

NAME: MAVIA ALAM KHAN(2303.KHI.DEG.017)

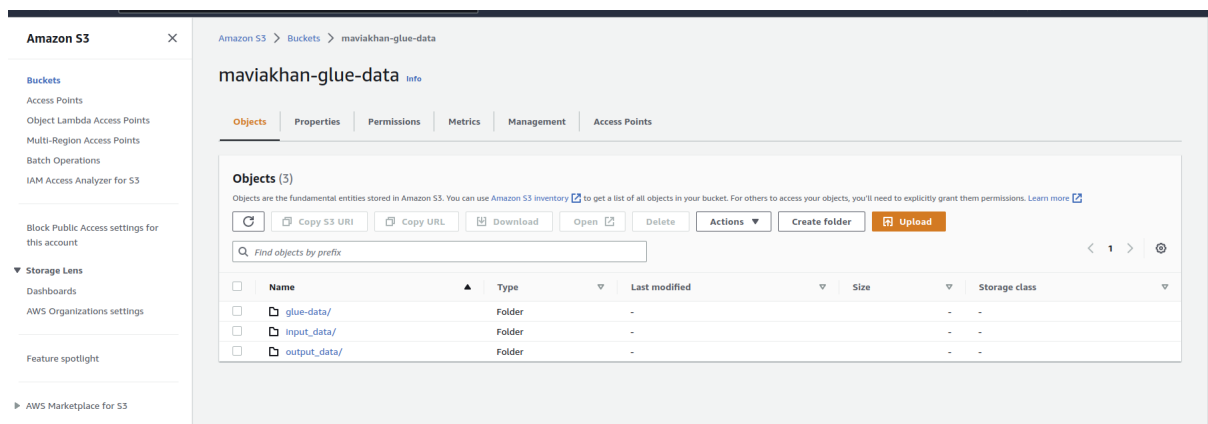
PAIRING WITH :Mohammad Hussam (2303.KHI.DEG.020)

&

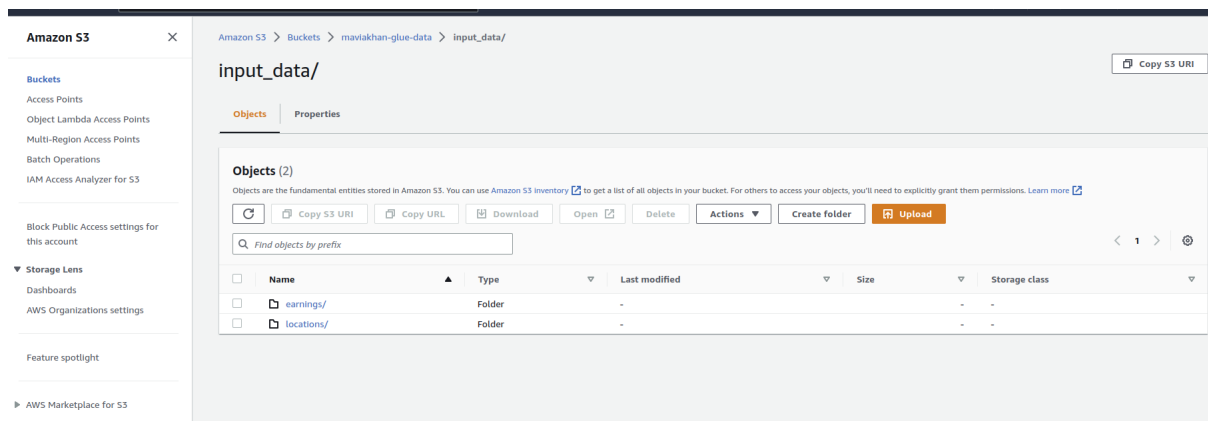
AQSA TAUHEED(2303.KHI.DEG.011)

ASSIGNMENT NO :5.2

First create a folder in s3 bucket name (input and output)



In input folder we store a two dataset earning.csv and location.csv



Now we create a crawler and extract the meta data

AWS Glue

Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)

Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings

Data Integration and ETL
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Interactive Sessions
Data classification tools
Sensitive data detection
Record Matching

Crawler successfully starting
The following crawler is now starting: "maviakhan_s3_earnings_crawler"

mavia_combined_employee_earnings_crawler

Last updated (UTC)
May 19, 2023 at 20:56:09

Crawler properties

Name mavia_combined_employee_earnings_crawler	IAM role maviakhan-glue-role	Database mavia-glue-database	State READY
Description -	Security configuration -	Lake Formation configuration -	Table prefix mavia_
Maximum table threshold -			

Advanced settings

Crawler runs | Schedule | Data sources | Classifiers | Tags

Crawler runs (5)
The list of crawler runs for this crawler.

Filter data | Filter by a date and time range

	Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
	May 19, 2023 at 16:52:55	May 19, 2023 at 16:53:44	48 s	Completed	0.065	1 table change, 0 partition changes
	May 19, 2023 at 11:55:06	May 19, 2023 at 11:55:56	49 s	Completed	0.067	3 table changes, 0 partition changes

AWS Glue

Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)

Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings

Data Integration and ETL
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Interactive Sessions
Data classification tools
Sensitive data detection
Record Matching

Crawler successfully starting
The following crawler is now starting: "maviakhan_s3_earnings_crawler"

maviakhan_s3_earnings_crawler

Last updated (UTC)
May 19, 2023 at 20:56:35

Crawler properties

Name maviakhan_s3_earnings_crawler	IAM role maviakhan-glue-role	Database mavia-glue-database	State READY
Description -	Security configuration -	Lake Formation configuration -	Table prefix maviakhan
Maximum table threshold -			

Advanced settings

Crawler runs | Schedule | Data sources | Classifiers | Tags

Crawler runs (2)
The list of crawler runs for this crawler.

Filter data | Filter by a date and time range

	Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
	May 19, 2023 at 16:55:51	May 19, 2023 at 16:56:41	49 s	Completed	0.060	1 table change, 1 partition change
	May 19, 2023 at 04:26:04	May 19, 2023 at 04:26:54	50 s	Completed	0.068	1 table change, 1 partition change

And after create the glue crawler we start creating the job

AWS Glue

Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)

Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings

Data Integration and ETL
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Interactive Sessions
Data classification tools
Sensitive data detection
Record Matching
Triggers
Workflows (orchestration)
Blueprints

Visual | Script | Job details | Runs | Schedules | Version Control

Source | Action | Target | Undo | Redo | Remove

Data source properties - S3 | Output schema | Data preview

Name
AmazonLocation

S3 source type
S3 location
Choose a file or folder in an S3 bucket.
Data Catalog table

S3 URL
Recursive
Read files in all subdirectories.
Data format
CSV
Delimiter
Comma (,)
Escape character - optional
Enter a character to use for escaping
The character which immediately follows is used as-is, except for a small set of well-known escapes (\n, \r, \t, and \0)
Quote character
Double quote (")
First line of source file contains column headers

Workflow Diagram:

```
graph TD
    A[Data source - S3 bucket AmazonLocation] --> B[Transform - Join Join]
    C[Data source - S3 bucket AmazonEarning] --> B
    B --> D[Transform - SQL Query SQL Query]
    D --> E[Data target - S3 bucket datatarget]
```

Now we create two s3 source one is employee earning data and location,we perform inner join on both data on emp_id and after that prepare for querying

The screenshot shows the AWS Glue console interface for a workflow named 'Assignmnt_5.2'. The 'Visual' tab is active, displaying a workflow diagram with two data sources: 'Data source - S3 bucket AmazonLocation' and 'Data source - S3 bucket AmazonEarning'. These sources are connected to a 'Transform - Join' node. Below the join node is a 'Transform - SQL Query' node, which is connected to a 'Data target - S3 bucket datatarget' node. The right-hand pane shows the configuration for the 'Join' node. The 'Name' field is set to 'Join'. Under 'Node parents', the 'AmazonLocation' and 'AmazonEarning' nodes are selected. A warning message states: 'The parents of this node have overlapping field names. AWS Glue Studio can add an Apply Mapping node to rename them and avoid downstream issues.' The 'Join type' is set to 'Inner join'. Under 'Join conditions', the 'emp_id' field from both parent nodes is selected with an equals sign (=) between them. The 'Transform' tab is active in the right-hand pane.

The screenshot shows the AWS Glue console interface for the same workflow 'Assignmnt_5.2'. The 'Visual' tab is active, showing the workflow diagram. The 'Transform - SQL Query' node is now selected. The right-hand pane shows the configuration for this node. The 'Name' field is set to 'SQL Query'. Under 'Node parents', the 'Join' node is selected. The 'Input sources' section shows 'Join' as the source and 'myDataSource' as the SQL alias. The 'SQL query' section contains the following SQL statement:

```

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

```

The 'Transform' tab is active in the right-hand pane.

The screenshot shows the AWS Glue console interface for the same workflow 'Assignmnt_5.2'. The 'Visual' tab is active, showing the workflow diagram. The 'Data target - S3 bucket datatarget' node is now selected. The right-hand pane shows the configuration for this node. The 'Name' field is set to 'datatarget'. Under 'Node parents', the 'SQL Query' node is selected. The 'Format' is set to 'Parquet' and the 'Compression Type' is set to 'Snappy'. The 'S3 Target Location' section shows the path 's3://maviakhan-glue-data/output_data/earningswithLocatio'. The 'Data Catalog update options' section shows the option 'Create a table in the Data Catalog and on subsequent runs, update the schema and add new partitions' selected. The 'Data target properties - S3' tab is active in the right-hand pane.

Now here the queries based on the salaries and percentage of these locations.

Last modified on 5/19/2023, 10:26:41 PM

Try new UI

End session

Actions

Save

Run

ules

Version Control

Remove

Data source - S3 bucket

AmazonEarning

Transform

Output schema

Data preview

Data preview (5)

Info

Previewing 3 of 3 fields

Filter sample dataset

location	average_earnings	raise_percentage
B	6286.75	155.14407467532467
C	5576.95	129.78780387309433
A	5926.05	191.49286768322676
D	5889.7	185.07744433688285
E	5599.2	158.74306839186693

Now finally we load the data and show in the output folder

Amazon S3

Buckets

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

AWS Organizations settings

Feature spotlight

AWS Marketplace for S3

Amazon S3 > Buckets > maviakhan-glue-data > output_data/ > earningswithLocationTarget/

earningswithLocationTarget/

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](#)

Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

Find objects by prefix

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	run-1684517258119-part-block-0-r-00002-snappy.parquet	parquet	May 19, 2023, 22:27:46 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	run-1684517258119-part-block-0-r-00014-snappy.parquet	parquet	May 19, 2023, 22:27:45 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	run-1684517258119-part-block-0-r-00021-snappy.parquet	parquet	May 19, 2023, 22:27:45 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	run-1684517258119-part-block-0-r-00025-snappy.parquet	parquet	May 19, 2023, 22:27:45 (UTC+05:00)	599.0 B	Standard
<input type="checkbox"/>	run-1684517258119-part-block-0-r-00031-snappy.parquet	parquet	May 19, 2023, 22:27:44 (UTC+05:00)	599.0 B	Standard