Data Engineering Cohort 1

Module 5

Assignment 5.2

Glue Jobs in AWS

Member #1

Name: Muhammad Humza

Roll Number: 2211-018-KHI-DEG

Member #2

Name: Muhammad Ovais

Roll Number: 2211-023-KHI-DEG

Task

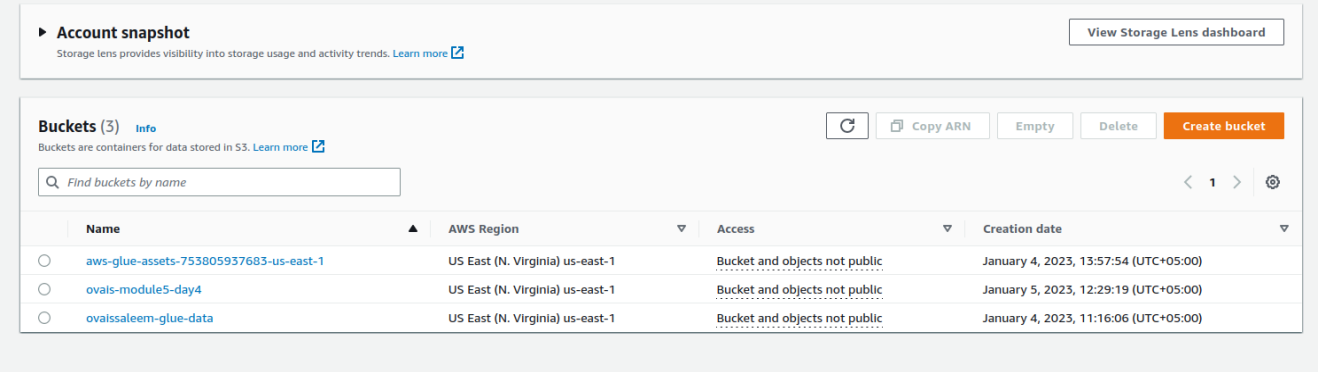
Using the earnings CSV as a base, prepare a new data file with employees' office locations. Make sure there are 5-6 distinct locations that are shared between employees.

Create a Glue job that aggregates the data based on the office location to calculate average salaries and raise percentages for these locations.

Solution

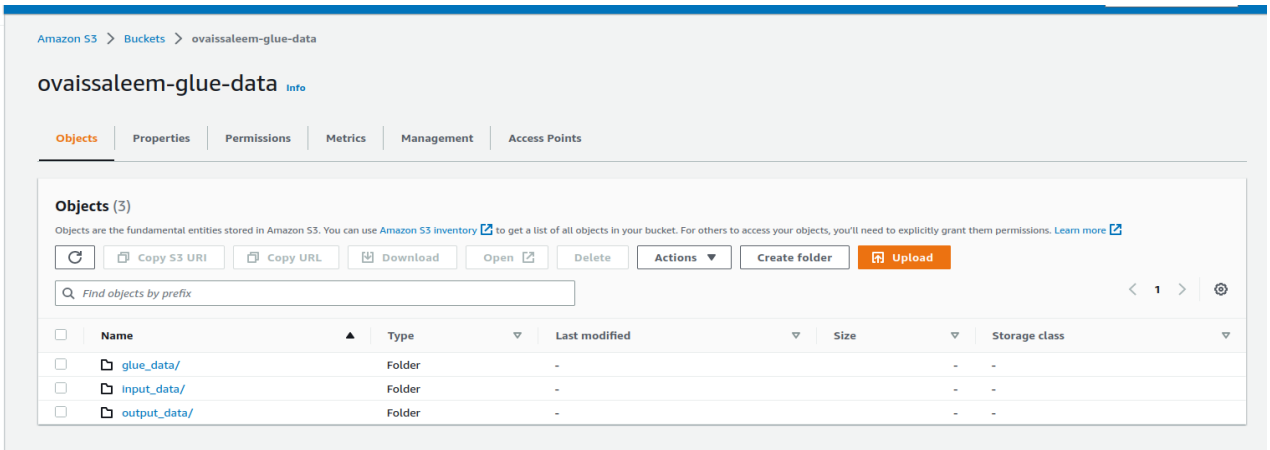
Step 1: Creating a bucket for Glue Job

Here we've created a bucket in AWS-S3 with our name a



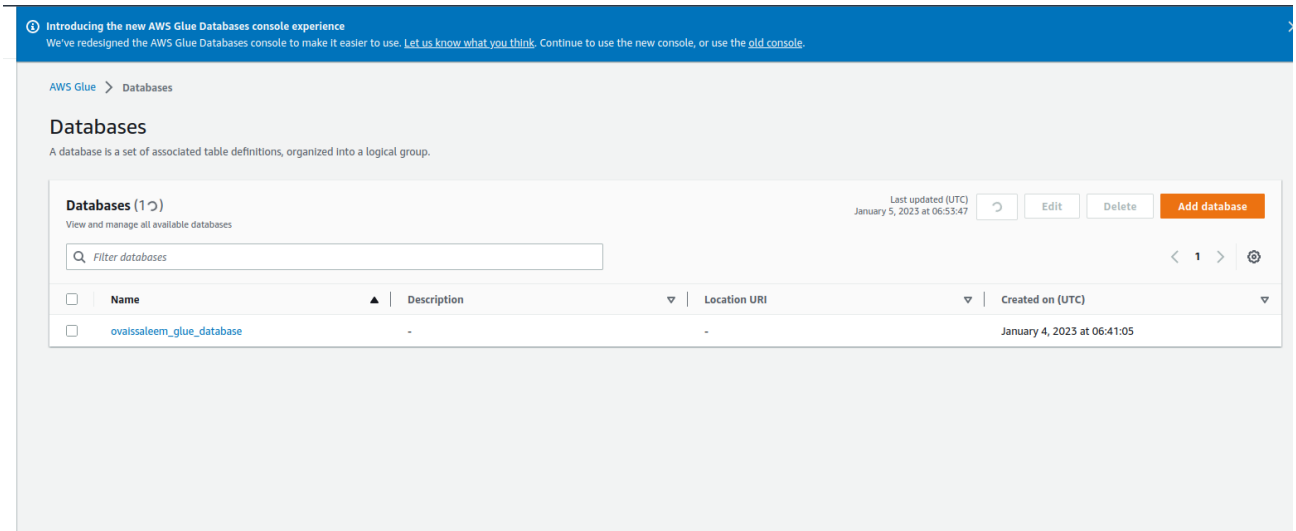
Step 2: Creating directories in bucket

After creating the bucket, we added three directories in it according to the Readme file.



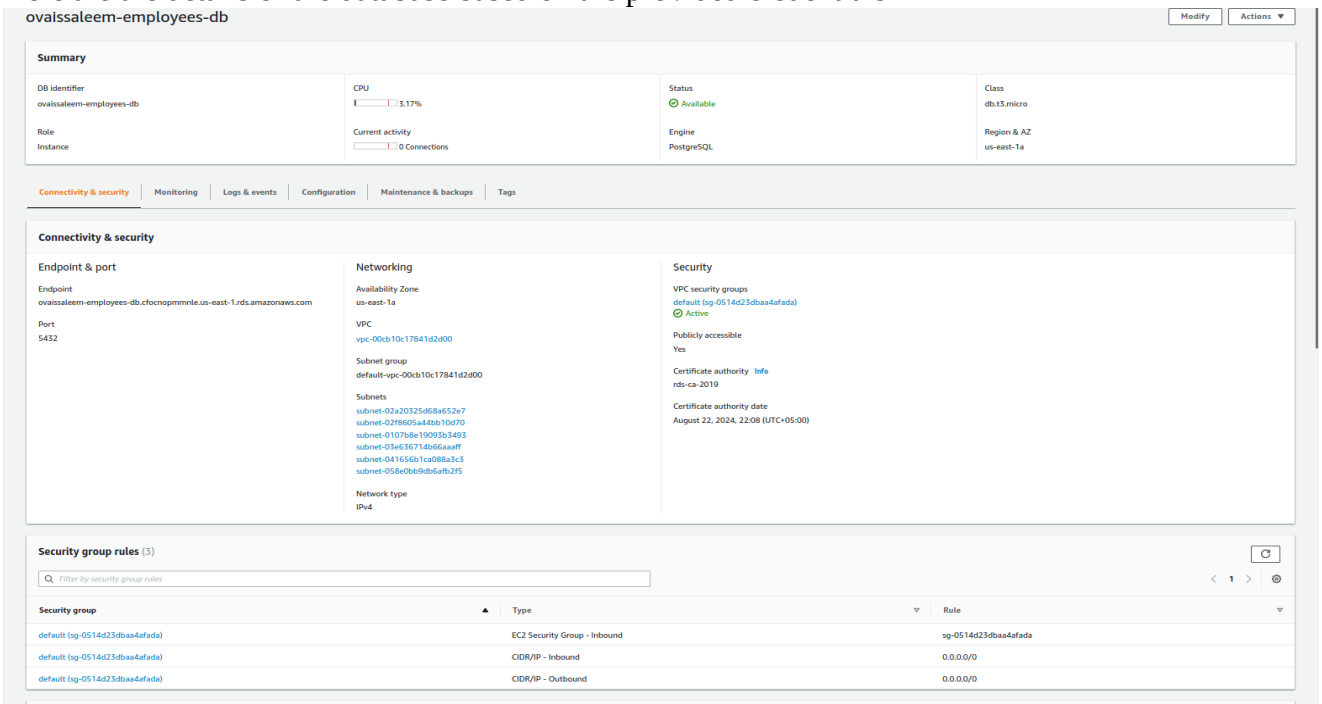
Step 3.1: Creating database in RDS

Here we create a database in RDS.



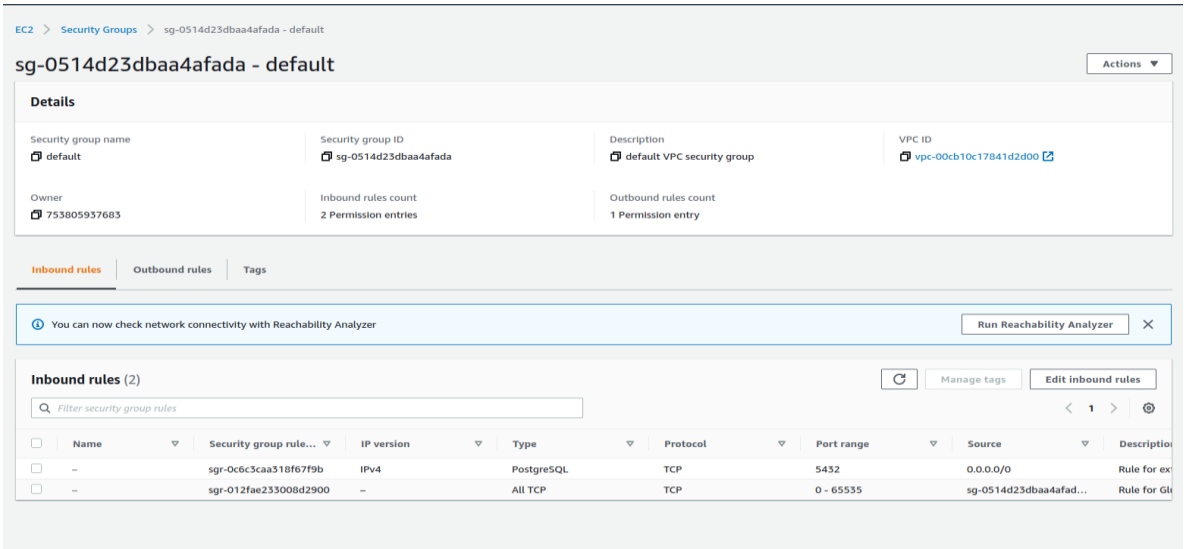
Step 3.2: Database Credentials in RDS

Here are the details of the database based on the provided credentials



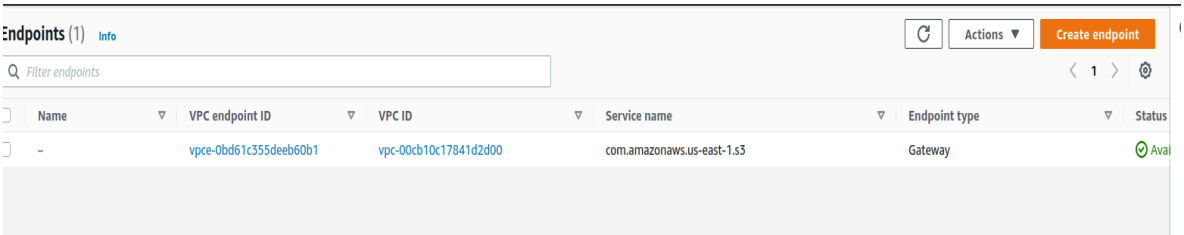
Step 4: Creating Security Groups

Here we create the security group with rules for our database



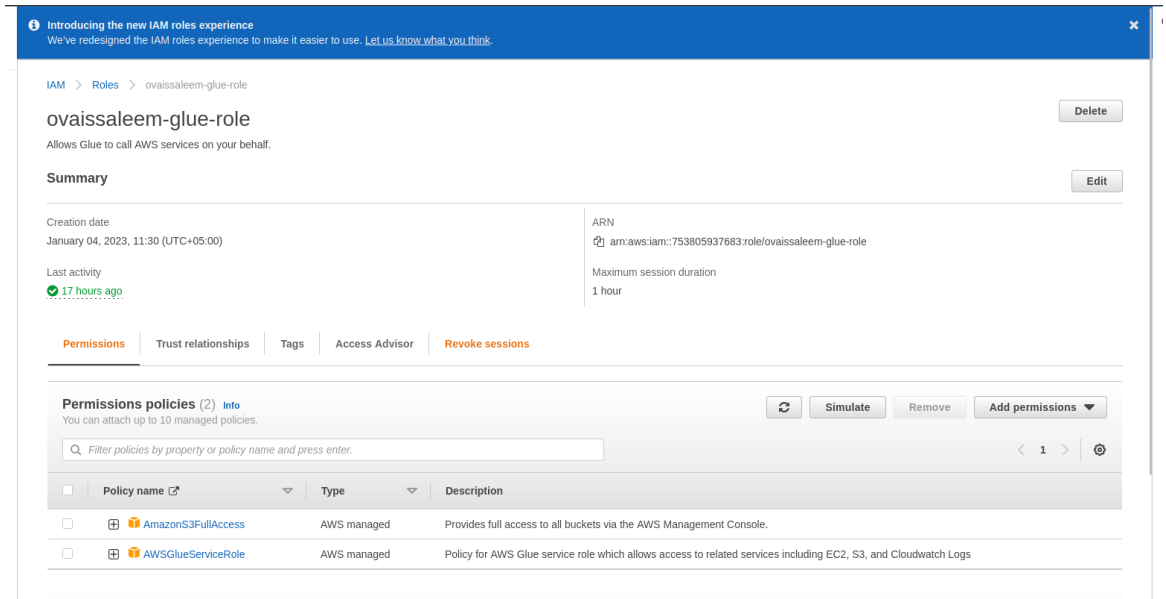
Step 5: Creating Endpoint

Here we created the endpoint for our database



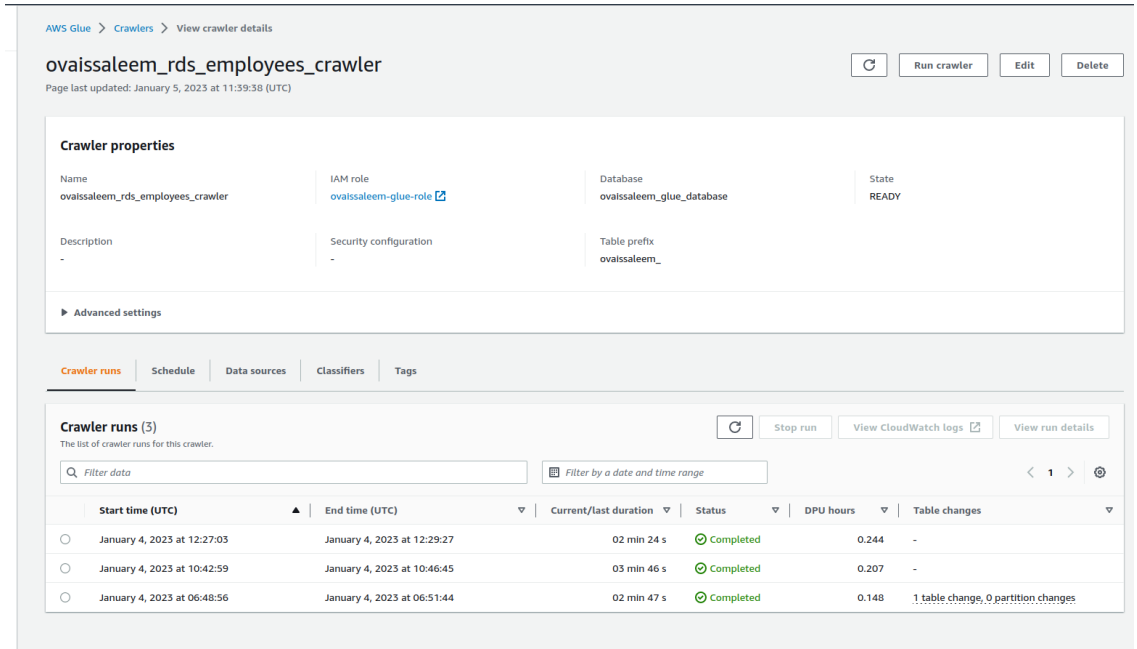
Step 6: Creating Role

Here we create a role from IAMRole in AWS that will be used in crawler.



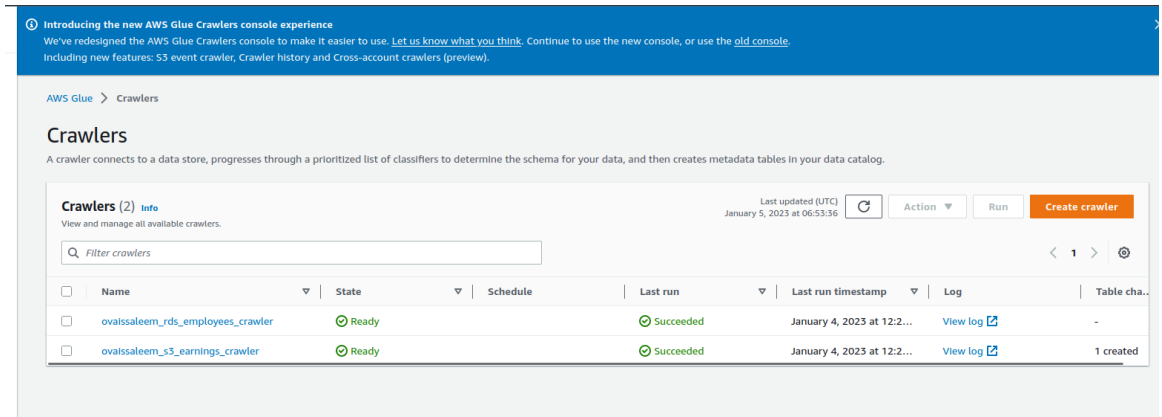
Step 7.1: Creating the crawler

Here we created the crawler with the provided credentials



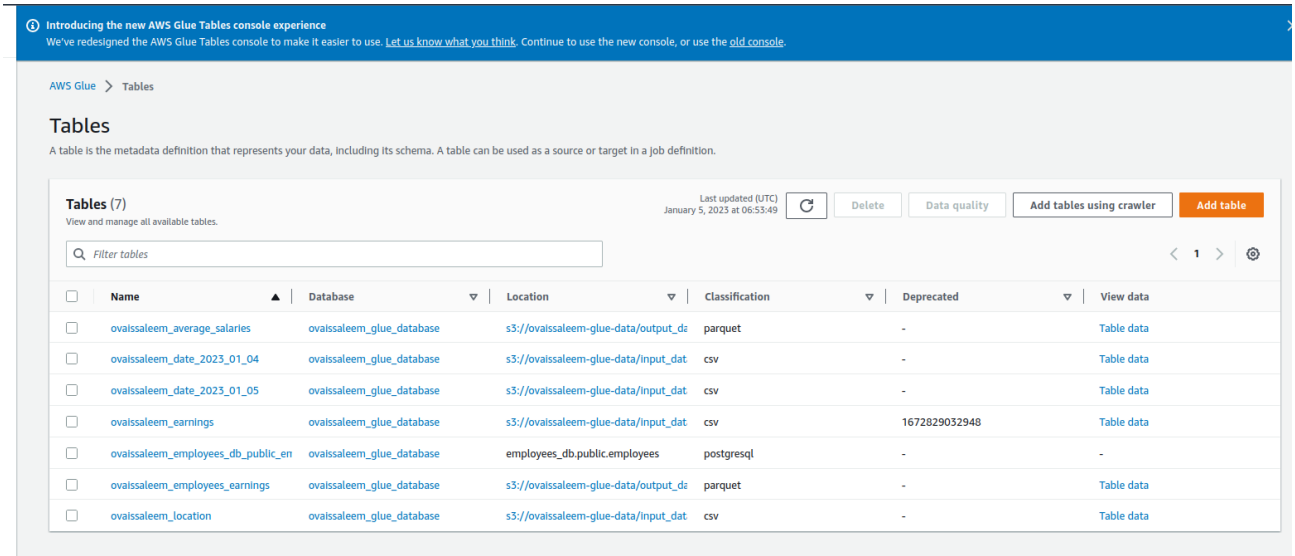
Step 7.2: Running the crawler

Here we can see that the crawler runs successfully for the glue job



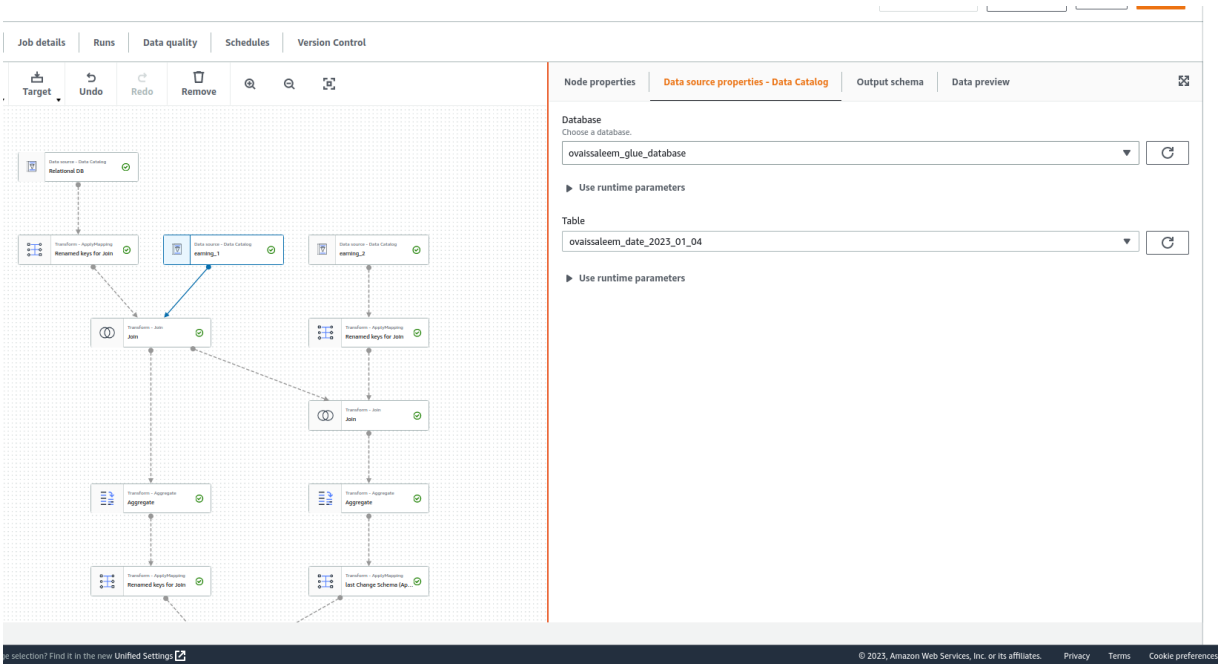
Step 8: Checking the tables

Here we check that the tables are fetched properly



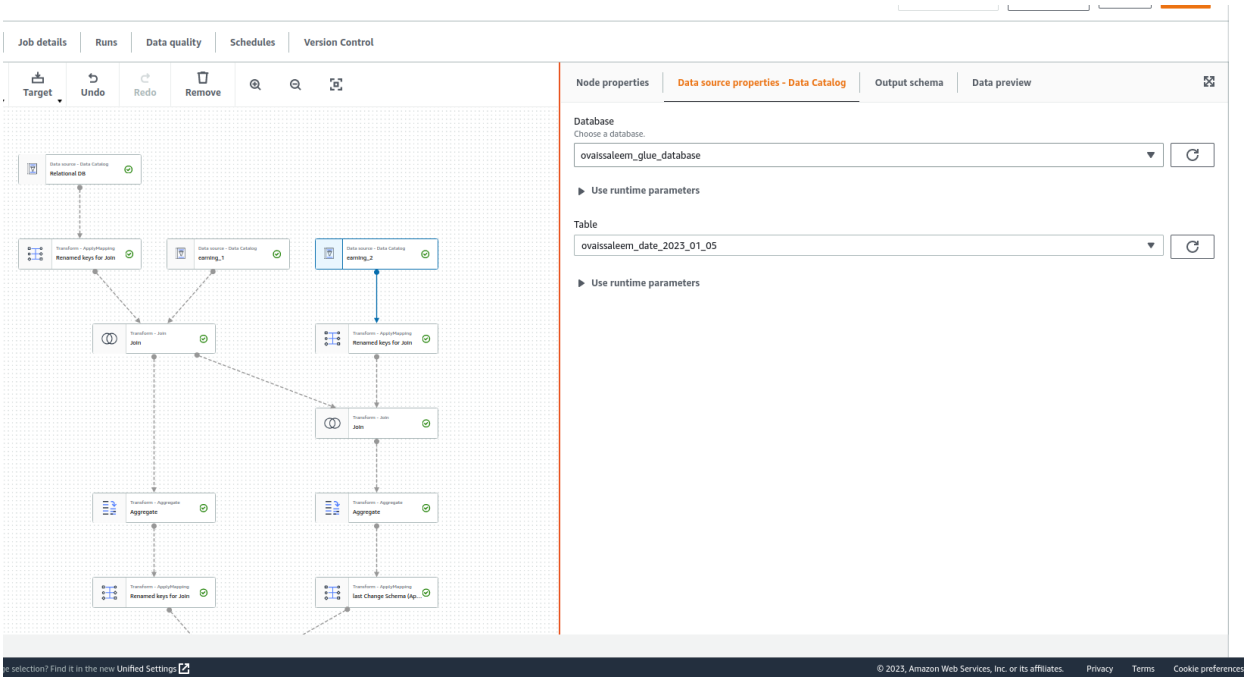
Step 9: Checking the table earning 1

Here we check the values of table earning 1 in query editor



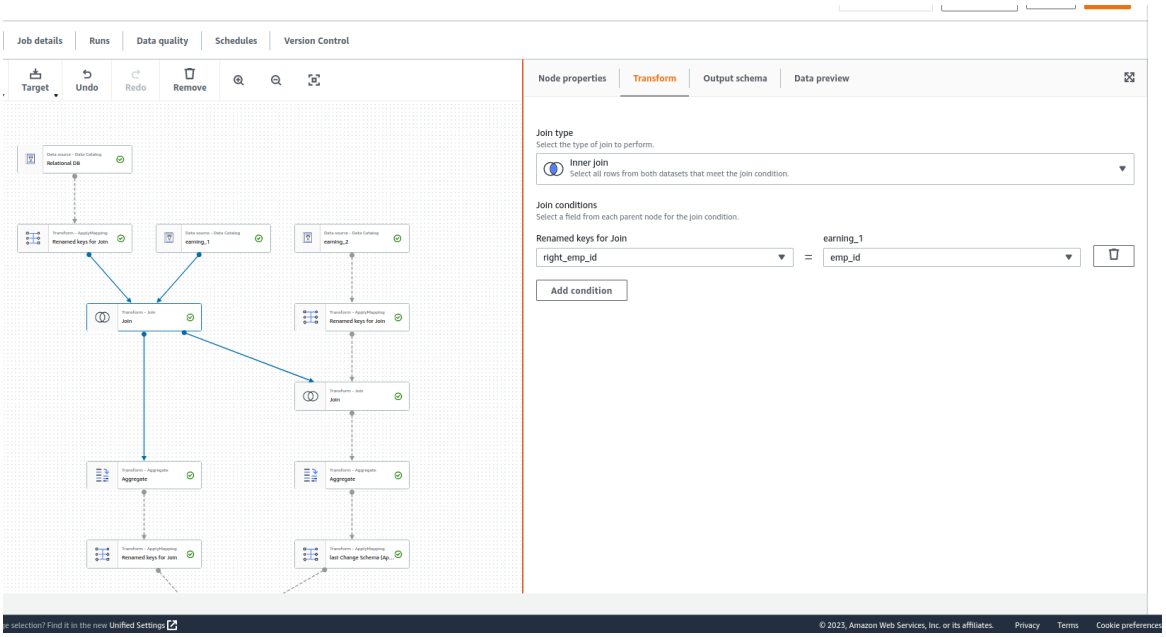
Step 10: Checking the table earning 2

Here we check the values of table earning 2 in query editor



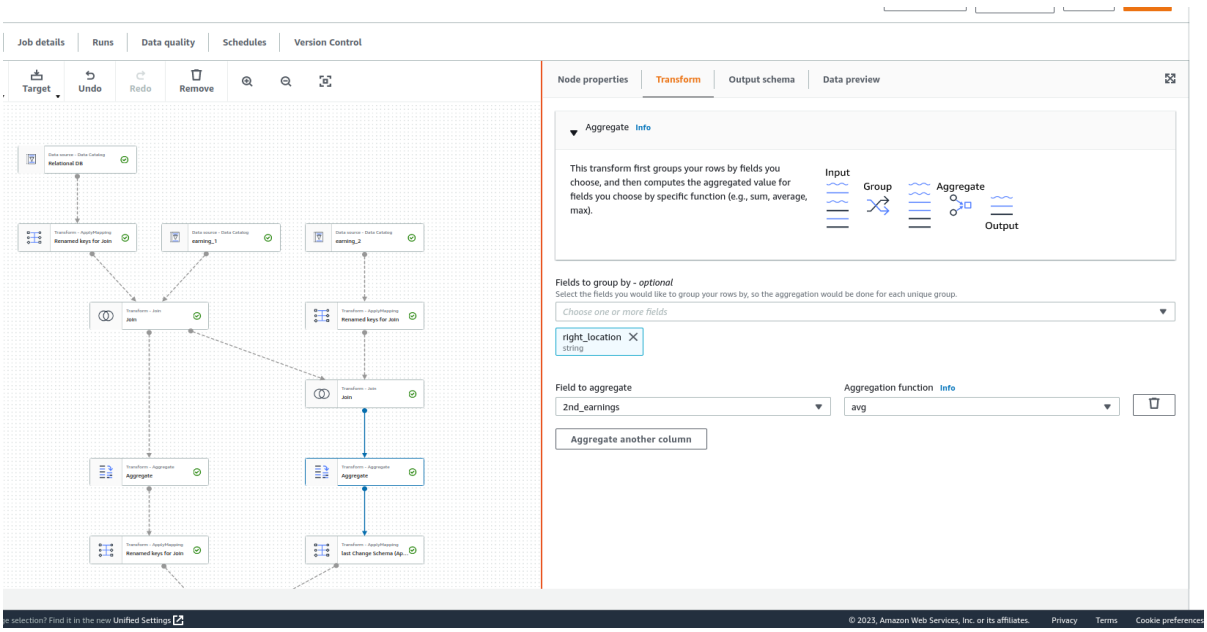
Step 11: Applying Joins

Here multiple joins were applied.



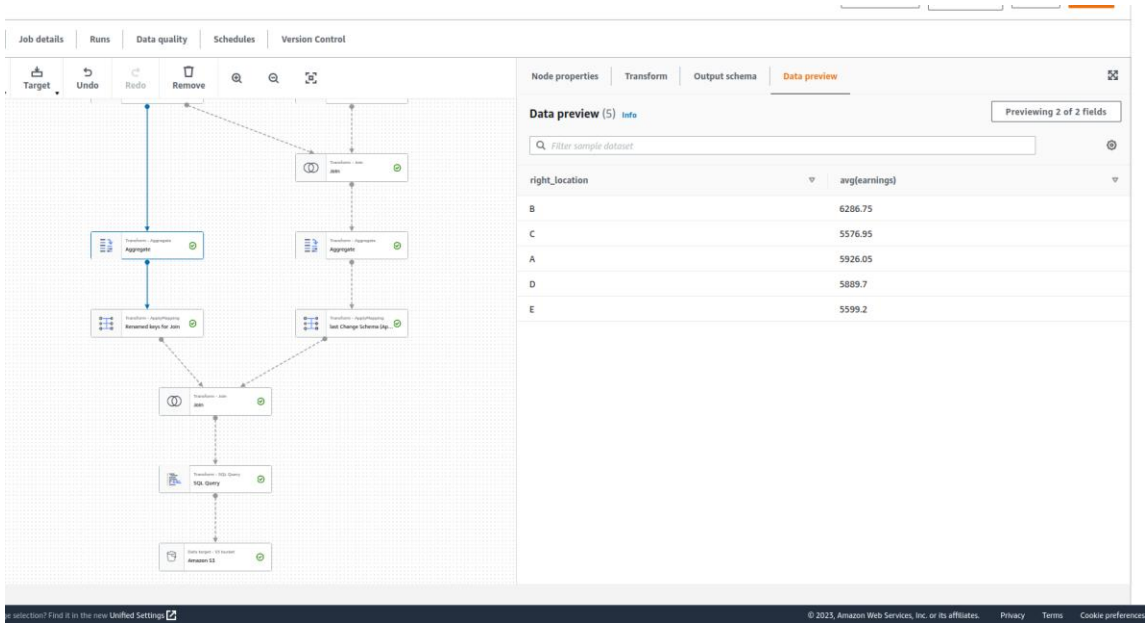
Step 12: Applying Joins and Aggregation

Here multiple joins and aggregation for average were applied.



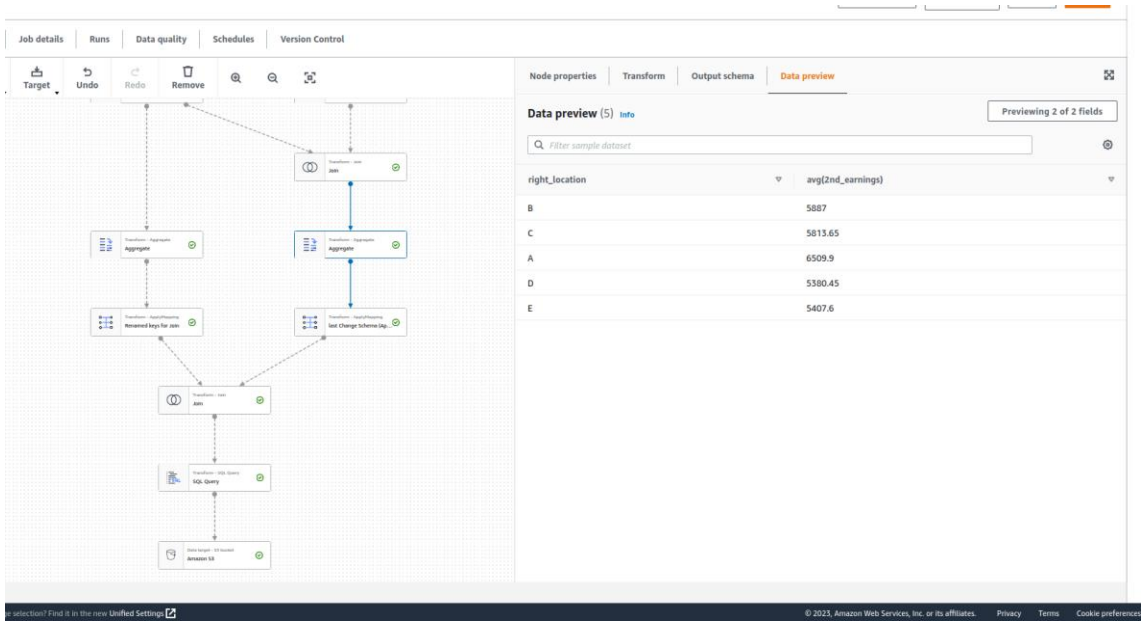
Step 13: Aggregation for earning 1

Here aggregation for earning 1 was applied



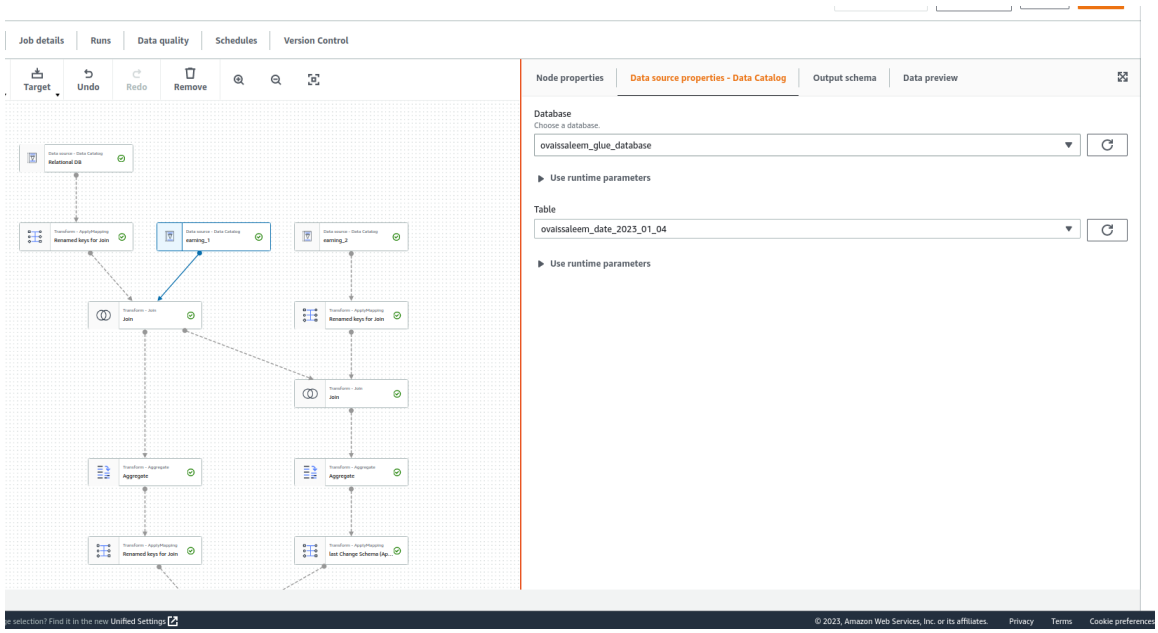
Step 14: Aggregation for earning 2

Here aggregation for earning 2 was applied



Step 15: Combined Aggregation

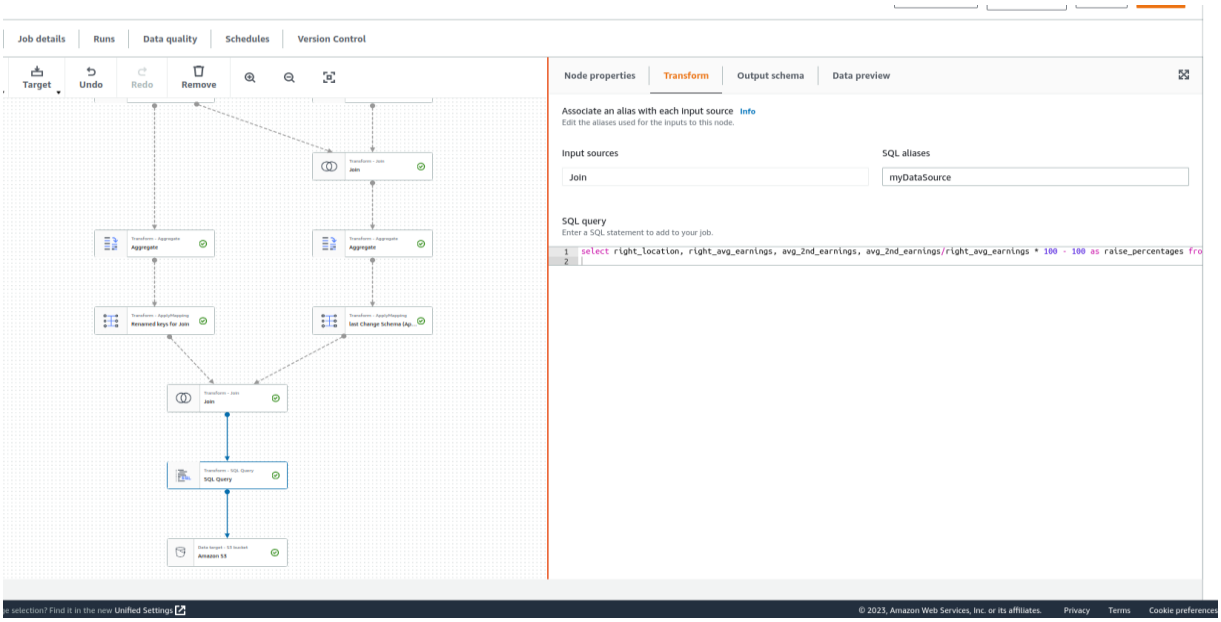
Here combined aggregation of both earnings is shown.



Step 16: Calculating the earning raise

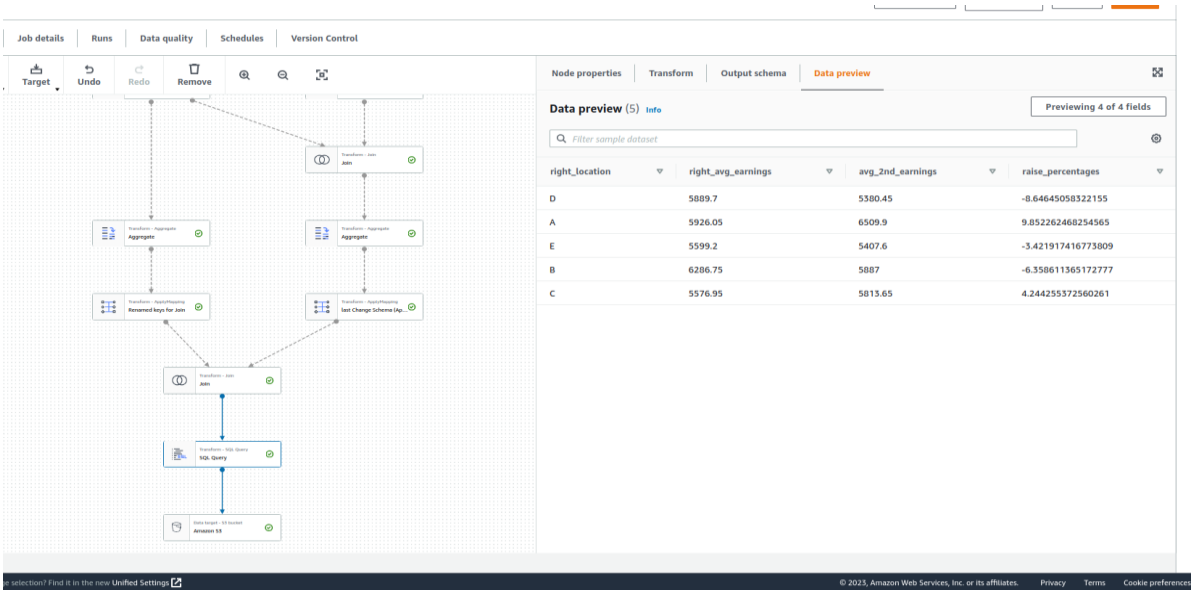
Here we calculated the raise in earnings

Query: select right_location, right_avg_earnings, avg_2nd_earnings, avg_2nd_earnings/right_avg_earnings * 100 - 100 as raise_percentages from myDataSource



Step 17: The raise between earnings

Here we can see the raise in earnings between earning 1 and earning 2



Step 18: The raise between earnings

Here we can see the raise in earnings between earning 1 and earning 2 that increased

Query: select right_location, right_avg_earnings, avg_2nd_earnings, avg_2nd_earnings/right_avg_earnings * 100 - 100 as raise_percentages from myDataSource where (avg_2nd_earnings/right_avg_earnings * 100 - 100) > 0