource: my-example-project
```

In questo caso, Mateo chiede al suo amministratore di aggiornare le policy per poter accedere alla risorsa *`my-example-project`* mediante l'operazione `databrew:`*`GetProject`*.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un errore che indica che non sei autorizzato a eseguire l'operazione `iam:PassRole`, le tue policy devono essere aggiornate per poter passare un ruolo a DataBrew.

Alcuni Servizi AWS consentono di passare un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

L'errore di esempio seguente si verifica quando un utente IAM denominato `marymajor` cerca di utilizzare la console per eseguire un'operazione in DataBrew. Tuttavia, l'operazione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per trasmettere il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire l'accesso a persone esterne al mio AWS account per accedere alle mie DataBrew risorse

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per servizi che supportano policy basate su risorse o liste di controllo degli accessi (ACL), utilizzare tali policy per concedere alle persone l'accesso alle proprie risorse.

Per maggiori informazioni, consulta gli argomenti seguenti:

- Per sapere se DataBrew supporta queste funzionalità, consultaIn che modo AWS Glue DataBrew funziona con IAM.
- Per scoprire come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta Fornire l'accesso a soggetti Account AWS di proprietà di terze parti nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta Fornire l'accesso a utenti autenticati esternamente (federazione delle identità) nella Guida per l'utente IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente di IAM.

Registrazione e monitoraggio DataBrew

Il monitoraggio è un elemento importante per mantenere l'affidabilità, la disponibilità e le prestazioni delle vostre DataBrew AWS soluzioni. È necessario raccogliere i dati di monitoraggio da tutte le parti della AWS soluzione in modo da poter eseguire più facilmente il debug di un errore multipunto, se si verifica.AWS fornisce diversi strumenti per monitorare le DataBrew risorse e rispondere a potenziali incidenti:

CloudWatch Allarmi Amazon

> Utilizzando Amazon CloudWatch alarms, controlli una singola metrica per un periodo di tempo specificato. Se la metrica supera una determinata soglia, viene inviata una notifica a un argomento o una policy di Amazon SNS.AWS Auto Scaling CloudWatch gli allarmi non richiamano azioni perché si trovano in uno stato particolare. È necessario invece cambiare lo stato e mantenerlo per un numero di periodi specificato.

AWS CloudTrail Registri

CloudTrail fornisce un registro delle azioni intraprese da un utente, un ruolo o un AWS servizio in DataBrew. Utilizzando le informazioni raccolte da CloudTrail, è possibile determinare a quale richiesta è stata inviata DataBrew, l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta, quando è stata effettuata e dettagli aggiuntivi.

Convalida della conformità per AWS Glue DataBrew

Third-party i revisori valutano la sicurezza e la conformità nell'AWS Glue DataBrew ambito di più programmi di AWS conformità. Questi includono SOC, PCI, FedRAMP, HIPAA e altri.

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione Scope by Compliance Program Servizi AWS e scegli il programma di conformità che ti interessa. Per informazioni generali, consulta Programmi di AWS conformità Programmi di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricamento dei report in AWS Artifact .

La vostra responsabilità di conformità durante l'utilizzo Servizi AWSè determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. Per ulteriori informazioni sulla responsabilità di conformità durante l'utilizzo Servizi AWS, consulta la Documentazione AWS sulla sicurezza.

Resilienza in AWS Glue DataBrew

L'infrastruttura AWS globale è costruita attorno a AWS regioni e zone di disponibilità.AWS Le regioni forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità è possibile progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

Ti suggeriamo infatti di configurare i tuoi lavori in modo da utilizzare uno o più tentativi.AWS Glue DataBrew Il numero di tentativi per un processo è configurato nella DataBrew console in Impostazioni avanzate del processo.

Per ulteriori informazioni su AWS regioni e zone di disponibilità, vedere AWS Global Infrastructure.

Sicurezza dell'infrastruttura in AWS Glue DataBrew

Come parte di un servizio gestito,AWS Glue DataBrewè protetto dalle procedure di sicurezza di rete AWS globali descritte nel white paper Amazon Web Services: Overview of Security Processes.

Utilizzi chiamate API AWS pubblicate per accedere DataBrew tramite la rete. I client devono supportare Transport Layer Security (TLS) 1.0 o versioni successive. È consigliabile TLS 1.2 o versioni successive. I client devono inoltre supportare suite di crittografia con Perfect Forward Secrecy (PFS) come Ephemeral (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Diffie-Hellman La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. In alternativa è possibile utilizzare Servizio di token di sicurezza AWS (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

Argomenti

- Utilizzo AWS Glue DataBrew con il tuo VPC
- Utilizzo AWS Glue DataBrew con endpoint VPC

Utilizzo AWS Glue DataBrew con il tuo VPC

Se utilizzi Amazon VPC per ospitare AWS le tue risorse, puoi configurare AWS Glue DataBrew l'instradamento del traffico attraverso il tuo cloud privato virtuale (VPC) basato sul servizio Amazon VPC. DataBrew esegue questa operazione effettuando innanzitutto il provisioning di un'interfaccia di rete elastica nella sottorete specificata. DataBrew quindi collega il gruppo di sicurezza specificato a quell'interfaccia di rete per controllare l'accesso. Il gruppo di sicurezza specificato deve disporre di regole autoreferenziali in entrata e in uscita per tutto il traffico. Inoltre, il tuo VPC deve avere i nomi host e la risoluzione DNS attivati. Per ulteriori informazioni, consulta Configurazione di un VPC per la connessione agli archivi dati JDBC nella Guida per gli sviluppatori.AWS Glue

Per AWS Glue Data Catalog i set di dati, le informazioni VPC vengono configurate quando si crea AWS Glue una connessione nel Data Catalog. Per creare tabelle Data Catalog per questa connessione, esegui un crawler dalla console.AWS GluePer ulteriori informazioni, consulta Populating the AWS Glue Data Catalog nella Developer Guide.AWS Glue

Per i set di dati del database, specifica le informazioni sul VPC quando crei la connessione dalla DataBrew console.

Per utilizzarla AWS Glue DataBrew con una sottorete VPC senza NAT, è necessario disporre di un endpoint VPC gateway per Amazon S3 e un endpoint VPC per l'interfaccia.AWS Glue Per ulteriori informazioni, consulta Creare un endpoint gateway e Interfacciare gli endpoint VPC AWS PrivateLink() nella documentazione di Amazon VPC. L'interfaccia elastica fornita da DataBrew non dispone di un indirizzo IPv4 pubblico e quindi non supporta l'uso di un Internet Gateway VPC.

Gli endpoint dell'interfaccia Amazon S3 non sono supportati al momento. Se stai usando AWS Secrets Manager per archiviare il tuo segreto, hai bisogno di un percorso per Secrets Manager. Se si utilizza la crittografia, è necessario un percorso verso AWS Key Management Service(AWS KMS).

Utilizzo AWS Glue DataBrew con endpoint VPC

Se utilizzi Amazon VPC per ospitare AWS le tue risorse, puoi stabilire una connessione privata tra il tuo VPC e DataBrew fornire un endpoint VPC. Utilizzando questo endpoint VPC, puoi effettuare DataBrew chiamate API.

Non è necessario utilizzare un endpoint DataBrew VPC con DataBrew il tuo VPC. Per ulteriori informazioni, consulta Utilizzo AWS Glue DataBrew con il tuo VPC.

Puoi utilizzarlo AWS Glue con gli endpoint VPC in tutte le AWS regioni che supportano sia gli endpoint VPC che gli endpoint AWS Glue VPC.

Per ulteriori informazioni, consulta questi argomenti nella Guida per l'utente di Amazon VPC:

- What Is Amazon VPC?
- Creazione di un endpoint di interfaccia

Analisi della configurazione e delle vulnerabilità in AWS Glue DataBrew

La configurazione e i controlli IT sono una responsabilità condivisa tra voi AWS e voi, nostri clienti. Per ulteriori informazioni, consulta il modello di responsabilitàAWS condivisa.

Monitoraggio AWS Glue DataBrew

Il monitoraggio è un elemento importante per mantenere l'affidabilità, la disponibilità e le prestazioni delle AWS Glue DataBrew altre AWS soluzioni esistenti.AWS fornisce i seguenti strumenti di monitoraggio per osservare DataBrew, segnalare quando qualcosa non va e intraprendere azioni automatiche quando necessario:

- Amazon CloudWatch monitora AWS le tue risorse e le applicazioni su cui esegui AWS in tempo reale. È possibile raccogliere e tenere traccia dei parametri, creare pannelli di controllo personalizzati e impostare allarmi per inviare una notifica o intraprendere azioni quando un parametro specificato raggiunge una determinata soglia. Ad esempio, puoi tenere CloudWatch traccia dell'utilizzo della CPU o di altri parametri delle tue istanze Amazon EC2 e avviare automaticamente nuove istanze quando necessario. Per ulteriori informazioni, consulta la Amazon CloudWatch User Guide.
- Amazon CloudWatch Events ti consente di configurare notifiche automatiche per eventi specifici in DataBrew. Gli eventi di DataBrew vengono trasmessi a CloudWatch Events quasi in tempo reale. È possibile configurare CloudWatch Events per monitorare gli eventi e richiamare le destinazioni in risposta a eventi che indicano modifiche alle condivisioni di risorse. Le modifiche a una condivisione di risorse attivano eventi sia per il proprietario della condivisione di risorse che per i principali a cui è stato concesso l'accesso alla condivisione di risorse. Per ulteriori informazioni, consulta la Amazon CloudWatch Events User Guide.
- Amazon CloudWatch Logs ti consente di monitorare, archiviare e accedere ai tuoi file di log da istanze Amazon EC2 e altre CloudTrail fonti. CloudWatch I log possono monitorare le informazioni nei file di registro e avvisarti quando vengono raggiunte determinate soglie. Puoi inoltre archiviare i dati del log in storage estremamente durevole. Per ulteriori informazioni, consulta la Amazon CloudWatch Logs User Guide.
- AWS CloudTrailacquisisce le chiamate API e gli eventi correlati effettuati da o per conto del tuo AWS account. Distribuisce quindi i file di log a un bucket Amazon S3 specificato. È possibile identificare quali utenti e account hanno effettuato le chiamate AWS, l'indirizzo IP di origine da cui sono state effettuate le chiamate e quando sono avvenute le chiamate. Per ulteriori informazioni, consulta la Guida per l'utente AWS CloudTrail.

Argomenti

- Monitoraggio DataBrew con CloudWatch log
- Registrazione delle chiamate API con DataBrew AWS CloudTrail
- Utilizzo AWS Notifiche utente con AWS Glue Databrew

Monitoraggio DataBrew con Amazon CloudWatch

È possibile monitorare DataBrew l'utilizzo CloudWatch, che raccoglie dati grezzi e li elabora in metriche leggibili e quasi in tempo reale. Queste statistiche vengono conservate per un periodo di 15 mesi, per permettere l'accesso alle informazioni storiche e offrire una prospettiva migliore sulle prestazioni del servizio o dell'applicazione web. È anche possibile impostare allarmi che controllano determinate soglie e inviare notifiche o intraprendere azioni quando queste soglie vengono raggiunte. Per ulteriori informazioni, consulta la Amazon CloudWatch User Guide.

AWS Glue DataBrew riporta le seguenti metriche nel `AWS/DataBrew` namespace.

Metrica	Description
`SessionCount`	Il numero totale di DataBrew sessioni nell'account del cliente Dimensioni valide: LogGroupName Statistiche valide: Sum Unità: numero

Automazione DataBrew con eventi CloudWatch

Amazon CloudWatch Events ti consente di automatizzare AWS i tuoi servizi e rispondere automaticamente a eventi di sistema come problemi di disponibilità delle applicazioni o modifiche delle risorse. Gli eventi derivanti dai AWS servizi vengono trasmessi a CloudWatch Events quasi in tempo reale. Puoi compilare regole semplici che indichino quali eventi sono considerati di interesse per te e quali azioni automatizzate intraprendere quando un evento corrisponde a una regola. Le azioni che possono essere attivate automaticamente includono le seguenti:

- Richiamo del comando run di Amazon EC2
- Inoltro dell'evento a Amazon Kinesis Data Streams

- Attivazione di una macchina a stati AWS Step Functions
- Notifica di un argomento Amazon SNS o di una coda Amazon SQS

DataBrew segnala un evento a CloudWatch Events ogni volta che lo stato di una risorsa nel tuo AWS account cambia. Gli eventi vengono emessi secondo il principio del massimo sforzo.

Di seguito sono riportati alcuni esempi di diversi eventi, che mostrano i vari stati di un DataBrew lavoro: `SUCCEEDED`,`FAILED`,`TIMEOUT`, e`STOPPED`.

```
{
  "version": "0",
  "id": "abcdef00-1234-5678-9abc-def012345678",
  "detail-type": "DataBrew Job State Change",
  "source": "aws.databrew",
  "account": "123456789012",
  "time": "2017-09-07T18:57:21Z",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "jobName": "MyJob",
    "severity": "INFO",
    "state": "SUCCEEDED",
    "jobRunId": "db_abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789",
    "message": "Job run succeeded"
  }
}

{
  "version": "0",
  "id": "abcdef01-1234-5678-9abc-def012345678",
  "detail-type": "DataBrew Job State Change",
  "source": "aws.databrew",
  "account": "123456789012",
  "time": "2017-09-07T06:02:03Z",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "jobName": "MyJob",
    "severity": "ERROR",
    "state": "FAILED",
    "jobRunId": "db_0123456789abcdef0123456789abcdef0123456789abcdef0123456789abcdef",
```

```
    "message": "AnalysisException: 'Path does not exist: s3://MyBucket/MyFile;'"
  }
}

{
  "version": "0",
  "id": "abcdef00-1234-5678-9abc-def012345678",
  "detail-type": "DataBrew Job State Change",
  "source": "aws.databrew",
  "account": "123456789012",
  "time": "2017-11-20T20:22:06Z",
  "region": "us-east-2",
  "resources": [],
  "detail": {
    "jobName": "MyJob",
    "severity": "WARN",
    "state": "TIMEOUT",
    "jobRunId": "db_abc0123456789abcdef0123456789abcdef0123456789abcdef0123456789def",
    "message": "Job run timed out"
  }
}

{
  "version": "0",
  "id": "abcdef00-1234-5678-9abc-def012345678",
  "detail-type": "DataBrew Job State Change",
  "source": "aws.databrew",
  "account": "123456789012",
  "time": "2017-11-20T20:22:06Z",
  "region": "us-east-2",
  "resources": [],
  "detail": {
    "jobName": "MyJob",
    "severity": "INFO",
    "state": "STOPPED",
    "jobRunId": "db_abc0123456789abcdef0123456789abcdef0123456789abcdef0123456789def",
    "message": "Job run stopped"
  }
}
```

Per ulteriori informazioni, consulta la Amazon CloudWatch Events User Guide.

Monitoraggio DataBrew con CloudWatch log

È possibile monitorare i DataBrew lavori utilizzando CloudWatch Logs, che raccoglie informazioni dettagliate dal sottosistema dei DataBrew lavori e le rende disponibili per la revisione. Questi log possono essere utili se desideri ottenere informazioni dettagliate sulle risorse utilizzate dal tuo profilo e dai lavori di elaborazione delle ricette o per la risoluzione dei problemi. Per ulteriori informazioni, consulta la Amazon CloudWatch Logs User Guide.

Registrazione delle chiamate API con DataBrew AWS CloudTrail

DataBrew è integrato con AWS CloudTrail, un servizio che fornisce una registrazione delle azioni intraprese da un utente, un ruolo o un AWS servizio in DataBrew. CloudTrail acquisisce tutte le chiamate API DataBrew come eventi. Le chiamate acquisite includono chiamate dalla DataBrew console e chiamate di codice alle operazioni DataBrew API. Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon S3, inclusi gli eventi per. DataBrew Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella CloudTrail console nella cronologia degli eventi. Utilizzando le informazioni raccolte da CloudTrail, puoi determinare la richiesta a cui è stata inviata DataBrew. Puoi determinare l'indirizzo IP da cui è stata effettuata la richiesta, l'autore della richiesta, il momento in cui è stata effettuata e i dettagli aggiuntivi.

Per ulteriori informazioni CloudTrail, consulta la Guida AWS CloudTrail per l'utente.

DataBrew Informazioni in CloudTrail

CloudTrail è abilitato sul tuo AWS account al momento della creazione dell'account. Quando si verifica un'attività in DataBrew, tale attività viene registrata in un CloudTrail evento insieme ad altri eventi AWS di servizio nella cronologia degli eventi. Puoi visualizzare, cercare e scaricare gli eventi recenti nel tuo AWS account. Per ulteriori informazioni, consulta Visualizzazione degli eventi con cronologia degli CloudTrail eventi nella Guida AWS CloudTrail per l'utente.

Per una registrazione continua degli eventi nel tuo AWS account, inclusi gli eventi per DataBrew, crea un percorso. Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Per impostazione predefinita, quando crei un percorso nella console, il percorso si applica a tutte le AWS regioni. Il trail registra gli eventi di tutte le regioni della AWS partizione e consegna i file di log al bucket Amazon S3 specificato. Inoltre, puoi configurare altri AWS servizi per analizzare ulteriormente e agire in base ai dati sugli eventi raccolti nei log. CloudTrail Per ulteriori informazioni, consultare gli argomenti seguenti nella Guida per l'utente di AWS CloudTrail:

- Panoramica della creazione di un trail
- CloudTrail Servizi e integrazioni supportati
- Configurazione delle notifiche Amazon SNS per CloudTrail
- Ricezione di file di CloudTrail registro da più regioni e ricezione di file di CloudTrail registro da più account

Tutte DataBrew le azioni vengono registrate CloudTrail e documentate nell'API Reference. Ad esempio, le chiamate a `UpdateRecipe` e `CreateDataset` le `StartJobRun` azioni generano voci nei file di CloudTrail registro.

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con credenziali utente o root.
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro AWS servizio.

Per ulteriori informazioni, consulta Elemento CloudTrail userIdentity.

Comprensione delle DataBrew voci dei file di registro

Ancora una volta, un CloudTrail trail è una configurazione che consente la consegna di eventi come file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta proveniente da qualsiasi fonte e include informazioni sull'azione richiesta, la data e l'ora dell'azione, i parametri della richiesta e così via. CloudTrail i file di registro non sono una traccia ordinata dello stack delle chiamate API pubbliche, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di CloudTrail registro che dimostra l'`CreateProfileJob`operazione.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
```

```
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::1234567890:user/joe",
        "accountId": "1234567890",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "joe"
    },
    "eventTime": "2020-11-09T18:54:44Z",
    "eventSource": "databrew.amazonaws.com",
    "eventName": "CreateProfileJob",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "requestParameters": {
        "OutputLocation": {
            "Bucket": "bucketName",
            "Key": "keyName"
        },
        "DatasetName": "my-chess-dataset",
        "RoleArn": "arn:aws:iam::1234567890:role/custom-role",
        "Name": "my-profile-job"
    },
    "responseElements": {
        "Name": "my-profile-job"
    },
    "requestID": "993bc3b8-3980-48dd-961e-c1c8529eb248",
    "eventID": "f8128dfa-df29-458b-a2d5-34805b46eefd",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "1234567890"
}
```


Utilizzo AWS Notifiche utente con AWS Glue Databrew

Puoi utilizzare le notifiche AWS utente per configurare i canali di distribuzione per ricevere notifiche sugli eventi di AWS Glue Databrew. L'utente riceverà una notifica quando un evento corrisponde a una regola specificata. È possibile ricevere notifiche per gli eventi tramite più canali, tra cui e-mail, Amazon Q Developer in applicazioni chat, notifiche chat, o notifiche push AWS Console Mobile Application. È anche possibile visualizzare le notifiche nel Centro notifiche della console.AWS Le notifiche utente supportano l'aggregazione, che può ridurre il numero di notifiche ricevute durante eventi specifici.

Fase della ricetta e riferimento alla funzione

In questo riferimento, puoi trovare le descrizioni dei passaggi e delle funzioni della ricetta che puoi utilizzare a livello di codice, utilizzando AWS CLI o utilizzando uno degli SDK.AWS In DataBrew pratica, una fase della ricetta è un'azione che trasforma i dati grezzi in un modulo pronto per essere utilizzato dalla pipeline di dati. Una DataBrew funzione è un tipo speciale di fase della ricetta che esegue un calcolo basato su parametri.

Le categorie per le trasformazioni nell'interfaccia utente includono quanto segue:

- Colonna di base: passaggi relativi alla ricetta
  - Filtro
  - Colonna
- Fasi della ricetta per la pulizia dei dati
  - Formato
  - Elimina
  - Estrarre
- Fasi della ricetta per la qualità dei dati
  - Mancante
  - Non valido
  - Duplicato
  - Valori anomali
- Fasi della ricetta relative alle informazioni di identificazione personale (PII)
  - Maschera le informazioni personali
  - Sostituisci le informazioni personali
  - Crittografa le informazioni personali
  - Mischiare le righe
- Struttura delle colonne: passaggi della ricetta
  - Split
  - Unione
  - Crea
- Fasi della ricetta per la formattazione delle colonne

  - Precisione decimale
  - Separatore delle migliaia
  - Numeri abbreviati
- Fasi della ricetta della struttura dei dati
  - Nest-Unnest
  - Pivot
  - Gruppo
  - Join
  - Union
- Fasi della ricetta della scienza dei dati
  - Testo
  - Dimensionare
  - Mapping
  - Encode
- Funzioni
  - Funzioni matematiche
  - Funzioni di aggregazione
  - Funzioni di testo
  - Funzioni di data e ora
  - Funzioni finestra
  - Funzioni Web
  - Altre funzioni

Per ulteriori informazioni su come questi passaggi e funzioni della ricetta vengono utilizzati in una ricetta (incluso l'uso delle espressioni condizionali), vedereDefinizione della struttura di una ricetta.

Le sezioni seguenti descrivono i passaggi e le funzioni della ricetta, organizzati in base al loro scopo.

Argomenti

Passaggi base della ricetta della colonna

Utilizza queste azioni di base relative alla composizione delle colonne per eseguire semplici trasformazioni sui dati.

Argomenti

- TO_DOUBLE_COLUMN
- TO_NUMBER_COLUMN
- TO_STRING_COLUMN

CHANGE_DATA_TYPE

Modifica il tipo di dati di una colonna esistente.

Se il valore di una colonna non può essere convertito nel nuovo tipo, verrà sostituito con NULL. Ciò può accadere quando una colonna di stringhe viene convertita in una colonna intera. Ad esempio, la stringa «123" diventerà il numero intero 123, ma la stringa «ABC» non può diventare un numero, quindi verrà sostituita con un valore NULL.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Nuovo tipo di colonna. Sono supportati i tipi di dati seguenti:
  - byte: numeri interi con segno a 1 byte. L'intervallo di numeri è compreso tra -128 e 127.
  - breve: numeri interi con segno a 2 byte. L'intervallo di numeri è compreso tra -32768 e 32767.
  - int: numeri interi con segno a 4 byte. L'intervallo di numeri va da -2147483648 a 2147483647.
  - long: numeri interi con segno a 8 byte. L'intervallo di numeri va da -9223372036854775808 a 9223372036854775807.
  - float: numeri in virgola mobile a precisione singola a 4 byte.
  - double: numeri in virgola mobile a doppia precisione da 8 byte.
  - decimali: numeri decimali firmati con un massimo di 38 cifre in totale e 18 cifre dopo la virgola decimale.
  - string: valori della stringa di caratteri.
  - booleano: il tipo booleano ha uno dei due valori possibili: `true` e `false` o `yes` e `no`.
  - timestamp: Valori che comprendono i campi anno, mese, giorno, ora, minuto e secondo.
  - data: Valori che comprendono i campi anno, mese e giorno.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "CHANGE_DATA_TYPE",
        "Parameters": {
            "sourceColumn": "columnName",
            "columnDataType": "boolean"
        }
    }
}
```

DELETE

Rimuove una colonna dal set di dati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DELETE",
        "Parameters": {
            "sourceColumn": "extra_data"
        }
    }
}
```

DUPLICARE

Crea una nuova colonna con un nome diverso, ma con tutti gli stessi dati. Sia la vecchia che la nuova colonna vengono mantenute nel set di dati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`— Un nome per la colonna duplicata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DUPLICATE",
        "Parameters": {
            "sourceColumn": "last_name",
            "targetColumn": "copy_of_last_name"
        }
    }
}
```

JSON_TO_STRUCTS

Converte una stringa JSON in strutture tipizzate staticamente. Durante la conversione, rileva lo schema di ogni oggetto JSON e li unisce per ottenere lo schema più generico che rappresenti l'intera stringa JSON. Il parametro «UnnestLevel» specifica quanti livelli di oggetti JSON convertire in strutture.

Parameters

- `sourceColumns`— Un elenco di colonne di origine.
- `regexColumnSelector` –Un'espressione regolare per selezionare le colonne.
- `removeSourceColumn`— Un valore booleano. In `true` tal caso rimuovi la colonna di origine; in caso contrario, conservala.
- `unnestLevel`— Il numero di livelli da eliminare.
- `conditionExpressions`— Espressioni condizionali.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "JSON_TO_STRUCTS",
        "Parameters": {
            "sourceColumns": "[\"address\"]",
            "removeSourceColumn": "true",
            "unnestLevel": "2"
       }
    }
}
```

MOVE_AFTER

Sposta una colonna nella posizione immediatamente successiva a un'altra colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`— Il nome di un'altra colonna. La colonna specificata da `sourceColumn` verrà spostata immediatamente dopo la colonna specificata da`targetColumn`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MOVE_AFTER",
        "Parameters": {
            "sourceColumn": "rating",
            "targetColumn": "height_cm"
        }
    }
}
```

MOVE_BEFORE

Sposta una colonna nella posizione immediatamente precedente a un'altra colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`— Il nome di un'altra colonna. La colonna specificata da `sourceColumn` verrà spostata immediatamente dopo la colonna specificata da`targetColumn`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MOVE_BEFORE",
        "Parameters": {
```

```
            "sourceColumn": "height_cm",
            "targetColumn": "weight_kg"
        }
    }
}
```

MOVE_TO_END

Sposta una colonna nella posizione finale (ultima colonna) nel set di dati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MOVE_TO_END",
        "Parameters": {
            "sourceColumn": "height_cm"
        }
    }
}
```

MOVE_TO_INDEX

Sposta una colonna in una posizione specificata da un numero.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetIndex`— La nuova posizione per la colonna. Le posizioni iniziano con 0, quindi, ad esempio, 1 si riferisce alla seconda colonna, 2 alla terza colonna e così via.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "MOVE_TO_INDEX",
        "Parameters": {
            "sourceColumn": "nationality",
            "targetIndex": "5"
        }
    }
}
```

MOVE_TO_START

Sposta una colonna nella posizione iniziale (prima colonna) nel set di dati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MOVE_TO_START",
        "Parameters": {
            "sourceColumn": "first_name"
        }
    }
}
```

RENAME

Crea una nuova colonna con un nome diverso, ma con tutti gli stessi dati. La vecchia colonna viene quindi rimossa dal set di dati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`— Un nuovo nome per la colonna.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "RENAME",
        "Parameters": {
            "sourceColumn": "date_of_birth",
            "targetColumn": "birth_date"
        }
    }
}
```

SORT

Ordina i dati in una o più colonne di un set di dati in ordine crescente, decrescente o personalizzato.

Parameters

- `expressions`— Una stringa che contiene una o più JSON-encoded stringhe che rappresentano espressioni di ordinamento.
  - `sourceColumn`— Una stringa che contiene il nome di una colonna esistente.
  - `ordering`— L'ordinamento può essere CRESCENTE o DECRESCENTE.
  - `nullsOrdering`— L'ordinamento dei valori Null può essere NULLS_TOP o NULLS_BOTTOM per inserire valori nulli o mancanti all'inizio o alla fine della colonna.
  - `customOrder`— Un elenco di stringhe che definisce un ordine personalizzato per l'ordinamento delle stringhe. Per impostazione predefinita, le stringhe vengono ordinate alfabeticamente.
  - `isCustomOrderCaseSensitive`: booleano. Il valore predefinito è `false`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SORT",
        "Parameters": {
            "expressions": "[{\"sourceColumn\": \"A\", \"ordering\": \"ASCENDING\",
 \"nullsOrdering\": \"NULLS_TOP\"}]",
       }
    }
 }
```

Example Esempio di ordinamento personalizzato

Nell'esempio seguente, la stringa di espressione CustomOrder ha il formato di un elenco di oggetti. Ogni oggetto descrive un'espressione di ordinamento per una colonna.

```
[
  {
    "sourceColumn": "A",
    "ordering": "ASCENDING",
    "nullsOrdering": "NULLS_TOP",
  },
  {
    "sourceColumn": "B",
    "ordering": "DESCENDING",
    "nullsOrdering": "NULLS_BOTTOM",
    "customOrder": ["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"],
    "isCustomOrderCaseSensitive": false,
  }
]
```

TO_BOOLEAN_COLUMN

Modifica il tipo di dati di una colonna esistente in BOOLEAN.

> **Note**
>
> Ti consigliamo di utilizzare l'azione di ricetta CHANGE_DATA_TYPE anziché TO_BOOLEAN_COLUMN.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Un valore `boolean` che deve essere.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "TO_BOOLEAN_COLUMN",
        "Parameters": {
            "columnDataType": "boolean",
            "sourceColumn": "is_present"
        }
    }
}
```

TO_DOUBLE_COLUMN

Modifica il tipo di dati di una colonna esistente in DOUBLE.

> **Note**
>
> Consigliamo di utilizzare l'azione di ricetta CHANGE_DATA_TYPE anziché TO_DOUBLE_COLUMN.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Un valore `number` che deve essere.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "TO_DOUBLE_COLUMN",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "hourly_rate"
        }
    }
}
```

TO_NUMBER_COLUMN

Modifica il tipo di dati di una colonna esistente in NUMBER.

> **Note**
>
> Consigliamo di utilizzare l'azione di ricetta CHANGE_DATA_TYPE anziché TO_NUMBER_COLUMN.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Un valore `number` che deve essere.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "TO_NUMBER_COLUMN",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "hours_worked"
        }
    }
}
```

TO_STRING_COLUMN

Modifica il tipo di dati di una colonna esistente in STRING.

> **Note**
>
> Consigliamo di utilizzare l'azione di ricetta CHANGE_DATA_TYPE anziché TO_STRING_COLUMN.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Un valore `string` che deve essere.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "TO_STRING_COLUMN",
        "Parameters": {
            "columnDataType": "string",
            "sourceColumn": "age"
        }
    }
}
```

Fasi della ricetta per la pulizia dei dati

Utilizza questi passaggi della ricetta di pulizia dei dati per eseguire semplici trasformazioni sui dati esistenti.

Argomenti

- SOSTITUIRE_BETWEEN_POSITIONS
- SOSTITUIRE_TEXT

CAPITAL_CASE

Modifica ogni stringa in una colonna per rendere maiuscola ogni parola. In maiuscolo, la prima lettera di ogni parola viene trasformata in maiuscolo e il resto della parola viene trasformato in minuscolo. Un esempio è: The Quick Brown Fox Jumped Over The Fence.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "CAPITAL_CASE",
        "Parameters": {
            "sourceColumn": "last_name"
        }
    }
}
```

FORMAT_DATE

Restituisce una colonna in cui una stringa di data viene convertita in un valore formattato.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetDateFormat`— Uno dei seguenti formati di data:
  - `mm/dd/yyyy`
  - `mm-dd-yyyy`
  - `dd month yyyy`
  - `month yyyy`
  - `dd month`

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FORMAT_DATE",
        "Parameters": {
            "sourceColumn": "birth_date",
            "targetDateFormat": "mm-dd-yyyy"
        }
    }
}
```

MINUSCOLO

Trasforma ogni stringa di una colonna in minuscolo, ad esempio: la rapida volpe bruna saltò oltre la recinzione

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "LOWER_CASE",
        "Parameters": {
            "sourceColumn": "nationality"
        }
    }
}
```

MAIUSCOLO_MINUSCOLO

Trasforma ogni stringa in maiuscolo in una colonna, ad esempio: THE QUICK BROWN FOX JUMPED OVER THE FENCE

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UPPER_CASE",
        "Parameters": {
            "sourceColumn": "nationality"
        }
    }
}
```

SENTENCE_CASE

Cambia ogni stringa in una colonna in maiuscole e minuscole. In maiuscole e minuscole, la prima lettera di ogni frase viene trasformata in maiuscolo e il resto della frase viene trasformato in minuscolo. Un esempio è: La volpe bruna veloce. È saltato. La recinzione

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SENTENCE_CASE",
        "Parameters": {
            "sourceColumn": "description"
        }
    }
}
```

ADD_DOUBLE_QUOTES

Racchiude i caratteri in una colonna tra virgolette doppie.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADD_DOUBLE_QUOTES",
        "Parameters": {
            "sourceColumn": "info_url"
        }
    }
}
```

ADD_PREFIX

Aggiunge uno o più caratteri, concatenandoli come prefisso all'inizio di una colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `pattern`— Il carattere o i caratteri da inserire all'inizio dei valori delle colonne.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADD_PREFIX",
        "Parameters": {
            "pattern": "aaa",
            "sourceColumn": "info_url"
        }
    }
}
```

ADD_SINGLE_QUOTES

Racchiude i caratteri in una colonna tra virgolette singole.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADD_SINGLE_QUOTES",
        "Parameters": {
            "sourceColumn": "info_url"
        }
    }
}
```

ADD_SUFFIX

Aggiunge un altro carattere concatenandolo come suffisso alla fine di una colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `pattern`— Il carattere o i caratteri da posizionare alla fine della colonna.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADD_SUFFIX",
        "Parameters": {
            "pattern": "bbb",
            "sourceColumn": "info_url"
        }
    }
}
```

EXTRACT_BETWEEN_DELIMITERS

Crea una nuova colonna, basata su delimitatori, dai valori di una colonna esistente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- `targetColumn`: il nome della nuova colonna da creare.
- `startPattern`— Un'espressione regolare, che indica il carattere o i caratteri che iniziano i valori delimitati.
- `endPattern`— Un'espressione regolare, che indica il carattere o i caratteri delimitatori che terminano i valori delimitati.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "EXTRACT_BETWEEN_DELIMITERS",
        "Parameters": {
            "endPattern": "\\/",
            "sourceColumn": "info_url",
            "startPattern": "\\/\\/",
            "targetColumn": "raw_url"
        }
    }
}
```

EXTRACT_BETWEEN_POSITIONS

Crea una nuova colonna, in base alla posizione dei caratteri, dai valori di una colonna esistente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.
- `startPosition`— La posizione del carattere in cui eseguire l'estrazione.
- `endPosition`— La posizione del carattere in cui terminare l'estrazione.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "EXTRACT_BETWEEN_POSITIONS",
```

```
        "Parameters": {
            "endPosition": "9",
            "sourceColumn": "last_name",
            "startPosition": "3",
            "targetColumn": "characters_3_to_9"
        }
    }
}
```

EXTRACT_PATTERN

Crea una nuova colonna, basata su un'espressione regolare, dai valori di una colonna esistente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.
- `pattern`— Un'espressione regolare che indica il carattere o i caratteri da cui estrarre e creare la nuova colonna.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "EXTRACT_PATTERN",
        "Parameters": {
            "pattern": "^....*...$",
            "sourceColumn": "last_name",
            "targetColumn": "first_and_last_few_characters"
        }
    }
}
```

EXTRACT_VALUE

Crea una nuova colonna con un valore estratto da un percorso specificato dall'utente. Se la colonna di origine è di tipo Map, Array o Struct, ogni campo del percorso deve essere eliminato utilizzando i segni di spunta inversa (ad esempio, `name`).

Parameters

- `targetColumn`— Il nome della colonna di destinazione.
- `sourceColumn`— Nome della colonna di origine da cui estrarre il valore.
- `path`— Il percorso della chiave specifica che l'utente desidera estrarre. Se la colonna di origine è di tipo Map, Array o Struct, ogni campo del percorso deve essere eliminato utilizzando i segni di spunta inversa (ad esempio, `name`).

  Considerate il seguente esempio di informazioni sull'utente:

```
user {
   name: “Ammy”
   address: {
      state: "CA",
      zipcode: 12345
   },
   phoneNumber:{"home": "123123123", "work": "456456456"}
   citizenship: ["Canada", "USA", "Mexico", "India"]
}
```

  Di seguito sono riportati alcuni esempi dei percorsi da fornire, a seconda del tipo di colonna di origine:

  - Se la colonna di origine è del tipo mappa, il percorso per estrarre il numero di telefono di casa è:

    `` `user`.`phoneNumber`.`home` ``
  - Se la colonna di origine è del tipo array, il percorso per estrarre il secondo valore di «cittadinanza» è:

    `` `user`.`citizenship`[1] ``
  - Se la colonna di origine è del tipo struct, il percorso per l'estrazione del codice postale è:

    `` `user`.`address`.`zipcode` ``

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "EXTRACT_VALUE",
        "Parameters": {
            "sourceColumn": "age",
            "targetColumn": "columnName",
            "path": "`age`.`name`",
        }
    }
}
```

REMOVE_COMBINED

Rimuove uno o più caratteri da una colonna, in base a quanto specificato dall'utente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `collapseConsecutiveWhitespace`— If`true`, sostituisce due o più caratteri di spazio bianco con esattamente un carattere di spazio bianco.
- `removeAllPunctuation`— Set`true`, rimuove tutti i seguenti caratteri: `. ! , ?`
- `removeAllQuotes`— Set`true`, rimuove tutte le virgolette singole e doppie.
- `removeAllWhitespace`— Set`true`, rimuove tutti gli spazi vuoti.
- `customCharacters`— Uno o più personaggi su cui è possibile agire.
- `customValue`— Un valore su cui si può agire.
- `removeCustomCharacters`— Set`true`, rimuove tutti i caratteri specificati dal `customCharacters` parametro.
- `removeCustomValue`— Set`true`, rimuove tutti i caratteri specificati dal `customValue` parametro.
- `punctuationally`— If`true`, rimuove i seguenti caratteri se si trovano all'inizio o alla fine del valore: `. ! , ?`
- `antidisestablishmentarianism`— Set`true`, rimuove le virgolette singole e le virgolette doppie dall'inizio e dalla fine del valore.
- `removeLeadingAndTrailingWhitespace`— Set`true`, rimuove tutti gli spazi bianchi dall'inizio e dalla fine del valore.
- `removeLetters`— Set`true`, rimuove tutti i caratteri alfabetici maiuscoli e minuscoli (Afino a; fino a). `Z a z`

- `removeNumbers`— Set`true`, rimuove tutti i caratteri numerici (fino a). `0` `9`
- `removeSpecialCharacters`— Set`true`, rimuove tutti i seguenti caratteri: `! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [ \ ] ^ _ ` { | } ~`

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "REMOVE_COMBINED",
        "Parameters": {
            "collapseConsecutiveWhitespace": "false",
            "removeAllPunctuation": "false",
            "removeAllQuotes": "false",
            "removeAllWhitespace": "false",
            "removeCustomCharacters": "false",
            "removeCustomValue": "false",
            "removeLeadingAndTrailingPunctuation": "false",
            "removeLeadingAndTrailingQuotes": "false",
            "removeLeadingAndTrailingWhitespace": "false",
            "removeLetters": "false",
            "removeNumbers": "false",
            "removeSpecialCharacters": "true",
            "sourceColumn": "info_url"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REMOVE_COMBINED",
        "Parameters": {
            "collapseConsecutiveWhitespace": "false",
            "customCharacters": "¶",
            "removeAllPunctuation": "false",
            "removeAllQuotes": "false",
            "removeAllWhitespace": "false",
            "removeCustomCharacters": "true",
            "removeCustomValue": "false",
            "removeLeadingAndTrailingPunctuation": "false",
            "removeLeadingAndTrailingQuotes": "false",
            "removeLeadingAndTrailingWhitespace": "false",
```

```
            "removeLetters": "false",
            "removeNumbers": "false",
            "removeSpecialCharacters": "false",
            "sourceColumn": "info_url"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REMOVE_COMBINED",
        "Parameters": {
            "collapseConsecutiveWhitespace": "true",
            "customValue": "M",
            "removeAllPunctuation": "true",
            "removeAllQuotes": "false",
            "removeAllWhitespace": "false",
            "removeCustomCharacters": "false",
            "removeCustomValue": "true",
            "removeLeadingAndTrailingPunctuation": "false",
            "removeLeadingAndTrailingQuotes": "true",
            "removeLeadingAndTrailingWhitespace": "true",
            "removeLetters": "true",
            "removeNumbers": "true",
            "removeSpecialCharacters": "false",
            "sourceColumn": "info_url"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REMOVE_COMBINED",
        "Parameters": {
            "collapseConsecutiveWhitespace": "false",
            "removeAllPunctuation": "false",
            "removeAllQuotes": "false",
            "removeAllWhitespace": "false",
            "removeCustomCharacters": "false",
            "removeCustomValue": "false",
            "removeLeadingAndTrailingPunctuation": "false",
            "removeLeadingAndTrailingQuotes": "false",
            "removeLeadingAndTrailingWhitespace": "false",
```

```
            "removeLetters": "false",
            "removeNumbers": "true",
            "removeSpecialCharacters": "false",
            "sourceColumn": "first_name"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REMOVE_COMBINED",
        "Parameters": {
            "collapseConsecutiveWhitespace": "false",
            "removeAllPunctuation": "false",
            "removeAllQuotes": "false",
            "removeAllWhitespace": "false",
            "removeCustomCharacters": "false",
            "removeCustomValue": "false",
            "removeLeadingAndTrailingPunctuation": "false",
            "removeLeadingAndTrailingQuotes": "false",
            "removeLeadingAndTrailingWhitespace": "false",
            "removeLetters": "false",
            "removeNumbers": "true",
            "removeSpecialCharacters": "false",
            "sourceColumn": "first_name"
        }
    }
}
```

REPLACE_BETWEEN_DELIMITERS

Sostituisce i caratteri tra due delimitatori con testo specificato dall'utente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `startPattern`— Carattere o caratteri o un'espressione regolare, che indica da dove deve iniziare la sostituzione.
- `endPattern`— Carattere o caratteri o un'espressione regolare, che indica dove deve terminare la sostituzione.
- `value`— Il carattere o i caratteri sostitutivi da sostituire.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_BETWEEN_DELIMITERS",
        "Parameters": {
            "endPattern": ">",
            "sourceColumn": "last_name",
            "startPattern": "&lt;",
            "value": "?"
        }
    }
}
```

SOSTITUIRE_BETWEEN_POSITIONS

Sostituisce i caratteri tra due posizioni con il testo specificato dall'utente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `startPosition`— Un numero che indica da quale posizione del carattere nella stringa deve iniziare la sostituzione.
- `endPosition`— Un numero che indica in quale posizione del carattere nella stringa deve terminare la sostituzione.
- `value`— Il carattere o i caratteri sostitutivi da sostituire.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_BETWEEN_POSITIONS",
        "Parameters": {
            "endPosition": "20",
            "sourceColumn": "nationality",
            "startPosition": "10",
            "value": "E"
        }
```

```
    }
}
```

SOSTITUIRE_TEXT

Sostituisce una sequenza di caratteri specificata con un'altra.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `pattern`— Carattere o caratteri o un'espressione regolare, che indica quali caratteri devono essere sostituiti nella colonna di origine.
- `value`— Il carattere o i caratteri sostitutivi da sostituire.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "REPLACE_TEXT",
        "Parameters": {
            "pattern": "x",
            "sourceColumn": "first_name",
            "value": "a"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REPLACE_TEXT",
        "Parameters": {
            "pattern": "[0-9]",
            "sourceColumn": "nationality",
            "value": "!"
        }
    }
}
```

Fasi della ricetta per la qualità dei dati

Utilizza queste istruzioni sulla qualità dei dati per inserire i valori mancanti, rimuovere i dati non validi o rimuovere i duplicati.

Argomenti

- SOSTITUIRE_WITH_MOST_FREQUENT
- SOSTITUIRE_WITH_NULL
- SOSTITUIRE_WITH_ROLLING_AVERAGE
- SOSTITUIRE_WITH_ROLLING_SUM
- SOSTITUIRE_WITH_SUM

ADVANCED_DATATYPE_FILTER

Filtra la colonna di origine corrente in base al rilevamento avanzato del tipo di dati. Ad esempio, data una colonna DataBrew identificata come contenente codici postali, questa trasformazione può filtrare la colonna in base al fuso orario. I dettagli che è possibile estrarre dipendono dal modello rilevato, come descritto nelle Note di seguito.

Parameters

- `sourceColumn`— Il nome di una colonna sorgente di stringa.
- `pattern`— Lo schema da estrarre.
- `advancedDataType`— Può essere uno tra telefono, codice postale, data e ora, stato, carta di credito, URL, e-mail, SSN o sesso.
- `filter values`— Elenco di valori di stringa in base ai quali l'utente desidera filtrare la colonna.
- `strategy`— KEEP_ROWS o DISCARD_ROWS o CLEAR_FILTERS o CLEAR_OTHERS.
- `clearWithEmpty`— `true` Booleano o, per cancellare le righe con invece di. `false` `empty` `null`

Note

- Se advanced DataType è Phone, il pattern può essere AREA_CODE, TIME_ZONE o COUNTRY_CODE.
- Se advanced DataType è Zip Code, il pattern può essere TIME_ZONE, COUNTRY, STATE, CITY, TYPE o REGION.
- Se advanced DataType è Date Time, il pattern può essere DAY, MONTH, MONTH_NAME, WEEK, QUARTER o YEAR.
- Se advanced DataType è State, il pattern può essere TIME_ZONE.
- Se advanced DataType è Credit Card, il pattern può essere LENGTH o NETWORK.
- Se advanced DataType è URL, il pattern può essere PROTOCOL, TLD o DOMAIN.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADVANCED_DATATYPE_FILTER",
        "Parameters": {
            "pattern": "AREA_CODE",
            "sourceColumn": "phoneColumn",
            "advancedDataType": "Phone",
            "filterValues": ['Ohio'],
            "strategy": "KEEP_ROWS"
        }
    }
}
```

ADVANCED_DATATYPE_FLAG

Crea una nuova colonna di bandiera in base ai valori della colonna di origine corrente. Ad esempio, data una colonna di origine contenente codici postali, questa trasformazione può essere utilizzata per contrassegnare i valori come `true` o in `false` base a un particolare fuso orario. I dettagli che è possibile estrarre dipendono dal modello rilevato, come descritto nelle Note di seguito.

Parameters

- `sourceColumn`— Il nome di una colonna sorgente di stringa.
- `pattern`— Lo schema da estrarre.
- `targetColumn`— Il nome della colonna di destinazione.
- `advancedDataType`— Può essere uno tra telefono, codice postale, data e ora, stato, carta di credito, URL, e-mail, SSN o sesso.
- `filter values`— Elenco di valori di stringa in base ai quali l'utente desidera filtrare la colonna.
- `trueString`— Il `true` valore per la colonna di destinazione.
- `falseString`— Il `false` valore per la colonna di destinazione.

Note

- Se advanced DataType è Phone, il pattern può essere AREA_CODE, TIME_ZONE o COUNTRY_CODE.

- Se advanced DataType è Zip Code, il pattern può essere TIME_ZONE, COUNTRY, STATE, CITY, TYPE o REGION.
- Se advanced DataType è Date Time, il pattern può essere DAY, MONTH, MONTH_NAME, WEEK, QUARTER o YEAR.
- Se advanced DataType è State, il pattern può essere TIME_ZONE.
- Se advanced DataType è Credit Card, il pattern può essere LENGTH o NETWORK.
- Se advanced DataType è URL, il pattern può essere PROTOCOL, TLD o DOMAIN.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADVANCED_DATATYPE_FLAG",
        "Parameters": {
            "pattern": "AREA_CODE",
            "sourceColumn": "phoneColumn",
            "advancedDataType": "Phone",
            "filterValues": ['Ohio'],
            "targetColumn": "targetColumnName",
            "trueString": "trueValue",
            "falseString": "falseValue"
        }
    }
}
```

DELETE_DUPLICATE_ROWS

Elimina qualsiasi riga che corrisponde esattamente a una riga precedente del set di dati. L'occorrenza iniziale non viene eliminata perché non corrisponde a una riga precedente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DELETE_DUPLICATE_ROWS"
    }
}
```

EXTRACT_ADVANCED_DATATYPE_DETAILS

Estrae i dettagli per il tipo di dati avanzato. I dettagli che è possibile estrarre dipendono dal modello rilevato, come descritto nelle Note di seguito.

Parameters

- `sourceColumn`— Il nome di una colonna sorgente di stringa.
- `pattern`— Lo schema da estrarre.
- `targetColumn`— Il nome della colonna di destinazione.
- `advancedDataType`— Può essere uno tra telefono, codice postale, data e ora, stato, carta di credito, URL, e-mail, SSN o sesso.

Note

- Se advanced DataType è Phone, il pattern può essere AREA_CODE, TIME_ZONE o COUNTRY_CODE.
- Se advanced DataType è Zip Code, il pattern può essere TIME_ZONE, COUNTRY, STATE, CITY, TYPE o REGION.
- Se advanced DataType è Date Time, il pattern può essere DAY, MONTH, MONTH_NAME, WEEK, QUARTER o YEAR.
- Se advanced DataType è State, il pattern può essere TIME_ZONE.
- Se advanced DataType è Credit Card, il pattern può essere LENGTH o NETWORK.
- Se advanced DataType è URL, il pattern può essere PROTOCOL, TLD o DOMAIN.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "EXTRACT_ADVANCED_DATATYPE_DETAILS",
        "Parameters": {
            "pattern": "TIMEZONE"
            "sourceColumn": "zipCode",
            "targetColumn": "timeZoneFromZipCode",
            "advancedDataType": "ZipCode"
```

```
        }
    }
}
```

FILL_WITH_AVERAGE

Restituisce una colonna con i dati mancanti sostituiti dalla media di tutti i valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_AVERAGE",
        "Parameters": {
            "sourceColumn": "age"
        }
    }
}
```

FILL_WITH_CUSTOM

Restituisce una colonna con dati mancanti sostituiti da un valore specifico.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati per la colonna. Questo tipo deve essere `datenumber`,`boolean`,`unsupported`,`string`, `otimestamp`.
- `value`— Il valore personalizzato da inserire. Il tipo di dati deve corrispondere al valore sceltocolumnDataType.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_CUSTOM",
        "Parameters": {
            "columnDataType": "string",
            "sourceColumn": "last_name",
            "value": "No last name provided"
        }
    }
}
```

RIEMPITO_CON_VUOTO

Restituisce una colonna con dati mancanti sostituita da una stringa vuota.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_EMPTY",
        "Parameters": {
            "sourceColumn": "wind_direction"
        }
    }
}
```

FILL_WITH_LAST_VALID

Restituisce una colonna con dati mancanti sostituiti dal valore valido più recente per quella colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati per la colonna. Questo tipo deve essere `datenumber`,`boolean`,`unsupported`,`string`, `otimestamp`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_LAST_VALID",
        "Parameters": {
            "columnDataType": "string",
            "sourceColumn": "birth_date"
        }
    }
}
```

FILL_WITH_MEDIAN

Restituisce una colonna con dati mancanti sostituita dalla mediana di tutti i valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_MEDIAN",
        "Parameters": {
            "sourceColumn": "age"
        }
    }
}
```

FILL_WITH_MODE

Restituisce una colonna con dati mancanti sostituita dalla modalità di tutti i valori.

È inoltre possibile specificare la logica di spareggio, in cui alcuni dei valori sono identici. Ad esempio, considerate i seguenti valori:

```
1 2 2 3 3 4
```

Una `modeType` delle `MINIMUM` cause `FILL_WITH_MODE` per cui viene restituito 2 come valore della modalità. In caso `modeType` `MAXIMUM` affermativo, la modalità è 3. Per `AVERAGE`, la modalità è 2,5.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `modeType`: come risolvere i valori pari nei dati. Questo valore deve essere `MINIMUMNONE`,`AVERAGE`, o`MAXIMUM`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_MODE",
        "Parameters": {
            "modeType": "MAXIMUM",
            "sourceColumn": "age"
        }
    }
}
```

RIEMPI_CON_MOST_FREQUENT

Restituisce una colonna con dati mancanti sostituiti dal valore più frequente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_MOST_FREQUENT",
        "Parameters": {
            "sourceColumn": "position"
        }
    }
```

```
}
```

FILL_WITH_NULL

Restituisce una colonna con i valori dei dati sostituiti da null.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_NULL",
        "Parameters": {
            "sourceColumn": "rating"
        }
    }
}
```

FILL_WITH_SUM

Restituisce una colonna con i dati mancanti sostituiti dalla somma di tutti i valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FILL_WITH_SUM",
        "Parameters": {
            "sourceColumn": "age"
        }
    }
```

```
}
```

FLAG_DUPLICATE_ROWS

Restituisce una nuova colonna con un valore specificato in ogni riga che indica se quella riga corrisponde esattamente a una riga precedente nel set di dati. Quando vengono trovate corrispondenze, i valori vengono contrassegnati come duplicati. L'occorrenza iniziale non viene contrassegnata poiché non corrisponde a una riga precedente.

Parameters

- `trueString`: valore da inserire se la riga corrisponde a una riga precedente.
- `falseString`: valore da inserire se la riga è univoca.
- `targetColumn`: nome della nuova colonna inserita nel set di dati.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FLAG_DUPLICATE_ROWS",
        "Parameters": {
            "trueString": "TRUE",
            "falseString": "FALSE",
            "targetColumn": "Flag"
        }
    }
}
```

FLAG_DUPLICATES_IN_COLUMN

Restituisce una nuova colonna con un valore specificato in ogni riga che indica se il valore nella colonna di origine della riga corrisponde a un valore in una riga precedente della colonna di origine. Quando vengono trovate corrispondenze, i valori vengono contrassegnati come duplicati. L'occorrenza iniziale non viene contrassegnata poiché non corrisponde a una riga precedente.

Parameters

- `sourceColumn`: nome della colonna di origine.

- `targetColumn`: nome della colonna di destinazione.
- `trueString`: stringa da inserire nella colonna di destinazione quando per un valore della colonna di origine è presente un duplicato di un valore precedente in tale colonna.
- `falseString`: stringa da inserire nella colonna di destinazione quando un valore della colonna di origine è diverso dai valori precedenti in tale colonna.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FLAG_DUPLICATES_IN_COLUMN",
        "Parameters": {
            "sourceColumn": "Name",
            "targetColumn": "Duplicate",
            "trueString": "TRUE",
            "falseString": "FALSE"
        }
    }
}
```

GET_ADVANCED_DATATYPE

Data una colonna di stringhe, identifica l'eventuale tipo di dati avanzato della colonna.

Parameters

- `columnName`— Il nome della colonna di stringhe.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "GET_ADVANCED_DATATYPE",
        "Parameters": {
            "sourceColumn": "columnName"
        }
    }
}
```

REMOVE_DUPLICATES

Elimina un'intera riga, se viene rilevato un valore duplicato in una colonna di origine selezionata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REMOVE_DUPLICATES",
        "Parameters": {
            "sourceColumn": "nationality"
        }
    }
}
```

REMOVE_INVALID

Elimina un'intera riga se viene rilevato un valore non valido in una colonna di quella riga.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna.
- `advancedDataType`— Tipi di dati speciali DataBrew rilevati da una colonna con il tipo di dati`string`. I tipi che DataBrew è possibile rilevare all'interno di una `string` colonna includono SSN, e-mail, numero di telefono, sesso, carta di credito, URL DateTime, indirizzo IP, valuta ZipCode, Paese, regione, stato e città.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REMOVE_INVALID",
```

```
        "Parameters": {
            "columnDataType": "string",
            "sourceColumn": "help_url"
        }
    }
}
```

REMOVE_MISSING

Restituisce solo le righe in cui non mancano dati in una colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REMOVE_MISSING",
        "Parameters": {
            "sourceColumn": "last_name"
        }
    }
}
```

REPLACE_WITH_AVERAGE

Sostituisce ogni valore non valido in una colonna con la media di tutti gli altri valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna. Questo tipo deve esser`number`.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "REPLACE_WITH_AVERAGE",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "age"
        }
    }
}
```

REPLACE_WITH_CUSTOM

Sostituisci le entità rilevate con un valore personalizzato.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `sourceColumns`— Un elenco di nomi di colonne esistenti.
- `columnDataType`— Il tipo di dati della colonna.
- `value`— Il valore personalizzato da utilizzare per sostituire i valori non validi.
- `advancedDataType`— Tipi di dati speciali rilevati da DataBrew una colonna contenente il tipo di `string` dati. I tipi che DataBrew è possibile rilevare all'interno di una `string` colonna includono SSN, e-mail, numero di telefono, sesso, carta di credito, URL DateTime, indirizzo IP, valuta ZipCode, Paese, regione, stato e città.

> **Note**
>
> Usa uno `sourceColumn` o entrambi`sourceColumns`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_CUSTOM",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "",
            "sourceColumns": ["column1", "column2"],
```

```
            "value": 0
        }
    }
}
```

REPLACE_WITH_EMPTY

Sostituisce ogni valore non valido in una colonna con un valore vuoto.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna.
- `advancedDataType`— Tipi di dati speciali DataBrew rilevati da una colonna con il tipo di dati`string`. I tipi che DataBrew è possibile rilevare all'interno di una `string` colonna includono SSN, e-mail, numero di telefono, sesso, carta di credito, URL DateTime, indirizzo IP, valuta ZipCode, Paese, regione, stato e città.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_EMPTY",
        "Parameters": {
            "columnDataType": "string",
            "sourceColumn": "nationality"
        }
    }
}
```

SOSTITUIRE_WITH_LAST_VALID

Sostituisce ogni valore non valido in una colonna con l'ultimo valore valido.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna.

- `advancedDataType`— Tipi di dati speciali DataBrew rilevati da una colonna con il tipo di dati`string`. I tipi che DataBrew è possibile rilevare all'interno di una `string` colonna includono SSN, e-mail, numero di telefono, sesso, carta di credito, URL DateTime, indirizzo IP, valuta ZipCode, Paese, regione, stato e città.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_LAST_VALID",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "rating"
        }
    }
}
```

SOSTITUIRE_WITH_MEDIAN

Sostituisce ogni valore non valido in una colonna con la mediana di tutti gli altri valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna. Questo tipo deve esserenumber.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_MEDIAN",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "games_won"
        }
    }
}
```

REPLACE_WITH_MODE

Sostituisce ogni valore non valido in una colonna con la modalità di tutti gli altri valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna. Questo tipo deve esserenumber.
- `modeType`: come risolvere i valori pari nei dati. Questo valore deve essere `MINIMUMNONE`,`AVERAGE`, o`MAXIMUM`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_MODE",
        "Parameters": {
            "columnDataType": "number",
            "modeType": "MAXIMUM",
            "sourceColumn": "height_cm"
        }
    }
}
```

SOSTITUIRE_WITH_MOST_FREQUENT

Sostituisce ogni valore non valido in una colonna con il valore di colonna più frequente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna.
- `advancedDataType`— Tipi di dati speciali DataBrew rilevati da una colonna con il tipo di dati`string`. I tipi che DataBrew è possibile rilevare all'interno di una `string` colonna includono SSN, e-mail, numero di telefono, sesso, carta di credito, URL DateTime, indirizzo IP, valuta ZipCode, Paese, regione, stato e città.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_MOST_FREQUENT",
        "Parameters": {
            "columnDataType": "string",
            "sourceColumn": "wind_direction"
        }
    }
}
```

SOSTITUIRE_WITH_NULL

Sostituisce ogni valore non valido in una colonna con un valore nullo.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna.
- `advancedDataType`— Tipi di dati speciali DataBrew rilevati da una colonna con il tipo di dati`string`. I tipi che DataBrew è possibile rilevare all'interno di una `string` colonna includono SSN, e-mail, numero di telefono, sesso, carta di credito, URL DateTime, indirizzo IP, valuta ZipCode, Paese, regione, stato e città.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_NULL",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "weight_kg"
        }
    }
}
```

SOSTITUIRE_WITH_ROLLING_AVERAGE

Sostituisce ogni valore in una colonna con la media mobile di una «finestra» di righe precedente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna. Questo tipo deve esserenumber.
- `period`- — La dimensione della finestra. Ad esempio, se `period` è 10, la media mobile viene calcolata utilizzando le 10 righe precedenti.

Example Esempio

```
{
    "RecipeStep": {
        "Action": {
            "Operation": "REPLACE_WITH_ROLLING_AVERAGE",
            "Parameters": {
                "sourceColumn": "created_at",
                "columnDataType": "number",
                "period": "2"
            }
        }
    }
}
```

SOSTITUIRE_WITH_ROLLING_SUM

Sostituisce ogni valore in una colonna con la somma progressiva di una «finestra» di righe precedente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna. Questo tipo deve esserenumber.
- `period`- — La dimensione della finestra. Ad esempio, se `period` è 10, la somma variabile viene calcolata utilizzando le 10 righe precedenti.

Example Esempio

```
{
```

```
    "RecipeStep": {
        "Action": {
            "Operation": "REPLACE_WITH_ROLLING_SUM",
            "Parameters": {
                "sourceColumn": "created_at",
                "columnDataType": "number",
                "period": "2"
            }
        }
    }
}
```

SOSTITUIRE_WITH_SUM

Sostituisce ogni valore non valido in una colonna con la somma di tutti gli altri valori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `columnDataType`— Il tipo di dati della colonna. Questo tipo deve esserenumber.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_SUM",
        "Parameters": {
            "columnDataType": "number",
            "sourceColumn": "games_won"
        }
    }
}
```

Fasi della ricetta relative alle informazioni di identificazione personale (PII)

Utilizza questi passaggi della ricetta per eseguire trasformazioni sulle informazioni di identificazione personale (PII) in un set di dati.

> **Note**
>
> Oltre ai passaggi illustrati in questa sezione, è possibile utilizzare per gestire DataBrew le PII anche istruzioni non progettate specificamente per le informazioni personali. Un esempio è DELETE una procedura di base relativa alla ricetta di una colonna che elimina una colonna.

Argomenti

HASH CRITTOGRAFICO

Applica un algoritmo ai valori hash nella colonna.

Parameters

- `sourceColumns`: una matrice di colonne esistenti.
- `secretId`: l'ARN della chiave segreta di Secrets Manager. La chiave utilizzata nell'algoritmo del prefisso HMAC (Hash-based Message Authentication Code) per eseguire l'hash delle colonne di origine, oppure `databrew!default` è l'output decodificato in base64 per il valore della chiave segreta Secrets Manager.
- `secretVersion` : Opzionale. Per impostazione predefinita, utilizza la versione più recente del segreto.

- `entityTypeFilter`— Matrice opzionale di tipi di entità. Può essere utilizzata per crittografare solo le informazioni di identificazione personale (PII) rilevate nella colonna a testo libero.
- `createSecretIfMissing`: booleano facoltativo. Se true, cercherà di creare il segreto per conto del chiamante.
- `algorithm`: l'algoritmo utilizzato per eseguire l'hash dei dati. Valori enum validi: MD5, SHA1, SHA256, SHA512, HMAC_MD5, HMAC_SHA1, HMAC_SHA256, HMAC_SHA512

  Ogni opzione si riferisce a un algoritmo di hashing diverso. Le opzioni con il prefisso «HMAC» si riferiscono a un algoritmo di hashing con chiave e richiedono il parametro. `secretId` Per le opzioni senza il prefisso «HMAC», il parametro non è obbligatorio. `secretId`

  Se non si fornisce un algoritmo hash, il servizio utilizza come impostazione predefinita «HMAC_SHA256".

```
{
   "sourceColumns": ["phonenumber"],
   "secretId": "arn:aws:secretsmanager:us-east-1:012345678901:secret:mysecret",
   "entityTypeFilter": ["USA_ALL"]
}
```

Quando lavora nell'esperienza interattiva, oltre al ruolo del progetto, l'utente della console deve avere l'autorizzazione per `secretsmanager:GetSecretValue` il segreto di Secrets Manager fornito.

Politica di esempio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:012345678901:secret:mysecret"
            ]
        }
```

```
    ]
}
```

Puoi anche scegliere di utilizzare il segreto DataBrew-created predefinito passando `databrew!default` come secretID e il parametro `createSecretIfMissing` come true. Questo non è consigliato per la produzione. Chiunque abbia il AwsGlueDataBrewFullAccessPolicyruolo può utilizzare il segreto predefinito.

DECIFRARE

È possibile utilizzare la trasformazione DECRYPT per decrittografare all'interno di. DataBrew I tuoi dati possono essere decrittografati anche all'esterno con Encryption SDK. DataBrew AWS Se l'ARN della chiave KMS fornito non corrisponde a quello utilizzato per crittografare la colonna, l'operazione di decrittografia non riesce. Per ulteriori informazioni sull'AWS Encryption SDK, consulta Cos'è l'AWS Encryption SDK nella Guida per gli sviluppatori.AWS Encryption SDK

Parameters

- `sourceColumns`: una matrice di colonne esistenti.
- `kmsKeyArn`— L'ARN della chiave del servizio di gestione delle AWS chiavi da utilizzare per decrittografare le colonne di origine. Per ulteriori informazioni sull'ARN chiave, consulta Key ARN nella Developer Guide.AWS Key Management Service

```
{
   "sourceColumns": ["phonenumber"],
   "kmsKeyArn": "arn:aws:kms:us-east-1:012345678901:key/<kms-key-id>"
}
```

Quando lavora nell'esperienza interattiva, oltre al ruolo del progetto, l'utente della console deve disporre dell'autorizzazione `kms:GenerateDataKey` e `kms:Decrypt` della chiave KMS fornita.

Politica di esempio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
```

```
    {
        "Effect": "Allow",
        "Action": [
            "kms:GenerateDataKey",
            "kms:Decrypt"
        ],
        "Resource": [
            "arn:aws:kms:us-east-1:012345678901:key/kms-key-id"
        ]
    }
  ]
}
```

DETERMINISTIC_DECRYPT

Decripta i dati crittografati con DETERMINISTIC_ENCRYPT.

Questa trasformazione è impossibile se l'ID e la versione segreti forniti non corrispondono a quelli utilizzati per crittografare la colonna.

Parameters

- `sourceColumns`: una matrice di colonne esistenti.
- `secretId`— L'ARN della chiave segreta di Secrets Manager da utilizzare per decrittografare le colonne di origine.
- `secretVersion` : Opzionale. Per impostazione predefinita, utilizza la versione più recente del segreto.

Esempio

```
{
   "sourceColumns": ["phonenumber"],
   "secretId": "arn:aws:secretsmanager:us-east-1:012345678901:secret:mysecret",
   "secretVersion": "adfe-1232-7563-3123"
}
```

Quando lavora nell'esperienza interattiva, oltre al ruolo del progetto, l'utente della console deve avere l'autorizzazione a secretsmanager: GetSecretValue sul segreto di Secrets Manager fornito.

Politica di esempio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:012345678901:secret:mysecret"
            ]
        }
    ]
}
```

DETERMINISTIC_ENCRYPT

Crittografa la colonna utilizzando una chiave a 256 AES-GCM-SIV bit. I dati crittografati con DETERMINISTIC_ENCRYPT possono essere decrittografati solo all'interno di con la trasformazione DETERMINISTIC_DECRYPT. DataBrew Questa trasformazione non utilizza l'Encryption SDK e utilizza invece la AWS libreria AWS KMS github LC.AWS

Può crittografare fino a 400 KB per cella. Non conserva il tipo di dati durante la decrittografia.

> **Note**
>
> Nota: l'uso di un segreto per più di un anno è sconsigliato.

Parameters

- `sourceColumns`: una matrice di colonne esistenti.
- `secretId`— L'ARN della chiave segreta di Secrets Manager da usare per crittografare le colonne di origine, o databrew! impostazione predefinita.
- `secretVersion` : Opzionale. Per impostazione predefinita, utilizza la versione più recente del segreto.

- `entityTypeFilter`— Matrice opzionale di tipi di entità. Può essere utilizzata per crittografare solo le informazioni di identificazione personale (PII) rilevate nella colonna a testo libero.
- `createSecretIfMissing`: booleano facoltativo. Se true, cercherà di creare il segreto per conto del chiamante.

Esempio

```
{
   "sourceColumns": ["phonenumber"],
   "secretId": "arn:aws:secretsmanager:us-east-1:012345678901:secret:mysecret",
   "secretVersion": "adfe-1232-7563-3123",
   "entityTypeFilter": ["USA_ALL"]
}
```

Quando lavora nell'esperienza interattiva, oltre al ruolo del progetto, l'utente della console deve avere l'autorizzazione per `secretsmanager:GetSecretValue` il segreto di Secrets Manager fornito.

Politica di esempio

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:012345678901:secret:mysecret"
            ]
        }
    ]
}
```

CIFRARE

Crittografa i valori nelle colonne di origine con AWS Encryption SDK. La trasformazione DECRYPT può essere utilizzata per decrittografare all'interno di. DataBrew Puoi anche decrittografare i dati all'esterno utilizzando Encryption SDK. DataBrew AWS

La trasformazione ENCRYPT può crittografare fino a 128 MiB per cella. Cercherà di mantenere il formato durante la decrittografia. Per mantenere il tipo di dati, i metadati correlati devono essere serializzati a meno di 1 KB. In caso contrario, è necessario impostare il parametro `preserveDataType` su false. I metadati del tipo di dati saranno archiviati in formato di testo semplice nel contesto della crittografia. Per ulteriori informazioni sul contesto di crittografia, consulta Encryption context nella Developer Guide.AWS Key Management Service

Parameters

- `sourceColumns`: una matrice di colonne esistenti.
- `kmsKeyArn`— La chiave ARN della chiave del servizio di gestione delle AWS chiavi da utilizzare per crittografare le colonne di origine. Per ulteriori informazioni sull'ARN chiave, consulta Key ARN nella Developer Guide.AWS Key Management Service
- `entityTypeFilter`— Serie opzionale di tipi di entità. Può essere utilizzata per crittografare solo le informazioni di identificazione personale (PII) rilevate nella colonna a testo libero.
- `preserveDataType`: booleano facoltativo. Il valore predefinito è true. Se false, il tipo di dati non sarà archiviato.

Nell'esempio seguente, `entityTypeFilter` e `preserveDataType` sono opzionali.

Esempio

```
{
    "sourceColumns": ["phonenumber"],
    "kmsKeyArn": "arn:aws:kms:us-east-1:012345678901:key/kms-key-id",
    "entityTypeFilter": ["USA_ALL"],
    "preserveDataType": "true"
}
```

Quando lavora nell'esperienza interattiva, oltre al ruolo del progetto, l'utente della console deve avere l'autorizzazione per `kms:GenerateDataKey` utilizzare la AWS KMS chiave fornita.

Politica di esempio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "kms:GenerateDataKey"
        ],
        "Resource": [
            "arn:aws:kms:us-east-1:012345678901:key/kms-key-id"
        ]
    }
  ]
}
```

MASK_CUSTOM

Maschera i caratteri che corrispondono a un valore personalizzato fornito.

Parameters

- `sourceColumns`— Un elenco di nomi di colonne esistenti.
- `maskSymbol`— Un simbolo che verrà utilizzato per sostituire i caratteri specificati.
- `regex`— Se vero, viene `customValue` considerato come un modello regex da abbinare.
- `customValue`— Tutte le occorrenze (o corrispondenze regex) di `customValue` verranno mascherate nella stringa.
- `entityTypeFilter`— Matrice opzionale di tipi di entità. Può essere utilizzata per crittografare solo le informazioni di identificazione personale (PII) rilevate nella colonna a testo libero.

Example Esempio

```
// Mask all occurrences of 'amazon' in the column
{
    "RecipeAction": {
        "Operation": "MASK_CUSTOM",
```

```
        "Parameters": {
            "sourceColumns": ["company"],
            "maskSymbol": "#",
            "customValue": "amazon"
        }
    }
}
```

MASK_DATE

Maschera i componenti di una data con un simbolo di maschera specificato dall'utente.

Parameters

- `sourceColumns`— Un elenco di nomi di colonne esistenti.
- `maskSymbol`— Un simbolo che verrà utilizzato per sostituire i caratteri specificati.
- `redact`— Una matrice di enumerazioni di componenti di data da mascherare. Valori enum validi: ANNO, MESE, GIORNO, ORA, MINUTO, SECONDO, MILLISECONDO.
- `locale`— Tag di lingua IETF BCP 47 opzionale. L'impostazione predefinita è `en`. Il locale da usare per la formattazione della data.

Example Esempio

```
// Mask year
{
    "RecipeAction": {
        "Operation": "MASK_DATE",
        "Parameters": {
            "sourceColumns": ["birthday"],
            "maskSymbol": "#",
            "redact": ["YEAR"]
        }
    }
}
```

MASK_DELIMITER

Maschera i caratteri tra due delimitatori con un simbolo di mascheramento specificato dall'utente.

Parameters

- `sourceColumns`— Un elenco di nomi di colonne esistenti.
- `maskSymbol`— Un simbolo che verrà utilizzato per sostituire i caratteri specificati.
- `startDelimiter`— Un carattere che indica da dove deve iniziare il mascheramento. L'omissione di questo parametro applicherà la maschera a partire dall'inizio della stringa.
- `endDelimiter`— Un carattere che indica dove deve finire il mascheramento. L'omissione di questo parametro applicherà il mascheramento da startDelimiter alla fine della stringa.
- `preserveDelimiters`— Se vero, applica la maschera ai delimitatori.
- `alphabet`— Una serie di set di caratteri da conservare durante il mascheramento. Valori enum validi: SYMBOLS, WHITESPACE.
- `entityTypeFilter`— Matrice opzionale di tipi di entità. Può essere utilizzata per crittografare solo le informazioni di identificazione personale (PII) rilevate nella colonna a testo libero.

Example Esempio

```
// Mask string between '<' and '>', ignoring white spaces, symbols, and lowercase
 letters
{
    "RecipeAction": {
        "Operation": "MASK_DELIMITER",
        "Parameters": {
            "sourceColumns": ["name"],
            "maskSymbol": "#",
            "startDelimiter": "<",
            "endDelimiter": ">",
            "preserveDelimiters": false,
            "alphabet": ["WHITESPACE", "SYMBOLS"]
        }
    }
}
```

MASK_RANGE

Maschera i caratteri tra due posizioni con un simbolo di mascheramento specificato dall'utente.

Parameters

- `sourceColumns`— Un elenco di nomi di colonne esistenti.
- `maskSymbol`— Un simbolo che verrà utilizzato per sostituire i caratteri specificati.
- `start`— Un numero che indica da quale posizione del carattere deve iniziare il mascheramento (indicizzato a 0, incluso). L'indicizzazione negativa è consentita. L'omissione di questo parametro applicherà la maschera dall'inizio della stringa fino a 'stop'.
- `stop`— Un numero che indica in quale posizione del carattere deve terminare il mascheramento (indicizzato 0, esclusivo). L'indicizzazione negativa è consentita. L'omissione di questo parametro applicherà la maschera dall'inizio alla fine della stringa.
- `alphabet`— Una serie di enumerazioni di set di caratteri da conservare durante il mascheramento. Valori enum validi: SYMBOLS, WHITESPACE.
- `entityTypeFilter`— Matrice opzionale di tipi di entità. Può essere utilizzata per crittografare solo le informazioni di identificazione personale (PII) rilevate nella colonna a testo libero.

Example Esempio

```
// Mask entire string
{
    "RecipeAction": {
        "Operation": "MASK_RANGE",
        "Parameters": {
            "sourceColumns": ["firstName", "lastName"],
            "maskSymbol": "#"
        }
    }
}
```

SOSTITUIRE_WITH_RANDOM_BETWEEN

Sostituisce i valori con un numero casuale.

Parameters

- `lowerBound`— Il limite inferiore dell'intervallo di numeri casuali.
- `sourceColumns`— Un elenco di nomi di colonne esistenti.

- `upperBound`— Il limite superiore dell'intervallo di numeri casuali.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_RANDOM_BETWEEN",
        "Parameters": {
            "lowerBound": "1",
            "sourceColumns": ["column1", "column2"],
            "upperBound": "100"
        }
    }
}
```

SOSTITUIRE_CON_RANDOM_DATE_BETWEEN

Sostituisce i valori con una data casuale.

Parameters

- `startDate`— L'inizio dell'intervallo di date a partire dal quale verrà presa una data casuale.
- `sourceColumns`— Un elenco di nomi di colonne esistenti.
- `endDate`— La fine dell'intervallo di date a partire dal quale verrà presa una data casuale.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "REPLACE_WITH_RANDOM_DATE_BETWEEN",
        "Parameters": {
            "startDate": "2020-12-12 12:12:12",
            "sourceColumns": ["column1", "column2"],
            "endDate": "2021-12-12 12:12:12"
        }
    }
}
```

SHUFFLE_ROWS

Mescola i valori in una determinata colonna. Il rimescolamento può avvenire con valori raggruppati in base a una colonna secondaria.

Parameters

- `sourceColumns`: una matrice di colonne esistenti.
- `groupByColumns`— Una matrice di colonne in base alla quale raggruppare le colonne di origine durante il mescolamento.

Example Esempio

```
{
   "sourceColumns": ["age"],
   "*groupByColumns*": ["country"]
}
```

Rilevamento e gestione dei valori anomali: fasi della ricetta

Utilizza questi passaggi della ricetta per lavorare con i valori anomali nei dati ed eseguire trasformazioni avanzate su di essi.

Argomenti

- FLAG_OUTLIERS
- REMOVE_OUTLIERS
- REPLACE_OUTLIERS
- RESCALE_OUTLIERS_CON_Z_SCORE
- RESCALE_OUTLIERS_CON_SKEW

FLAG_OUTLIERS

Restituisce una nuova colonna contenente un valore personalizzabile in ogni riga che indica se il valore della colonna di origine è un valore anomalo.

Parameters

- `sourceColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.
- `targetColumn`— specifica il nome di una nuova colonna in cui inserire i risultati della strategia di valutazione dei valori anomali.
- `outlierStrategy`— Speciifica l'approccio da utilizzare per il rilevamento dei valori anomali. I valori validi includono quanto segue:
  - `Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla media di oltre la soglia di deviazione standard.
  - `MODIFIED_Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla mediana di oltre la soglia di deviazione assoluta mediana.
  - `IQR`— Identifica un valore come valore anomalo quando supera il primo e l'ultimo quartile dei dati della colonna. L'intervallo interquartile (IQR) misura dove si trova il 50% medio dei punti dati.
- `threshold`— Speciifica il valore di soglia da utilizzare per il rilevamento dei valori anomali. Il `sourceColumn` valore viene identificato come valore anomalo se il punteggio calcolato con il supera questo numero. `outlierStrategy` L'impostazione predefinita è 3.
- `trueString`— specifica il valore della stringa da utilizzare se viene rilevato un valore anomalo. L'impostazione predefinita è «True».
- `falseString`— specifica il valore della stringa da utilizzare se non viene rilevato alcun valore anomalo. L'impostazione predefinita è «False».

Negli esempi seguenti viene visualizzata la sintassi per una singola RecipeActionoperazione. Una ricetta contiene almeno RecipeStepun'operazione e una fase della ricetta contiene almeno un'azione di ricetta. Un'azione di ricetta esegue la trasformazione dei dati specificata. Un gruppo di azioni di ricetta viene eseguito in ordine sequenziale per creare il set di dati finale.

JSON

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi JSON. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in JSON

```
{
```

```
    "Action": {
        "Operation": "FLAG_OUTLIERS",
        "Parameters": {
            "sourceColumn": "name-of-existing-column",
            "targetColumn": "name-of-new-column",
            "outlierStrategy": "IQR",
            "threshold": "1.5",
            "trueString": "Yes",
            "falseString": "No"
        }
    }
}
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo UpdateRecipe. Puoi utilizzare queste e altre operazioni API nel tuo codice.

YAML

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi YAML. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in YAML

```
- Action:
  Operation: FLAG_OUTLIERS
  Parameters:
    sourceColumn: name-of-existing-column
    targetColumn: name-of-new-column
    outlierStrategy: IQR
    trueString: Outlier
    falseString: No
    threshold: '1.5'
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo. UpdateRecipe Puoi utilizzare queste e altre operazioni API nel tuo codice.

REMOVE_OUTLIERS

Rimuove i punti dati classificati come valori anomali, in base alle impostazioni dei parametri.

Parameters

- `sourceColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.
- `outlierStrategy`— Speciifica l'approccio da utilizzare per il rilevamento dei valori anomali. I valori validi includono quanto segue:
  - `Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla media di oltre la soglia di deviazione standard.
  - `MODIFIED_Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla mediana di oltre la soglia di deviazione assoluta mediana.
  - `IQR`— Identifica un valore come valore anomalo quando supera il primo e l'ultimo quartile dei dati della colonna. L'intervallo interquartile (IQR) misura dove si trova il 50% medio dei punti dati.
- `threshold`— Speciifica il valore di soglia da utilizzare per il rilevamento dei valori anomali. Il `sourceColumn` valore viene identificato come valore anomalo se il punteggio calcolato con il supera questo numero. `outlierStrategy` L'impostazione predefinita è 3.
- `removeType`— Speciifica il modo di rimuovere i dati. I valori validi includono `DELETE_ROWS` e `CLEAR`.
- `trimValue`— Speciifica se rimuovere tutti o alcuni dei valori anomali. Il valore booleano predefinito è. `FALSE`
  - `FALSE`— Rimuove tutti i valori anomali
  - `TRUE`— Rimuove i valori anomali che si collocano al di fuori della soglia percentile specificata in and. `minValue` `maxValue`
- `minValue`— Indica il valore percentile minimo per l'intervallo dei valori anomali. L'intervallo valido è compreso tra 0 e 100.
- `maxValue`— Indica il valore percentile massimo per l'intervallo dei valori anomali. L'intervallo valido è compreso tra 0 e 100.

Negli esempi seguenti viene visualizzata la sintassi per una singola operazione. RecipeAction Una ricetta contiene almeno RecipeStepun'operazione e una fase della ricetta contiene almeno un'azione di ricetta. Un'azione di ricetta esegue la trasformazione dei dati specificata. Un gruppo di azioni di ricetta viene eseguito in ordine sequenziale per creare il set di dati finale.

JSON

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi JSON. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in JSON

```
{
    "Action": {
        "Operation": "REMOVE_OUTLIERS",
        "Parameters": {
            "sourceColumn": "name-of-existing-column",
            "outlierStrategy": "Z_SCORE",
            "threshold": "3",
            "removeType": "DELETE_ROWS",
            "trimValue": "TRUE",
            "minValue": "5",
            "maxValue": "95"
        }
    }
}
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo UpdateRecipe. Puoi utilizzare queste e altre operazioni API nel tuo codice.

YAML

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi YAML. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in YAML

```
- Action:
  Operation: REMOVE_OUTLIERS
  Parameters:
    sourceColumn: name-of-existing-column
    outlierStrategy: Z_SCORE
    threshold: '3'
    removeType: DELETE_ROWS
    trimValue: 'TRUE'
```

```
    minValue: '5'
    maxValue: '95'
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo. UpdateRecipe Puoi utilizzare queste e altre operazioni API nel tuo codice.

REPLACE_OUTLIERS

Aggiorna i valori dei punti dati che vengono classificati come valori anomali, in base alle impostazioni dei parametri.

Parameters

- `sourceColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.
- `outlierStrategy`— Speciifica l'approccio da utilizzare per il rilevamento dei valori anomali. I valori validi includono quanto segue:
  - `Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla media di oltre la soglia di deviazione standard.
  - `MODIFIED_Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla mediana di oltre la soglia di deviazione assoluta mediana.
  - `IQR`— Identifica un valore come valore anomalo quando supera il primo e l'ultimo quartile dei dati della colonna. L'intervallo interquartile (IQR) misura dove si trova il 50% medio dei punti dati.
- `threshold`— Speciifica il valore di soglia da utilizzare per il rilevamento dei valori anomali. Il `sourceColumn` valore viene identificato come valore anomalo se il punteggio calcolato con il supera questo numero. `outlierStrategy` L'impostazione predefinita è 3.
- `replaceType`— specifica il metodo da utilizzare per la sostituzione dei valori anomali. I valori validi includono quanto segue:
  - `WINSORIZE_VALUES`— specifica l'utilizzo del percentile minimo e massimo per limitare i valori.
  - `REPLACE_WITH_CUSTOM`
  - `REPLACE_WITH_EMPTY`
  - `REPLACE_WITH_NULL`
  - `REPLACE_WITH_MODE`
  - `REPLACE_WITH_AVERAGE`
  - `REPLACE_WITH_MEDIAN`

  - REPLACE_WITH_SUM
  - REPLACE_WITH_MAX
- modeType— Indica il tipo di funzione modale da utilizzare quando è. replaceType REPLACE_WITH_MODE I valori validi includono i seguenti: MINMAX, eAVERAGE.
- minValue— Indica il valore percentile minimo per l'intervallo di valori anomali da applicare quando trimValue viene utilizzato. L'intervallo valido è compreso tra 0 e 100.
- maxValue— Indica il valore percentile massimo per l'intervallo di valori anomali da applicare quando viene utilizzato. trimValue L'intervallo valido è compreso tra 0 e 100.
- value— Speciifica il valore da inserire durante l'utilizzo. REPLACE_WITH_CUSTOM
- trimValue— Speciifica se rimuovere tutti o alcuni dei valori anomali. Questo valore booleano è impostato su TRUE quando replaceType è, o. REPLACE_WITH_NULL REPLACE_WITH_MODE WINSORIZE_VALUES L'impostazione predefinita è per FALSE tutti gli altri.
  - FALSE— Rimuove tutti i valori anomali
  - TRUE— Rimuove i valori anomali che si collocano al di fuori della soglia massima percentile specificata in and. minValue maxValue

Negli esempi seguenti viene visualizzata la sintassi per una singola operazione. RecipeAction Una ricetta contiene almeno RecipeStepun'operazione e una fase della ricetta contiene almeno un'azione di ricetta. Un'azione di ricetta esegue la trasformazione dei dati specificata. Un gruppo di azioni di ricetta viene eseguito in ordine sequenziale per creare il set di dati finale.

JSON

Di seguito viene mostrato un esempio RecipeAction da utilizzare come membro di un esempio RecipeStep per una DataBrew ricetta, utilizzando la sintassi JSON. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in JSON

```
{
    "Action": {
        "Operation": "REPLACE_OUTLIERS",
        "Parameters": {
            "maxValue": "95",
            "minValue": "5",
            "modeType": "AVERAGE",
```

```
            "outlierStrategy": "Z_SCORE",
            "replaceType": "REPLACE_WITH_MODE",
            "sourceColumn": "name-of-existing-column",
            "threshold": "3",
            "trimValue": "TRUE"
        }
    }
}
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo UpdateRecipe. Puoi utilizzare queste e altre operazioni API nel tuo codice.

YAML

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi YAML. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in YAML

```
- Action:
  Operation: REMOVE_OUTLIERS
  Parameters:
    sourceColumn: name-of-existing-column
    outlierStrategy: Z_SCORE
    threshold: '3'
    replaceType: REPLACE_WITH_MODE
    modeType: AVERAGE
    minValue: '5'
    maxValue: '95'
    trimValue: 'TRUE'
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo. UpdateRecipe Puoi utilizzare queste e altre operazioni API nel tuo codice.

RESCALE_OUTLIERS_CON_Z_SCORE

Restituisce una nuova colonna con un valore anomalo ridimensionato in ogni riga, in base alle impostazioni dei parametri. Questa azione applica anche Z-score la normalizzazione ai valori dei dati in scala lineare in modo che abbiano una media (μ) di 0 e una deviazione standard (μ) di 1. Consigliamo questa azione per gestire i valori anomali.

Parameters

- `sourceColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.
- `targetColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.
- `outlierStrategy`— Speciifica l'approccio da utilizzare per il rilevamento dei valori anomali. I valori validi includono quanto segue:
  - `Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla media di oltre la soglia di deviazione standard.
  - `MODIFIED_Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla mediana di oltre la soglia di deviazione assoluta mediana.
  - `IQR`— Identifica un valore come valore anomalo quando supera il primo e l'ultimo quartile dei dati della colonna. L'intervallo interquartile (IQR) misura dove si trova il 50% medio dei punti dati.
- `threshold`— Il valore di soglia da utilizzare per rilevare i valori anomali. Il `sourceColumn` valore viene identificato come valore anomalo se il punteggio calcolato con il supera questo numero. `outlierStrategy` L'impostazione predefinita è 3.

Negli esempi seguenti viene visualizzata la sintassi per una singola operazione. RecipeAction Una ricetta contiene almeno RecipeStepun'operazione e una fase della ricetta contiene almeno un'azione di ricetta. Un'azione di ricetta esegue la trasformazione dei dati specificata. Un gruppo di azioni di ricetta viene eseguito in ordine sequenziale per creare il set di dati finale.

JSON

Di seguito viene illustrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per un'operazione DataBrew Recipe, utilizzando la sintassi JSON. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in JSON

```
{
    "Action": {
        "Operation": "RESCALE_OUTLIERS_WITH_Z_SCORE",
        "Parameters": {
            "sourceColumn": "name-of-existing-column",
```

```
            "targetColumn": "name-of-new-column",
            "outlierStrategy": "Z_SCORE",
            "threshold": "3"
        }
    }
}
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo UpdateRecipe. Puoi utilizzare queste e altre operazioni API nel tuo codice.

YAML

Di seguito viene illustrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per un'operazione DataBrew Recipe, utilizzando la sintassi YAML. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in YAML

```
- Action:
  Operation: REMOVE_OUTLIERS
  Parameters:
    sourceColumn: name-of-existing-column
    targetColumn: name-of-new-column
    outlierStrategy: Z_SCORE
    threshold: '3'
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo. UpdateRecipe Puoi utilizzare queste e altre operazioni API nel tuo codice.

RESCALE_OUTLIERS_CON_SKEW

Restituisce una nuova colonna con un valore anomalo ridimensionato in ogni riga, in base alle impostazioni dei parametri. Questa azione serve a ridurre l'asimmetria della distribuzione applicando la trasformazione di log o root specificata. Consigliamo questa azione per la gestione di dati distorti.

Parameters

- `sourceColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.

- `targetColumn`— specifica il nome di una colonna numerica esistente che potrebbe contenere valori anomali.
- `outlierStrategy`— Speciifica l'approccio da utilizzare per il rilevamento dei valori anomali. I valori validi includono quanto segue:
  - `Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla media di oltre la soglia di deviazione standard.
  - `MODIFIED_Z_SCORE`— Identifica un valore come valore anomalo quando si discosta dalla mediana di oltre la soglia di deviazione assoluta mediana.
  - `IQR`— Identifica un valore come valore anomalo quando supera il primo e l'ultimo quartile dei dati della colonna. L'intervallo interquartile (IQR) misura dove si trova il 50% medio dei punti dati.
- `threshold`— Speciifica il valore di soglia da utilizzare per il rilevamento dei valori anomali. Il `sourceColumn` valore viene identificato come valore anomalo se il punteggio calcolato con il supera questo numero. `outlierStrategy` L'impostazione predefinita è 3.
- `skewFunction`— specifica il metodo da utilizzare per la sostituzione dei valori anomali. I valori validi includono quanto segue:
  - LOG — Applica una forte trasformazione per ridurre l'inclinazione positiva e negativa. Si tratta di un logaritmo naturale (2,718281828).
  - ROOT (with`value = 3`) — Applica una trasformazione abbastanza forte per ridurre l'inclinazione positiva e negativa. (Radice cubica)
  - ROOT (con`value = 2`) — Applica una trasformazione moderata per ridurre solo l'inclinazione positiva. (Radice quadrata)
  - SQUARE: applica una trasformazione moderata per ridurre l'inclinazione negativa. (Quadrato)
  - Trasformazione personalizzata: applica la `ROOT` trasformazione specificata `LOG` o utilizza il numero personalizzato fornito nel `value` parametro.
- `value`— specifica il valore da utilizzare per la trasformazione personalizzata. Se `skewFunction` è LOG, questo valore rappresenta la base del registro. Se `skewFunction` è ROOT, questo valore rappresenta la potenza della radice.

Gli esempi seguenti mostrano la sintassi per una singola RecipeActionoperazione. Una ricetta contiene almeno RecipeStepun'operazione e una fase della ricetta contiene almeno un'azione di ricetta. Un'azione di ricetta esegue la trasformazione dei dati specificata. Un gruppo di azioni di ricetta viene eseguito in ordine sequenziale per creare il set di dati finale.

JSON

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi JSON. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in JSON

```
{
    "Action": {
        "Operation": "RESCALE_OUTLIERS_WITH_SKEW",
        "Parameters": {
            "outlierStrategy": "Z_SCORE",
            "threshold": "3",
            "skewFunction": "ROOT",
            "sourceColumn": "name-of-existing-column",
            "targetColumn": "name-of-new-column",
            "value": "4"
        }
    }
}
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo UpdateRecipe. Puoi utilizzare queste e altre operazioni API nel tuo codice.

YAML

Di seguito viene mostrato un esempio `RecipeAction` da utilizzare come membro di un esempio `RecipeStep` per una DataBrew ricetta, utilizzando la sintassi YAML. Per esempi di sintassi che mostrano un elenco di azioni relative alle ricette, vedere. Definizione della struttura di una ricetta

Example Esempio in YAML

```
- Action:
  Operation: RESCALE_OUTLIERS_WITH_SKEW
  Parameters:
    outlierStrategy: Z_SCORE
    threshold: '3'
    skewFunction: ROOT
    sourceColumn: name-of-existing-column
    targetColumn: name-of-new-column
    value: '4'
```

Per ulteriori informazioni sull'utilizzo di questa azione di ricetta in un'operazione API, consulta CreateRecipeo. UpdateRecipe Puoi utilizzare queste e altre operazioni API nel tuo codice.

Fasi della ricetta per la struttura a colonne

Usa questi passaggi della ricetta per modificare la struttura delle colonne dei tuoi dati.

Argomenti

OPERAZIONE_BOOLEANA

Crea una nuova colonna, in base al risultato della condizione logica IF. Restituisce il valore vero se l'espressione booleana è vera, il valore falso se l'espressione booleana è falsa o restituisce un valore personalizzato.

Parameters

- `trueValueExpression`— Risultato quando viene soddisfatta la condizione.
- `falseValueExpression`— Risultato quando la condizione non è soddisfatta.
- `valueExpression`— Condizione booleana.
- `withExpressions`— Configurazione per risultati aggregati.

- `targetColumn`: un nome per la nuova colonna creata.

È possibile utilizzare valori costanti, riferimenti di colonna e risultati aggregati in trueValueExpression, false ValueExpression e ValueExpression.

Example Esempio: valori costanti

Valori che rimangono invariati, come un numero o una frase.

```
{
  "RecipeStep": {
    "Action": {
      "Operation": "BOOLEAN_OPERATION",
      "Parameters": {
        "trueValueExpression": "It is true.",
        "falseValueExpression": "It is false.",
        "valueExpression": "`column.1` < 2000",
        "targetColumn": "result.column"
      }
    }
  }
}
```

Example Esempio: riferimenti alle colonne

Valori che sono colonne nel set di dati.

```
{
  "RecipeStep": {
    "Action": {
      "Operation": "BOOLEAN_OPERATION",
      "Parameters": {
        "trueValueExpression": "`column.2`",
        "falseValueExpression": "`column.3`",
        "valueExpression": "`column.1` < `column.4`",
        "targetColumn": "result.column"
      }
    }
  }
```

```
}
```

Example Esempio: risultati aggregati

Valori calcolati da funzioni aggregate. Una funzione di aggregazione esegue un calcolo su una colonna e restituisce un singolo valore.

```
{
  "RecipeStep": {
    "Action": {
      "Operation": "BOOLEAN_OPERATION",
      "Parameters": {
        "trueValueExpression": "`:mincolumn.2`",
        "falseValueExpression": "`:maxcolumn.3`",
        "valueExpression": "`column.1` < `:avgcolumn.4`",
        "withExpressions": "[{\"name\":\"mincolumn.2\",\"value\":\"min(`column.2`)\",
\"type\":\"aggregate\"},{\"name\":\"maxcolumn.3\",\"value\":\"max(`column.3`)\",\"type
\":\"aggregate\"},{\"name\":\"avgcolumn.4\",\"value\":\"avg(`column.4`)\",\"type\":
\"aggregate\"}]",
        "targetColumn": "result.column"
      }
    }
  }
}
```

Gli utenti devono convertire il codice JSON in una stringa tramite escape.

Nota che i nomi dei parametri in true ValueExpressionValueExpression, false e valueExpression devono corrispondere ai nomi in withExpressions. Per utilizzare i risultati aggregati di alcune colonne, è necessario creare i relativi parametri e fornire le funzioni di aggregazione.

Example Esempio:

```
{
  "RecipeStep": {
    "Action": {
      "Operation": "BOOLEAN_OPERATION",
      "Parameters": {
        "trueValueExpression": "It is true.",
        "falseValueExpression": "It is false.",
        "valueExpression": "`column.1` < 2000",
```

```
        "targetColumn": "result.column"
      }
    }
  }
}
```

Example Esempio: and/or

È possibile utilizzare and e/o per combinare più condizioni.

```
{
  "RecipeStep": {
    "Action": {
      "Operation": "BOOLEAN_OPERATION",
      "Parameters": {
        "trueValueExpression": "It is true.",
        "falseValueExpression": "It is false.",
        "valueExpression": "`column.1` < 2000 and `column.2` >= `column.3",
        "targetColumn": "result.column"
      }
    }
  }
}
{
  "RecipeStep": {
    "Action": {
      "Operation": "BOOLEAN_OPERATION",
      "Parameters": {
        "trueValueExpression": "`column.4`",
        "falseValueExpression": "`column.5`",
        "valueExpression": "startsWith(`column1`, 'value1') or endsWith(`column2`,
 'value2')",
        "targetColumn": "result.column"
      }
    }
  }
}
```

Funzioni aggregate valide

La tabella seguente mostra tutte le funzioni aggregate valide che possono essere utilizzate in un'operazione booleana.

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
Numerico	Somma	`:sum.column.1`	`[ { "name": "sum.column.1", "value": "sum(`column.1`)", "type": "aggregate" } ]`	Restituisce la somma di `column.1`
	Media	`:mean.column.1`	`[ { "name": "mean.column.1", "value": "avg(`column.1`)", "type": "aggregate" } ]`	Restituisce la media di `column.1`
	Deviazione media assoluta	`:deviazione assoluta media.column.1`	`[ { "name":`	Restituisce la deviazione

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito

			<pre>"meanabsolutedeviation.column.1",
			
			"value": "mean_absolute_deviation(`column.1`)",
			
			"type": "aggregate"
			}
			]</pre>
				media assoluta di `column.1`

	Mediana	`:median.column.1`	<pre>[
			{
			
			"name": "median.column.1",
			
			"value": "median(`column.1`)",
			
			"type": "aggregate"
			}
			]</pre>
				Restituisce la mediana di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Prodotto	`:product .column.1`	`[ { "name": "product. column.1", "value": "product( `column.1 `)", "type": "aggregat e" } ]`	Restituisce il prodotto di `column.1`
	Deviazione standard	`:standar ddeviatio n.column.1`	`[ { "name": "standard deviation .column.1 ", "value": "stddev(` column.1` )", "type": "aggregat e" } ]`	Restituisce la deviazion e standard di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Varianza	`:varianc e.column.1`	`[ { "name": "variance .column.1 ", "value": "variance (`column. 1`)", "type": "aggregat e" } ]`	Restituisce la varianza di `column.1`
	Errore standard di media	`:standar derrorofm ean.column.1`	`[ { "name": "standard errorofme an.column .1", "value": "standard _error_of _mean(`co lumn.1`)", "type": "aggregat e" } ]`	Restituisce l'errore standard della media di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Asimmetria	`:skewness.column.1`	`[ { "name": "skewness.column.1", "value": "skewness(`column.1`)", "type": "aggregate" } ]`	Restituisce l'asimmetria di `column.1`
	Curtosi	`:curtosis.column.1`	`[ { "name": "kurtosis.column.1", "value": "kurtosis(`column.1`)", "type": "aggregate" } ]`	Restituisce la curtosi di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
Datetime/ Numeric/Text	Conteggio	`:count.column.1`	`[ { "name": "count.column.1", "value": "count(`column.1`)", "type": "aggregate" } ]`	Restituisce il numero totale di righe in `column.1`
	Conteggio distinto	`:countdistinct.column.1`	`[ { "name": "count.column.1", "value": "count(distinct `column.1`)", "type": "aggregate" } ]`	Restituisce il numero totale di righe distinte in `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Min	`:min.column.1`	`[ { "name": "min.column.1", "value": "min(`column.1`)", "type": "aggregate" } ]`	Restituisce il valore minimo di `column.1`
	Max	`:max.column.1`	`[ { "name": "max.column.1", "value": "max(`column.1`)", "type": "aggregate" } ]`	Restituisce il valore massimo di `column.1`

Condizioni valide in una ValueExpression

La tabella seguente mostra le condizioni supportate e le espressioni di valore che è possibile utilizzare.

Tipo di colonna	Condizione	ValueExpression	Description
Stringa	Contiene	contiene (`colonna`, 'testo')	Condizione per verificare se il valore nella colonna contiene testo
	Non contiene	! contiene (`colonna`, 'testo')	Condizione per verificare se il valore nella colonna non contiene testo
	Corrispondenze	match (`column`, 'pattern')	Condizione per verificare se il valore nella colonna corrisponde al modello
	Non corrisponde	! corrispondenze (`colonna`, 'modello')	Condizione per verificare se il valore nella colonna non corrisponde al modello
	Inizia con	startsWith (`column`, 'text')	Condizione per verificare se il valore nella colonna inizia con testo
	Non inizia con	! startsWith (`column`, 'text')	Condizione per verificare se il valore nella colonna non inizia con il testo
	Ends with	EndsWith (`column`, 'text')	Condizione per verificare se il valore nella colonna termina con testo

Tipo di colonna	Condizione	ValueExpression	Description
	Non termina con	! EndsWith (`colonna`, 'testo')	Condizione per verificare se il valore nella colonna non termina con il testo
Numerico	Less than	`colonna` < numero	Condizione per verificare se il valore nella colonna è inferiore al numero
	Minore o uguale a	`colonna` <= numero	Condizione per verificare se il valore nella colonna è minore o uguale al numero
	Greater than	`colonna` > numero	Condizione per verificare se il valore nella colonna è maggiore del numero
	Maggiore o uguale a	`colonna` >= numero	Condizione per verificare se il valore nella colonna è maggiore o uguale al numero
	È compreso tra	isBetween (`column` , minNumber, maxNumber)	Condizione per verificare se il valore nella colonna è compreso tra minNumber e maxNumber

Tipo di colonna	Condizione	ValueExpression	Description
	Non è compreso tra	! isBetween (`column` , minNumber, maxNumber)	Condizione per verificare se il valore nella colonna non è compreso tra minNumber e maxNumber
Booleano	È vero	`column` = VERO	Condizione per verificare se il valore nella colonna è booleano TRUE
	È falso	`column` = FALSE	Condizione per verificare se il valore nella colonna è booleano FALSE
Date/Timestamp	Prima di	`colonna` < 'data'	Condizione per verificare se il valore nella colonna è precedente alla data
	Anteriore o uguale a	`colonna` <= 'data'	Condizione per verificare se il valore nella colonna è precedente o uguale alla data
	Più tardi di	`colonna` > 'data'	Condizione per verificare se il valore nella colonna è successivo alla data

Tipo di colonna	Condizione	ValueExpression	Description
	Più tardi o uguale a	`colonna` >= 'data'	Condizione per verificare se il valore nella colonna è successivo o uguale alla data
String/Numeric/Date/Timestamp	È esattamente	`column` = 'valore'	Condizione per verificare se il valore nella colonna è esattamente il valore
	Is not (Non è)	`colonna`! = 'valore'	Condizione per verificare se il valore nella colonna non è un valore
	Manca	Manca (`colonna`)	Condizione per verificare se manca il valore nella colonna
	Non manca	! manca (`colonna`)	Condizione per verificare se il valore nella colonna non manca
	È valido	isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna è valido (il valore è di tipo dati o può essere convertito in tipo di dati)

Tipo di colonna	Condizione	ValueExpression	Description
	Non è valido	! isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna non è valido (il valore è di tipo dati o può essere convertito in tipo di dati)
Annidato	Manca	Manca (`colonna`)	Condizione per verificare se manca il valore nella colonna
	Non manca	! manca (`colonna`)	Condizione per verificare se il valore nella colonna non manca
	È valido	isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna è valido (il valore è di tipo dati o può essere convertit o in tipo di dati)
	Non è valido	! isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna non è valido (il valore è di tipo dati o può essere convertito in tipo di dati)

CASE_OPERATION

Crea una nuova colonna, in base al risultato della condizione logica CASE. L'operazione case esamina le condizioni del caso e restituisce un valore quando viene soddisfatta la prima condizione. Quando una condizione è vera, l'operazione interrompe la lettura e restituisce il risultato. Se nessuna condizione è vera, restituisce il valore predefinito.

Parameters

- `valueExpression`— Condizioni.
- `withExpressions`— Configurazione per risultati aggregati.
- `targetColumn`— Nome della colonna appena creata.

Example Esempio

```
{
  "RecipeStep": {
    "Action": {
      "Operation": "CASE_OPERATION",
      "Parameters": {
        "valueExpression": "case when `column11` < `column.2` then 'result1' when
 `column2` < 'value2' then 'result2' else 'high' end",
        "targetColumn": "result.column"
      }
    }
  }
}
```

Funzioni aggregate valide

La tabella seguente mostra tutte le funzioni aggregate valide che possono essere utilizzate in un'operazione di case.

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
Numerico	Somma	`:sum.column.1`	`[ { `	Restituisce la somma di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito

			<pre>"name": "sum.column.1",
			"value": "sum(`column.1`)",
			"type": "aggregate"
			}
			]</pre>	

	Media	`:mean.column.1`	<pre>[
			{
			"name": "mean.column.1",
			"value": "avg(`column.1`)",
			"type": "aggregate"
			}
			]</pre>
				Restituisce la media di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Deviazione media assoluta	`:deviazione assoluta media.column.1`	`[ { "name": "meanabsolutedeviation.column.1", "value": "mean_absolute_deviation(`column.1`)", "type": "aggregate" } ]`	Restituisce la deviazione media assoluta di `column.1`
	Mediana	`:median.column.1`	`[ { "name": "median.column.1", "value": "median(`column.1`)", "type": "aggregate" } ]`	Restituisce la mediana di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Prodotto	`:product .column.1`	`[ { "name": "product. column.1", "value": "product( `column.1 `)", "type": "aggregat e" } ]`	Restituisce il prodotto di `column.1`
	Deviazione standard	`:standar ddeviatio n.column.1`	`[ { "name": "standard deviation .column.1 ", "value": "stddev(` column.1` )", "type": "aggregat e" } ]`	Restituisce la deviazion e standard di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Varianza	`:varianc e.column.1`	`[ { "name": "variance .column.1 ", "value": "variance (`column. 1`)", "type": "aggregat e" } ]`	Restituisce la varianza di `column.1`
	Errore standard di media	`:standar derrorofm ean.column.1`	`[ { "name": "standard errorofme an.column .1", "value": "standard _error_of _mean(`co lumn.1`)", "type": "aggregat e" } ]`	Restituisce l'errore standard della media di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Asimmetria	`:skewness.column.1`	`[ { "name": "skewness.column.1", "value": "skewness(`column.1`)", "type": "aggregate" } ]`	Restituisce l'asimmetria di `column.1`
	Curtosi	`:curtosis.column.1`	`[ { "name": "kurtosis.column.1", "value": "kurtosis(`column.1`)", "type": "aggregate" } ]`	Restituisce la curtosi di `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
Datetime/ Numeric/Text	Conteggio	`:count.column.1`	`[ { "name": "count.column.1", "value": "count(`column.1`)", "type": "aggregate" } ]`	Restituisce il numero totale di righe in `column.1`
	Conteggio distinto	`:countdistinct.column.1`	`[ { "name": "count.column.1", "value": "count(distinct `column.1`)", "type": "aggregate" } ]`	Restituisce il numero totale di righe distinte in `column.1`

Tipo di colonna	Condizione	ValueExpression	Con Expressions	Valore restituito
	Min	`:min.column.1`	`[ { "name": "min.column.1", "value": "min(`column.1`)", "type": "aggregate" } ]`	Restituisce il valore minimo di `column.1`
	Max	`:max.column.1`	`[ { "name": "max.column.1", "value": "max(`column.1`)", "type": "aggregate" } ]`	Restituisce il valore massimo di `column.1`

Condizioni valide in una ValueExpression

La tabella seguente mostra le condizioni supportate e le espressioni di valore che è possibile utilizzare.

Tipo di colonna	Condizione	ValueExpression	Description
Stringa	Contiene	contiene (`colonna`, 'testo')	Condizione per verificare se il valore nella colonna contiene testo
	Non contiene	! contiene (`colonna`, 'testo')	Condizione per verificare se il valore nella colonna non contiene testo
	Corrispondenze	match (`column`, 'pattern')	Condizione per verificare se il valore nella colonna corrisponde al modello
	Non corrisponde	! corrispondenze (`colonna`, 'modello')	Condizione per verificare se il valore nella colonna non corrisponde al modello
	Inizia con	startsWith (`column`, 'text')	Condizione per verificare se il valore nella colonna inizia con testo
	Non inizia con	! startsWith (`column`, 'text')	Condizione per verificare se il valore nella colonna non inizia con il testo
	Ends with	EndsWith (`column`, 'text')	Condizione per verificare se il valore nella colonna termina con testo

Tipo di colonna	Condizione	ValueExpression	Description
	Non termina con	! EndsWith (`colonna`, 'testo')	Condizione per verificare se il valore nella colonna non termina con il testo
Numerico	Less than	`colonna` < numero	Condizione per verificare se il valore nella colonna è inferiore al numero
	Minore o uguale a	`colonna` <= numero	Condizione per verificare se il valore nella colonna è minore o uguale al numero
	Greater than	`colonna` > numero	Condizione per verificare se il valore nella colonna è maggiore del numero
	Maggiore o uguale a	`colonna` >= numero	Condizione per verificare se il valore nella colonna è maggiore o uguale al numero
	È compreso tra	isBetween (`column` , minNumber, maxNumber)	Condizione per verificare se il valore nella colonna è compreso tra minNumber e maxNumber

Tipo di colonna	Condizione	ValueExpression	Description
	Non è compreso tra	! isBetween (`column` , minNumber, maxNumber)	Condizione per verificare se il valore nella colonna non è compreso tra minNumber e maxNumber
Booleano	È vero	`column` = VERO	Condizione per verificare se il valore nella colonna è booleano TRUE
	È falso	`column` = FALSE	Condizione per verificare se il valore nella colonna è booleano FALSE
Date/Timestamp	Prima di	`colonna` < 'data'	Condizione per verificare se il valore nella colonna è precedente alla data
	Anteriore o uguale a	`colonna` <= 'data'	Condizione per verificare se il valore nella colonna è precedente o uguale alla data
	Più tardi di	`colonna` > 'data'	Condizione per verificare se il valore nella colonna è successivo alla data

Tipo di colonna	Condizione	ValueExpression	Description
	Successivo o uguale a	`colonna` >= 'data'	Condizione per verificare se il valore nella colonna è successivo o uguale alla data
String/Numeric/Date/Timestamp	È esattamente	`column` = 'valore'	Condizione per verificare se il valore nella colonna è esattamente il valore
	Is not (Non è)	`colonna`! = 'valore'	Condizione per verificare se il valore nella colonna non è un valore
	Manca	Manca (`colonna`)	Condizione per verificare se manca il valore nella colonna
	Non manca	! manca (`colonna`)	Condizione per verificare se il valore nella colonna non manca
	È valido	isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna è valido (il valore è di tipo dati o può essere convertito in tipo di dati)

Tipo di colonna	Condizione	ValueExpression	Description
	Non è valido	! isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna non è valido (il valore è di tipo dati o può essere convertito in tipo di dati)
Annidato	Manca	Manca (`colonna`)	Condizione per verificare se manca il valore nella colonna
	Non manca	! manca (`colonna`)	Condizione per verificare se il valore nella colonna non manca
	È valido	isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna è valido (il valore è di tipo dati o può essere convertit o in tipo di dati)
	Non è valido	! isValid (`column`, tipo di dati)	Condizione per verificare se il valore nella colonna non è valido (il valore è di tipo dati o può essere convertito in tipo di dati)

FLAG_COLUMN_FROM_NULL

Crea una nuova colonna, in base alla presenza di valori nulli in una colonna esistente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`— Il nome di una nuova colonna da creare.
- `flagType`— Un valore che deve essere impostato su`Null values`.
- `trueString`— Un valore per la nuova colonna, se nell'origine viene trovato un valore nullo. Se nessun valore è specificato, il valore predefinito è `True`.
- `falseString`— Un valore per la nuova colonna, se nell'origine viene trovato un valore diverso da nullo. Se nessun valore è specificato, il valore predefinito è `False`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FLAG_COLUMN_FROM_NULL",
        "Parameters": {
            "flagType": "Null values",
            "sourceColumn": "weight_kg",
            "targetColumn": "is_weight_kg_missing"
        }
    }
}
```

FLAG_COLUMN_FROM_PATTERN

Crea una nuova colonna, in base alla presenza di un pattern specificato dall'utente in una colonna esistente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`— Il nome di una nuova colonna da creare.
- `flagType`— Un valore che deve essere impostato su`Pattern`.
- `pattern`— Un'espressione regolare, che indica il modello da valutare.
- `trueString`— Un valore per la nuova colonna, se nell'origine viene trovato un valore nullo. Se nessun valore è specificato, il valore predefinito è `True`.

- `falseString`— Un valore per la nuova colonna, se nell'origine viene trovato un valore diverso da nullo. Se nessun valore è specificato, il valore predefinito è `False`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FLAG_COLUMN_FROM_PATTERN",
        "Parameters": {
            "falseString": "No",
            "flagType": "Pattern",
            "pattern": "N.*",
            "sourceColumn": "wind_direction",
            "targetColumn": "northerly",
            "trueString": "yes"
        }
    }
}
```

MERGE

Unisce due o più colonne in una nuova colonna.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di una o più colonne da unire.
- `delimiter`— Un separatore opzionale tra i valori, da visualizzare nella colonna di destinazione.
- `targetColumn`— Il nome della colonna unita da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MERGE",
        "Parameters": {
            "delimiter": " ",
```

```
            "sourceColumns": "[\"first_name\",\"last_name\"]",
            "targetColumn": "Merged Column 1"
        }
    }
}
```

SPLIT_COLUMN_BETWEEN_DELIMITER

Divide una colonna in tre nuove colonne, in base a un delimitatore iniziale e finale.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `patternOption1`— Una JSON-encoded stringa che rappresenta uno o più caratteri che indicano il primo delimitatore.
- `patternOption2`— Una JSON-encoded stringa che rappresenta uno o più caratteri che indicano il secondo delimitatore.
- `pattern`— Uno o più caratteri da utilizzare come separatore per la suddivisione dei dati.
- `includeInSplit`— Se vero, include il pattern nella nuova colonna; in caso contrario, il pattern viene scartato.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_BETWEEN_DELIMITER",
        "Parameters": {
            "patternOption1": "{\"pattern\":\"H\",\"includeInSplit\":true}",
            "patternOption2": "{\"pattern\":\"M\",\"includeInSplit\":true}",
            "sourceColumn": "last_name"
        }
    }
}
```

SPLIT_COLUMN_BETWEEN_POSITIONS

Divide una colonna in tre nuove colonne, in base agli offset specificati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `startPosition`— La posizione del carattere da cui deve iniziare la divisione.
- `endPosition`— La posizione del personaggio in cui deve terminare la divisione.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_BETWEEN_POSITIONS",
        "Parameters": {
            "endPosition": "12",
            "sourceColumn": "last_name",
            "startPosition": "2"
        }
    }
}
```

SPLIT_COLUMN_FROM_END

Divide una colonna in due nuove colonne, con uno scostamento dalla fine della stringa.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `position`— La posizione del carattere, a partire dall'estremità destra della stringa, in cui deve avvenire la divisione.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_FROM_END",
        "Parameters": {
            "position": "1",
            "sourceColumn": "nationality"
        }
```

```
    }
}
```

SPLIT_COLUMN_FROM_START

Divide una colonna in due nuove colonne, con uno scostamento rispetto all'inizio della stringa.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `position`— La posizione del carattere, a partire dall'estremità sinistra della stringa, in cui deve avvenire la divisione.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_FROM_START",
        "Parameters": {
            "position": "1",
            "sourceColumn": "first_name"
        }
    }
}
```

SPLIT_COLUMN_MULTIPLE_DELIMITER

Divide una colonna in base a più delimitatori.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `patternOptions`— Una JSON-encoded stringa che rappresenta uno o più pattern che determinano i criteri di divisione.
- `pattern`— Uno o più caratteri da utilizzare come separatore per la suddivisione dei dati.
- `limit`— Quante divisioni eseguire. Il minimo è 1, il massimo è 20.
- `includeInSplit`— Se vero, include il motivo nella nuova colonna; in caso contrario, il motivo viene scartato.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_MULTIPLE_DELIMITER",
        "Parameters": {
            "limit": "1",
            "patternOptions": "[{\"pattern\":\",\",\"includeInSplit\":true},{\"pattern
\":\" \",\"includeInSplit\":true}]",
            "sourceColumn": "description"
        }
    }
}
```

SPLIT_COLUMN_SINGLE_DELIMITER

Divide una colonna in una o più nuove colonne, in base a un delimitatore specifico.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `pattern`— Uno o più caratteri da utilizzare come separatore, quando si dividono i dati.
- `limit`— Quante divisioni eseguire. Il minimo è 1, il massimo è 20.
- `includeInSplit`— Se vero, include il motivo nella nuova colonna; in caso contrario, il motivo viene scartato.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_SINGLE_DELIMITER",
        "Parameters": {
            "includeInSplit": "true",
            "limit": "1",
            "pattern": "/",
            "sourceColumn": "info_url"
        }
    }
}
```

SPLIT_COLUMN_WITH_INTERVALS

Divide una colonna a intervalli di n caratteri, dove si specifica n.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `startPosition`— La posizione del carattere da cui deve iniziare la divisione.
- `interval`— Il numero di caratteri da saltare prima della divisione successiva.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SPLIT_COLUMN_WITH_INTERVALS",
        "Parameters": {
            "interval": "4",
            "sourceColumn": "nationality",
            "startPosition": "1"
        }
    }
}
```

Fasi della ricetta per la formattazione delle colonne

Utilizza i passaggi della procedura di formattazione delle colonne per modificare il formato dei dati nelle colonne.

Argomenti

- FORMATO_NUMERICO
- FORMATO_NUMERO_TELEFONICO

FORMATO_NUMERICO

Restituisce una colonna in cui un valore numerico viene convertito in una stringa formattata.

Parameters

- `sourceColumn` – Stringa. Il nome di una colonna esistente.
- `decimalPlaces`— Numero intero. Il valore del numero di cifre dopo il separatore decimale.
- `numericDecimalSeparator` – Stringa. Uno dei seguenti valori che indica il separatore decimale:
  - "."
  - ","
- `numericThousandSeparator` – Stringa. Uno dei seguenti valori che indica il separatore delle migliaia:
  - nullo. Indica che il separatore dei mille non è abilitato.
  - ","
  - " "
  - "."
  - "\\"
- `numericAbbreviatedUnit` – Stringa. Uno dei seguenti valori che indica l'unità di abbreviazione:
  - nullo. Indica che un'unità di abbreviazione non è abilitata.
  - «MILLE»
  - «MILIONI»
  - «MILIARDO»
  - «TRILIONI»
- `numericUnitAbbreviation` – Stringa. Uno dei seguenti valori o qualsiasi valore personalizzato, indicante l'abbreviazione dell'unità:
  - nullo. Indica che l'abbreviazione delle unità non è abilitata.
  -

Unità di abbreviazione	Opzioni
Migliaia	K, k, M, mille, personalizzato
Milioni	M, m, MM, milioni, personalizzato
Miliardi	miliardi, miliardi, personalizzati
Trilioni	T, tn, trilioni, personalizzato

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "NUMBER_FORMAT",
        "Parameters": {
            "sourceColumn": "income",
            "decimalPlaces": "2",
            "numericDecimalSeparator": ".",
            "numericThousandSeparator": ",",
            "numericAbbreviatedUnit": "THOUSAND",
            "numericUnitAbbreviation": "K"
        }
    }
}
```

FORMATO_NUMERO_TELEFONICO

Restituisce una colonna in cui una stringa di numeri di telefono viene convertita in un valore formattato.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `phoneNumberFormat`: il formato in cui convertire il numero di telefono. Se non viene specificato alcun formato, il valore predefinito è `E.164`, un formato standard per i numeri di telefono riconosciuto a livello internazionale. I valori validi includono quanto segue:
  - `E164`(ometti il periodo successivoE)
- `defaultRegion`: un codice regionale valido costituito da due o tre lettere maiuscole che specifica la regione del numero di telefono quando non è presente alcun prefisso nel numero stesso. Può essere indicato al massimo uno di `defaultRegion` o `defaultRegionColumn`.
- `defaultRegionColumn`— Il nome di una colonna del tipo `Country` di dati avanzato. Il codice regionale della colonna specificata viene utilizzato per determinare il prefisso del numero di telefono quando non è presente nel numero stesso. Può essere indicato al massimo uno di `defaultRegion` o `defaultRegionColumn`.

Note

- Gli input che non possono essere formattati con un numero di telefono valido rimangono invariati.
- Se non viene fornita alcuna regione predefinita e un numero di telefono non inizia con un simbolo più (+) e il prefisso internazionale, il numero di telefono non viene formattato.

Example

Esempio: area predefinita fissa

```
{
    "Action": {
        "Operation": "FORMAT_PHONE_NUMBER",
        "Parameters": {
            "sourceColumn": "Phone Number",
            "defaultRegion": "US"
        }
    }
}
```

Esempio: opzione predefinita per la colonna relativa alla regione

```
{
    "Action": {
        "Operation": "FORMAT_PHONE_NUMBER",
        "Parameters": {
            "sourceColumn": "Phone Number",
            "defaultRegionColumn": "Country Code"
        }
    }
}
```

Fasi della ricetta della struttura dei dati

Utilizza questi passaggi della ricetta per tabulare e riepilogare i dati da diverse prospettive o per eseguire funzioni avanzate.

Argomenti

NEST_TO_ARRAY

Converte le colonne selezionate dall'utente in valori di matrice. L'ordine delle colonne selezionate viene mantenuto durante la creazione dell'array risultante. I diversi tipi di dati delle colonne vengono convertiti in un tipo comune che supporta i tipi di dati di tutte le colonne.

Parameters

- `sourceColumns`— Elenco delle colonne di origine.
- `targetColumn`— Il nome della colonna di destinazione.
- `removeSourceColumns`— Contiene il valore `true` o `false` indica se l'utente desidera rimuovere o meno le colonne di origine selezionate.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "NEST_TO_ARRAY",
        "Parameters": {
```

```
            "sourceColumns": "[\"age\",\"weight_kg\",\"height_cm\"]",
            "targetColumn": "columnName",
            "removeSourceColumns": "true"
        }
    }
}
```

NEST_TO_MAP

Converte le colonne selezionate dall'utente in coppie chiave-valore, ciascuna con una chiave che rappresenta il nome della colonna e un valore che rappresenta il valore della riga. L'ordine della colonna selezionata non viene mantenuto durante la creazione della mappa risultante. I diversi tipi di dati delle colonne vengono convertiti in un tipo comune che supporta i tipi di dati di tutte le colonne.

Parameters

- `sourceColumns`— Elenco delle colonne di origine.
- `targetColumn`— Il nome della colonna di destinazione.
- `removeSourceColumns`— Contiene il valore `true` o `false` indica se l'utente desidera rimuovere o meno le colonne di origine selezionate.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "NEST_TO_MAP",
        "Parameters": {
            "sourceColumns": "[\"age\",\"weight_kg\",\"height_cm\"]",
            "targetColumn": "columnName",
            "removeSourceColumns": "true"
        }
    }
}
```

NEST_TO_STRUCT

Converte le colonne selezionate dall'utente in coppie chiave-valore, ciascuna con una chiave che rappresenta il nome della colonna e un valore che rappresenta il valore della riga. L'ordine delle colonne selezionate e il tipo di dati di ogni colonna vengono mantenuti nella struttura risultante.

Parameters

- `sourceColumns`— Elenco delle colonne di origine.
- `targetColumn`— Il nome della colonna di destinazione.
- `removeSourceColumns`— Contiene il valore `true` o `false` indica se l'utente desidera rimuovere o meno le colonne di origine selezionate.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "NEST_TO_STRUCT",
        "Parameters": {
            "sourceColumns": "[\"age\",\"weight_kg\",\"height_cm\"]",
            "targetColumn": "columnName",
            "removeSourceColumns": "true"
        }
    }
}
```

UNNEST_ARRAY

Snidifica una colonna di tipo `array` in una nuova colonna. Se l'array contiene più di un valore, viene generata una riga corrispondente a ciascun elemento. Questa funzione disintegra solo un livello di una colonna di matrice.

Parameters

- `sourceColumn`— Il nome di una colonna esistente. Questa colonna deve essere di `struct` tipo.
- `targetColumn`— Nome della colonna di destinazione generata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UNNEST_ARRAY",
        "Parameters": {
```

```
            "sourceColumn": "address",
            "targetColumn": "address"
        }
    }
}
```

UNNEST_MAP

Snidifica una colonna di tipo `map` e genera una colonna per la chiave e il valore. Se è presente più di una coppia chiave-valore, viene generata una riga corrispondente a ciascun valore chiave. Questa funzione disintegra solo un livello di una colonna della mappa.

Parameters

- `sourceColumn`— Il nome di una colonna esistente. Questa colonna deve essere di `struct` tipo.
- `removeSourceColumn`— Se`true`, la colonna di origine viene eliminata dopo il completamento della funzione.
- `targetColumn`— Se fornita, ciascuna colonna generata inizierà con questo come prefisso.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UNNEST_MAP",
        "Parameters": {
            "sourceColumn": "address",
            "removeSourceColumn": "false",
            "targetColumn": "address"
        }
    }
}
```

UNNEST_STRUCT

Annidifica una colonna di tipo `struct` e genera una colonna per ciascuna delle chiavi presenti nella struttura. Questa funzione smonta solo la struttura di primo livello.

Parameters

- `sourceColumn`— Il nome di una colonna esistente. Questa colonna deve essere di tipo strutturale.
- `removeSourceColumn`— Se`true`, la colonna di origine viene eliminata dopo il completamento della funzione.
- `targetColumn`— Se fornita, ciascuna colonna generata inizierà con questo come prefisso.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UNNEST_STRUCT",
        "Parameters": {
            "sourceColumn": "address",
            "removeSourceColumn": "false"
            "targetColumn": "add"
        }
    }
}
```

UNNEST_STRUCT_N

Crea una nuova colonna per ogni campo di una colonna di tipo selezionato. `struct`

Ad esempio, data la seguente struttura:

```
            user {
               name: “Ammy”
               address: {
                  state: "CA",
                  zipcode: 12345
               }
            }
```

Questa funzione crea 3 colonne:

nome.utente	user.address.state	user.address.zip code
Ammy	CA	12345

Parameters

- `sourceColumns`— Elenco delle colonne di origine.
- `regexColumnSelector`— Un'espressione regolare per selezionare le colonne da separare.
- `removeSourceColumn`— Un valore booleano. Se vero, rimuovi la colonna di origine; altrimenti conservala.
- `unnestLevel`— Il numero di livelli da disnidificare.
- `delimiter`— Il delimitatore viene utilizzato nel nome della colonna appena creata per separare i diversi livelli della struttura. Ad esempio: se il delimitatore è «/», il nome della colonna avrà la seguente forma: «/state». user/address
- `conditionExpressions`— Espressioni condizionali.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UNNEST_STRUCT_N",
        "Parameters": {
            "sourceColumns": "[\"address\"]",
            "removeSourceColumn": "true",
            "unnestLevel": "2",
            "delimiter": "/"
        }
    }
}
```

GROUP_BY

Riepiloga i dati raggruppando le righe per una o più colonne e quindi applicando una funzione di aggregazione a ciascun gruppo.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne che costituiscono la base di ogni gruppo.
- `groupByAggFunctions`— Una JSON-encoded stringa che rappresenta un elenco di funzioni di aggregazione da applicare. (Se non si desidera l'aggregazione, specificare`UNAGGREGATED`.)
- `useNewDataFrame`— Se vero, i risultati di GROUP_BY vengono resi disponibili nella sessione del progetto, sostituendone il contenuto corrente.

Example Esempio

```
[
  {
    "Action": {
      "Operation": "GROUP_BY",
      "Parameters": {
        "groupByAggFunctionOptions": "[{\"sourceColumnName\":\"all_votes\",
\"targetColumnName\":\"all_votes_count\",\"targetColumnDataType\":\"number\",
\"functionName\":\"COUNT\"}]",
        "sourceColumns": "[\"year\",\"state_name\"]",
        "useNewDataFrame": "true"
      }
    }
  }
]
```

JOIN

Esegue un'operazione di unione su due set di dati.

Parameters

- `joinKeys`— Una JSON-encoded stringa che rappresenta un elenco di colonne di ogni set di dati che fungono da chiavi di unione.
- `joinType`— Il tipo di join da eseguire. Deve essere uno dei seguenti: `INNER_JOIN` `LEFT_JOIN` | `RIGHT_JOIN` | `OUTER_JOIN` | `LEFT_EXCLUDING_JOIN` | `RIGHT_EXCLUDING_JOIN` | `OUTER_EXCLUDING_JOIN`

- `leftColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne del set di dati attivo corrente.
- `rightColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne di un altro set di dati (secondario) da unire a quello corrente.
- `secondInputLocation`— Un URL Amazon S3 che si risolve nel file di dati per il set di dati secondario.
- `secondaryDatasetName`— Il nome del set di dati secondario.

Example Esempio

```
{
    "Action": {
        "Operation": "JOIN",
        "Parameters": {
            "joinKeys": "[{\"key\":\"assembly_session\",\"value\":\"assembly_session
\"},{\"key\":\"state_code\",\"value\":\"state_code\"}]",
            "joinType": "INNER_JOIN",
            "leftColumns": "[\"year\",\"assembly_session\",\"state_code\",\"state_name
\",\"all_votes\",\"yes_votes\",\"no_votes\",\"abstain\",\"idealpoint_estimate
\",\"affinityscore_usa\",\"affinityscore_russia\",\"affinityscore_china\",
\"affinityscore_india\",\"affinityscore_brazil\",\"affinityscore_israel\"]",
            "rightColumns": "[\"assembly_session\",\"vote_id\",\"resolution\",
\"state_code\",\"state_name\",\"member\",\"vote\"]",
            "secondInputLocation": "s3://databrew-public-datasets-us-east-1/votes.csv",
            "secondaryDatasetName": "votes"
        }
    }
}
```

PIVOT

Converte tutti i valori di riga in una colonna selezionata in singole colonne con valori.

Parameters

- `sourceColumn`— Il nome di una colonna esistente. La colonna può avere un massimo di 10 valori distinti.
- `valueColumn`— Il nome di una colonna esistente. La colonna può avere un massimo di 10 valori distinti.
- `aggregateFunction`— Il nome di una funzione di aggregazione. Se non desideri l'aggregazione, usa la parola chiave. `COLLECT_LIST`

Example Esempio

```
{
    "Action": {
        "Operation": "PIVOT",
        "Parameters": {
            "aggregateFunction": "SUM",
            "sourceColumn": "state_name",
            "valueColumn": "all_votes"
        }
    }
}
```

SCALE

Ridimensiona o normalizza l'intervallo di dati in una colonna numerica.

Parameters

- `sourceColumn`— Il nome di una colonna esistente.
- `strategy`— L'operazione da applicare ai valori delle colonne:
  - `MIN_MAX`— Ridimensiona i valori in un intervallo di [0,1].
  - `SCALE_BETWEEN`— Ridimensiona i valori in un intervallo di due valori specificati.
  - `MEAN_NORMALIZATION`— Ridimensiona i dati in modo che abbiano una media (μ) di 0 e una deviazione standard (μ) di 1 entro un intervallo di [-1, 1].
  - `Z_SCORE`— Ridimensiona linearmente i valori dei dati in modo che abbiano una media (μ) di 0 e una deviazione standard (π) di 1. Ideale per gestire i valori anomali.

- `targetColumn`— Il nome di una colonna che contiene i risultati.

Example Esempio

```
{
    "Action": {
        "Operation": "NORMALIZATION",
        "Parameters": {
            "sourceColumn": "all_votes",
            "strategy": "MIN_MAX",
            "targetColumn": "all_votes_normalized"
        }
    }
}
```

TRASPORRE

Converte tutte le righe selezionate in colonne e le colonne in righe.

Column 1	Column A	Column B	Column C
Row A	Value A	Value B	Value C
Row B	Value A1	Value B1	Value C1

New column	Row A	Row B
Column A	Value A	Value A1
Column B	Value B	Value B1
Column C	Value C	Value C1

Parameters

- `pivotColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne le cui righe verranno convertite in nomi di colonne.
- `valueColumns`— Una JSON-encoded stringa che rappresenta un elenco di una o più colonne da convertire in righe.
- `aggregateFunction`— Il nome di una funzione di aggregazione. Se non desideri l'aggregazione, usa la parola chiave. `COLLECT_LIST`

- `newColumn`— La colonna per contenere le colonne trasposte come valori.

Example Esempio

```
{
    "Action": {
        "Operation": "TRANSPOSE",
        "Parameters": {
            "pivotColumns": "[\"Teacher\"]",
            "valueColumns": "[\"Tom\",\"John\",\"Harry\"]",
            "aggregateFunction": "COLLECT_LIST",
            "newColumn": "Student"
        }
    }

}
```

UNION

Combina le righe di due o più set di dati in un unico risultato.

Parameters

- `datasetsColumns`— Una JSON-encoded stringa che rappresenta un elenco di tutte le colonne nei set di dati.
- `secondaryDatasetNames`— Una JSON-encoded stringa che rappresenta un elenco di uno o più set di dati secondari.
- `secondaryInputs`— Una JSON-encoded stringa che rappresenta un elenco di bucket Amazon S3 e nomi di chiavi di oggetti che indicano DataBrew dove trovare i set di dati secondari.
- `targetColumnNames`— Una JSON-encoded stringa che rappresenta un elenco di nomi di colonna per i risultati.

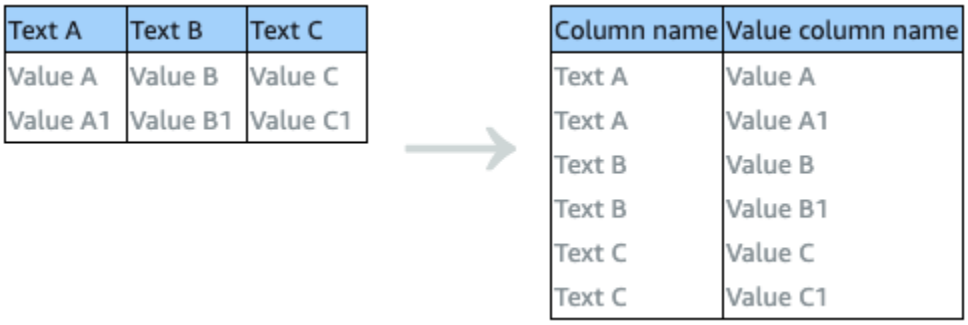
Example Esempio

```
{
    "Action": {
```

```
        "Operation": "UNION",
        "Parameters": {
            "datasetsColumns": "[[\"assembly_session\",\"state_code\",
\"state_name\",\"year\",\"all_votes\",\"yes_votes\",\"no_votes\",\"abstain
\",\"idealpoint_estimate\",\"affinityscore_usa\",\"affinityscore_russia\",
\"affinityscore_china\",\"affinityscore_india\",\"affinityscore_brazil\",
\"affinityscore_israel\"],[\"assembly_session\",\"state_code\",\"state_name
\",null,null,null,null,null,null,null,null,null,null,null,null]]",
            "secondaryDatasetNames": "[\"votes\"]",
            "secondaryInputs": "[{\"S3InputDefinition\":{\"Bucket\":\"databrew-public-
datasets-us-east-1\",\"Key\":\"votes.csv\"}}]",
            "targetColumnNames": "[\"assembly_session\",\"state_code\",\"state_name\",
\"year\",\"all_votes\",\"yes_votes\",\"no_votes\",\"abstain\",\"idealpoint_estimate
\",\"affinityscore_usa\",\"affinityscore_russia\",\"affinityscore_china\",
\"affinityscore_india\",\"affinityscore_brazil\",\"affinityscore_israel\"]"
        }
    }
}
```

UNPIVOT

Converte tutti i valori di colonna in una riga selezionata in singole righe con valori.



Text A	Text B	Text C
Value A	Value B	Value C
Value A1	Value B1	Value C1

Column name	Value column name
Text A	Value A
Text A	Value A1
Text B	Value B
Text B	Value B1
Text C	Value C
Text C	Value C1

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di una o più colonne da rimuovere dal pivot.
- `unpivotColumn`— La colonna dei valori per l'operazione unpivot.
- `valueColumn`— La colonna per contenere i valori non pivotati.

Example Esempio

```
{
    "Action": {
        "Operation": "UNPIVOT",
        "Parameters": {
            "sourceColumns": "[\"idealpoint_estimate\"]",
            "unpivotColumn": "unpivoted_idealpoint_estimate",
            "valueColumn": "unpivoted_column_values"
        }
    }
}
```

Fasi della ricetta della scienza dei dati

Usa queste istruzioni per tabulare e riepilogare i dati da diverse prospettive o per eseguire trasformazioni avanzate.

Argomenti

- BINARIZZAZIONE
- BUCKETIZZAZIONE
- CATEGORICAL_MAPPING
- ONE_HOT_ENCODING
- SCALE
- ASIMMETRIA
- TOKENIZZAZIONE

BINARIZZAZIONE

Prende tutti i valori in una colonna sorgente numerica selezionata, li confronta con un valore di soglia e restituisce una nuova colonna con 1 o 0 per ogni riga.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  `targetColumn`: il nome della nuova colonna da creare.

  `threshold`— Numero che indica la soglia per l'assegnazione del valore 0 o 1.

`flip`— Opzione per invertire l'assegnazione binaria in modo che ai valori inferiori venga assegnato 1 e ai valori più alti venga assegnato 0. Quando il parametro flip è vero, i valori inferiori o uguali al valore di soglia restituiscono 1 e i valori maggiori del valore di soglia restituiscono 0.

Example Esempio

```
{
    "Action": {
        "Operation": "BINARIZATION",
        "Parameters": {
            "sourceColumn": "level",
            "targetColumn": "bin",
            "threshold": "100.0",
            "flip": "false"
        }
    }
}
```

BUCKETIZZAZIONE

La bucketizzazione (chiamata Binning nella console) prende gli elementi in una colonna di valori numerici, li raggruppa in contenitori definiti da intervalli numerici e genera una nuova colonna che visualizza il contenitore per ogni riga. La bucketizzazione può essere effettuata utilizzando suddivisioni o percentuali. Il primo esempio seguente utilizza le suddivisioni e il secondo esempio utilizza una percentuale.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  `targetColumn`: il nome della nuova colonna da creare.

  `bucketNames`— Elenco dei nomi dei bucket.

  `splits`— Elenco dei livelli dei bucket. I bucket sono consecutivi e un limite superiore per un bucket sarà un limite inferiore per il bucket successivo.

  `percentage`— Ogni bucket verrà descritto come percentuale.

Example Esempio di utilizzo delle suddivisioni

```
{
    "Action": {
        "Operation": "BUCKETIZATION",
        "Parameters": {
            "sourceColumn": "level",
            "targetColumn": "bin",
            "bucketNames": "[\"Bin1\",\"Bin2\",\"Bin3\"]",
            "splits": "[\"-Infinity\",\"2\",\"20\",\"Infinity\"]"
        }
    }
}
```

Example Esempio di utilizzo di una percentuale

```
{
    "Action": {
        "Operation": "BUCKETIZATION",
        "Parameters": {
            "sourceColumn": "level",
            "targetColumn": "bin",
            "bucketNames": "[\"Bin1\",\"Bin2\"]",
            "percentage": "50"
        }
    }
}
```

CATEGORICAL_MAPPING

Mappa uno o più valori categoriali su valori numerici o di altro tipo

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  `categoryMap`— Una JSON-encoded stringa che rappresenta una mappa di valori in categorie.

  `deleteOtherRows`— Se`true`, tutte le righe non mappate verranno rimosse dal set di dati.

  `other`— Quando fornito, tutti i valori non mappati verranno sostituiti da questo valore.

`keepOthers`— Se vero, tutti i valori non mappati rimarranno gli stessi.

`mapType`— Il tipo di dati della colonna mappata.

`targetColumn`— Il nome di una colonna che contiene i risultati.

Example Esempio

```
{
    "Action": {
        "Operation": "CATEGORICAL_MAPPING",
        "Parameters": {
            "categoryMap": "{\"United States of America\":\"1\",\"Canada\":\"2\",\"Cuba
\":\"3\",\"Haiti\":\"4\",\"Dominican Republic\":\"5\"}",
            "deleteOtherRows": "false",
            "keepOthers": "true",
            "mapType": "NUMERIC",
            "sourceColumn": "state_name",
            "targetColumn": "state_name_mapped"
        }
    }
}
```

ONE_HOT_ENCODING

Crea n colonne numeriche, dove n è il numero di valori univoci in una variabile categoriale selezionata.

Ad esempio, si consideri una colonna denominata. `shirt_size` Le camicie sono disponibili nelle taglie S, M, L o XL. I dati della colonna potrebbero essere simili ai seguenti.

```
shirt_size
-----------
L
XL
M
S
M
M
S
```

```
XL
M
L
XL
M
```

In questo scenario, esistono quattro valori distinti per`shirt_size`. Pertanto, `ONE_HOT_ENCODING` genera quattro nuove colonne. A ogni nuova colonna viene assegnato un nome`shirt_size_`*x*, dove *x* rappresenta un `shirt_size` valore distinto.

I risultati `shirt_size` e le quattro colonne generate hanno questo aspetto.

```
shirt_size    shirt_size_S    shirt_size_M    shirt_size_L    shirt_size_XL
------------    ------------    ------------    ------------    -------------
L               0               0               1               0
XL              0               0               0               1
M               0               1               0               0
S               1               0               0               0
M               0               1               0               0
M               0               1               0               0
S               1               0               0               0
XL              0               0               0               1
M               0               1               0               0
L               0               0               1               0
XL              0               0               0               1
M               0               1               0               0
```

La colonna specificata `ONE_HOT_ENCODING` può avere un massimo di dieci (10) valori distinti.

Parameters

- `sourceColumn`: il nome di una colonna esistente. La colonna può avere un massimo di 10 valori distinti.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ONE_HOT_ENCODING",
        "Parameters": {
            "sourceColumn": "shirt_size"
```

```
        }
    }
}
```

SCALE

Ridimensiona o normalizza l'intervallo di dati in una colonna numerica.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `strategy`— L'operazione da applicare ai valori delle colonne:
  - `MIN_MAX`— Ridimensiona i valori in un intervallo di [0,1]
  - `SCALE_BETWEEN`— Ridimensiona i valori in un intervallo di 2 valori specificati.
  - `MEAN_NORMALIZATION`— Ridimensiona i dati in modo che abbiano una media (μ) di 0 e una deviazione standard (μ) di 1 entro un intervallo di [-1, 1]
  - `Z_SCORE`— Scala linearmente i valori dei dati in modo che abbiano una media (μ) di 0 e una deviazione standard (π) di 1. Ideale per gestire i valori anomali.
- `targetColumn`— Il nome di una colonna che contiene i risultati.

Example Esempio

```
{
    "Action": {
        "Operation": "NORMALIZATION",
        "Parameters": {
            "sourceColumn": "all_votes",
            "strategy": "MIN_MAX",
            "targetColumn": "all_votes_normalized"
        }
    }
}
```

ASIMMETRIA

Applica trasformazioni ai valori dei dati per modificare la forma di distribuzione e la relativa inclinazione.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  `targetColumn`: il nome della nuova colonna da creare.

  `skewFunction`
  - `ROOT`— estrae value-root. La radice può essere fornita nel parametro. `value`

    `LOG`— valore base del log. La base del log può essere fornita nel `value` parametro.

    `SQUARE`— funzione quadrata

  `value`— Argomento della funzione skewFunction.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SKEWNESS",
        "Parameters": {
            "sourceColumn": "level",
            "targetColumn": "bin",
            "skewFunction": "LOG",
            "value": "2.718281828"
        }
    }
}
```

TOKENIZZAZIONE

Divide il testo in unità più piccole, o token, ad esempio singole parole o termini.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `delimiter`— Un delimitatore personalizzato che appare tra le parole tokenizzate. (Il comportamento predefinito consiste nel separare ogni token con uno spazio.)
- `expandContractions`— Se`ENABLED`, espande le parole contratte. Ad esempio: «non» diventa «non farlo».

- `stemmingMode`— Divide il testo in unità o token più piccoli, ad esempio singole parole o termini minuscoli. Sono disponibili due modalità di derivazione: |. `PORTER` `LANCASTER`
- `stopWordRemovalMode`— Rimuove parole comuni come a, an, the e altre.
- `customStopWords`— Per`StopWordRemovalMode`, consente di specificare un elenco personalizzato di parole chiave.
- `targetColumn`— Il nome di una colonna per contenere i risultati.

Example Esempio

```
{
    "Action": {
        "Operation": "TOKENIZATION",
        "Parameters": {
            "customStopWords": "[]",
            "delimiter": "- ",
            "expandContractions": "ENABLED",
            "sourceColumn": "dimensions",
            "stemmingMode": "PORTER",
            "stopWordRemovalMode": "DEFAULT",
            "targetColumn": "dimensions_tokenized"
        }
    }

}
```

Funzioni matematiche

Di seguito, trovate gli argomenti di riferimento per le funzioni matematiche che funzionano con le azioni delle ricette.

Argomenti

- ABSOLUTE
- ADD
- CEILING
- DEGREES
- DIVIDERE

- ESPONENTE
- FLOOR
- È_PARI
- IS_ODD
- LN
- LOG
- MOD
- MOLTIPLICARE
- NEGARE
- PI
- POWER
- RADIANS
- RANDOM
- RANDOM_BETWEEN
- ROUND
- SIGN
- SQUARE_ROOT
- TOGLIERE

ABSOLUTE

Restituisce il valore assoluto del numero inserito in una nuova colonna. Il valore assoluto indica la distanza tra il numero e lo zero, indipendentemente dal fatto che sia positivo o negativo

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "ABSOLUTE",
        "Parameters": {
            "sourceColumn": "freezingTemps",
            "targetColumn": "absValueOfFreezingTemps"
        }
    }
}
```

ADD

Somma i valori della colonna di input in una nuova colonna, utilizzando (`sourceColumn1+sourceColumn2`) o (`sourceColumn1+value1`).

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `value1`— Un valore numerico.
- `sourceColumn2`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ADD",
        "Parameters": {
            "sourceColumn1": "weight_kg",
            "sourceColumn2": "height_cm",
            "targetColumn": "weight_plus_height"
        }
    }
}
```

CEILING

Restituisce il numero intero più piccolo maggiore o uguale ai numeri decimali immessi in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value1`— Un valore numerico.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "CEILING",
        "Parameters": {
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_CEILING"
        }
    }
}
```

DEGREES

Converte i radianti di un angolo in gradi e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DEGREES",
        "Parameters": {
            "sourceColumn": "height_cm",
            "targetColumn": "height_cm_DEGREES"
        }
    }
}
```

DIVIDERE

Divide un numero di input per un altro e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `value1`— Un valore numerico.
- `sourceColumn2`: il nome di una colonna esistente.
- `value2`— Un valore numerico.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DIVIDE",
        "Parameters": {
            "sourceColumn1": "height_cm",
            "targetColumn": "divide_by_2",
            "value2": "2"
        }
    }
}
```

ESPONENTE

Restituisce il numero di Eulero elevato all'ennesimo grado in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
```

```
        "Operation": "EXPONENT",
        "Parameters": {
            "sourceColumn": "age",
            "targetColumn": "age_EXPONENT"
        }
    }
}
```

FLOOR

Restituisce il numero integrale più grande maggiore o uguale al numero immesso in una nuova colonna.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `value`— Un valore numerico.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FLOOR",
        "Parameters": {
            "targetColumn": "FLOOR Column 1",
            "value": "42"
        }
    }
}
```

È_PARI

Restituisce un valore booleano in una nuova colonna che indica se la colonna o il valore di origine è pari. Se la colonna o il valore di origine sono decimali, il risultato è false.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

- `trueString`: una stringa che indica se il valore è pari.
- `falseString`— Una stringa che indica se il valore non è pari.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "IS_EVEN",
        "Parameters": {
            "falseString": "Value is odd",
            "sourceColumn": "height_cm",
            "targetColumn": "height_cm_IS_EVEN",
            "trueString": "Value is even"
        }
    }
}
```

IS_ODD

Restituisce un valore booleano in una nuova colonna che indica se la colonna o il valore di origine è dispari. Se la colonna o il valore di origine sono decimali, il risultato è false.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.
- `trueString`— Una stringa che indica se il valore è dispari.
- `falseString`— Una stringa che indica se il valore non è dispari.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "IS_ODD",
        "Parameters": {
            "falseString": "Value is even",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_IS_ODD",
```

```
            "trueString": "Value is odd"
        }
    }
}
```

LN

Restituisce il logaritmo naturale (numero di Eulero) di un valore in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "LN",
        "Parameters": {
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_LN"
        }
    }
}
```

LOG

Restituisce il logaritmo di un valore in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.
- `base`— La base del logaritmo. Il valore predefinito è 10.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "LOG",
        "Parameters": {
            "base": "10",
            "sourceColumn": "age",
            "targetColumn": "age_LOG"
        }
    }
}
```

MOD

Restituisce la percentuale di un numero rispetto a un altro numero in una nuova colonna.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MOD",
        "Parameters": {
            "sourceColumn1": "start_date",
            "sourceColumn2": "end_date",
            "targetColumn": "MOD Column 1"
        }
    }
}
```

MOLTIPLICARE

Moltiplica due numeri e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.

- `value1`— Un valore numerico.
- `sourceColumn2`: il nome di una colonna esistente.
- `value2`— Un valore numerico.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MULTIPLY",
        "Parameters": {
            "sourceColumn1": "hourly_rate",
            "sourceColumn2": "hours",
            "targetColumn": "total_pay"
        }
    }
}
```

NEGARE

Nega un valore e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "NEGATE",
        "Parameters": {
            "sourceColumn": "age",
            "targetColumn": "age_NEGATE"
        }
    }
}
```

PI

Restituisce il valore di pi (3,141592653589793) in una nuova colonna.

Parameters

- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "PI",
        "Parameters": {
            "targetColumn": "PI Column 1"
        }
    }
}
```

POWER

Restituisce il valore di un numero alla potenza dell'esponente in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`— Un numero il cui valore deve essere aumentato.
- `targetColumn`: il nome della nuova colonna da creare.
- `exponent`— La potenza a cui verrà innalzato il valore.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "POWER",
        "Parameters": {
            "exponent": "3",
            "sourceColumn": "age",
            "targetColumn": "age_cubed"
        }
    }
}
```

RADIANS

Converte i gradi in radianti (divide per 180/pi) e restituisce il valore in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "RADIANS",
        "Parameters": {
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_RADIANS"
        }
    }
}
```

RANDOM

Restituisce un numero casuale compreso tra 0 e 1 in una nuova colonna.

Parameters

- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "RANDOM",
        "Parameters": {
            "targetColumn": "RANDOM Column 1"
        }
    }
}
```

RANDOM_BETWEEN

In una nuova colonna, restituisce un numero casuale compreso tra un limite inferiore specificato (incluso) e un limite superiore specificato (incluso).

Parameters

- `lowerBound`— Il limite inferiore dell'intervallo di numeri casuali.
- `upperBound`— Il limite superiore dell'intervallo di numeri casuali.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "RANDOM_BETWEEN",
        "Parameters": {
            "lowerBound": "1",
            "targetColumn": "RANDOM_BETWEEN Column 1",
            "upperBound": "100"
        }
    }
}
```

ROUND

Arrotonda un valore numerico al numero intero più vicino in una nuova colonna. Arrotonda per eccesso quando la frazione è pari o superiore a 0,5.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ROUND",
        "Parameters": {
            "sourceColumn": "rating",
            "targetColumn": "rating_ROUND"
        }
    }
}
```

SIGN

Restituisce una nuova colonna con -1 se il valore è inferiore a 0, 0 se il valore è 0 e +1 se il valore è maggiore di 0.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SIGN",
        "Parameters": {
            "sourceColumn": "age",
            "targetColumn": "age_SIGN"
        }
    }
}
```

SQUARE_ROOT

Restituisce la radice quadrata di un valore in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SQUARE_ROOT",
        "Parameters": {
            "sourceColumn": "age",
            "targetColumn": "age_SQUARE_ROOT"
        }
    }
}
```

TOGLIERE

Sottrae un numero da un altro e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `value1`— Un valore numerico.
- `sourceColumn2`: il nome di una colonna esistente.
- `value2`— Un valore numerico.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "SUBTRACT",
        "Parameters": {
            "sourceColumn1": "weight_kg",
            "targetColumn": "weight_minus_10_kg",
            "value2": "10"
        }
    }
}
```

Funzioni di aggregazione

Di seguito, trova gli argomenti di riferimento per le funzioni di aggregazione che funzionano con le azioni delle ricette.

Argomenti

- ANY
- AVERAGE
- COUNT
- COUNT_DISTINCT
- KTH_LARGER
- KTH_LARGEST_UNIQUE
- MAX
- MEDIAN
- MIN
- MODE
- DEVIAZIONE_STANDARD
- SUM
- VARIANCE

ANY

Restituisce tutti i valori delle colonne di origine selezionate in una nuova colonna. I valori vuoti e nulli vengono ignorati.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ANY",
        "Parameters": {
            "sourceColumns": "[\"age\",\"last_name\"]",
            "targetColumn": "ANY Column 1"
        }
    }
}
```

AVERAGE

Calcola la media dei valori nelle colonne di origine e restituisce il risultato in una nuova colonna. Qualsiasi elemento diverso da un numero viene ignorato.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "AVERAGE",
        "Parameters": {
            "sourceColumns": "[\"age\",\"weight_kg\",\"height_cm\"]",
            "targetColumn": "AVERAGE Column 1"
        }
    }
}
```

COUNT

Restituisce il numero di valori dalle colonne di origine selezionate in una nuova colonna. I valori vuoti e nulli vengono ignorati.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "COUNT",
        "Parameters": {
            "sourceColumns": "[\"ANY Column 1\",\"birth_date\",\"last_name\"]",
            "targetColumn": "COUNT Column 1"
        }
    }
}
```

COUNT_DISTINCT

Restituisce il numero totale di valori distinti dalle colonne di origine selezionate in una nuova colonna. I valori vuoti e nulli vengono ignorati.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "COUNT_DISTINCT",
```

```
        "Parameters": {
            "sourceColumns": "[\"long_name\",\"weight_kg\"]",
            "targetColumn": "COUNT_DISTINCT Column 1"
        }
    }
}
```

KTH_LARGER

Restituisce il k-esimo numero più grande dalle colonne di origine selezionate in una nuova colonna.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.
- `value`— Un numero che rappresenta k.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "KTH_LARGEST",
        "Parameters": {
            "sourceColumns": "[\"height_cm\",\"weight_kg\",\"age\"]",
            "targetColumn": "KTH_LARGEST Column 1",
            "value": "2"
        }
    }
}
```

KTH_LARGEST_UNIQUE

Restituisce il k-esimo numero univoco più grande dalle colonne di origine selezionate in una nuova colonna.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.

- `targetColumn`: un nome per la nuova colonna creata.

  `value`— Un numero che rappresenta k.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "KTH_LARGEST_UNIQUE",
        "Parameters": {
            "sourceColumns": "[\"age\",\"height_cm\",\"weight_kg\"]",
            "targetColumn": "KTH_LARGEST_UNIQUE Column 1",
            "value": "3"
        }
    }
}
```

MAX

Restituisce il valore numerico massimo dalle colonne di origine selezionate in una nuova colonna. Qualsiasi elemento diverso da un numero viene ignorato.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MAX",
        "Parameters": {
            "sourceColumns": "[\"age\",\"height_cm\",\"weight_kg\"]",
            "targetColumn": "MAX Column 1"
        }
    }
}
```

MEDIAN

Restituisce la mediana, il numero centrale di un gruppo ordinato di numeri, dalle colonne di origine selezionate in una nuova colonna. Qualsiasi elemento diverso da un numero viene ignorato.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MEDIAN",
        "Parameters": {
            "sourceColumns": "[\"age\",\"years_in_service\"]",
            "targetColumn": "MEDIAN Column 1"
        }
    }
}
```

MIN

Restituisce il valore minimo dalle colonne di origine selezionate in una nuova colonna. Qualsiasi elemento diverso da un numero viene ignorato.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MIN",
```

```
        "Parameters": {
            "sourceColumns": "[\"age\",\"height_cm\",\"weight_kg\"]",
            "targetColumn": "MIN Column 1"
        }
    }
}
```

MODE

Restituisce la modalità, il numero che appare più spesso, dalle colonne di origine selezionate in una nuova colonna. Qualsiasi elemento diverso da un numero viene ignorato. Per più modalità, la modalità viene calcolata con la funzione modale.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MODE",
        "Parameters": {
            "modeType": "MINIMUM",
            "sourceColumns": "[\"years_in_service\",\"age\"]",
            "targetColumn": "MODE Column 1"
        }
    }
}
```

DEVIAZIONE_STANDARD

Restituisce la deviazione standard dalle colonne di origine selezionate in una nuova colonna.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "STANDARD_DEVIATION",
        "Parameters": {
            "sourceColumns": "[\"years_in_sservice\",\"age\"]",
            "targetColumn": "STANDARD_DEVIATION Column 1"
        }
    }
}
```

SUM

Restituisce la somma dei valori delle colonne di origine selezionate in una nuova colonna. Qualsiasi elemento diverso da un numero viene considerato come 0.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SUM",
        "Parameters": {
            "sourceColumns": "[\"age\",\"years_in_service\"]",
            "targetColumn": "SUM Column 1"
        }
    }
}
```

VARIANCE

Restituisce la varianza dalle colonne di origine selezionate in una nuova colonna. La varianza è definita come. `Var(X) = [Sum ((X – mean(X))^2)]/Count(X)`

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta un elenco di colonne esistenti.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "VARIANCE",
        "Parameters": {
            "sourceColumns": "[\"age\",\"years_in_service\"]",
            "targetColumn": "VARIANCE Column 1"
        }
    }
}
```

Funzioni di testo

Di seguito, trovate gli argomenti di riferimento per le funzioni di testo che funzionano con le azioni delle ricette.

Argomenti

- CHAR
- ENDS_WITH
- PRECISO
- TROVARE
- LEFT
- LEN
- LOWER
- MERGE_COLUMNS_AND_VALUES
- CORRETTO
- REMOVE_SYMBOLS
- REMOVE_WHITESPACE
- REPEAT_STRING

- RIGHT
- RIGHT_FIND
- STARTS_WITH
- STRING_GREATER_THAN
- STRING_GREATER_THAN_EQUAL
- STRING_LESS_THAN
- STRING_LESS_THAN_EQUAL
- SUBSTRING
- TRIM
- UNICODE
- UPPER

CHAR

Restituisce in una nuova colonna il carattere Unicode per ogni numero intero nella colonna di origine o per un valore intero personalizzato.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`— Un numero intero che rappresenta un valore Unicode.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "CHAR",
        "Parameters": {
```

```
            "sourceColumn": "age",
            "targetColumn": "age_char"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "CHAR",
        "Parameters": {
            "value": 42,
            "targetColumn": "asterisk"
        }
    }
}
```

ENDS_WITH

Restituisce `true` in una nuova colonna se un numero specificato di caratteri all'estrema destra, o una stringa personalizzata, corrisponde a uno schema.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `pattern`— Un'espressione regolare che deve corrispondere alla fine della stringa.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "ENDS_WITH",
```

```
        "Parameters": {
            "sourceColumn": "nationality",
            "pattern": "[Ss]",
            "targetColumn": "nationality_ends_with"
        }
    }
}
```

PRECISO

Crea una nuova colonna popolata con uno dei seguenti elementi:

- `True`se una stringa in una colonna (o valore) corrisponde esattamente a un'altra stringa in una colonna (o valore) diverso.
- `False`se non c'è alcuna corrispondenza.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `value1`: una stringa di caratteri da valutare.
- `value2`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> È possibile specificare solo una delle seguenti combinazioni:
>
> - Entrambi`sourceColumn`*N*.
> - Uno `sourceColumn`*N* e uno dei`value`*N*.
> - Entrambi`value`*N*.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "EXACT",
        "Parameters": {
            "sourceColumn1": "nationality",
            "value2": "Argentina",
            "targetColumn": "nationality_exact"
        }
    }
}
```

TROVARE

Effettuando una ricerca da sinistra a destra, trova le stringhe che corrispondono a una stringa specificata dalla colonna di origine o da un valore personalizzato e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `pattern`— Un'espressione regolare da cercare.
- `position`— La posizione del carattere con cui iniziare, dall'estremità sinistra della stringa.
- `ignoreCase`— Se`true`, ignora le differenze tra maiuscole e minuscole tra lettere maiuscole e minuscole. Per imporre una corrispondenza rigorosa, usa invece. `false`
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "FIND",
        "Parameters": {
            "sourceColumn": "city",
            "pattern": "[AEIOU]",
            "position": "1",
            "ignoreCase": "false",
            "targetColumn": "begins_with_a_vowel"
        }
    }
}
```

LEFT

Dato un numero di caratteri, prende il numero più a sinistra della stringa dalla colonna di origine o dalla stringa personalizzata e restituisce il numero specificato di caratteri all'estrema sinistra in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `position`— La posizione del carattere con cui iniziare, dall'estremità sinistra della stringa.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "LEFT",
        "Parameters": {
            "position": "3",
            "sourceColumn": "city",
            "targetColumn": "city_left"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "LEFT",
        "Parameters": {
            "position": "5",
            "value": "How now brown cow",
            "targetColumn": "how_now_5_left_chars"
```

```
        }
    }
}
```

LEN

Restituisce in una nuova colonna la lunghezza delle stringhe della colonna di origine o delle stringhe personalizzate.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "LEN",
        "Parameters": {
            "sourceColumn": "last_name",
            "targetColumn": "last_name_len"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "LEN",
        "Parameters": {
            "value": "Hello",
            "targetColumn": "hello_len"
```

```
        }
    }
}
```

LOWER

Converte tutti i caratteri alfabetici delle stringhe nella colonna di origine o nelle stringhe personalizzate in lettere minuscole e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "LOWER",
        "Parameters": {
            "sourceColumn": "last_name",
            "targetColumn": "last_name_lower"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "LOWER",
        "Parameters": {
            "value": "GOODBYE",
            "targetColumn": "goodbye_lower"
```

```
        }
    }
}
```

MERGE_COLUMNS_AND_VALUES

Concatena le stringhe nelle colonne di origine e restituisce il risultato in una nuova colonna. È possibile inserire un delimitatore tra i valori uniti.

Parameters

- `sourceColumns`— I nomi di due o più colonne esistenti, in formato. JSON-encoded
- `delimiter` : Opzionale. Uno o più caratteri da inserire tra i due valori delle colonne di origine.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MERGE_COLUMNS_AND_VALUES",
        "Parameters": {
            "sourceColumns": "[\"last_name\",\"birth_date\"]",
            "delimiter": " was born on: ",
            "targetColumn": "merged_column"
        }
    }
}
```

CORRETTO

Converte tutti i caratteri alfabetici delle stringhe nella colonna di origine o nei valori personalizzati in maiuscole e restituisce il risultato in una nuova colonna.

Nel caso corretto, detto anche maiuscolo, la prima lettera di ogni parola viene scritta in maiuscolo e il resto della parola viene trasformato in minuscolo. Un esempio è: The Quick Brown Fox Jumped Over The Fence

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "PROPER",
        "Parameters": {
            "sourceColumn": "first_name",
            "targetColumn": "first_name_proper"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "PROPER",
        "Parameters": {
            "value": "MR. H. SMITH, ESQ.",
            "targetColumn": "formal_name_proper"
        }
    }
}
```

REMOVE_SYMBOLS

Rimuove i caratteri che non sono lettere, numeri, caratteri latini accentati o spazi bianchi dalle stringhe nella colonna di origine o nelle stringhe personalizzate e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "REMOVE_SYMBOLS",
        "Parameters": {
            "sourceColumn": "info_url",
            "targetColumn": "info_url_remove_symbols"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REMOVE_SYMBOLS",
        "Parameters": {
            "value": "$&#$&HEY!#@@",
            "targetColumn": "without_symbols"
        }
    }
}
```

REMOVE_WHITESPACE

Rimuove lo spazio bianco dalle stringhe nella colonna sorgente o nelle stringhe personalizzate e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "REMOVE_WHITESPACE",
        "Parameters": {
            "sourceColumn": "job_desc",
            "targetColumn": "job_desc_remove_whitespace"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REMOVE_WHITESPACE",
        "Parameters": {
            "value": "This string has spaces in it",
            "targetColumn": "string_without_spaces"
        }
    }
}
```

REPEAT_STRING

Ripete le stringhe nella colonna di origine o nel valore di input personalizzato un numero di volte specificato e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- `value`: una stringa di caratteri da valutare.
- `count`— Il numero di volte in cui ripetere la stringa.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "REPEAT_STRING",
        "Parameters": {
            "count": 3,
            "sourceColumn": "last_name",
            "targetColumn": "last_name_repeat_string"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "REPEAT_STRING",
        "Parameters": {
            "count": 80,
            "value": "*",
            "targetColumn": "80_stars"
        }
    }
}
```

RIGHT

Dato un numero di caratteri, prende il numero più a destra delle stringhe dalla colonna di origine o dalle stringhe personalizzate e restituisce il numero specificato di caratteri all'estrema destra in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `position`— La posizione del carattere con cui iniziare, dal lato destro della stringa.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "RIGHT",
        "Parameters": {
            "sourceColumn": "nationality",
            "position": "3",
            "targetColumn": "nationality_right"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "RIGHT",
        "Parameters": {
            "value": "United States of America",
            "position": "7",
            "targetColumn": "usa_right"
        }
    }
}
```

RIGHT_FIND

Effettuando una ricerca da destra a sinistra, trova le stringhe che corrispondono a una stringa specificata dalla colonna di origine o da un valore personalizzato e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `pattern`— Un'espressione regolare da cercare.
- `position`— La posizione del carattere con cui iniziare, dall'estremità destra della stringa.
- `ignoreCase`— Se`true`, ignora le differenze tra maiuscole e minuscole tra lettere maiuscole e minuscole. Per imporre una corrispondenza rigorosa, usa invece. `false`
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "RIGHT_FIND",
        "Parameters": {
            "sourceColumn": "nationality",
            "pattern": "s",
            "position": "1",
            "ignoreCase": "true",
            "targetColumn": "ends_with_an_s"
        }
    }
}
```

STARTS_WITH

Restituisce `true` in una nuova colonna se un numero specificato di caratteri all'estrema sinistra, o una stringa personalizzata, corrisponde a uno schema.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.

- `pattern`— Un'espressione regolare che deve corrispondere all'inizio della stringa.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "STARTS_WITH",
        "Parameters": {
            "sourceColumn": "nationality",
            "pattern": "[AEIOU]",
            "targetColumn": "nationality_starts_with"
        }
    }
}
```

STRING_GREATER_THAN

Crea una nuova colonna popolata con uno dei seguenti elementi:

- `True`se una stringa in una colonna (o valore) è maggiore di un'altra stringa in una colonna (o valore) diverso.
- `False`se non c'è corrispondenza.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `value1`: una stringa di caratteri da valutare.
- `value2`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> È possibile specificare solo una delle seguenti combinazioni:
>
> - Entrambi`sourceColumn`*N*.
> - Uno `sourceColumn`*N* e uno dei`value`*N*.
> - Entrambi`value`*N*.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "STRING_GREATER_THAN",
        "Parameters": {
            "sourceColumn1": "first_name",
            "sourceColumn2": "last_name",
            "targetColumn": "string_greater_than"
        }
    }
}
```

STRING_GREATER_THAN_EQUAL

Crea una nuova colonna popolata con uno dei seguenti elementi:

- `True`se una stringa in una colonna (o valore) è maggiore o uguale a un'altra stringa in una colonna (o valore) diverso.
- `False`se non c'è corrispondenza.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `value1`: una stringa di caratteri da valutare.
- `value2`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> È possibile specificare solo una delle seguenti combinazioni:
>
> - Entrambi`sourceColumn`*N*.
> - Uno `sourceColumn`*N* e uno dei`value`*N*.
> - Entrambi`value`*N*.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "STRING_GREATER_THAN_EQUAL",
        "Parameters": {
            "sourceColumn1": "nationality",
            "targetColumn": "string_greater_than_equal",
            "value2": "s"
        }
    }
}
```

STRING_LESS_THAN

Crea una nuova colonna popolata con uno dei seguenti elementi:

- `True`se una stringa in una colonna (o valore) è inferiore a un'altra stringa in una colonna (o valore) diverso.
- `False`se non c'è corrispondenza.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `value1`: una stringa di caratteri da valutare.
- `value2`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> È possibile specificare solo una delle seguenti combinazioni:
>
> - Entrambi`sourceColumn`*N*.
> - Uno `sourceColumn`*N* e uno dei`value`*N*.
> - Entrambi`value`*N*.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "STRING_LESS_THAN",
        "Parameters": {
            "sourceColumn1": "first_name",
            "sourceColumn2": "last_name",
            "targetColumn": "string_less_than"
        }
    }
}
```

STRING_LESS_THAN_EQUAL

Crea una nuova colonna popolata con uno dei seguenti elementi:

- `True`se una stringa in una colonna (o valore) è minore o uguale a un'altra stringa in una colonna (o valore) diverso.
- `False`se non c'è corrispondenza.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `value1`: una stringa di caratteri da valutare.
- `value2`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> È possibile specificare solo una delle seguenti combinazioni:
>
> - Entrambi`sourceColumn`*N*.
> - Uno `sourceColumn`*N* e uno dei`value`*N*.
> - Entrambi`value`*N*.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "STRING_LESS_THAN_EQUAL",
        "Parameters": {
            "sourceColumn1": "first_name",
            "targetColumn": "string_less_than_equal",
            "value2": "s"
        }
    }
}
```

SUBSTRING

Restituisce in una nuova colonna alcune o tutte le stringhe specificate nella colonna di origine, in base ai valori dell'indice iniziale e finale definiti dall'utente.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `startPosition`— La posizione del carattere con cui iniziare, dall'estremità sinistra della stringa.
- `endPosition`— La posizione del carattere con cui terminare, dall'estremità sinistra della stringa.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SUBSTRING",
        "Parameters": {
            "sourceColumn": "last_name",
            "startPosition": "5",
            "endPosition": "8",
            "targetColumn": "chars_5_through_8"
        }
    }
}
```

TRIM

Rimuove gli spazi bianchi iniziali e finali dalle stringhe nella colonna di origine o nelle stringhe personalizzate e restituisce il risultato in una nuova colonna. Gli spazi tra le parole non vengono rimossi.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "TRIM",
        "Parameters": {
            "sourceColumn": "nationality",
```

```
            "targetColumn": "nationality_trim"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "TRIM",
        "Parameters": {
            "value": "   This string should be trimmed       ",
            "targetColumn": "string_trimmed"
        }
    }
}
```

UNICODE

Restituisce in una nuova colonna il valore dell'indice Unicode per il primo carattere delle stringhe nella colonna di origine o per le stringhe personalizzate.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "UNICODE",
        "Parameters": {
```

```
            "sourceColumn": "first_name",
            "targetColumn": "first_name_unicode"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "UNICODE",
        "Parameters": {
            "value": "?",
            "targetColumn": "sixty_three"
        }
    }
}
```

UPPER

Converte tutti i caratteri alfabetici delle stringhe nella colonna di origine o nelle stringhe personalizzate in lettere maiuscole e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "UPPER",
```

```
        "Parameters": {
            "sourceColumn": "last_name",
            "targetColumn": "last_name_upper"
        }
    }
}
```

```
{
    "RecipeAction": {
        "Operation": "UPPER",
        "Parameters": {
            "value": "a string of lowercase letters",
            "targetColumn": "string_upper"
        }
    }
}
```

Funzioni di data e ora

Di seguito, trova gli argomenti di riferimento per le funzioni di data e ora che funzionano con le azioni delle ricette.

Argomenti

- CONVERT_TIMEZONE
- DATE
- DATE_ADD
- DATE_DIFF
- DATA_FORMAT
- DATA_ORA
- GIORNO
- ORA
- MILLISECOND
- MINUTO
- MESE
- NOME_MESE

- NOW
- TRIMESTRE
- SECOND
- TIME
- OGGI
- UNIX_TIME
- UNIX_TIME_FORMAT
- GIORNO_SETTIMANA
- NUMERO_SETTIMANA
- ANNO

CONVERT_TIMEZONE

Converte un valore temporale dalla colonna di origine in una nuova colonna basata su un fuso orario specificato.

Parameters

- `sourceColumn`: il nome di una colonna esistente. La colonna di origine può essere di tipo`string`, `date` o. `timestamp`
- `fromTimeZone`— Fuso orario del valore di origine. Se non viene specificato nulla, il fuso orario predefinito è UTC.
- `toTimeZone`— Fuso orario in cui convertire. Se non viene specificato nulla, il fuso orario predefinito è UTC.
- `targetColumn`— Un nome per la colonna appena creata.
- `dateTimeFormat` : Opzionale. Una stringa di formato per la data. Se il formato non è specificato, viene utilizzato il formato predefinito:`yyyy-mm-dd HH:MM:SS`.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "CONVERT_TIMEZONE",
```

```
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "fromTimeZone": "UTC+08:00",
            "toTimeZone": "UTC+08:00",
            "targetColumn": "DATETIME Column CONVERT_TIMEZONE",
            "dateTimeFormat": "yyyy-mm-dd HH:MM:SS"
        }
    }
}
```

DATE

Crea una nuova colonna contenente il valore della data, dalle colonne di origine o dai valori forniti.

Parameters

- `dateTimeFormat` : Opzionale. Una stringa di formato per la data, così come verrà visualizzata nella nuova colonna. Se questa stringa non è specificata, il formato predefinito è`yyyy-mm-dd HH:MM:SS`.
- `dateTimeParameters`— Una JSON-encoded stringa che rappresenta i componenti della data e dell'ora:
  - `year`
  - `value`
  - `month`
  - `day`
  - `hour`
  - `second`

  Ogni componente deve specificare uno dei seguenti elementi:
  - `sourceColumn`: il nome di una colonna esistente.
  - `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
```

```
    "RecipeAction": {
        "Operation": "DATE",
        "Parameters": {
            "dateTimeFormat": "mm/dd/yy",
            "dateTimeParameters": "{\"year\":{\"value\":\"2019\"},\"month\":{\"value\":
\"12\"},\"day\":{\"value\":\"31\"},\"hour\":{},\"minute\":{},\"second\":{}}",
            "targetColumn": "DATE Column 1"
        }
    }
}
```

DATE_ADD

Aggiunge un anno, un mese o un giorno alla data da una colonna o da un valore di origine e crea una nuova colonna contenente i risultati.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `units`— Un'unità di misura per la regolazione della data. I valori validi sono `MONTHS` `YEARSMILLISECONDS`,,`QUARTERS`,`HOURS`,`MICROSECONDS`,`WEEKS`, `SECONDSDAYS`, e`MINUTES`.
- `dateAddValue`— Il numero `units` di da aggiungere alla data.
- `dateTimeFormat` : Opzionale. Una stringa di formato per la data, così come verrà visualizzata nella nuova colonna. Se non è specificato, il formato predefinito è `yyyy-mm-dd HH:MM:SS`.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DATE_ADD",
```

```
        "Parameters": {
            "sourceColumn": "DATE Column 1",
            "units": "DAYS",
            "dateAddValue": "14",
            "dateTimeFormat": "mm/dd/yyyy",
            "targetColumn": "DATE Column 1_DATEADD"
        }
    }
}
```

DATE_DIFF

Crea una nuova colonna contenente la differenza tra due date.

Parameters

- `sourceColumn1`: il nome di una colonna esistente.
- `sourceColumn2`: il nome di una colonna esistente.
- `value1`: una stringa di caratteri da valutare.
- `value2`: una stringa di caratteri da valutare.
- `units`— Un'unità di misura per descrivere la differenza tra le date. I valori validi sono `MONTHS YEARSMILLISECONDS`,`QUARTERS`,`HOURS`,`MICROSECONDS`,`WEEKS`,`SECONDS`,`DAYS`, e`MINUTES`.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> È possibile specificare solo una delle seguenti combinazioni:
>
> - Entrambi `sourceColumn1` e`sourceColumn2`.
> - Uno di `sourceColumn1` o `sourceColumn2` e uno di `value1` o`value2`.
> - Entrambi `value1` e`value2`.

Example Esempio

```
{
    "RecipeAction": {
```

```
        "Operation": "DATE_DIFF",
        "Parameters": {
            "value1": "2020-01-01",
            "value2": "2020-10-06",
            "units": "DAYS",
            "targetColumn": "DATEDIFF Column 1"
        }
    }
}
```

DATA_FORMAT

Crea una nuova colonna contenente una data, in un formato specifico, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`— Una stringa da valutare.
- `dateTimeFormat` : Opzionale. Una stringa di formato per la data, così come verrà visualizzata nella nuova colonna. Se non è specificato, il formato predefinito è `yyyy-mm-dd HH:MM:SS`.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempi

```
{
    "RecipeAction": {
        "Operation": "DATE_FORMAT",
        "Parameters": {
            "sourceColumn": "DATE Column 1",
            "dateTimeFormat": "month*dd*yyyy",
            "targetColumn": "DATE Column 1_DATEFORMAT"
        }
    }
```

```
}
```

```
{
    "RecipeAction": {
        "Operation": "DATE_FORMAT",
        "Parameters": {
            "value": "22:10:47",
            "dateTimeFormat": "HH:MM:SS",
            "targetColumn": "formatted_date_value"
        }
    }
}
```

DATA_ORA

Crea una nuova colonna contenente il valore di data e ora, dalle colonne di origine o dai valori forniti.

Parameters

- `dateTimeFormat` : Opzionale. Una stringa di formato per la data, così come verrà visualizzata nella nuova colonna. Se questa stringa non è specificata, il formato predefinito è`yyyy-mm-dd HH:MM:SS`.
- `dateTimeParameters`— Una JSON-encoded stringa che rappresenta i componenti della data e dell'ora:
  - `year`
  - `value`
  - `month`
  - `day`
  - `hour`
  - `second`

  Ogni componente deve specificare uno dei seguenti elementi:
  - `sourceColumn`: il nome di una colonna esistente.
  - `value`: una stringa di caratteri da valutare.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DATE_TIME",
        "Parameters": {
            "dateTimeFormat": "yyyy-mm-dd HH:MM:SS",
            "dateTimeParameters": "{\"year\":{\"value\":\"2010\"},\"month\":{\"value\":
\"5\"},\"day\":{\"value\":\"21\"},\"hour\":{\"value\":\"13\"},\"minute\":{\"value\":
\"34\"},\"second\":{\"value\":\"25\"}}",
            "targetColumn": "DATETIME Column 1"
        }
    }
}
```

GIORNO

Crea una nuova colonna contenente il giorno del mese, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "DAY",
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_DAY"
        }
    }
}
```

ORA

Crea una nuova colonna contenente il valore dell'ora, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "HOUR",
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_HOUR"
        }
    }
}
```

MILLISECOND

Crea una nuova colonna contenente il valore in millisecondi proveniente da una colonna sorgente o da un valore di input.

Parameters

- `sourceColumn`: il nome di una colonna esistente. La colonna di origine può essere di tipo`string`, `date` o. `timestamp`
- `value`: una stringa di caratteri da valutare.
- `targetColumn`— Un nome per la colonna appena creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MILLISECOND",
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_MILLISECOND"
        }
    }
}
```

MINUTO

Crea una nuova colonna contenente il valore dei minuti, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MINUTE",
        "Parameters": {
```

```
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_MINUTE"
        }
    }
}
```

MESE

Crea una nuova colonna contenente il numero del mese, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MONTH",
        "Parameters": {
            "value": "2018-05-27",
            "targetColumn": "MONTH Column 1"
        }
    }
}
```

NOME_MESE

Crea una nuova colonna contenente il nome del mese, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "MONTH_NAME",
        "Parameters": {
            "value": "2018-05-27",
            "targetColumn": "MONTHNAME Column 1"
        }
    }
}
```

NOW

Crea una nuova colonna contenente la data e l'ora correnti nel formato`yyyy-mm-dd HH:MM:SS`.

Parameters

- `timeZone`— Il nome di un fuso orario. Se non viene specificato alcun fuso orario, l'impostazione predefinita è Universal Coordinated Time (UTC).
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "NOW",
        "Parameters": {
            "timeZone": "US/Pacific",
```

```
            "targetColumn": "NOW Column 1"
        }
}
```

TRIMESTRE

Crea una nuova colonna contenente il trimestre basato sulla data da una stringa che rappresenta una data.

> **Note**
>
> I trimestri sono indicati nella nuova colonna come 1, 2, 3 o 4.
>
> - 1 corrisponde a gennaio, febbraio e marzo.
> - 2 è aprile, maggio e giugno.
> - Il 3 è luglio, agosto e settembre.
> - Il 4 è ottobre, novembre e dicembre.

Parameters

- `sourceColumn`: il nome di una colonna esistente. La colonna di origine può essere di tipo `stringdate`, `otimestamp`.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`— Un nome per la colonna appena creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "QUARTER",
```

```
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_QUARTER"
        }
    }
}
```

SECOND

Crea una nuova colonna contenente il secondo valore, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "SECOND",
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_SECOND"
        }
    }
}
```

TIME

Crea una nuova colonna contenente il valore temporale, dalle colonne o dai valori di origine forniti.

Parameters

- `dateTimeFormat` : Opzionale. Una stringa di formato per la data, così come deve apparire nella nuova colonna. Se questa stringa non è specificata, il formato predefinito è`yyyy-mm-dd HH:MM:SS`.
- `dateTimeParameters`— Una JSON-encoded stringa che rappresenta i componenti della data e dell'ora:
  - `year`
  - `value`
  - `month`
  - `day`
  - `hour`
  - `second`

  Ogni componente deve specificare uno dei seguenti elementi:
  - `sourceColumn`: il nome di una colonna esistente.
  - `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "TIME",
        "Parameters": {
            "dateTimeFormat": "HH:MM:SS",
            "dateTimeParameters": "{\"year\":{},\"month\":{},\"day\":{},\"hour\":
{\"sourceColumn\":\"rand_hour\"},\"minute\":{\"sourceColumn\":\"rand_minute\"},\"second
\":{\"sourceColumn\":\"rand_second\"}}",
            "targetColumn": "TIME Column 1"
        }
    }
}
```

OGGI

Crea una nuova colonna contenente la data corrente nel formato`yyyy-mm-dd`.

Parameters

- `timeZone`— Il nome di un fuso orario. Se non viene specificato alcun fuso orario, l'impostazione predefinita è Universal Coordinated Time (UTC).
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "TODAY",
        "Parameters": {
            "timeZone": "US/Pacific",
            "targetColumn": "TODAY Column 1"
        }
    }
}
```

UNIX_TIME

Crea una nuova colonna contenente un numero che rappresenta l'ora dell'epoca (ora Unix), il numero di secondi dal 1 gennaio 1970, in base a una colonna sorgente o a un valore di input. Se è possibile dedurre il fuso orario, l'output è in quel fuso orario. Altrimenti, l'output è in formato UTC (Universal Coordinated Time).

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UNIX_TIME",
        "Parameters": {
            "sourceColumn": "TIME Column 1",
            "targetColumn": "TIME Column 1_UNIXTIME"
        }
    }
}
```

UNIX_TIME_FORMAT

Converte l'ora Unix per una colonna sorgente o un valore di input in un formato di data numerico specificato e restituisce il risultato in una nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`— Un numero intero che rappresenta un timestamp di epoca Unix.
- `dateTimeFormat` : Opzionale. Una stringa di formato per la data, così come deve apparire nella nuova colonna. Se non è specificato, il formato predefinito è `yyyy-mm-dd HH:MM:SS`.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "UNIX_TIME_FORMAT",
        "Parameters": {
            "value": "1601936554",
            "dateTimeFormat": "yyyy-mm-dd HH:MM:SS",
            "targetColumn": "UNIXTIMEFORMAT Column 1"
        }
    }
}
```

GIORNO_SETTIMANA

Crea una nuova colonna contenente il giorno della settimana, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "WEEK_DAY",
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_WEEKDAY"
        }
    }
}
```

NUMERO_SETTIMANA

Crea una nuova colonna contenente il numero della settimana (da 1 a 52), da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "WEEK_NUMBER",
        "Parameters": {
            "sourceColumn": "DATETIME Column 1",
            "targetColumn": "DATETIME Column 1_WEEK_NUMBER"
        }
    }
}
```

ANNO

Crea una nuova colonna contenente l'anno, da una stringa che rappresenta una data.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: un nome per la nuova colonna creata.

> **Note**
>
> Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "YEAR",
        "Parameters": {
            "value": "2019-06-12",
            "targetColumn": "YEAR Column 1"
        }
    }
}
```

Funzioni finestra

Di seguito, trovate gli argomenti di riferimento per le funzioni delle finestre che funzionano con le azioni delle ricette.

Argomenti

- FILL
- NEXT
- PRECEDENTE
- ROLLING_AVERAGE
- ROLLING_COUNT_A
- ROLLING_KTH_LARGER
- ROLLING_KTH_LARGEST_UNIQUE
- ROLLING_MAX
- ROLLING_MIN
- ROLLING_MODE
- ROLLING_STANDARD_DEVIATION
- ROLLING_SUM

- ROLLING_VARIANCE
- ROW_NUMBER
- SESSION

FILL

Restituisce una nuova colonna basata su una colonna di origine specificata. Per eventuali valori mancanti o nulli nella colonna di origine, `FILL` sceglie il valore non vuoto più recente da una finestra di righe prima e dopo il valore di origine in questione. Il valore scelto viene quindi inserito nella nuova colonna.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `numRowsBefore`— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "FILL",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "last_name",
            "targetColumn": "last_name_FILL"
        }
    }
}
```

NEXT

Restituisce una nuova colonna, in cui ogni valore rappresenta un valore che si trova n righe successive nella colonna di origine.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- numRows— Un valore che rappresenta n righe precedenti nella colonna di origine. Ad esempio, se numRows è 3, NEXT utilizza il terzo `sourceColumn` valore successivo come nuovo `targetColumn` valore.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "NEXT",
        "Parameters": {
            "numRows": "1",
            "sourceColumn": "age",
            "targetColumn": "age_NEXT"
        }
    }
}
```

PRECEDENTE

Restituisce una nuova colonna, in cui ogni valore rappresenta un valore che si trova n righe prima nella colonna di origine.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- numRows— Un valore che rappresenta n righe precedenti nella colonna di origine. Ad esempio, se numRows è 3, PREV utilizza il terzo `sourceColumn` valore precedente come nuovo `targetColumn` valore.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "PREV",
        "Parameters": {
            "numRows": "1",
            "sourceColumn": "age",
            "targetColumn": "age_PREV"
        }
    }
}
```

ROLLING_AVERAGE

Restituisce in una nuova colonna la media progressiva dei valori da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- numRowsBefore— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- numRowsAfter— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_AVERAGE",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_AVERAGE"
        }
    }
```

```
}
```

ROLLING_COUNT_A

Restituisce in una nuova colonna il conteggio progressivo dei valori non nulli da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `numRowsBefore`— Un numero di righe prima della riga di origine corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_COUNT_A",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_COUNT_A"
        }
    }
}
```

ROLLING_KTH_LARGER

Restituisce in una nuova colonna il k-esimo valore più grande compreso tra un numero di righe specificato prima e un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

- numRowsBefore— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- numRowsAfter— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- value— Il valore di k.
- targetColumn: un nome per la nuova colonna creata.

Example Esempio

```
{
  "Action": {
    "Operation": "ROLLING_KTH_LARGEST",
    "Parameters": {
      "sourceColumn": "weight_kg",
      "numRowsBefore": "5",
      "numRowsAfter": "5",
      "value": "3"
      "targetColumn": "weight_kg_ROLLING_KTH_LARGEST"
    }
  }
}
```

ROLLING_KTH_LARGEST_UNIQUE

Restituisce in una nuova colonna il k-esimo valore univoco rotante più grande compreso tra un numero di righe specificato prima e un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- sourceColumn: il nome di una colonna esistente.
- numRowsBefore— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- numRowsAfter— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- value— Il valore di k.
- targetColumn: un nome per la nuova colonna creata.

Example Esempio

```
{
  "Action": {
    "Operation": "ROLLING_KTH_LARGEST_UNIQUE",
    "Parameters": {
      "sourceColumn": "games_played",
      "numRowsBefore": "3",
      "numRowsAfter": "3",
      "value": "5",
      "targetColumn": "weight_kg_ROLLING_KTH_LARGEST_UNIQUE"
    }
  }
}
```

ROLLING_MAX

Restituisce in una nuova colonna il numero massimo di valori variabile da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  numRowsBefore— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- numRowsAfter— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_MAX",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
```

```
            "targetColumn": "weight_kg_ROLLING_MAX"
        }
    }
}
```

ROLLING_MIN

Restituisce in una nuova colonna il minimo progressivo dei valori da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  `numRowsBefore`— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_MIN",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_MIN"
        }
    }
}
```

ROLLING_MODE

Restituisce in una nuova colonna la modalità di rotazione (il valore più comune) da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `numRowsBefore`— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- modeType — La funzione modale da applicare alla finestra. I valori validi sono `NONE`, `MINIMUM`, `MAXIMUM` e `AVERAGE`.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_MODE",
        "Parameters": {
            "modeType": "MINIMUM",
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_MODE"
        }
    }
}
```

ROLLING_STANDARD_DEVIATION

Restituisce in una nuova colonna la deviazione standard mobile dei valori da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `numRowsBefore`— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.

- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_STDEV",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_STDEV"
        }
    }
}
```

ROLLING_SUM

Restituisce in una nuova colonna la somma progressiva dei valori da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.

  `numRowsBefore`— Un numero di righe prima della riga sorgente corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_SUM",
        "Parameters": {
            "numRowsAfter": "10",
```

```
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_SUM"
        }
    }
}
```

ROLLING_VARIANCE

Restituisce in una nuova colonna la varianza progressiva dei valori da un numero specificato di righe precedenti a un numero specificato di righe dopo la riga corrente nella colonna specificata.

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `numRowsBefore`— Un numero di righe prima della riga di origine corrente, che rappresenta l'inizio della finestra.
- `numRowsAfter`— Un numero di righe dopo la riga di origine corrente, che rappresenta la fine della finestra.
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROLLING_VAR",
        "Parameters": {
            "numRowsAfter": "10",
            "numRowsBefore": "10",
            "sourceColumn": "weight_kg",
            "targetColumn": "weight_kg_ROLLING_VAR"
        }
    }
}
```

ROW_NUMBER

Restituisce in una nuova colonna un identificatore di sessione basato su una finestra creata dai nomi delle colonne delle istruzioni «group by» e «order by».

Parameters

- `groupByColumns`— Una JSON-encoded stringa che descrive le colonne «raggruppa per».
- `orderByColumns`— Una JSON-encoded stringa che descrive le colonne «ordina per».
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "ROW_NUMBER",
        "Parameters": {
            "groupByColumns": "[\"is public domain\"]",
            "orderByColumns": "[\"dimensions\"]",
            "targetColumn": "Row number"
        }
    }
}
```

SESSION

Restituisce in una nuova colonna un identificatore di sessione basato su una finestra creata dai nomi delle colonne delle istruzioni «group by» e «order by».

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `units`— Un'unità di misura per descrivere la durata della sessione. I valori validi sono `MONTHS` `YEARSMILLISECONDS`,`QUARTERS`,`HOURS`,`MICROSECONDS`,`WEEKS`,`SECONDS`,`DAYS`, e`MINUTES`.
- `value`— Il numero di `units` per definire il periodo di tempo.
- `groupByColumns`— Una JSON-encoded stringa che descrive le colonne «raggruppa per».
- `orderByColumns`— Una JSON-encoded stringa che descrive le colonne «ordina per».
- `targetColumn`: un nome per la nuova colonna creata.

Example Esempio

```
{
    "Action": {
        "Operation": "SESSION",
        "Parameters": {
            "sourceColumn": "object number",
            "units": "MINUTES",
            "value": "10",
            "groupByColumns": "[\"is public domain\"]",
            "orderByColumns": "[\"dimensions\"]",
            "targetColumn": "object number_SESSION",
        }
    }
}
```

Funzioni Web

Di seguito, trovate gli argomenti di riferimento per le funzioni Web che funzionano con le azioni relative alle ricette.

Argomenti

- IP_TO_INT
- INT_TO_IP
- URL_PARAMS

IP_TO_INT

Converte il valore del protocollo Internet versione 4 (IPv4) della colonna di origine o altro valore nel valore intero corrispondente nella colonna di destinazione e restituisce il risultato in una nuova colonna. Questa funzione è disponibile solo per IPv4.

Ad esempio, si consideri il seguente indirizzo IP.

```
192.168.1.1
```

Se utilizzate questo valore come input per`IP_TO_INT`, il valore di output è il seguente.

```
3232235777
```

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "IP_TO_INT",
        "Parameters": {
            "sourceColumn": "my_ip_address",
            "targetColumn": "IP_TO_INT Column 1"
        }
    }
}
```

INT_TO_IP

Converte il valore intero della colonna di origine o di un altro valore nel valore IPv4 corrispondente nella colonna di destinazione e restituisce il risultato in una nuova colonna. Questa funzione funziona solo per IPv4.

Ad esempio, si consideri il seguente numero intero.

```
167772410
```

Se utilizzate questo valore come input per`INT_TO_IP`, il valore di output è il seguente.

```
10.0.0.250
```

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.

- `targetColumn`: il nome della nuova colonna da creare.

Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
[  {
    "RecipeAction": {
      "Operation": "INT_TO_IP",
      "Parameters": {
        "sourceColumn": "my_integer",
        "targetColumn": "INT_TO_IP Column 1"
      }
    }
  }
]
```

URL_PARAMS

Estrae i parametri di query da una stringa URL, li formatta come oggetto JSON e restituisce il risultato in una nuova colonna.

Ad esempio, si consideri il seguente URL.

```
https://example.com/?firstParam=answer&secondParam=42
```

Se utilizzate questo valore come input per`URL_PARAMS`, il valore di output è il seguente.

```
{"firstParam": ["answer"], "secondParam": ["42"]}
```

Parameters

- `sourceColumn`: il nome di una colonna esistente.
- `value`: una stringa di caratteri da valutare.
- `targetColumn`: il nome della nuova colonna da creare.

Puoi specificare `sourceColumn` o `value`, ma non entrambi.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "URL_PARAMS",
        "Parameters": {
            "sourceColumn": "my_url",
            "targetColumn": "URL_PARAMS Column 1"
        }
    }
}
```

Altre funzioni

Di seguito, trovate gli argomenti di riferimento per altre funzioni che funzionano con le azioni delle ricette.

Argomenti

- COALESCE
- GET_ACTION_RESULT
- GET_STEP_DATAFRAME

COALESCE

Restituisce in una nuova colonna il primo valore non nullo trovato nella matrice di colonne. L'ordine delle colonne elencate nella funzione determina l'ordine in cui vengono cercate.

Parameters

- `sourceColumns`— Una JSON-encoded stringa che rappresenta l'elenco delle colonne esistenti.
- `targetColumn`: il nome della nuova colonna da creare.

Example Esempio

```
{
    "RecipeAction": {
```

```
        "Operation": "COALESCE",
        "Parameters": {
            "sourceColumns": "[\"nation_position\",\"joined\"]",
            "targetColumn": "COALESCE Column 1"
        }
    }
}
```

GET_ACTION_RESULT

Recupera il risultato di un'azione inviata in precedenza. Da utilizzare solo nell'esperienza interattiva.

Parameters

- `actionId`— La risposta ActionId restituita nella SendProjectSessionAction risposta originale.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "GET_ACTION_RESULT",
        "Parameters": {
            "actionId": "7",
        }
    }
}
```

GET_STEP_DATAFRAME

Recupera il frame di dati da una fase della ricetta del progetto. Da utilizzare solo nell'esperienza interattiva. Utilizzato con il ViewFrame parametro per impaginare su un frame di dati di grandi dimensioni.

Parameters

- `stepIndex`— L'indice della fase della ricetta del progetto per la quale recuperare il frame di dati.

Example Esempio

```
{
    "RecipeAction": {
        "Operation": "GET_STEP_DATAFRAME",
        "Parameters": {
            "stepIndex": "0"
        }
    }
}
```

Quote per AWS Glue DataBrew

Puoi visualizzare le quote DataBrew di servizio nella console AWS Service Quotas. Puoi anche richiedere un aumento della quota, per qualsiasi quota regolabile.

Cronologia dei documenti per AWS Glue DataBrew Guida per gli sviluppatori

Versione attuale dell'API: databrew-2017-07-25

La tabella seguente descrive la documentazione per questa versione di.AWS Glue DataBrew Se desideri ricevere una notifica quando la AWS Glue DataBrew Developer Guide viene aggiornata, puoi iscriverti al feed RSS.

Modifica	Descrizione	Data
`glue:GetCustomEntityType` aggiunto alle politiche AWS gestite	Questa autorizzazione è necessaria per eseguire i lavori di AWS Glue DataBrew profilo con PII-identification enabled. Per ulteriori informazioni, consulta AWS Glue DataBrew gli aggiornamenti delle politiche AWS gestite.	20 marzo 2024
Support per più algoritmi di hashing nella trasformazione CRYPTOGRAPHIC_HASH	Ora puoi specificare un algoritmo di hashing quando esegui l'hashing dei valori in una colonna. Per ulteriori informazioni, vedere CRYPTOGRAPHIC_HASH.	11 agosto 2023
`glue:BatchGetCustomEntityTypes` aggiunto AWS alle politiche gestite	Questa autorizzazione è necessaria per eseguire i lavori di AWS Glue DataBrew profilo con PII-identification enabled. Per ulteriori informazioni, consulta AWS Glue DataBrew gli aggiornamenti delle politiche AWS gestite.	09 maggio 2022

Support per il formato di file Apache ORC	DataBrew ora supporta Apache ORC come formato di file per sorgenti di DataBrew dati e output. Per ulteriori informazioni, consulta Tipi di file supportati per le fonti di dati.	31 marzo 2022
Support per l'accesso ad AWS Glue Data Catalog Amazon S3 su più account	Ora puoi accedere alle tabelle AWS Glue Data Catalog S3 da altre tabelle Account AWS se nella console viene creata una politica delle risorse appropria ta.AWS Glue Dopo aver creato una policy, le tabelle S3 del Data Catalog pertinent i possono essere selezionate come fonti di input durante la creazione di un DataBrew set di dati. Per ulteriori informazi oni, consulta Connessioni supportate per sorgenti di dati e output.	11 marzo 2022

Support per l'integrazione nativa della console con Amazon AppFlow	DataBrew ora ha l'integrazione nativa della console con Amazon AppFlow. Questa integrazione significa che puoi connetterti ai dati di Salesforce, Zendesk, Slack e altre applicazioni ServiceNow Software-as-a-Service (SaaS). Puoi anche connetterti a dati provenienti da Servizi AWS Amazon S3 e Amazon Redshift. Per ulteriori informazioni, consulta Connessioni supportate per sorgenti e output di dati.	18 novembre 2021
Support per le regole di qualità dei dati	DataBrew ora supporta la creazione di regole di qualità dei dati, che sono controlli di convalida personalizzabili che definiscono i requisiti aziendali per dati specifici. Per ulteriori informazioni, vedere Convalida della qualità dei dati in.AWS Glue DataBrew	18 novembre 2021

Support per istruzioni SQL personalizzate	DataBrew ora supporta istruzioni SQL personalizzate per il recupero di dati da Amazon Redshift e Snowflake . Questo supporto significa che puoi utilizzare una query appositamente creata per selezionare e limitare i dati restituiti da tabelle di grandi dimensioni. Per ulteriori informazioni, consulta Connessioni supportate per sorgenti di dati e output.	18 novembre 2021
Support per il rilevamento delle informazioni PII	DataBrew ora supporta il rilevamento di informazioni di identificazione personale (PII). In questo modo è possibile mascherare le informazi oni personali durante la preparazione dei dati. Per ulteriori informazioni, consulta Identificazione e gestione delle informazioni di identificazione personale (PII).	18 novembre 2021
Support per altre AWS regioni	DataBrew ora supporta AWS regioni aggiuntive. Per un elenco delle regioni supportate, consulta AWS Glue DataBrew endpoint e quote.	5 ottobre 2021

Supporto per la scrittura di dati su tabelle Lake Formation-based Amazon S3	DataBrew ora supporta la scrittura di dati in tabelle AWS Glue Data Catalog S3 basate su.AWS Lake Formation DataBrew ora supporta anche la scrittura di dati nel formato Tableau Hyper. Per maggiori informazioni, consulta Creare e lavorare con i lavori di AWS Glue DataBrew ricetta.	13 agosto 2021
Support per la scrittura di dati in destinazioni JDBC	DataBrew ora supporta la scrittura di dati direttamente in JDBC-supported database e data warehouse. Questi includono Amazon Redshift, Snowflake, Microsoft SQL Server, MySQL, Oracle Database e PostgreSQL. Per ulteriori informazioni, consulta Creazione e utilizzo dei lavori di ricetta.AWS Glue DataBrew	23 luglio 2021
Support per specificare quali statistiche sulla qualità dei dati vengono generate per un profilo di lavoro	DataBrew ora supporta la specificazione di quali statistiche sulla qualità dei dati vengono generate automaticamente per i set di dati in un job di profilo. Per ulteriori informazioni, consulta Creazione e utilizzo dei lavori di ricetta.AWS Glue DataBrew	23 luglio 2021

Support per la scrittura di set di dati in AWS Glue Data Catalog	DataBrew ora include il supporto per la scrittura di set di dati direttamente in.AWS Glue Data Catalog Puoi scegliere di archiviare i set di dati creati da processi che eseguono le tue ricette di preparazione dei dati nelle tabelle Amazon S3, Amazon Redshift e Amazon RDS nel Data Catalog. Le tabelle RDS supportate includono quelle per Amazon Aurora, RDS per Oracle, RDS per Microsoft SQL Server, RDS per MySQL e RDS per PostgreSQL.	30 giugno 2021
Support per l'identificazione di tipi di dati avanzati	DataBrew ora include il supporto per identificare e contrassegnare automatic amente i tipi di dati avanzati per le colonne, il che semplific a la normalizzazione delle colonne che contengono determinati tipi di dati. Questi tipi di dati includono numero di previdenza sociale, indirizzo e-mail, numero di telefono, sesso, carta di credito, URL, indirizzo IP, data e ora, valuta, codice postale, paese, regione, stato e città.	30 giugno 2021

Support per l'utilizzo AppFlow di Amazon per trasferire dati da applicazioni SAAS	DataBrew ora supporta l'utilizzo di Amazon AppFlow per trasferire dati in Amazon S3 da applicazioni software-as-a-service (SaaS) di terze parti come Salesforce, Zendesk, Slack e. ServiceNow Per ulteriori informazioni, consulta Connessioni supportate per sorgenti e output di dati.	29 aprile 2021
Support per la creazione di DataBrew set di dati con input da database JDBC	DataBrew ora supporta la creazione di set di dati da JDBC-supported database e data warehouse, tra cui Amazon Redshift, Snowflake, Microsoft SQL Server, MySQL, Oracle Database e PostgreSQL. Per ulteriori informazioni, consulta Connessioni supportate per sorgenti e output di dati.	2 aprile 2021
Support per ulteriori Regioni AWS	DataBrew ora supporta ulteriori Regioni AWS. Per un elenco delle regioni supportate, consulta AWS Glue DataBrew endpoint e quote.	28 gennaio 2021

Nuove trasformazioni per la gestione della duplicazione	Sono state aggiunte quattro nuove trasformazioni per la gestione della duplicazione alla console e all' DataBrew API. Per ulteriori informazioni, consulta DELETE_DUPLICATE_ROWS, FLAG_DUPLICATE_ROWS, FLAG_DUPLICATES_IN_COLUMNe REMOVE_DUPLICATES nei passaggi della ricetta sulla qualità dei dati.	28 gennaio 2021
Delimitatori CSV aggiuntivi	DataBrew ora supporta delimitatori aggiuntivi oltre alle virgole nei file con valori separati da virgole (CSV) utilizzati per creare set di dati. DataBrew Per ulteriori informazioni, consulta Creazione e utilizzo dei set di dati.AWS Glue DataBrew	28 gennaio 2021
DataBrew estensione per JupyterLab	Ora puoi usarla AWS Glue DataBrew come estension e in JupyterLab. Per ulteriori informazioni, consulta Utilizzo DataBrew come estensione in JupyterLab.	20 novembre 2020
Nuovo strumento di preparazi one dei dati:AWS Glue DataBrew	Questa è la prima versione della Guida per sviluppatori di AWS Glue DataBrew.	11 novembre 2020

AWS Glossario

Per la AWS terminologia più recente, consultate il AWS glossario nella sezione Reference.Glossario AWS

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.