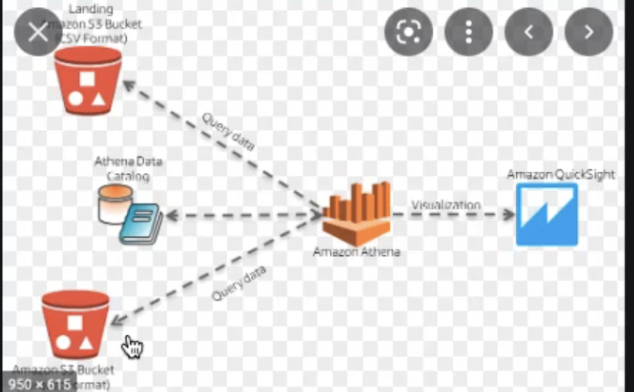
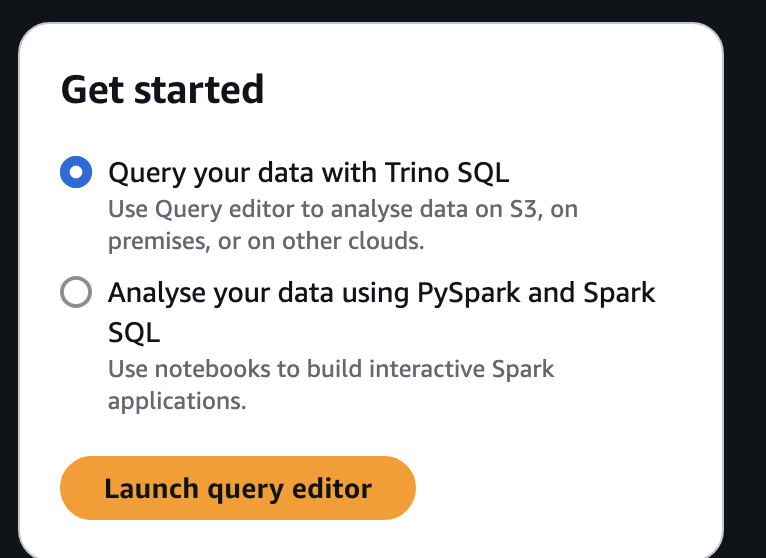
AWS Athena:

* Cost effective interactive query service
* We can create db and table via query editor as well
* Athena is built on top of presto
* 
* 

CREATE external TABLE `cust-db`.hqltb (

customerid string,

namestyle string,

title string,

firstname string,

middlename string,

lastname string,

suffix string,

companyname string,

salesperson string,

emailaddress string,

phone string,

passwordhash string,

passwordsalt string,

rowguid string,

modifieddate date)

PARTITIONED BY (

year string,

month string,

day string)

stored as parquet

LOCATION 's3://atul.data/hql-data/'

insert into hqltb select \* from parquetparquet\_landing;

Athena Commands:

aws athena help

aws athena list-databases help

aws athena list-databases \

--catalog-name AwsDataCatalog \

--region us-east-1

When we run queries from CLI certain workgroup setting needs to be enabled else it will not work

An **Athena Workgroup** is a feature in Amazon Athena that allows you to manage queries and their settings in a more organized way. Workgroups enable you to control query execution settings, monitor costs, and apply governance across users and workloads.

**Key Features of Athena Workgroups:**

1. **Query Segregation**:
   * Queries can be grouped based on use cases, teams, or environments (e.g., development, staging, production).
   * Each workgroup runs queries independently of others, ensuring that one group’s workload doesn't impact others.
2. **Settings Enforcement**:
   * You can configure specific settings for a workgroup, such as enabling/disabling query result caching or setting limits on the amount of data scanned per query.
3. **Cost Control**:
   * You can monitor and control costs at the workgroup level by tracking the amount of data scanned by queries.
   * Set thresholds to alert when a workgroup exceeds its predefined budget.
4. **Query Logs**:
   * Workgroups can be configured to send query logs to Amazon S3 for monitoring and debugging purposes.
5. **Permissions Management**:
   * Permissions for users and roles can be defined at the workgroup level to restrict or allow access to specific resources.

[Commands] Get help and list Athena databases using AWS CLI

aws athena help?

aws athena list-databases help

aws athena list-databases \

--catalog-name AwsDataCatalog \

--region us-east-1

[Commands] Managing Athena Workgroups using AWS CLI

aws athena list-work-groups \

--region us-east-1

aws athena get-work-group \

--work-group primary \

--region us-east-1

[Commands] Run Athena Queries using AWS CLI

aws athena start-query-execution help

aws athena start-query-execution \

--query-string "SELECT count(\*) FROM myretail.orders" \

--region us-east-1

# Capture query execution id

aws athena get-query-execution help

aws athena get-query-execution \

--query-execution-id 4631659e-e58d-49b8-b83e-8d137719d72e \

--region us-east-1

[Commands] Get Athena Table Metadata using AWS CLI

aws athena list-table-metadata help

aws athena list-table-metadata \

--catalog-name AwsDataCatalog \

--database-name myretail \

--region us-east-1

aws athena get-table-metadata help

aws athena get-table-metadata \

--catalog-name AwsDataCatalog \

--database-name myretail \

--table-name orders \

--region us-east-1

[Commands] Run Athena Queries with custom location

aws athena start-query-execution help

aws athena start-query-execution \

--query-string "SELECT count(\*) FROM myretail.orders" \

--result-configuration OutputLocation=s3://itv-retail/myretail/order\_count\_awscli \

--region us-east-1

aws athena get-query-execution \

--query-execution-id e303ff4d-bb46-4018-9c78-dc9bff90f718 \

--region us-east-1

[Commands] Drop Athena table using AWS CLI

aws athena start-query-execution \

--query-string "DROP TABLE myretail.order\_items" \

--region us-east-1

aws athena get-query-execution \

--query-execution-id 34a18dde-b71a-4ea3-847f-cb10bc9b5c5a \

--region us-east-1

aws s3 ls s3://itv-retail/myretail/order\_items

aws s3 rm s3://itv-retail/myretail/order\_items/ --recursive

aws s3 ls s3://itv-retail/myretail/

[Commands] Run CTAS under Athena using AWS CLI

aws athena start-query-execution \

--query-string "CREATE TABLE myretail.order\_items \

WITH ( \

format = 'TEXTFILE', \

external\_location = 's3://itv-retail/myretail/order\_items/', \

field\_delimiter = ',' \

) \

AS \

SELECT \* FROM retail\_db.order\_items" \

--region us-east-1

aws athena get-query-execution \

--query-execution-id 9eb94e4e-1613-43e7-a933-00453b169cf1 \

--region us-east-1

aws s3 ls s3://itv-retail/myretail/order\_items/

[Code] Getting Started with Managing Athena using Python boto3

import boto3

import os

os.environ.setdefault('AWS\_DEFAULT\_REGION', 'us-east-1')

athena\_client = boto3.client('athena')

athena\_client.list\_work\_groups?

athena\_client.list\_work\_groups()

[Code] List Amazon Athena Databases using Python boto3

import boto3

import os

os.environ.setdefault('AWS\_DEFAULT\_REGION', 'us-east-1')

athena\_client = boto3.client('athena')

athena\_client.list\_databases?

databases = athena\_client.list\_databases(CatalogName='AwsDataCatalog')

type(databases)

type(databases['DatabaseList'])

[database['Name'] for database in databases['DatabaseList']]

[Code] List Amazon Athena Tables using Python boto3

import boto3

import os

os.environ.setdefault('AWS\_DEFAULT\_REGION', 'us-east-1')

athena\_client = boto3.client('athena')

# Get metadata for all tables in given database

athena\_client.list\_table\_metadata?

athena\_client.list\_table\_metadata(

CatalogName='AwsDataCatalog',

DatabaseName='retail\_db'

)

tables = athena\_client.list\_table\_metadata(

CatalogName='AwsDataCatalog',

DatabaseName='retail\_db'

)

type(tables)

tables['TableMetadataList']

[table['Name'] for table in tables['TableMetadataList']]

# Get metadata for a given table in given database

athena\_client.get\_table\_metadata?

athena\_client.get\_table\_metadata(

CatalogName='AwsDataCatalog',

DatabaseName='retail\_db',

TableName='orders'

)

table\_metadata = athena\_client.get\_table\_metadata(

CatalogName='AwsDataCatalog',

DatabaseName='retail\_db',

TableName='orders'

)

table\_metadata

table\_metadata['TableMetadata']['TableType']

table\_metadata['TableMetadata']['Columns']

table\_metadata['TableMetadata']['Parameters']['location']

# Processing metadata of all the tables returned by list\_table\_metadata

tables = athena\_client.list\_table\_metadata(

CatalogName='AwsDataCatalog',

DatabaseName='retail\_db'

)

[table['Parameters']['location'] for table in tables['TableMetadataList']]

[Code] Run Amazon Athena Queries using Python boto3

import boto3

import os

os.environ.setdefault('AWS\_DEFAULT\_REGION', 'us-east-1')

athena\_client = boto3.client('athena')

athena\_client.start\_query\_execution?

athena\_client.list\_work\_groups()

athena\_client.get\_work\_group(WorkGroup='primary')

query\_execution = athena\_client.start\_query\_execution(

QueryString='SELECT count(\*) FROM myretail.orders'

)

query\_execution

athena\_client.get\_query\_execution(

QueryExecutionId=query\_execution['QueryExecutionId']

)

# s3 command to validate creation of the file using default output location

aws s3 ls s3://itvathena/wgprimary/79e570cb-6d51-4408-84f4-bc81556ec9d0.csv

[Code] Review Athena Query Results using Python boto3

import boto3

import os

os.environ.setdefault('AWS\_DEFAULT\_REGION', 'us-east-1')

athena\_client = boto3.client('athena')

athena\_client.get\_table\_metadata(

CatalogName='AwsDataCatalog',

DatabaseName='myretail',

TableName='orders'

)

query\_str = """

SELECT order\_status, count(\*) AS order\_count

FROM myretail.orders

GROUP BY order\_status

"""

query\_execution = athena\_client.start\_query\_execution(

QueryString=query\_str

)

athena\_client.get\_query\_execution(

QueryExecutionId=query\_execution['QueryExecutionId']

)

query\_results = athena\_client.get\_query\_results(

QueryExecutionId=query\_execution['QueryExecutionId']

)

query\_results

query\_results['ResultSet']

query\_results['ResultSet']['Rows']