CIS 602: Big Data – Homework 4

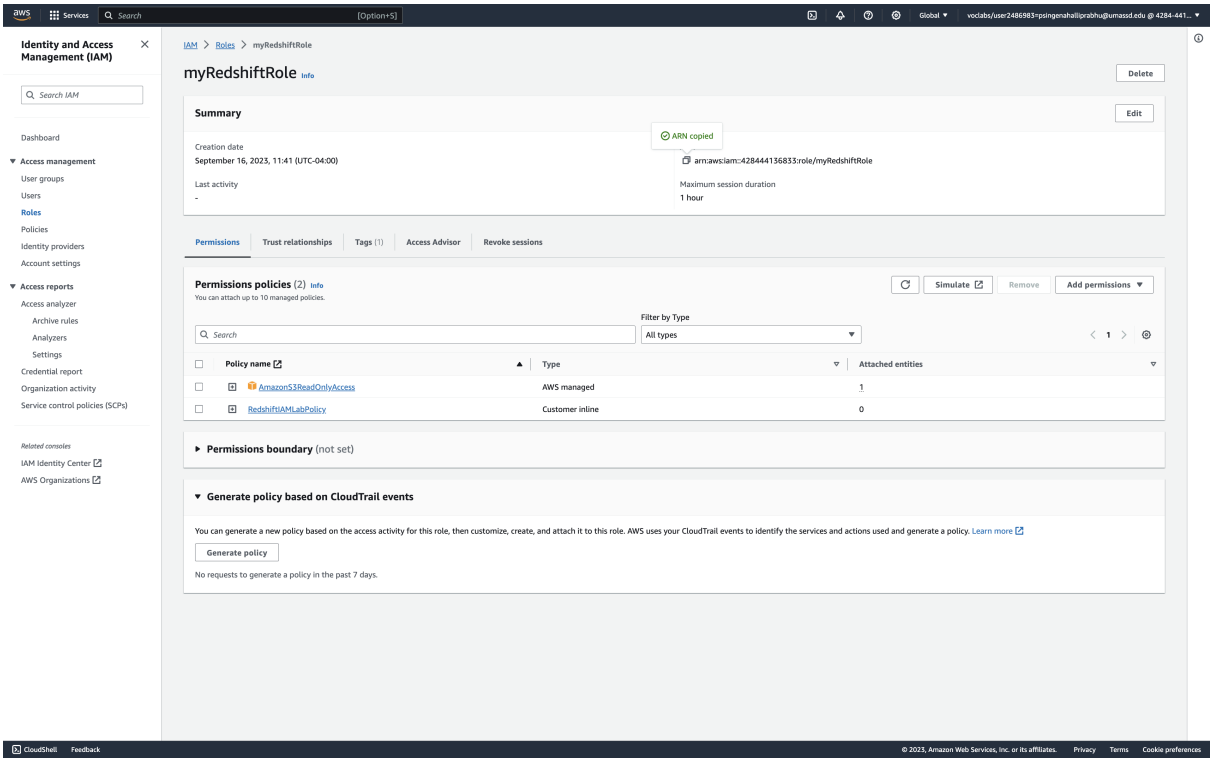
Pradyoth Singenahalli Prabhu - 02071847

Task 1: Reviewing the IAM role to access and configure Amazon Redshift

Accessing the IAM role and get the Amazon Resource Name (ARN):

I copied the ARN from the summary section for the role `MyRedshiftRole`

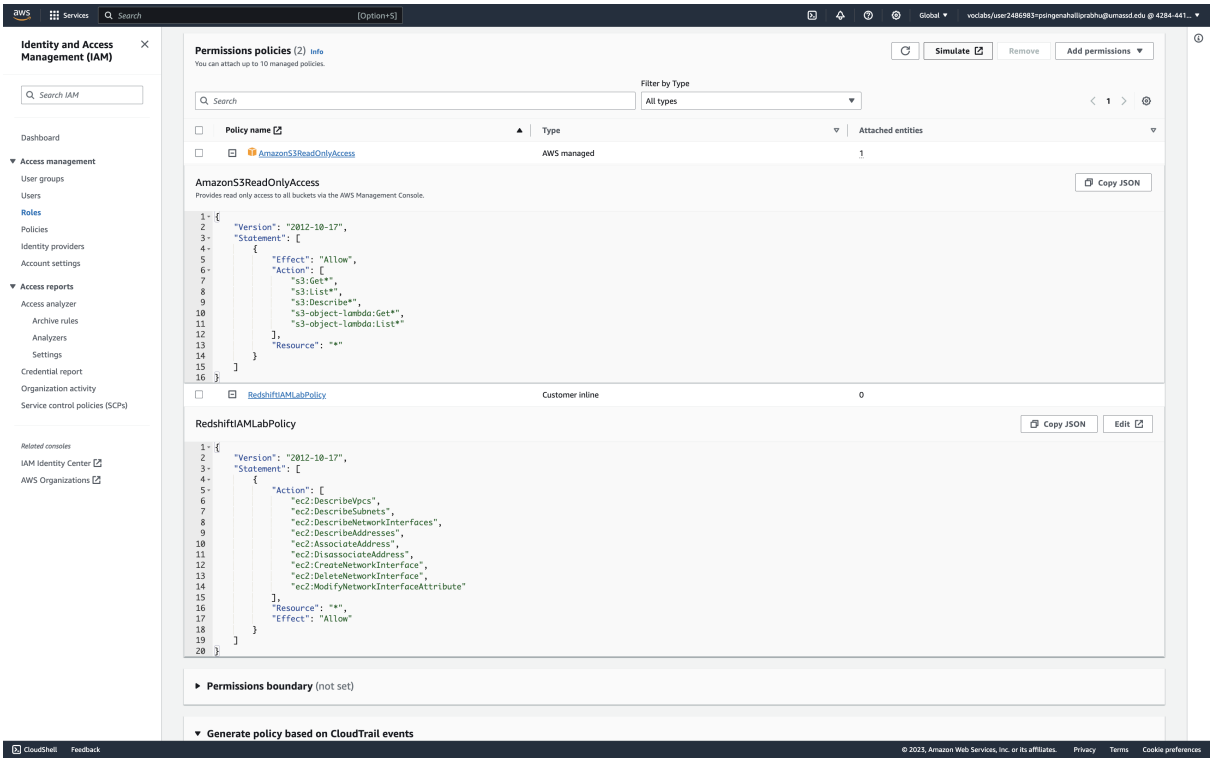
ARN: `arn:aws:iam::428444136833:role/myRedshiftRole`



Reviewing the policies that are associated with the role:

We are reviewing the 2 policies

1. `AmazonS3ReadOnlyAccess`
2. `RedshiftIAMLabPolicy`

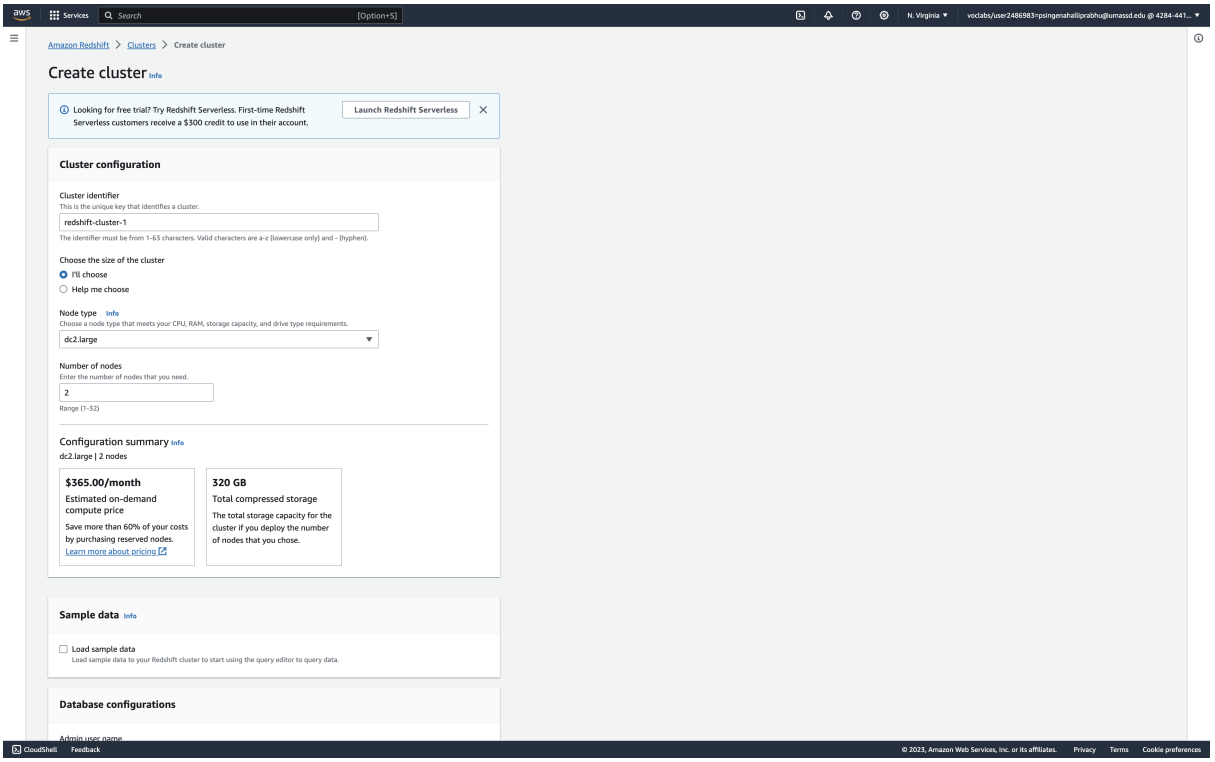


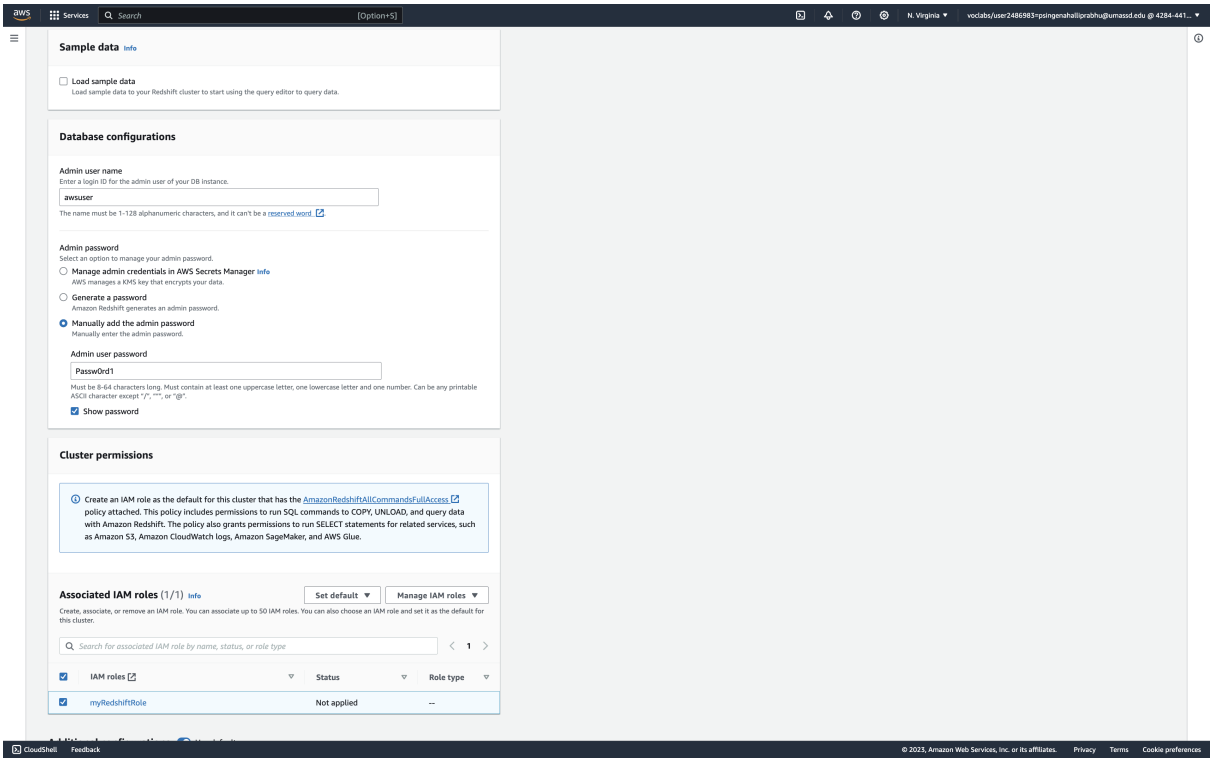
Task 2: Creating and configuring a Redshift cluster

Clusters are the main infrastructure component of an Amazon Redshift data warehouse. A *cluster* is made up of one or more compute nodes. If a cluster has more than one node, then one of the nodes is the ***leader node***, and the other nodes are ***compute nodes***. Client applications interact with the leader node.

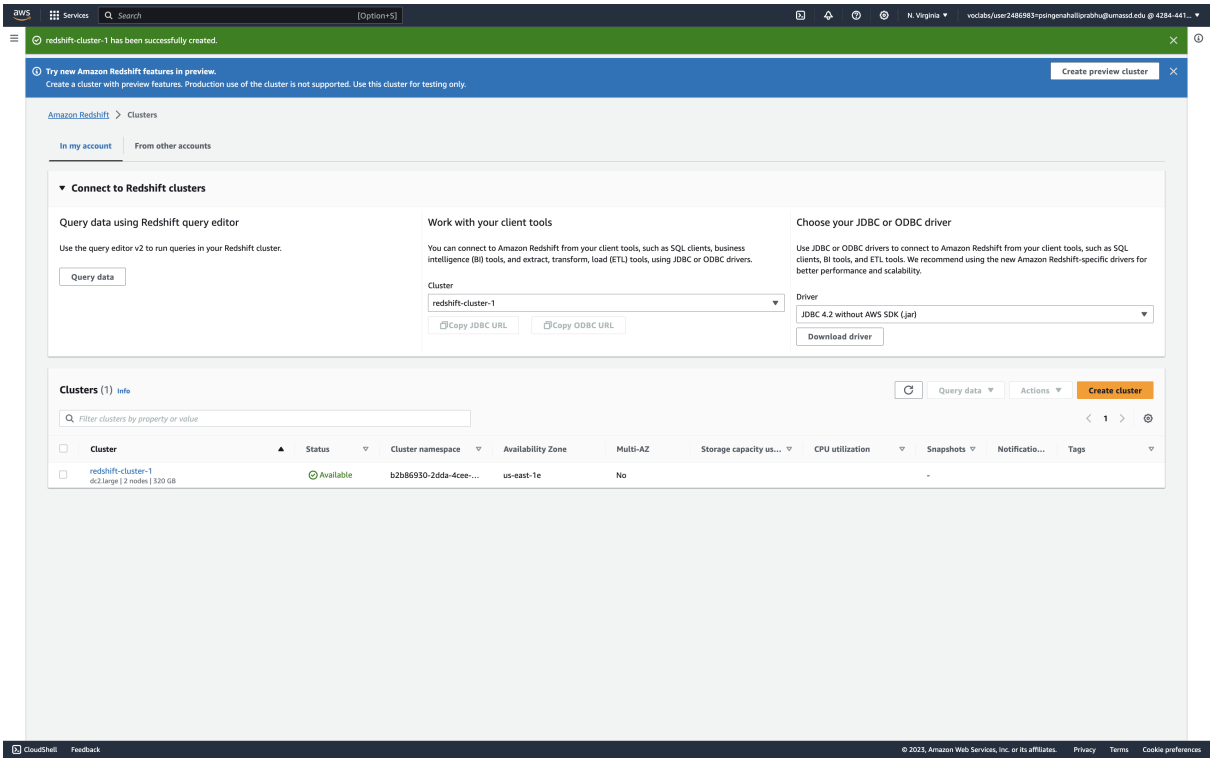
Accessing Amazon Redshift and initiate the Redshift cluster creation process:

Creating Amazon Redshift cluster:



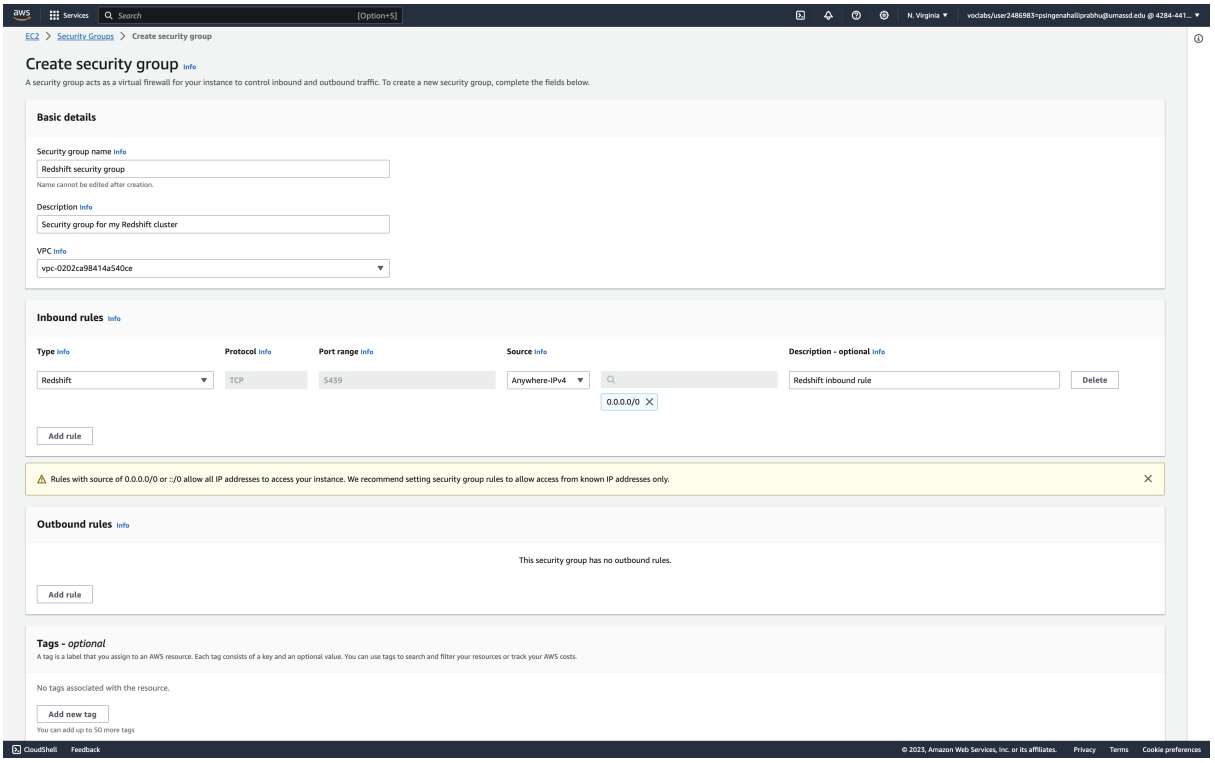


Successfully created Amazon Redshift Cluster:

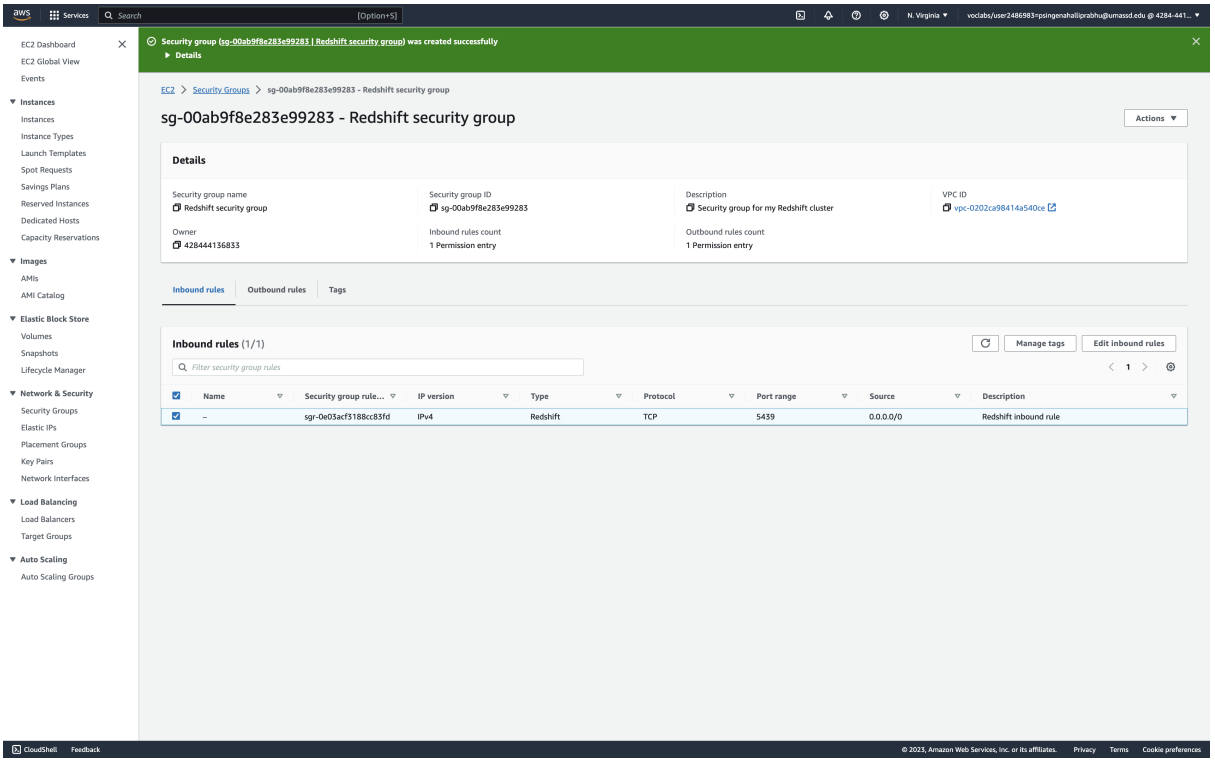


Creating the security group and configure access to the Redshift database:

Creating security group named `Redshift security group` in EC2:

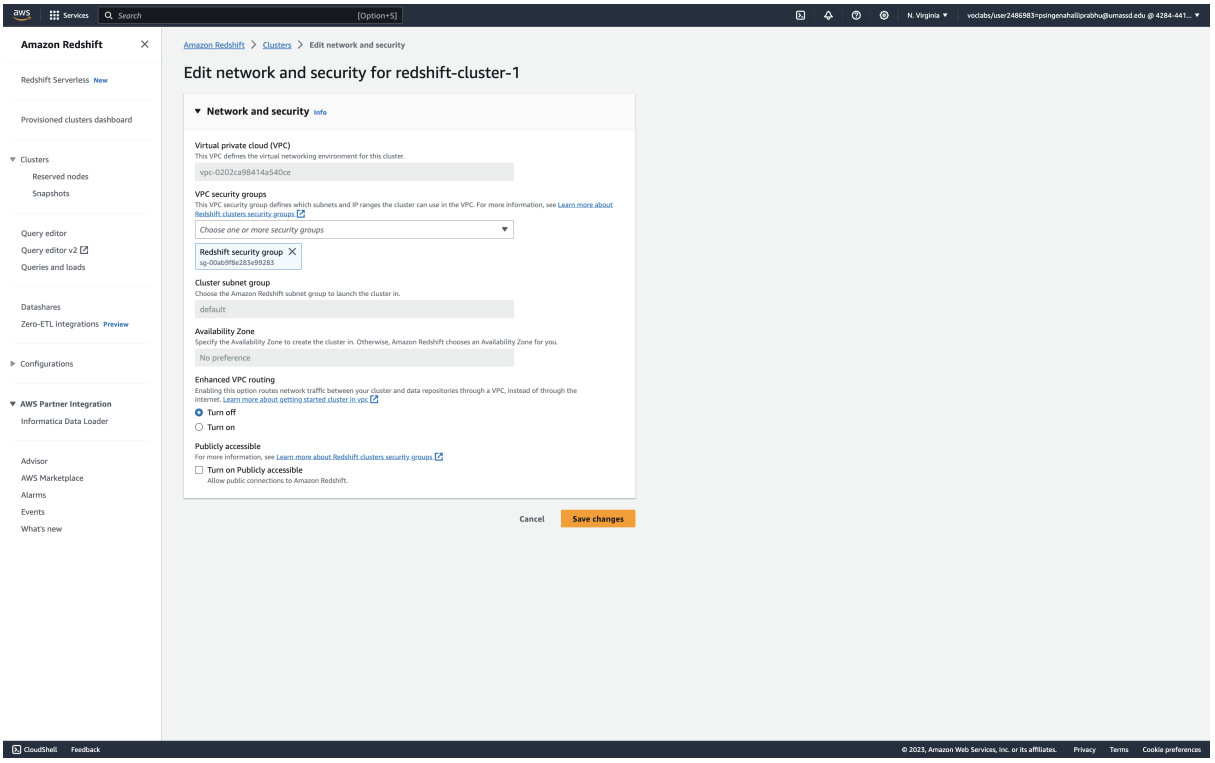


Successfully created the group:



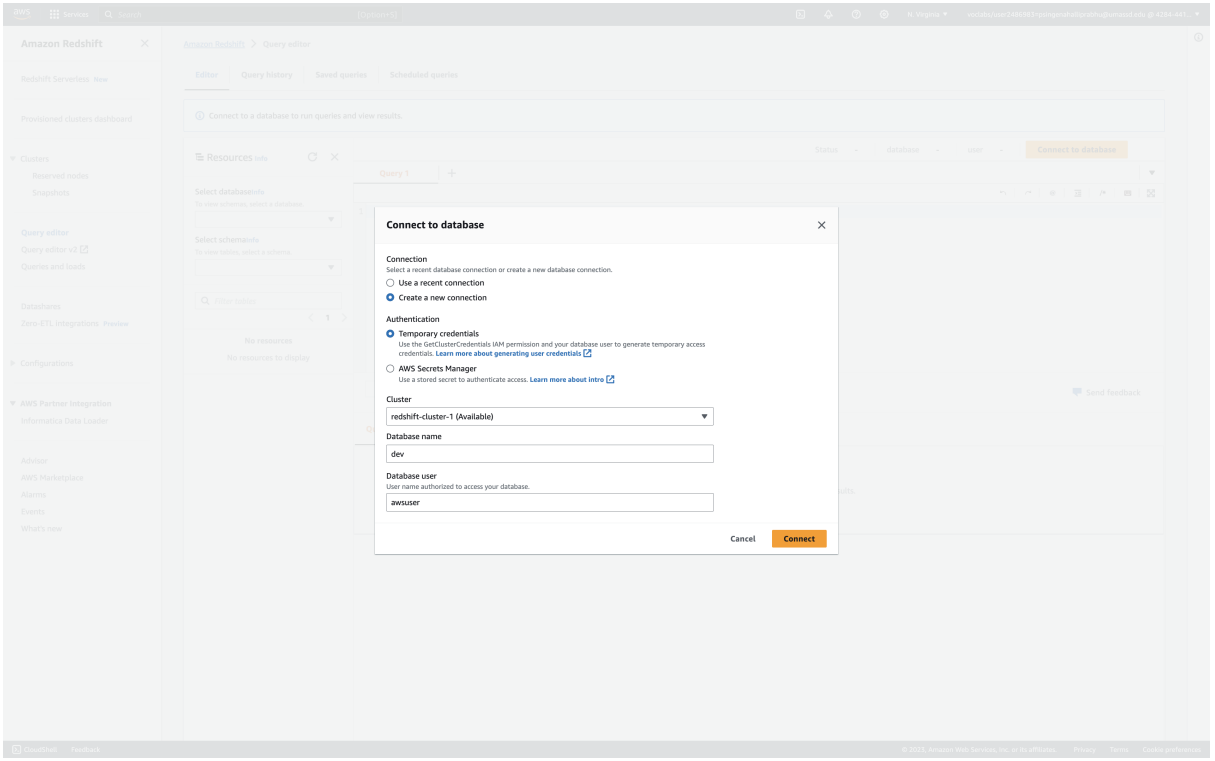
Configuring Redshift cluster:

In the **network and security,** added **Redshift security group** to VPC security groups.



Task 3: Creating tables in a database

Connecting to the Redshift cluster and accessing the query editor:

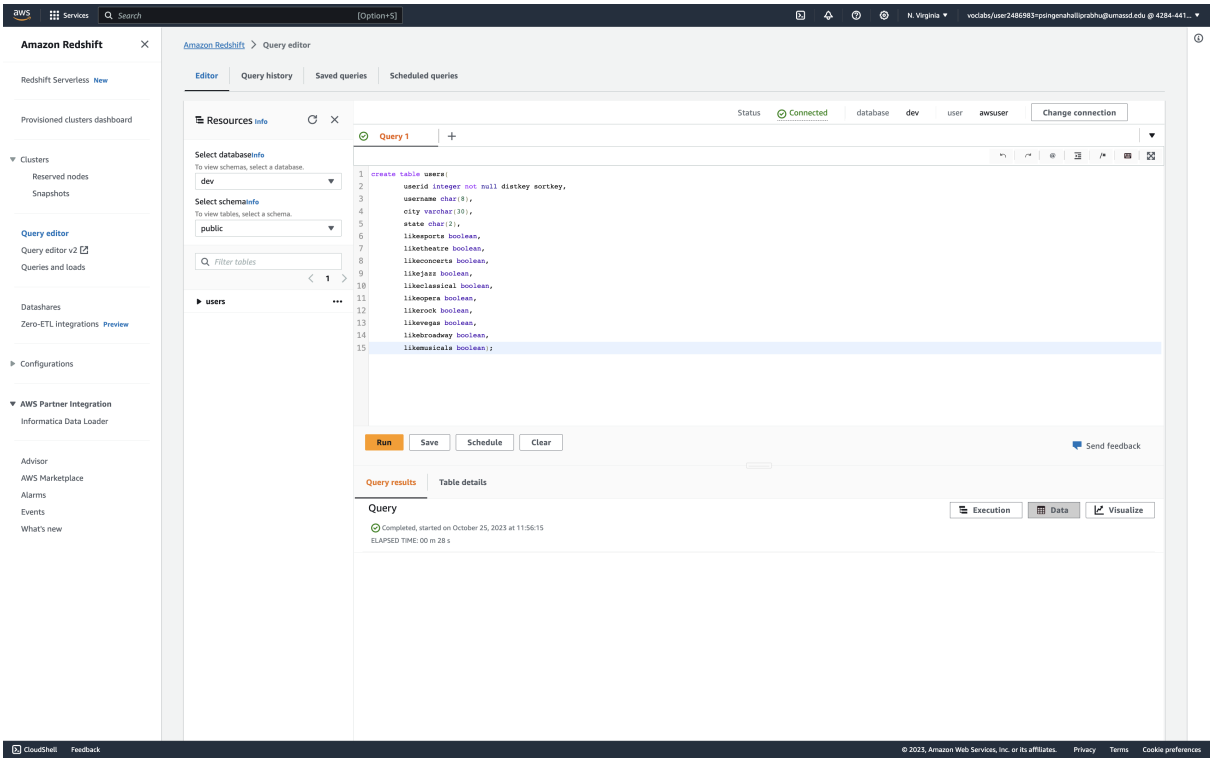


Creating the `users` table:

Query:

```
create table users(
        userid integer not null distkey sortkey,
        username char(8),
        city varchar(30),
        state char(2),
        likesports boolean,
        liketheatre boolean,
        likeconcerts boolean,
        likejazz boolean,
        likeclassical boolean,
        likeopera boolean,
        likerock boolean,
        likevegas boolean,
        likebroadway boolean,
        likemusicals boolean);
```

Created successfully:

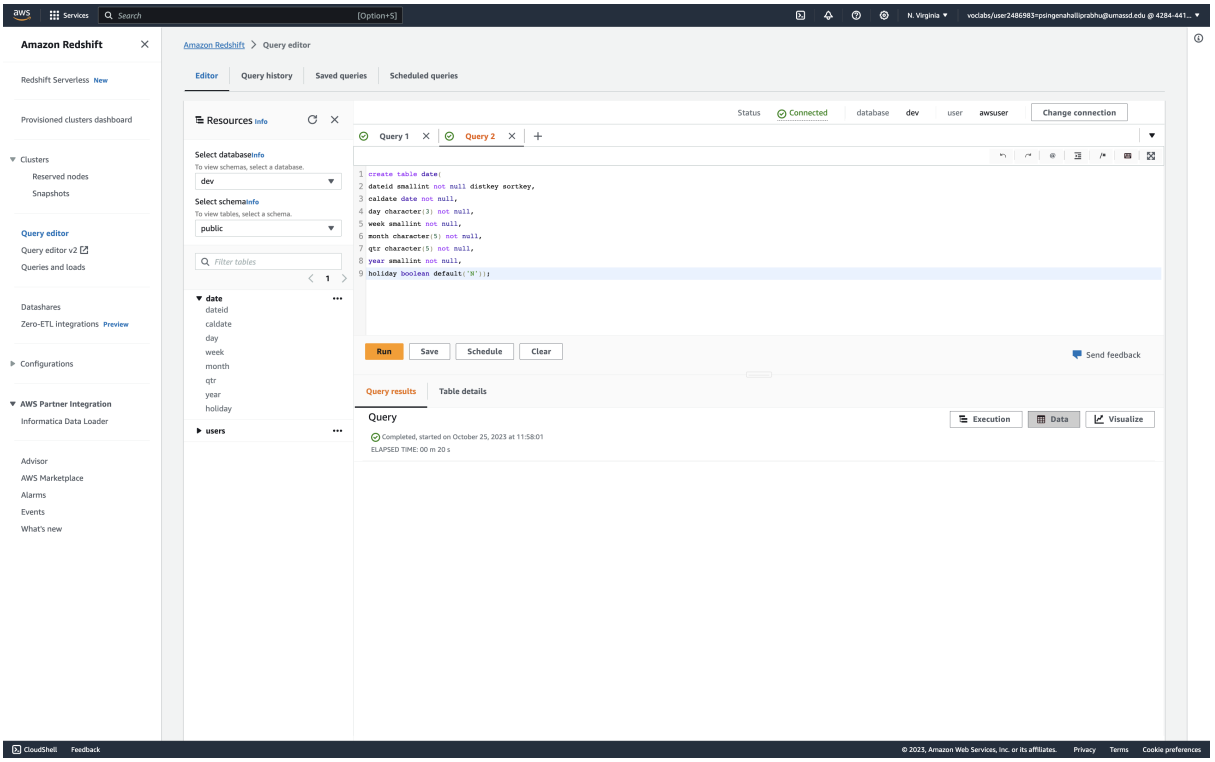


Creating the `data` table:

Query:

```
create table date(
dateid smallint not null distkey sortkey,
caldate date not null,
day character(3) not null,
week smallint not null,
month character(5) not null,
qtr character(5) not null,
year smallint not null,
holiday boolean default('N'));
```

Created successfully:

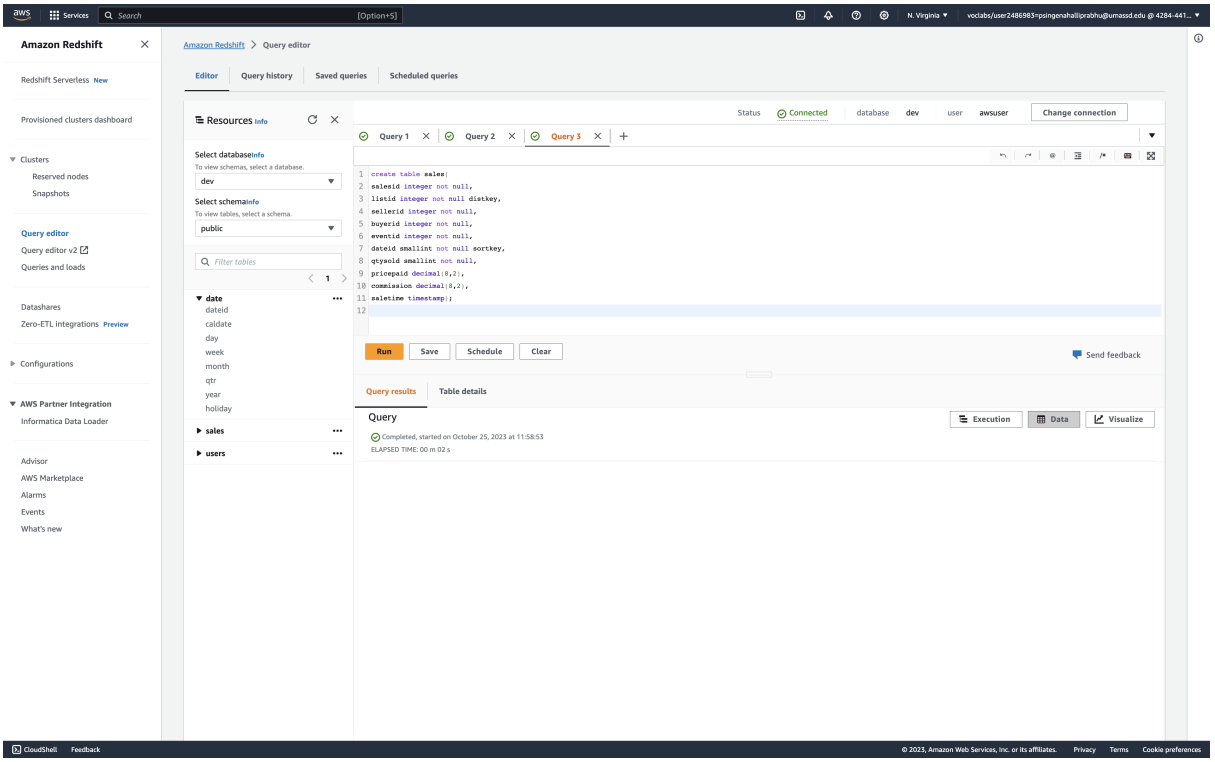


Creating the `sales` table:

Query:

```
create table sales(
salesid integer not null,
listid integer not null distkey,
sellerid integer not null,
buyerid integer not null,
eventid integer not null,
dateid smallint not null sortkey,
qtysold smallint not null,
pricepaid decimal(8,2),
commission decimal(8,2),
saletime timestamp);
```

Created successfully:



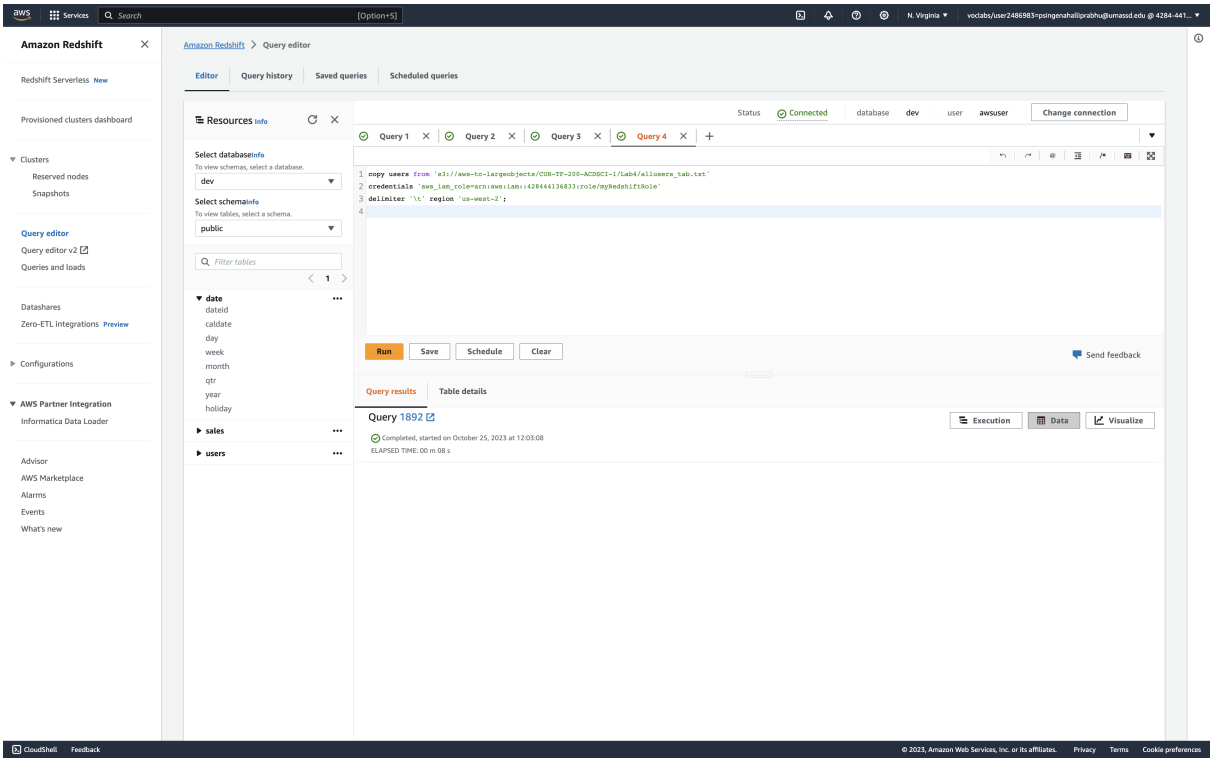
Task 4: Loading data from Amazon S3

To load data to the `users` table, running the following query:

Query:

```
copy users from 's3://aws-tc-largeobjects/CUR-TF-200-ACDSCI-1/Lab4/allusers_tab.txt'
credentials 'aws_iam_role=arn:aws:iam::428444136833:role/myRedshiftRole'
delimiter '\t' region 'us-west-2';
```

Successfully loaded the data:

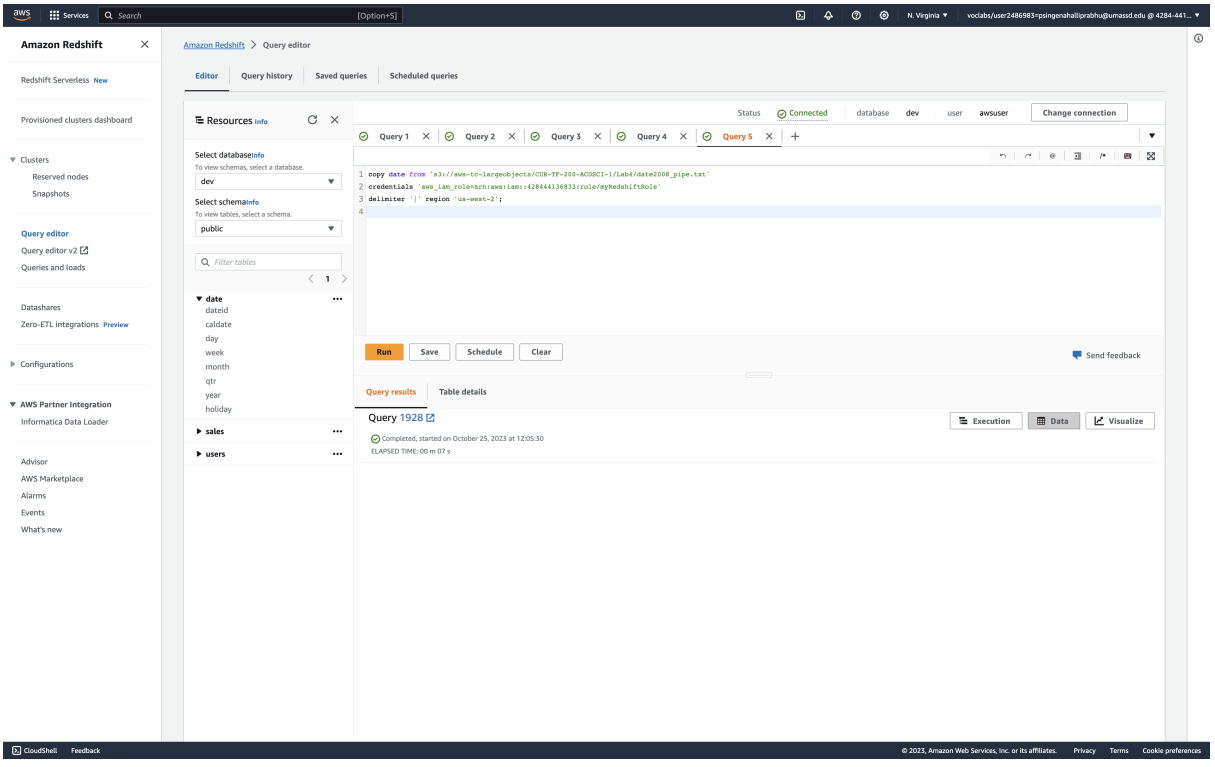


To load data to the `data` table, running the following query:

Query:

```
copy date from 's3://aws-tc-largeobjects/CUR-TF-200-ACDSCI-1/Lab4/date2008_pipe.txt'
credentials 'aws_iam_role=arn:aws:iam::428444136833:role/myRedshiftRole'
delimiter '|' region 'us-west-2';
```

Successfully loaded the data:

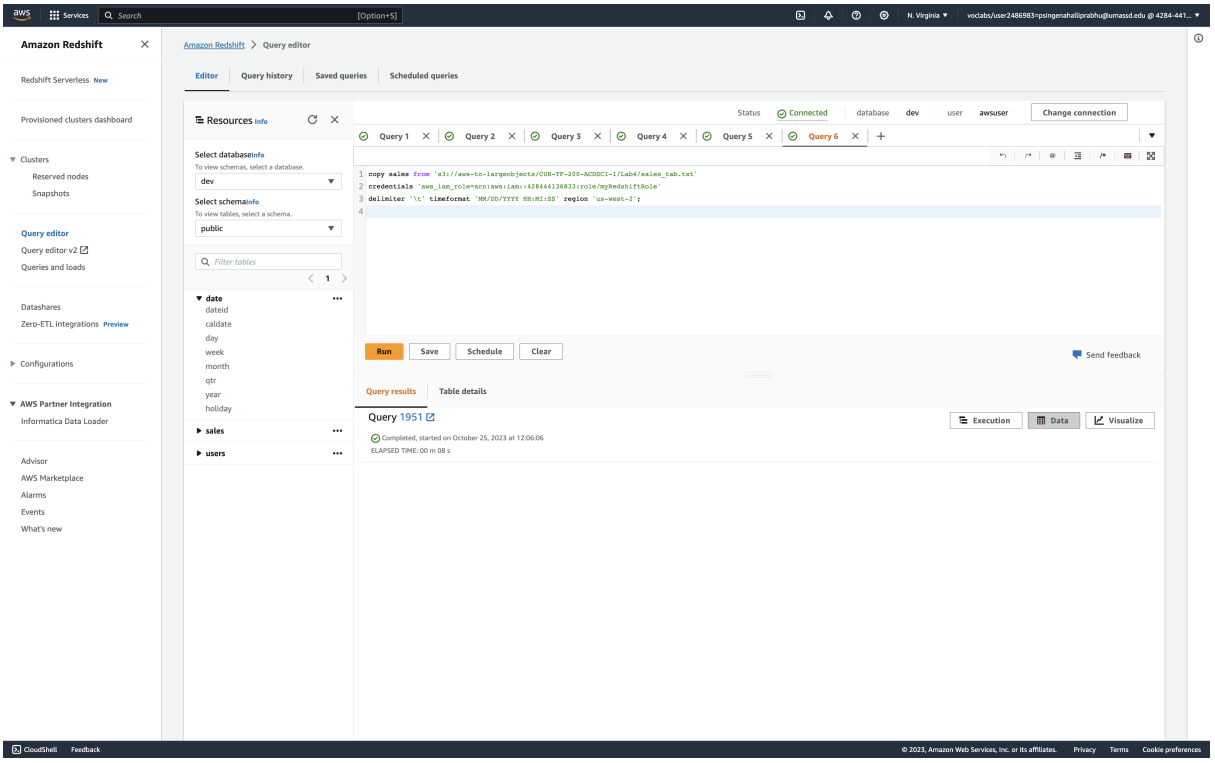


To load data to the `sales` table, running the following query:

Query:

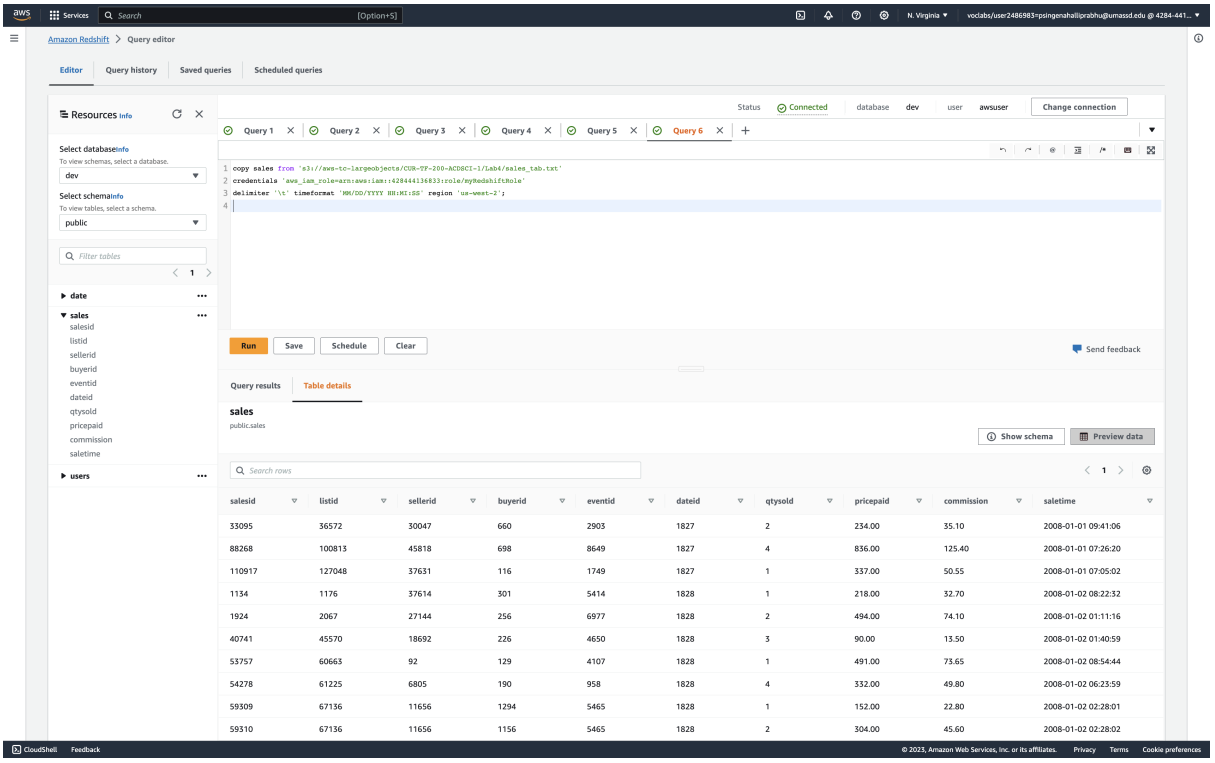
```
copy sales from 's3://aws-tc-largeobjects/CUR-TF-200-ACDSCI-1/Lab4/sales_tab.txt'
credentials 'aws_iam_role=arn:aws:iam::428444136833:role/myRedshiftRole'
delimiter '\t' timeformat 'MM/DD/YYYY HH:MI:SS' region 'us-west-2';
```

Successfully loaded the data:



Task 5: Querying the data

Previewing the `sales` table:

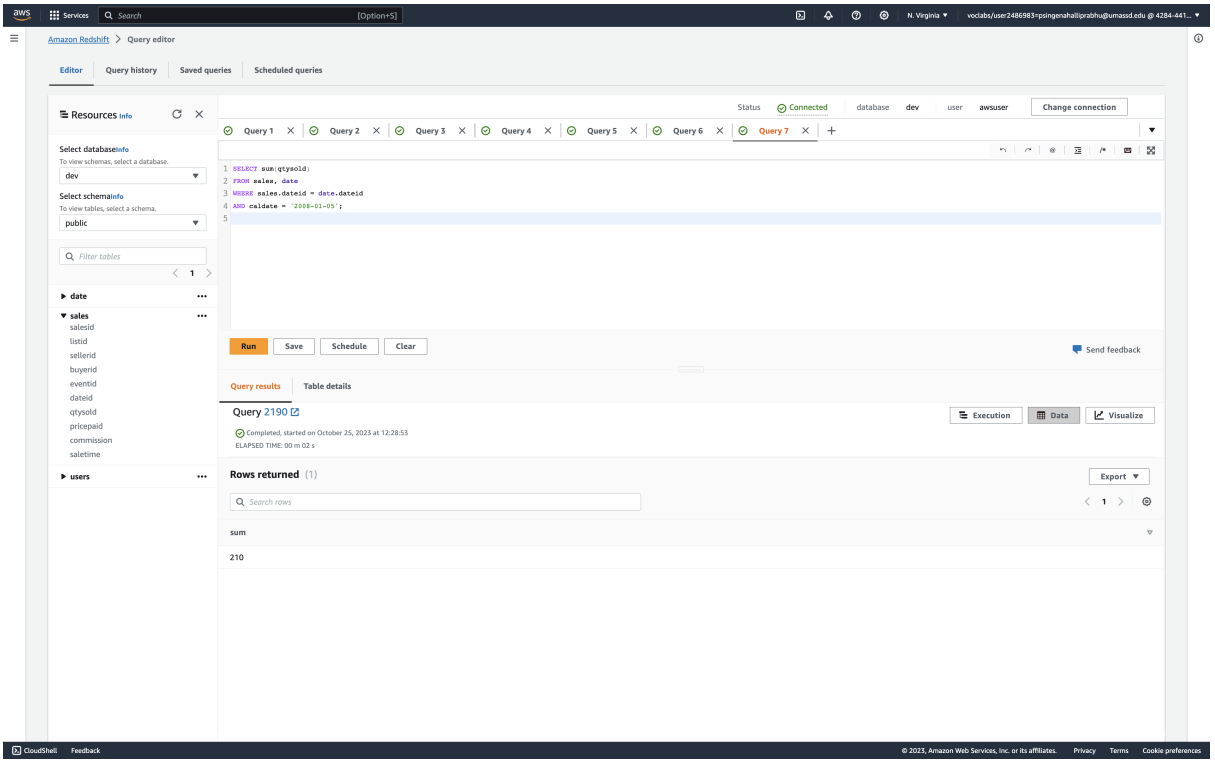


Querying to determine total number of items that were sold on a particular date:

Query:

```
SELECT sum(qtysold)
FROM sales, date
WHERE sales.dateid = date.dateid
AND caldate = '2008-01-05';
```

Result:

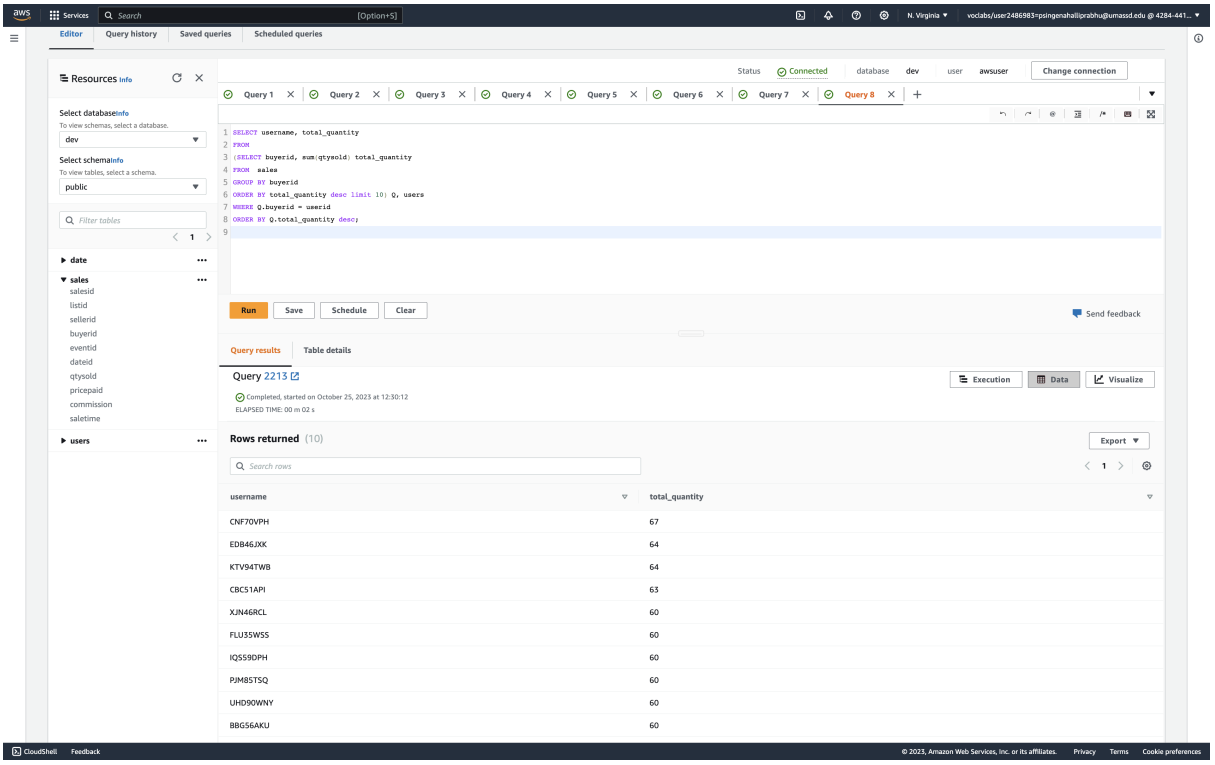


Query to find the top 10 buyers by quantity:

Query:

```
SELECT username, total_quantity
FROM
(SELECT buyerid, sum(qtysold) total_quantity
FROM  sales
GROUP BY buyerid
ORDER BY total_quantity desc limit 10) Q, users
WHERE Q.buyerid = userid
ORDER BY Q.total_quantity desc;
```

Result:



Task 6: Running queries from the AWS CLI

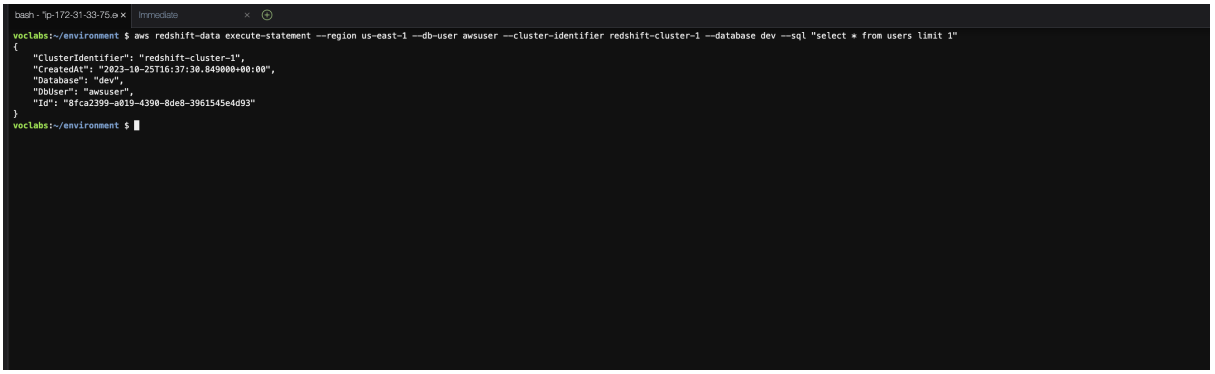
Running the query in Cloud9 Instance:

Query:

```
aws redshift-data execute-statement --region us-east-1 --db-user awsuser --cluster-identifier redshift-cluster-1 --database dev --sql '
```

Result:

```
{
    "ClusterIdentifier": "redshift-cluster-1",
    "CreatedAt": "2023-10-25T16:37:30.849000+00:00",
    "Database": "dev",
    "DbUser": "awsuser",
    "Id": "8fca2399-a019-4390-8de8-3961545e4d93"
}
```



Retrieving the results of the query:

Query:

```
aws redshift-data get-statement-result  --id 8fca2399-a019-4390-8de8-3961545e4d93 --region us-east-1
```

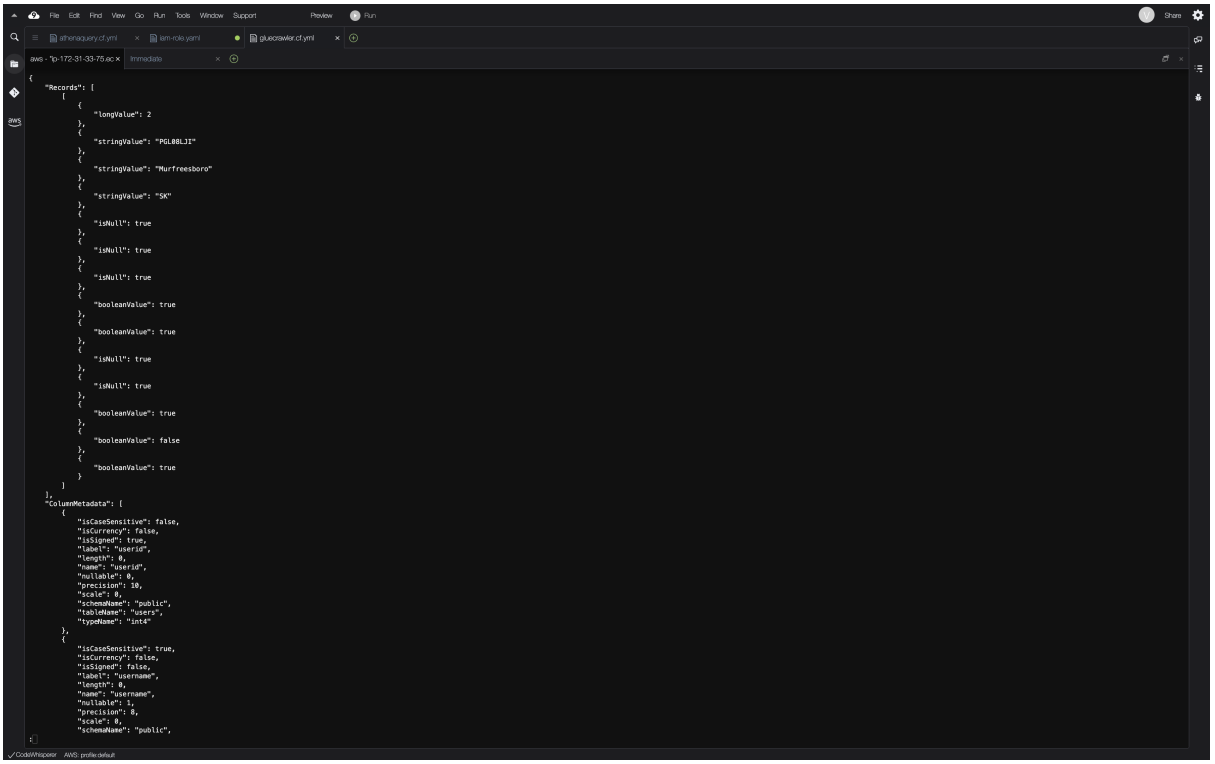
Results:

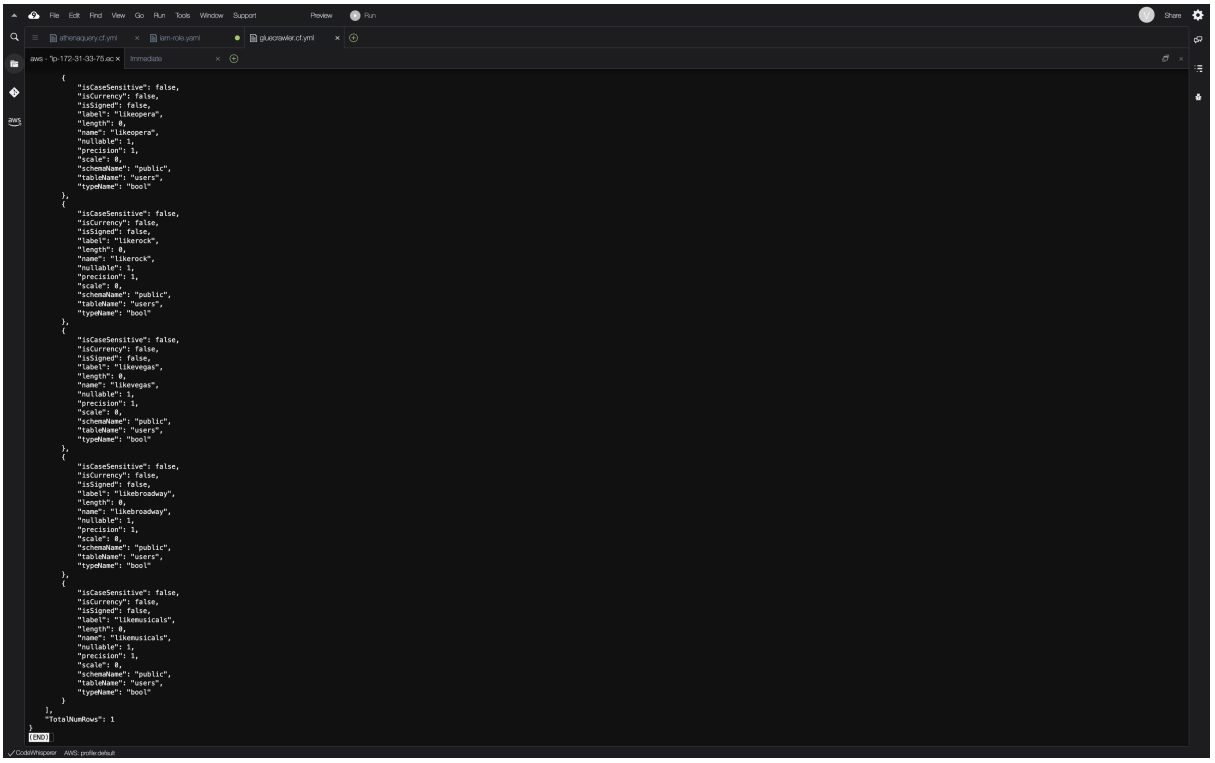

```
{
    "Records": [
        [
            {
                "longValue": 2
            },
            {
                "stringValue": "PGL08LJI"
            },
            {
                "stringValue": "Murfreesboro"
            },
            {
                "stringValue": "SK"
            },
            {
                "isNull": true
            },
            {
                "isNull": true
            },
            {
                "isNull": true
            },
            {
                "booleanValue": true
            },
            {
                "booleanValue": true
            },
            {
                "isNull": true
            },
            {
                "isNull": true
            },
            {
                "booleanValue": true
            },
            {
                "booleanValue": false
            },
            {
                "booleanValue": true
            }
        ]
    ],
    "ColumnMetadata": [
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": true,
            "label": "userid",
            "length": 0,
```

```
            "name": "userid",
            "nullable": 0,
            "precision": 10,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "int4"
        },
        {
            "isCaseSensitive": true,
            "isCurrency": false,
            "isSigned": false,
            "label": "username",
            "length": 0,
            "name": "username",
            "nullable": 1,
            "precision": 8,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bpchar"
        },
        {
            "isCaseSensitive": true,
            "isCurrency": false,
            "isSigned": false,
            "label": "city",
            "length": 0,
            "name": "city",
            "nullable": 1,
            "precision": 30,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "varchar"
        },
        {
            "isCaseSensitive": true,
            "isCurrency": false,
            "isSigned": false,
            "label": "state",
            "length": 0,
            "name": "state",
            "nullable": 1,
            "precision": 2,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bpchar"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likesports",
            "length": 0,
            "name": "likesports",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "liketheatre",
            "length": 0,
            "name": "liketheatre",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likeconcerts",
            "length": 0,
```

```
            "name": "likeconcerts",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likejazz",
            "length": 0,
            "name": "likejazz",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likeclassical",
            "length": 0,
            "name": "likeclassical",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likeopera",
            "length": 0,
            "name": "likeopera",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likerock",
            "length": 0,
            "name": "likerock",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likevegas",
            "length": 0,
            "name": "likevegas",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likebroadway",
            "length": 0,
```

```
            "name": "likebroadway",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        },
        {
            "isCaseSensitive": false,
            "isCurrency": false,
            "isSigned": false,
            "label": "likemusicals",
            "length": 0,
            "name": "likemusicals",
            "nullable": 1,
            "precision": 1,
            "scale": 0,
            "schemaName": "public",
            "tableName": "users",
            "typeName": "bool"
        }
    ],
    "TotalNumRows": 1
}
```



Conclusion Remarks:

The meticulous approach taken in the initial review of IAM roles demonstrated a keen awareness of the critical need for security measures within the AWS ecosystem, especially when dealing with sensitive data. By ensuring controlled access between Amazon Redshift and other integral AWS services, the foundation was set for a secure and well-regulated data environment. This emphasis on security echoes the growing importance of data protection and compliance, particularly in the context of modern data-driven businesses.

The subsequent creation and configuration of the Redshift cluster, along with the strategic establishment of security groups and VPC associations, not only solidified the security framework but also laid the groundwork for efficient data management and streamlined access. This approach underscores a thorough understanding of the intricacies involved in setting up and optimizing a data infrastructure, thereby showcasing a readiness to handle complex data requirements in a professional setting.

Moreover, the adept handling of data integration, demonstrated through the creation of tables in the database and the seamless loading of data from Amazon S3, points to a robust understanding of data manipulation techniques within the Redshift environment. The meticulous attention given to data consistency and integrity during the loading process, as evidenced by the careful utilization of the SQL COPY command with defined delimiters, reflects a dedication to data quality and accuracy, essential aspects of any successful data science and analysis practice.

The demonstration of executing queries through the AWS CLI highlights a pragmatic approach to data processing, emphasizing the need for a deeper understanding of programmatic access for comprehensive data analysis. Acknowledging SQL's flexibility in data querying signifies a clear recognition of its role in enabling effective and streamlined data analysis workflows, indicating a readiness to leverage advanced tools and techniques to extract meaningful insights from complex datasets.

In summary, the successful completion of the tasks not only showcases a solid grasp of Amazon Redshift's capabilities but also reflects a proactive approach to data management, security, and analysis, all of which are pivotal in contributing to the rapidly evolving landscape of technology-driven data science and analytics.