Getting started with AWS Glue DataBrew

You can use the following tutorial to guide you in creating your first DataBrew project. You load a sample dataset, run transformations on that dataset, build a recipe to capture those transformations, and run a job to write the transformed data to Amazon S3.

Topics

Prerequisites

Before you proceed, follow the applicable instructions in Setting up AWS Glue DataBrew. Then continue to Step 1: Create a project.

Step 1: Create a project

In this step, you use the DataBrew console to quickly get started with a sample project.

To create a project

1. Sign in to the AWS Management Console and open the DataBrew console at https://console.aws.amazon.com/databrew/.
2. Make sure that your AWS Region is selected at upper-right on the DataBrew console. For a list of AWS Regions supported by DataBrew, see DataBrew endpoints and quotas in the *AWS General Reference.*
3. On the navigation pane, choose **Projects**, and then choose **Create project**.
4. On the **Project details** pane, do the following:

- For **Project name**, enter `chess-project`.
- For **Attached recipe**, create a new recipe. A suggested name for the recipe is provided (`chess-project-recipe`).

5. On the **Select a dataset** pane, choose **Sample files**.
6. On the **Sample files** pane, choose **Famous chess game moves**. This dataset contains detailed information on more than 20,000 games of chess.

   For **Dataset name** a suggested name for the dataset is provided (`chess-games`).
7. On the **Access permissions** pane, choose `AwsGlueDataBrewDataAccessRole`. This is a service-linked role that lets DataBrew access your Amazon S3 buckets on your behalf.
8. Choose **Create project**, and wait until DataBrew finishes preparing the project. The window looks similar to the following.

   The data that you see represents a sample from the `chess-games` dataset. By default, the sample consists of the first 500 rows from the dataset. You can change this project setting later.

   The toolbar provides access to hundreds of data transforms that you can apply to the data.

   The recipe pane at right in the DataBrew console tracks the transformations you applied so far.

Step 2: Summarize the data

In this step, you build a DataBrew recipe—a set of transformations that can be applied to this dataset and others like it. When the recipe is complete, you publish it so that it's available for use.

In the game of chess, players can be rated based on how well they perform against other players. (For more information, see https://en.wikipedia.org/wiki/Chess_rating_system). For this tutorial, you focus on only the games where both players were Class A, meaning that their ratings were 1800 or more.

To summarize the data

1. On the transformation toolbar, choose **Filter**, **By Condition**, **Greater than or equal to**.
2. Set these options as follows:

   - **Source column** - `white_rating`

- **Filter condition** – Greater than or equal to 1800

   To see how the transform works, choose **Preview changes**. Then choose **Apply**.

3. Repeat the previous step, but this time set **Source column** to `black_rating`. After you apply your changes, the sample data contains only those games where the players on each side (black and white) were Class A or above.
4. Summarize the data to determine how many games were won by each side. To do this, on the transformation toolbar, choose **Group**.
5. For the **Group** properties, do the following:

   a. In the first row, choose `winner` for **Column name**. Leave **Aggregate** set to **Group by**.

   b. In the second row, choose `victory_status` for the **Column name**. Leave **Aggregate** set to **Group by**.

   c. Choose **Add another column**.

   d. In the third row, choose `winner` for **Column name**. Set **Aggregate** to **Count**.

   e. For **Group type**, choose **Group as new table**. The preview pane shows you what the result will look like.

   f. Choose **Finish**.
6. Choose **Publish** to save your work, at right on the recipe pane.
7. For **Version Description**, enter **First version of my recipe**. Then choose **Publish**.

Step 3: Add more transformations

In this step, you add more transformations to your recipe and publish another version of it. To refine our example, we use the information that not all chess games result in a clear winner; some games are played to a draw.

To add more recipe transformations and republish

1. From the transformation toolbar, choose **Filter**, **By Condition**, **Is not** to remove the games that were played to a draw.
2. Set these options as follows:

   - **Source column** - `victory_status`

- **Filter condition –** Is not `draw`

   To add this transform to your recipe, choose **Apply**.

3. Change the data in `victory_status` so that it's more meaningful. To do this, from the transformation toolbar choose **Clean**, **Replace**, **Replace value or pattern**.
4. Set these options as follows:

   - **Source column** - `victory_status`
   - **Specify values to replace –** Value or pattern
   - **Value to be replaced** - `mate`
   - **Replace with value** - `checkmate`

   To add this transform to your recipe, choose **Apply**.
5. Repeat the previous step, but change `resign` to `other player resigned`.
6. Repeat the previous step, but change `outoftime` to `time ran out`.
7. Choose **Publish** to save your work, at right on the recipe pane.

Step 4: Review your DataBrew resources

Now that you worked with a sample project, review the DataBrew resources you created so far.

To review your DataBrew resources

1. On the navigation pane, choose **Datasets**.

   When you created the sample project, DataBrew created a dataset for you (`chess-games`). The source data file is stored in Amazon S3, and is in Microsoft Excel format (`chess-games.xlsx`). The file contains metadata from over 20,000 games of chess. The `chess-games` dataset provides the information that DataBrew needs to read the data in that file.
2. On the navigation pane, choose **Projects**.

   You should see the project that you worked with in the previous steps (`chess-project`). Every project requires a dataset, in this case `chess-games`. Every project also requires a recipe, so that you can add data transformation steps as you go along. When you created this sample project, DataBrew created a new (empty) recipe for you, and attached it to the project.

3. On the navigation pane, choose **Recipes**, and in the **Recipe name** column, choose **chess-project-recipe**. This shows you the recipe that DataBrew created for your project, and that you've refined by adding transformation steps to it.
4. At left, view the recipe versions that have been published. Choose one of these to view its **Recipe steps** tab, which shows the recipe details and steps for that version.
5. View the **Data lineage** tab, which shows where the data came from and how it's being used. For more details, choose any of the icons in the diagram.

Step 5: Create a data profile

When you work with on a project, DataBrew displays statistics such as the number of rows in the sample and the distribution of unique values in each column. These statistics, and many more, represent a *profile* of the sample.

To request a data profile, create and run a profile job.

To profile a dataset

1. On the navigation pane, choose **Jobs**.
2. On the **Profile jobs** tab, choose **Create job**.
3. For **Job name**, enter `chess-data-profile`.
4. For **Job type**, choose **Create a profile job**.
5. On the **Job input** pane, do the following:
   - For **Run on**, choose **Dataset**.
   - Choose **Select a dataset** to view a list of available datasets, and choose `chess-games`.
6. On the **Job output settings** pane, do the following:
   - For **File type**, choose **JSON** (JavaScript Object Notation).
   - Choose **S3 location** to view a list of available Amazon S3 buckets, and choose the bucket to use. Then choose **Browse**. In the list of folders, choose `databrew-output`, and chose **Select**.
7. On the **Access permissions** pane, choose `AwsGlueDataBrewDataAccessRole`. This is a service linked role that lets DataBrew access your Amazon S3 buckets on your behalf.
8. Choose **Create and run job**. DataBrew creates a job with your settings, and then runs it.

9. On the **Job run history** pane, wait for the job status to change from `Running` to `Succeeded`.

10. To view the profile, choose **VIEW PROFILE**:

    The **DATASETS** window is shown. Take some time to explore the following tabs:

    - Dataset preview
    - Profile overview
    - Column statistics
    - Data lineage statistics

Step 6: Transform the dataset

Until now, you tested your recipe on only a sample of the dataset. Now it's time to transform the entire dataset by creating a DataBrew recipe job.

When the job runs, DataBrew applies your recipe to all of the data in the dataset, and writes the transformed data to an Amazon S3 bucket. The transformed data is separate from the original dataset. DataBrew doesn't alter the source data.

Before you proceed, ensure that you have an Amazon S3 bucket in your account that you can write to. In that bucket, create a folder to capture the job output from DataBrew. To do these steps, use the following procedure.

To create an S3 bucket and folder to capture job output

1. Sign in to the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/databrew/.

   If you already have an Amazon S3 bucket available, and you have write permissions for it, skip the next step.

2. If you don't have an Amazon S3 bucket, choose **Create bucket**. For **Bucket name**, enter a unique name for your new bucket. Choose **Create bucket**.

3. From the list of buckets, choose the one that you want to use.

4. Choose **Create folder**.

5. For **Folder name**, enter `databrew-output`, and choose **Create folder**.

After you create an Amazon S3 bucket and folder to contain the job, run your job by using the following procedure.

To create and run a recipe job

1. On the navigation pane, choose **Jobs**.
2. On the **Recipe jobs** tab, choose **Create job**.
3. For **Job name**, enter `chess-winner-summary`.
4. For **Job type**, choose **Create a recipe job**.
5. On the **Job input** pane, do the following:
   - For **Run on**, choose **Dataset**.
   - Choose **Select a dataset** to view a list of available datasets, and choose `chess-games`.
   - Choose **Select a recipe** to view a list of available recipes, and choose `chess-project-recipe`.
6. On the **Job output settings** pane, do the following:
   - **File type –** chose **CSV** (comma-separated values).
   - **S3 location** - choose this field to view a list of available Amazon S3 buckets, and choose the bucket to use. Then choose **Browse**. In the list of folders, choose `databrew-output`, and choose **Select**.
7. On the **Access permissions** pane, choose `AwsGlueDataBrewDataAccessRole`. This service-linked role lets DataBrew access your Amazon S3 buckets on your behalf.
8. Choose **Create and run job**. DataBrew creates a job with your settings, and then runs it.
9. On the **Job run history** pane, wait for the job status to change from `Running` to `Succeeded`.
10. Choose **Output** to access the Amazon S3 console. Choose your S3 bucket, and then choose the `databrew-output` folder to access the job output.
11. (Optional) Choose **Download** to download the file and view its contents.

Step 7: (Optional) Clean up

The walkthrough is complete. You can keep using the DataBrew and Amazon S3 resources that you created, or delete them.

To clean up resources

1. Open the DataBrew console at https://console.aws.amazon.com/databrew/, and on the navigation pane, choose **Projects**.
2. Choose your project (**Sample project**). For **Actions**, choose **Delete**.
3. On the **Delete Sample project** pane, choose **Delete attached recipe**. Then choose **Delete**. Your project, along with its recipe and jobs, will be deleted.
4. On the navigation pane, choose **Datasets**.
5. Choose your dataset (`chess-games`), and for **Actions**, choose **Delete**.
6. Open the Amazon S3 console at https://console.aws.amazon.com/s3/. Delete the `databrew-output` folder and its contents.

   (Optional) If you're sure that you no longer need your Amazon S3 bucket, you can delete it.