Lesson 04 Demo 10

Querying S3 Bucket Operations with Athena

Objective: To execute a query within AWS Athena for performing operations on a designated S3 bucket

Tools required: None

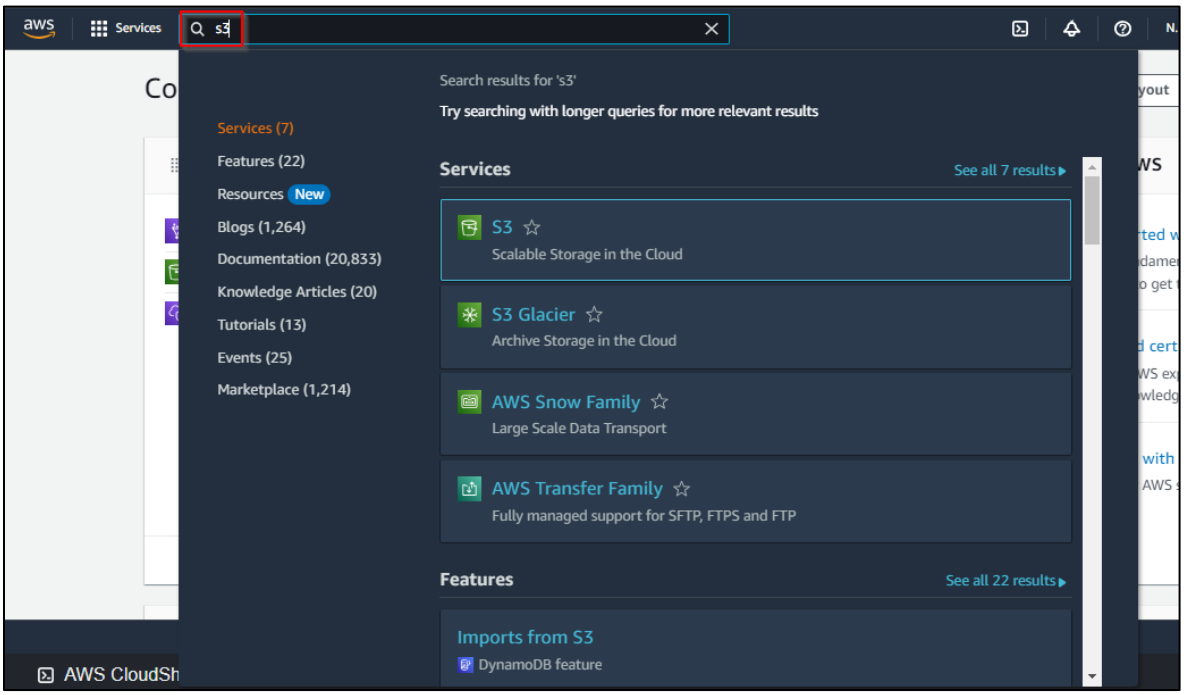
Prerequisites: AWS Lab access with an AWS account created

Steps to be followed:

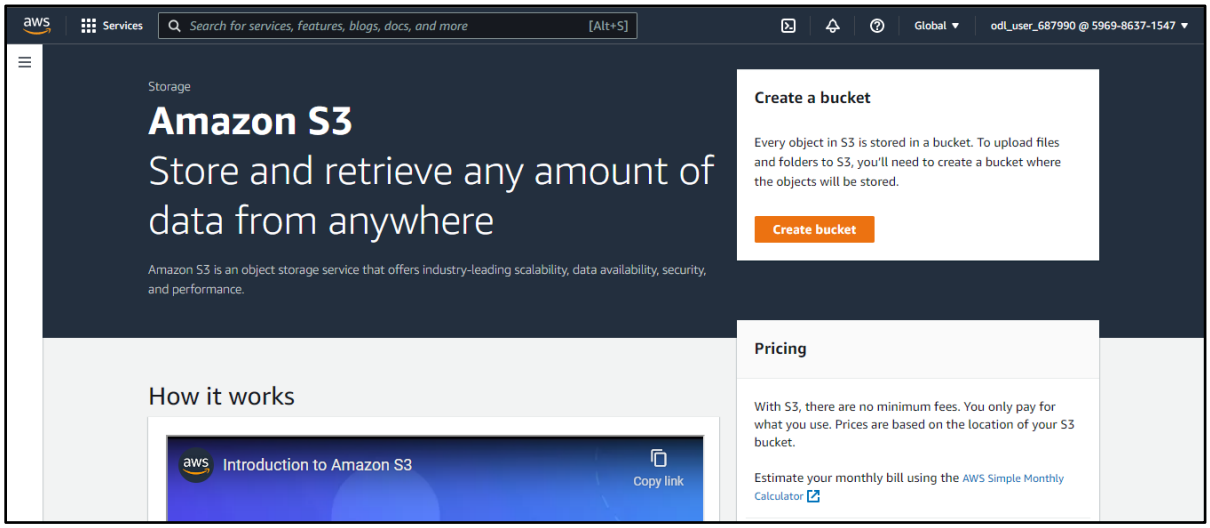
1. Create S3 buckets
2. Execute queries in Athena

Step 1: Create S3 buckets

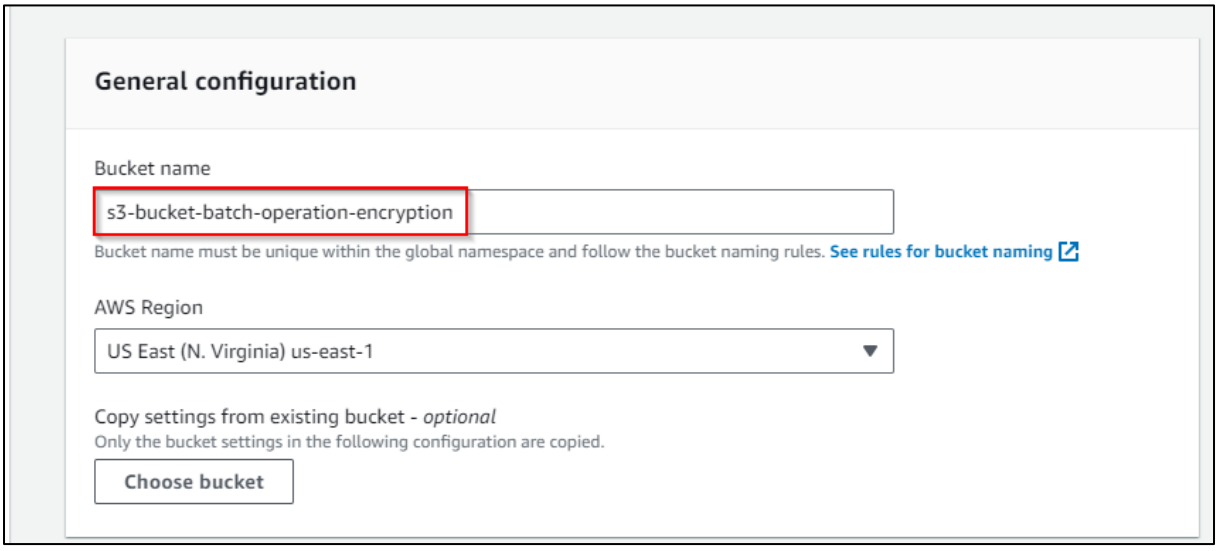
1.1 Open the AWS Management Console and search for **S3 bucket**

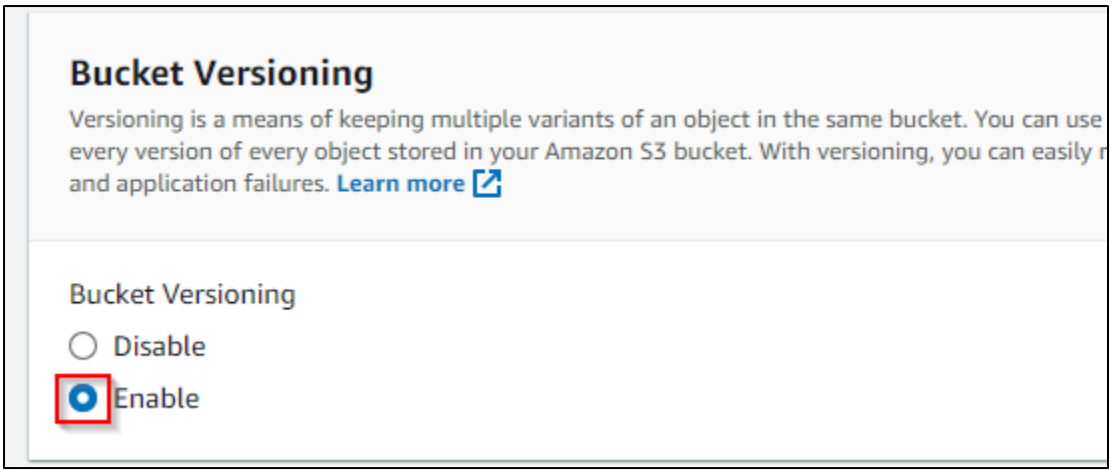


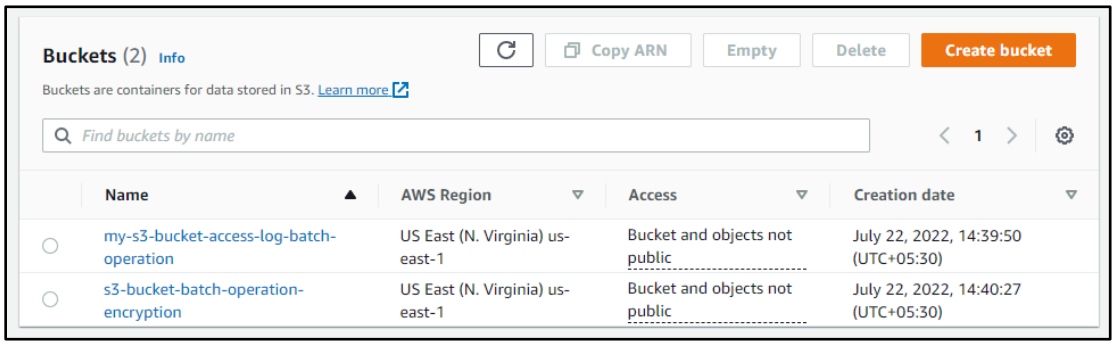
1.2 Click on **Create bucket**



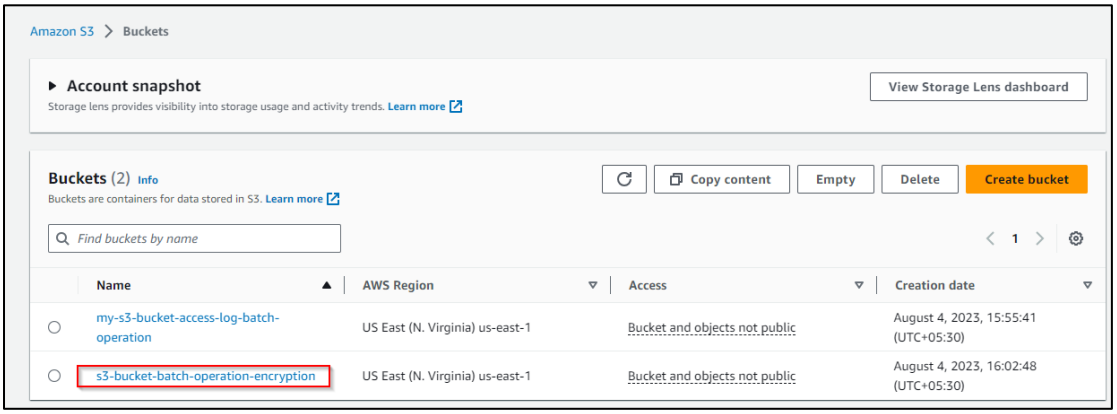
1.3 Create a bucket named **my-s3-bucket-access-log-batch-operation** to aggregate log files from the **s3-bucket-batch-operation-encryption** bucket. Enable **Bucket versioning**.



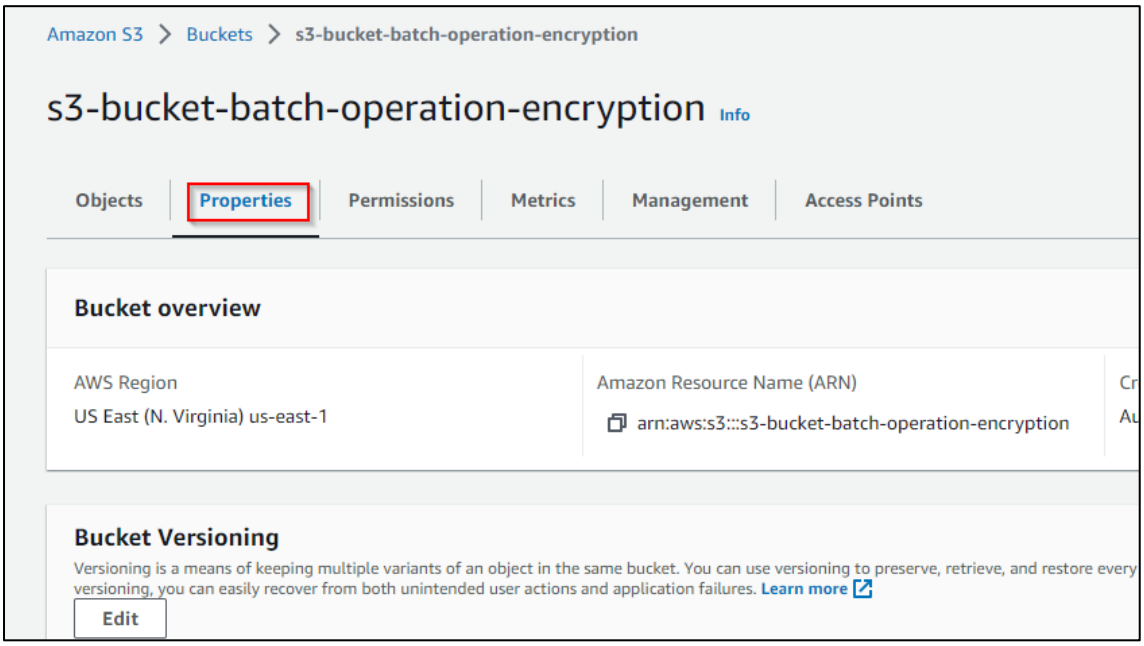




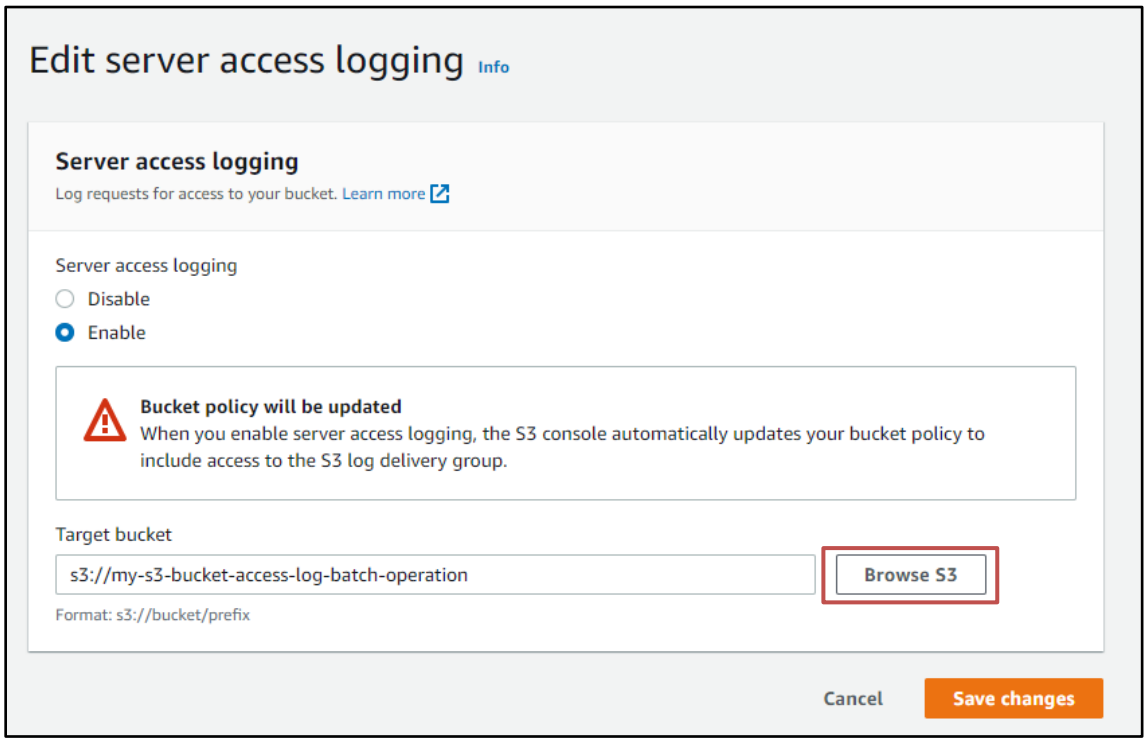
1.4 Access the **s3-bucket-batch-operation-encryption** bucket



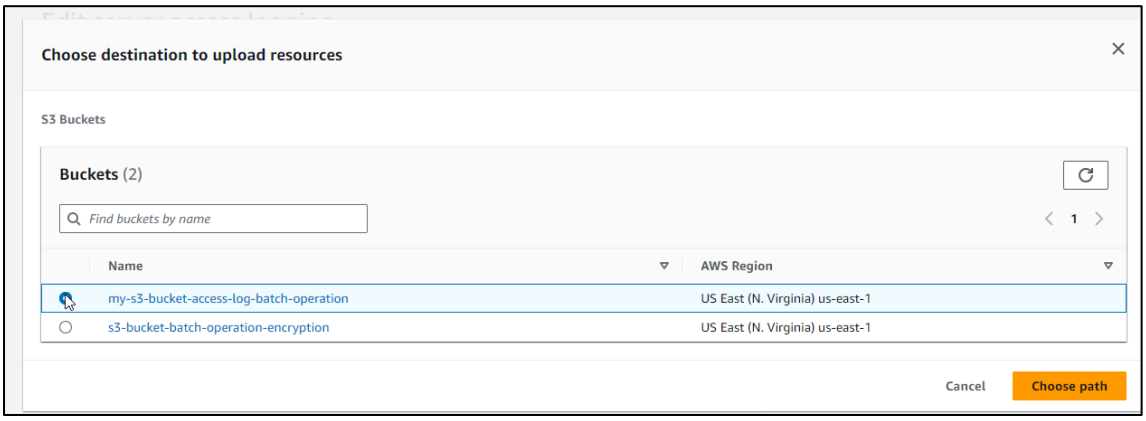
1.5 Navigate to **Properties**



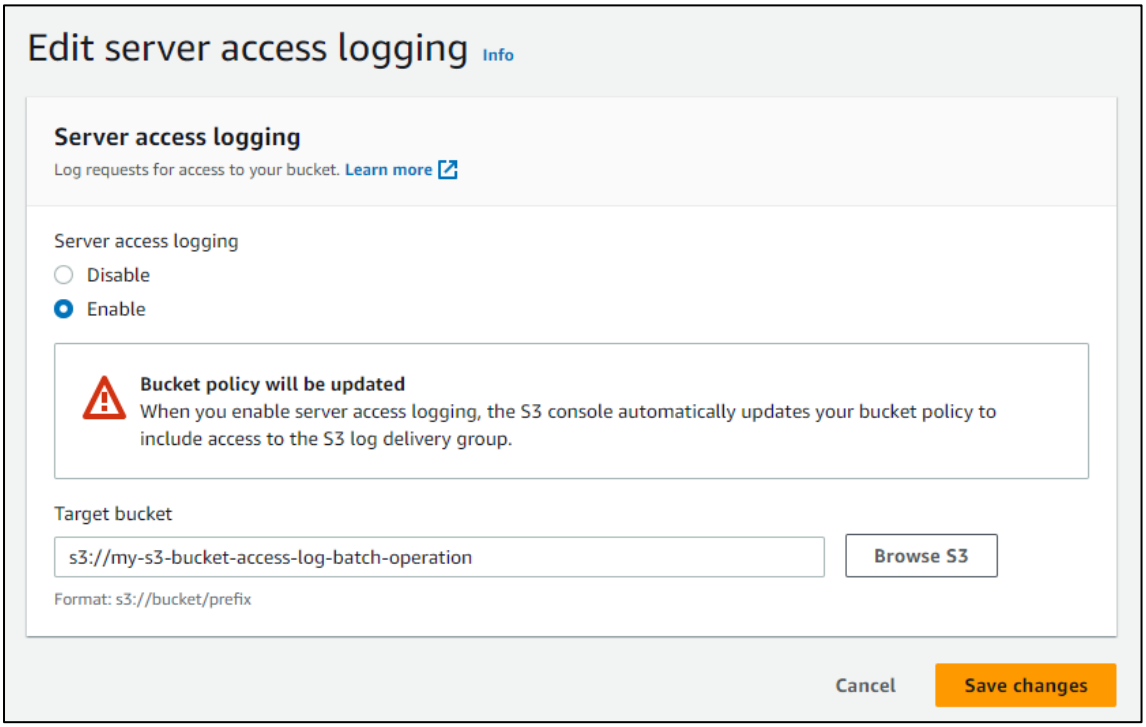
1.6 Scroll down and select **Server access logging**, and then click **Edit** and **Browse S3**



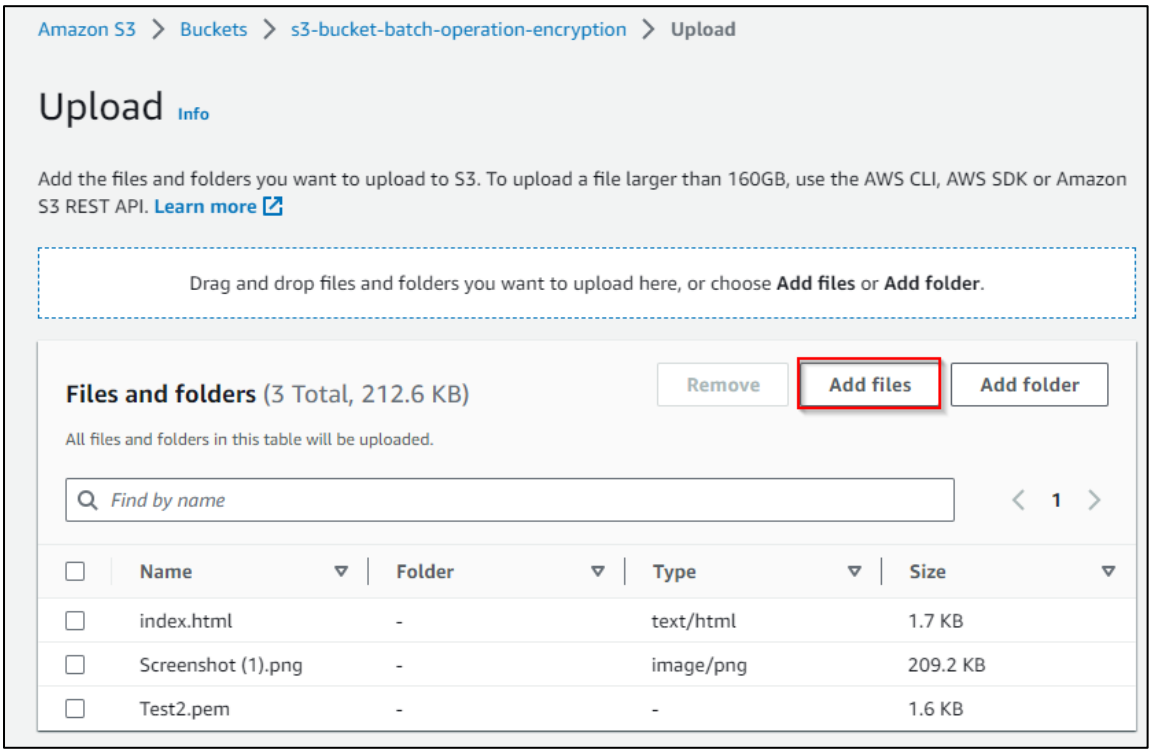
1.7 Choose the log-batch file path by clicking **Choose path**

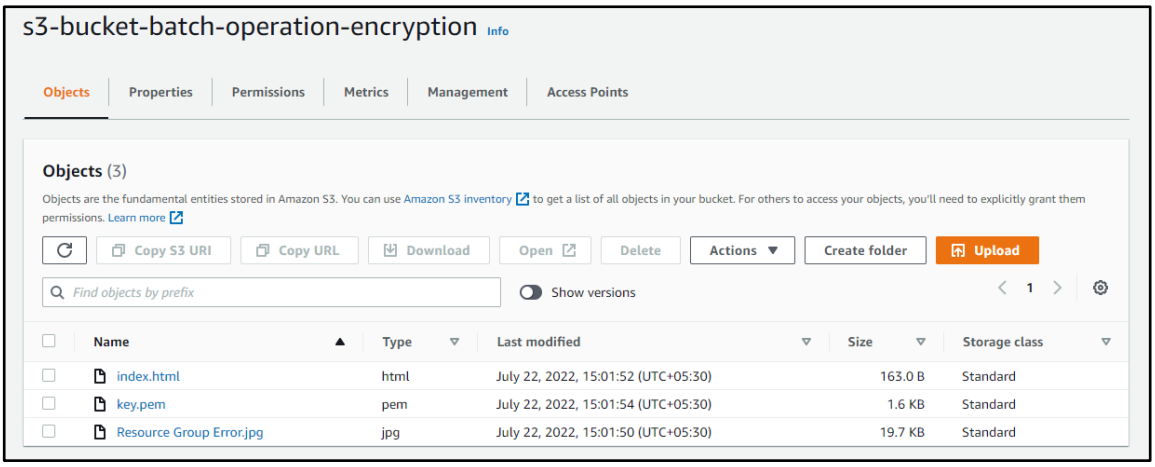


1.8 Click **Save changes**



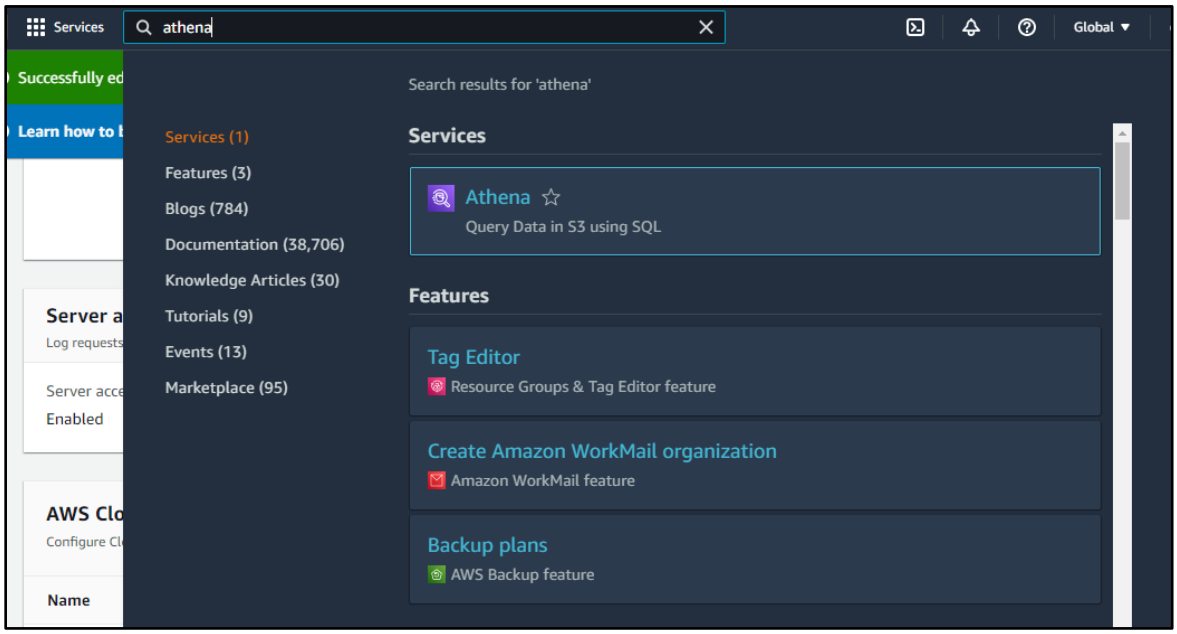
1.9 Add files (HTML, PNG, and PEM) to the bucket using **Add files**



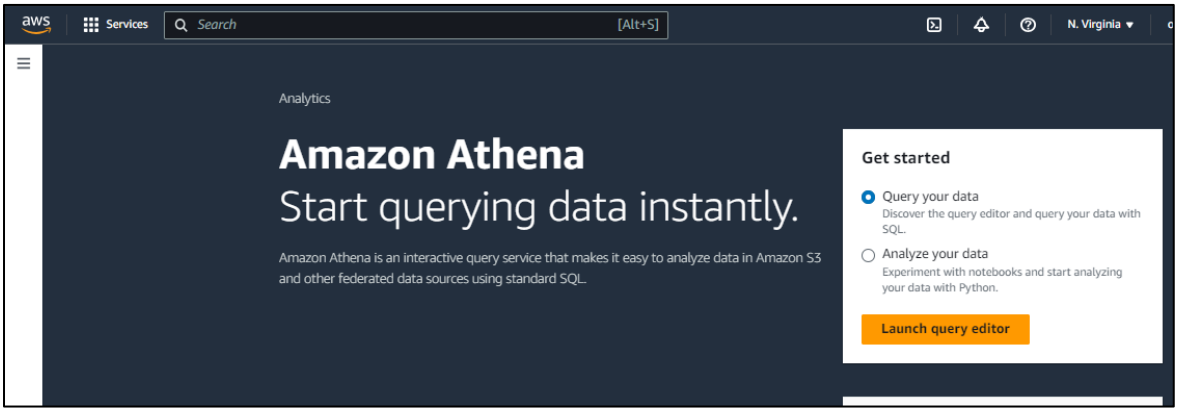


Step 2: Execute queries in Athena

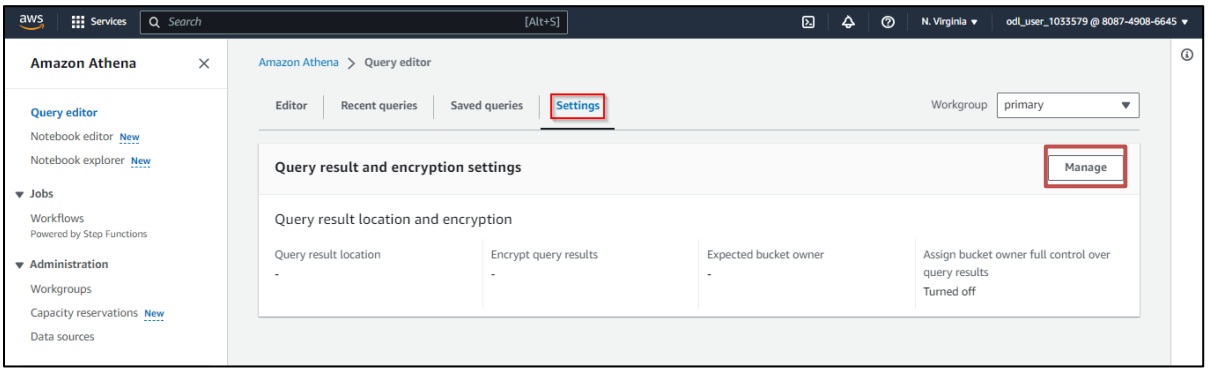
2.1 Navigate to the AWS Management Console and search for **Athena**



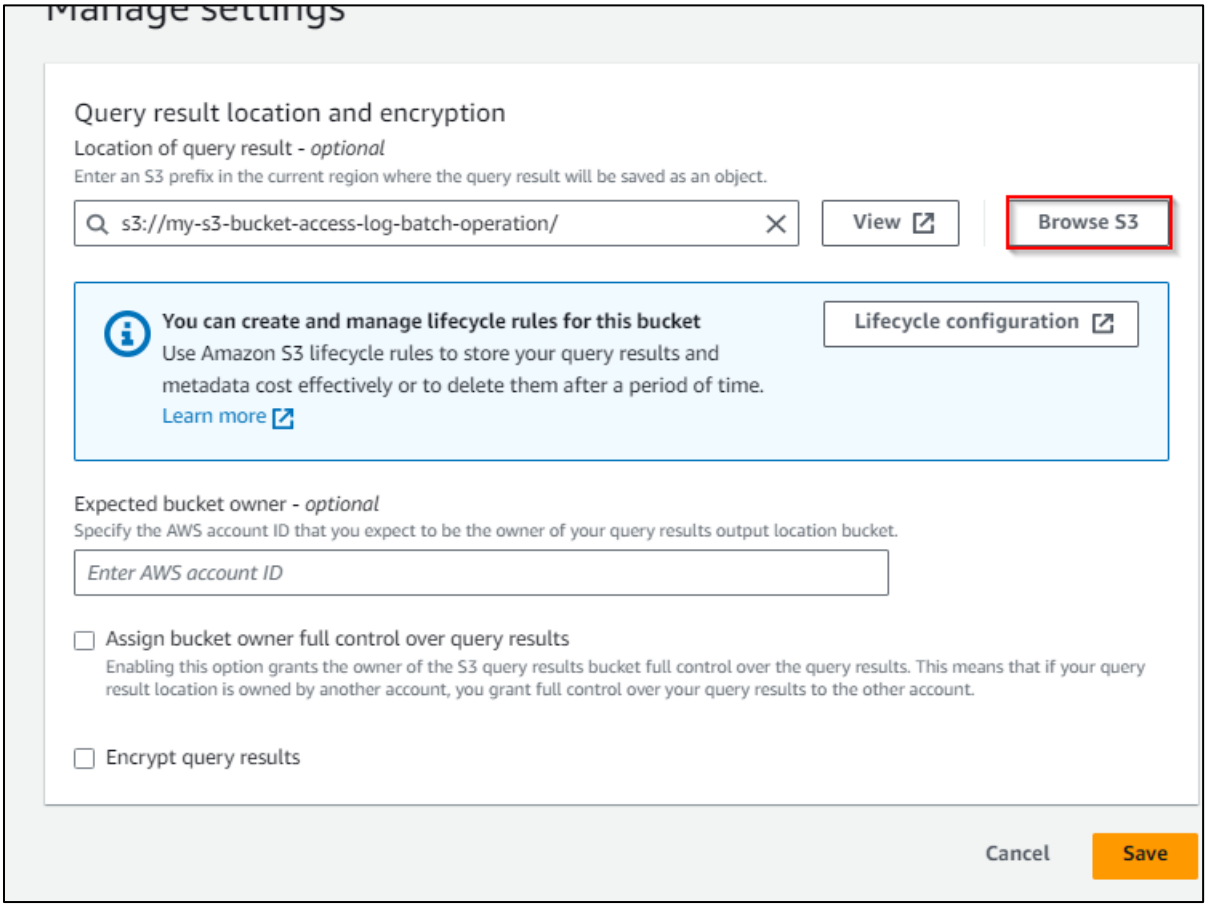
2.2 Launch the query editor in Athena

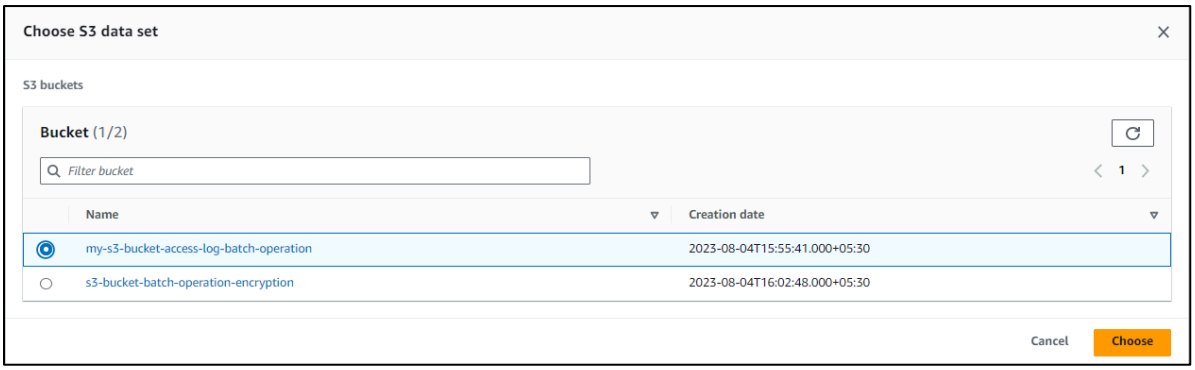


2.3 Click **Settings** > **Manage**

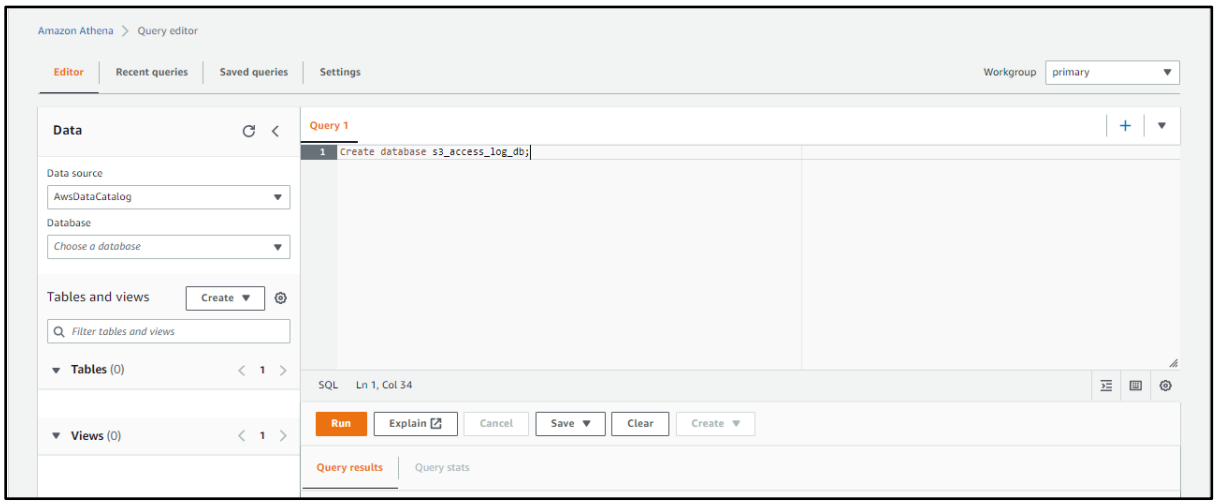


2.4 Click on **Browse S3**, select one of the path locations, and then click **Save**

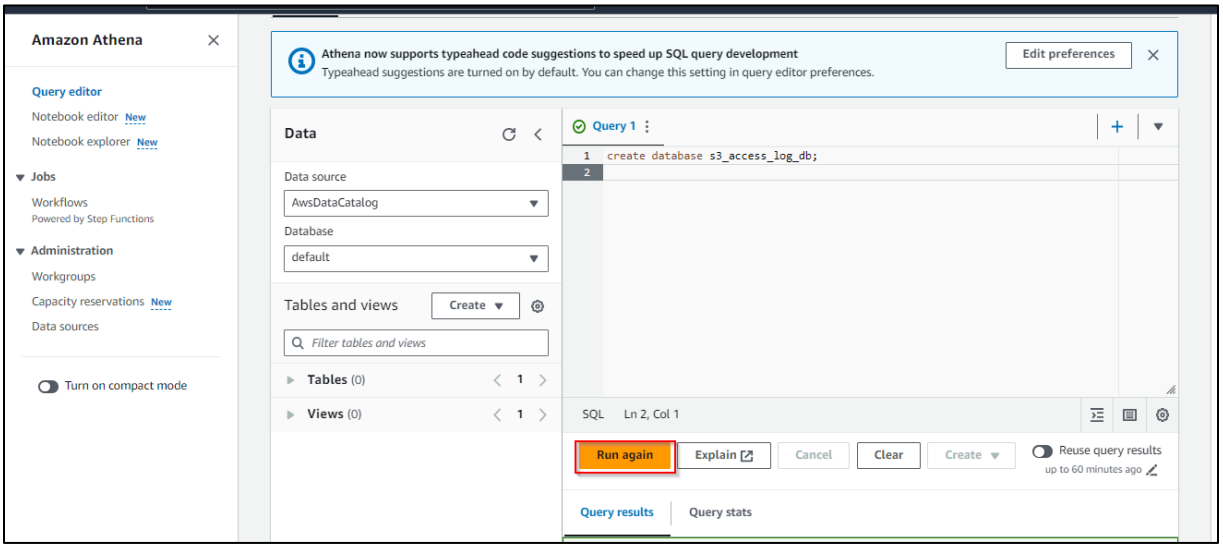




2.5 In the editor, execute the following query:
create database s3_access_log_db;



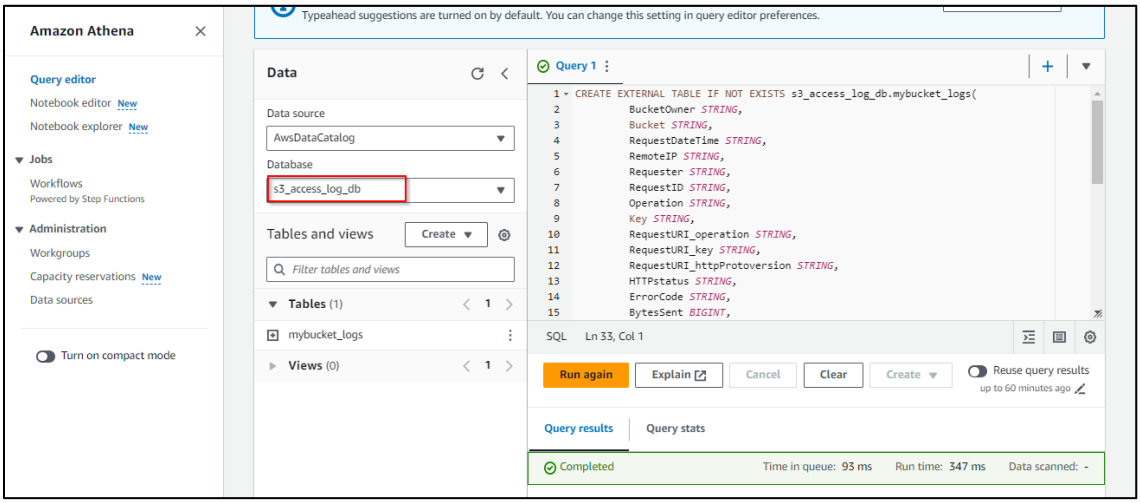
2.6 Click on **Run again**

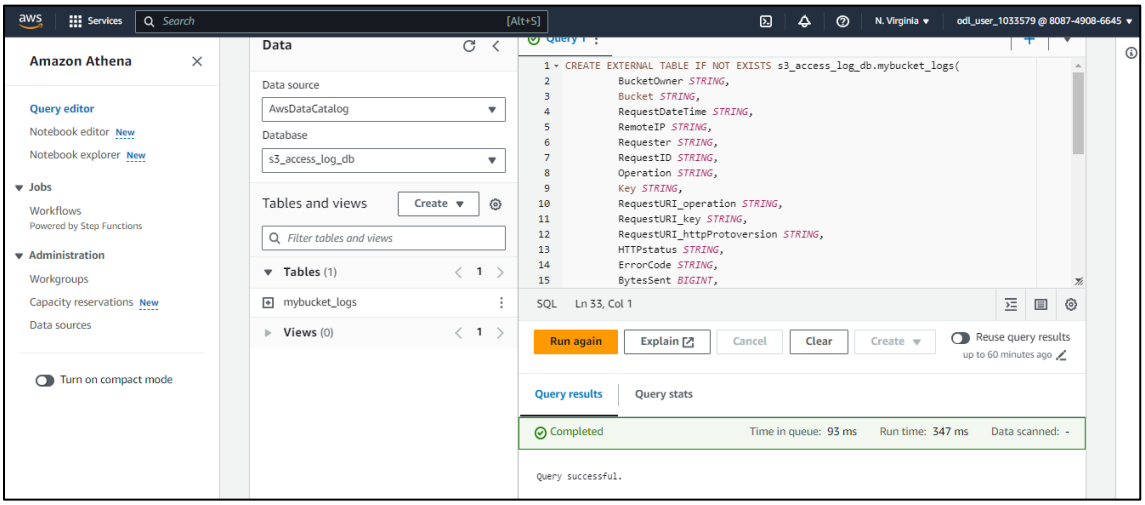


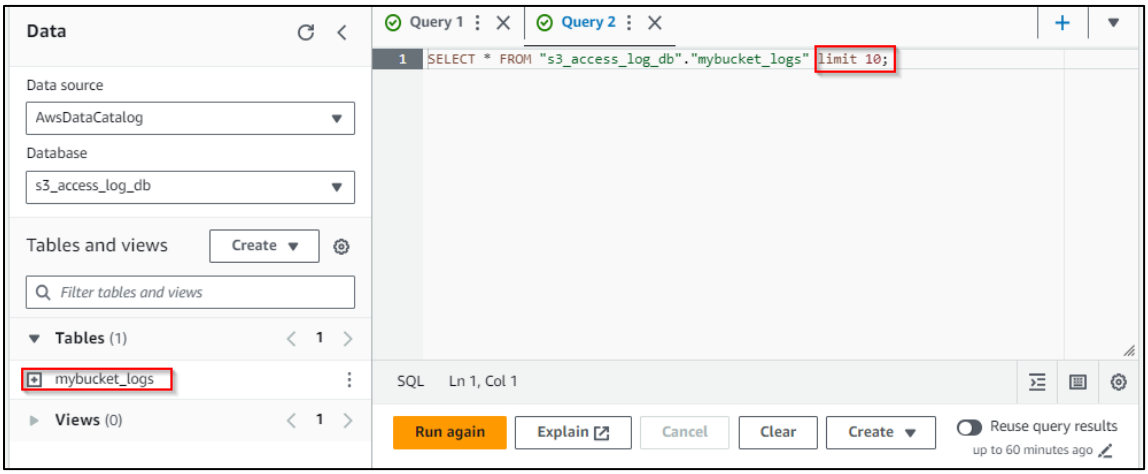
2.7 Execute the given query to create an external table:

```
CREATE EXTERNAL TABLE IF NOT EXISTS s3_access_log_db.mybucket_logs(
 BucketOwner STRING,
 Bucket STRING,
 RequestDateTime STRING,
 RemoteIP STRING,
 Requester STRING,
 RequestID STRING,
 Operation STRING,
 Key STRING,
 RequestURI_operation STRING,
 RequestURI_key STRING,
 RequestURI_httpProtoversion STRING,
 HTTPstatus STRING,
 ErrorCode STRING,
 BytesSent BIGINT,
 ObjectSize BIGINT,
 TotalTime STRING,
 TurnAroundTime STRING,
 Referrer STRING,
 UserAgent STRING,
 VersionId STRING,
 HostId STRING,
 SigV STRING,
 CipherSuite STRING,
 AuthType STRING,
 EndPoint STRING,
 TLSVersion STRING
)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.RegexSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1', 'input.regex' = '([^ ]*) ([^ ]*) \\[(.*?)\\] ([^ ]*) ([^ ]*) ([^ ]*)
([^ ]*) ([^ ]*) \\"([^ ]*) ([^ ]*) (- |[^ ]*)\\" (-|[0-9]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*)
([^ ]*) (\"[^\"]*\") ([^ ]*)(?: ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*))?.*$' )
LOCATION 's3://my-s3-bucket-access-log-batch-operation/'
```

2.8 Execute the code by selecting **s3_access_log_db**, and then click **Run again**



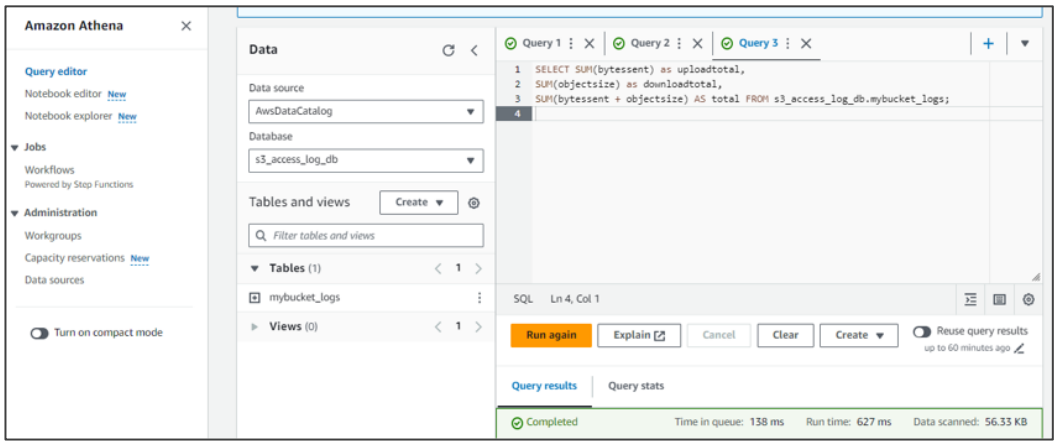




Preview the table: A predefined query with a limit of 10 will be available. Remove the limit to view the complete table data.

2.9 Execute the given query to find the exact amount of data uploaded to and downloaded from the monitored bucket:

SELECT SUM(bytessent) as uploadtotal,
SUM(objectsize) as downloadtotal,
SUM(bytessent + objectsize) AS total FROM s3_access_log_db.mybucket_logs;



By following these steps, you can now execute queries in AWS Athena, enabling you to analyze operations on a designated S3 bucket and get insights into data usage patterns.