

Name : Muhammad Safi (2303.khi.deg.023)

Assignment partners:

Huzaifa Ali (2303.khi.deg.016)

Shiekh Muhammad Sabih(2303.KHI.DEG.010)

1. first we add locations.csv file uploaded in the s3 bucket.

Amazon S3 > Buckets > muhammadsafi-glue-data > input_data/ > locations/

locations/ Copy S3 URI

Objects Properties

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 Inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Refresh Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	locations.csv	csv	May 17, 2023, 10:17:33 (UTC+05:00)	916.0 B	Standard

2. Now, we will setup a crawler in order to extract the meta data and generate a catalog.

AWS Glue > Crawlers > muhammadsafi_s3_locations_crawler

muhammadsafi_s3_locations_crawler Last updated (UTC) May 17, 2023 at 06:24:39 Refresh Run crawler Edit Delete

Crawler properties

Name muhammadsafi_s3_locations_crawler	IAM role muhammadsafi-glue-role	Database muhammadsafi_glue_database	State READY
Description -	Security configuration -	Lake Formation configuration -	Table prefix muhammadsafi_
Maximum table threshold -			

Advanced settings

Crawler runs | Schedule | Data sources | Classifiers | Tags

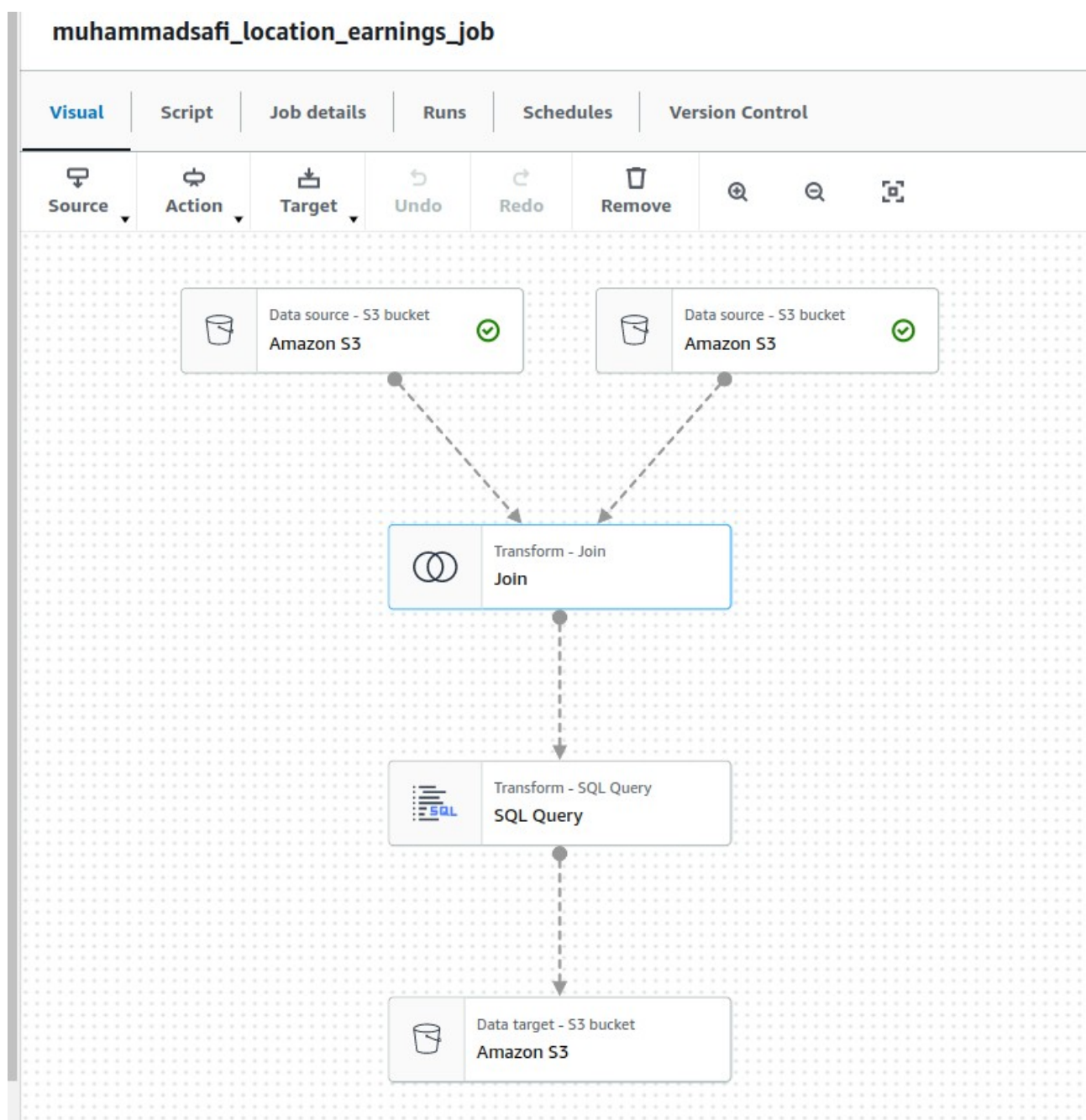
Crawler runs (2)

The list of crawler runs for this crawler.

Filter by a date and time range Refresh Stop run View CloudWatch logs View run details

<input type="radio"/>	Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
<input type="radio"/>	May 17, 2023 at 06:21:44	May 17, 2023 at 06:22:37	53 s	Completed	0.055	-
<input type="radio"/>	May 17, 2023 at 05:51:32	May 17, 2023 at 05:52:12	40 s	Completed	0.080	1 table change, 0 partition changes

3. We'll run the crawler and after its done, we'll start creating the job.



4. The job created here, takes in two tables employee earnings data and other s3 source takes in locations data, we perform an inner join on both data sources on emp_id and prepare the data for querying.

muhammadsafi_location_earnings_job

Last modified on 5/17/2023, 11:20:07 AM [Try new UI](#) [Actions](#) [Save](#) [Run](#)

[Visual](#) [Script](#) [Job details](#) [Runs](#) [Schedules](#) [Version Control](#)

[Source](#) [Action](#) [Target](#) [Undo](#) [Redo](#) [Remove](#) [Find](#) [Find](#) [Refresh](#)

```
graph TD; DS1[Data source - S3 bucket Amazon S3] --> T1[Transform - Join Join]; DS2[Data source - S3 bucket Amazon S3] --> T1; T1 --> T2[Transform - SQL Query SQL Query]; T2 --> DT[Data target - S3 bucket Amazon S3];
```

Transform | **Output schema** | **Data preview**

Name
SQL Query

Node parents
Choose which nodes will provide inputs for this one.
Choose one or more parent nodes

Join
Join - Transform

Associate an alias with each input source [Info](#)
Edit the aliases used for the inputs to this node.

Input sources
Join

SQL aliases
myDataSource

SQL query
Enter a SQL statement to add to your job.

```
1 -- select * from myDataSource
2 SELECT
3   location,
4   AVG(earnings) AS average_earnings,
5   (AVG(earnings) - MIN(earnings)) / MIN(earnings) * 100 AS raise_percentage
6 FROM
7   myDataSource
8 GROUP BY
9   location;
10
```

5. The query here aggregates the data based on location and calculates the salaries and percentages of these locations.

Transform	Output schema	Data preview	
Data preview (5) Info		Previewing 3 of 3 fields	
<input type="text" value="Filter sample dataset"/>			
location	average_earnings	raise_percentage	
B	6086.875	184.30056048575432	
C	5695.3	158.9949977262392	
A	6217.975	205.85218888342354	
D	5635.075	180.91101694915253	
E	5503.4	154.31608133086874	

6. Now, we'll save this data to our s3 bucket in its output and save this job.

AWS Glue > Jobs

AWS Glue Studio [Info](#)

Create job [Info](#) Create

☒ **Visual with a source and target**
Start with a source, ApplyMapping transform, and target.

☐ **Visual with a blank canvas**
Author using an interactive visual interface.

☐ **Spark script editor**
Write or upload your own Spark code.

☐ **Python Shell script editor**
Write or upload your own Python shell script.

☐ **Jupyter Notebook**
Write your own code in a Jupyter Notebook for interactive development.

Source
 Amazon S3
JSON, CSV, or Parquet files stored in S3.

→


Target
 Amazon S3
S3 bucket by specifying a bucket path as the data target.

Your jobs (2) [Info](#) Refresh Actions Run job

<input type="checkbox"/>	Job name	Type	Last modified	AWS Glue version
<input type="checkbox"/>	muhammadsafi_location_earnings_job	Glue ETL	5/17/2023, 11:20:07 AM	3.0
<input type="checkbox"/>	muhammadsafi_employee_earnings_job	Glue ETL	5/16/2023, 4:31:26 PM	3.0

7. After running this job, parquet files compressed with snappy are added to our output location in the s3 bucket.

Here we can see the output schema that is being saved in our output location of s3 bucket.

Data target properties - S3			Output schema	Data preview	
Schema					
Key		Data type		Partition	
location		string		-	
average_earnings		double		-	
raise_percentage		double		-	

Here we can see the saved files of the outputs generated.

Amazon S3 > Buckets > muhammadsaft-glue-data > output_data/ > location_earnings/

location_earnings/

Copy S3 URI

Objects

Properties

Objects (5)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 Inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Refresh

Copy S3 URI

Copy URL

Download

Open






Delete

Actions

Create folder

Upload

Find objects by prefix

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	 run-1684304478447-part-block-0-r-00002-snappy.parquet	parquet	May 17, 2023, 11:21:27 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	 run-1684304478447-part-block-0-r-00014-snappy.parquet	parquet	May 17, 2023, 11:21:26 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	 run-1684304478447-part-block-0-r-00021-snappy.parquet	parquet	May 17, 2023, 11:21:27 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	 run-1684304478447-part-block-0-r-00025-snappy.parquet	parquet	May 17, 2023, 11:21:28 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	 run-1684304478447-part-block-0-r-00031-snappy.parquet	parquet	May 17, 2023, 11:21:26 (UTC+05:00)	599.0 B	Standard