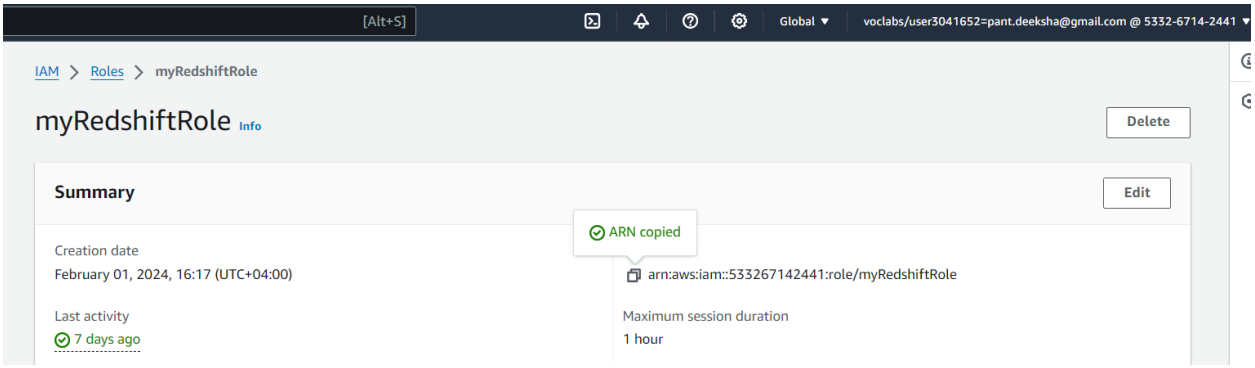
Creation of a Redshift Cluster

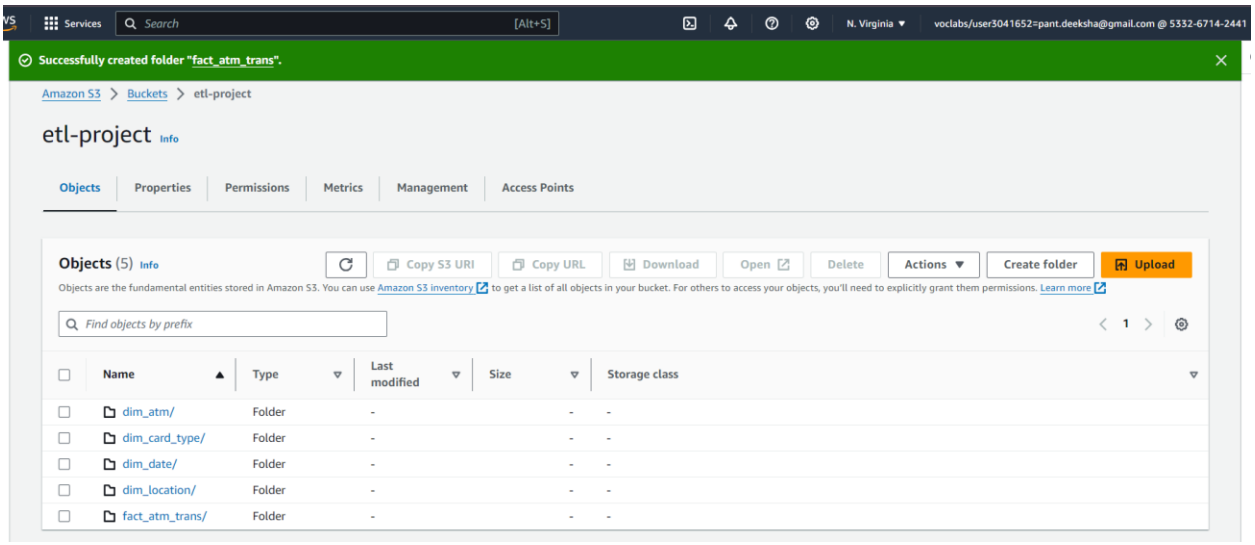
Screenshots of the configuration of the Redshift cluster that you have created:

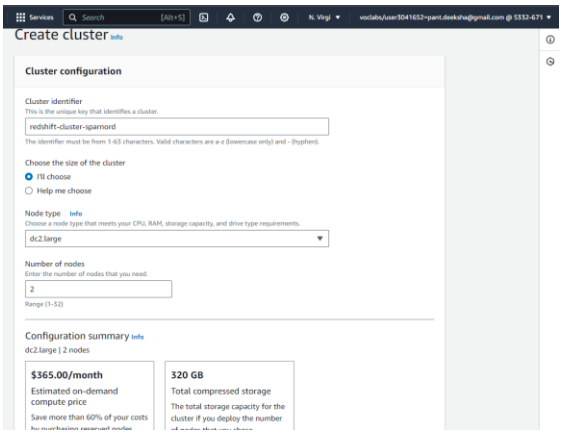
<Screenshot of the type of machine used along with number of nodes>

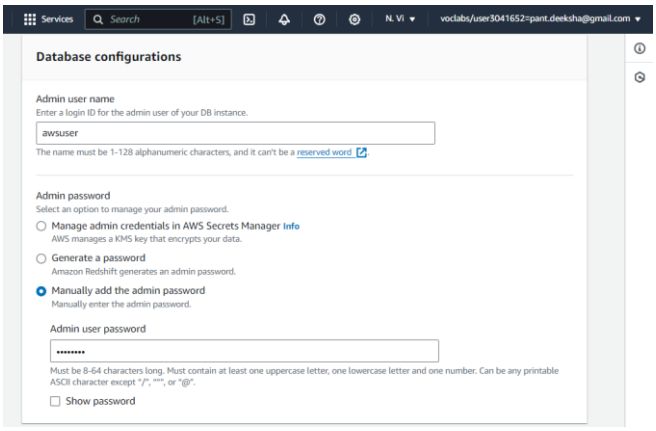
IAM Role used in Redshift Cluster creation:

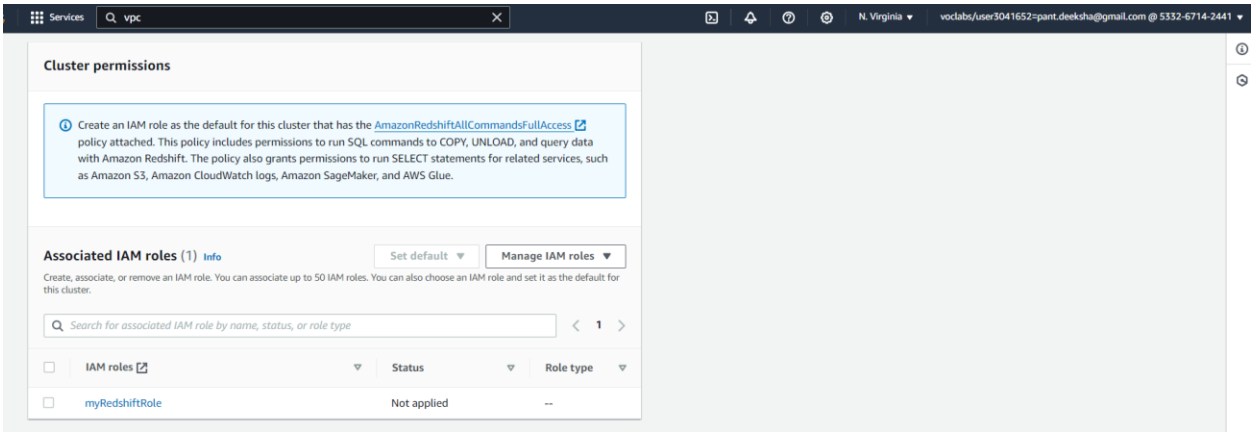


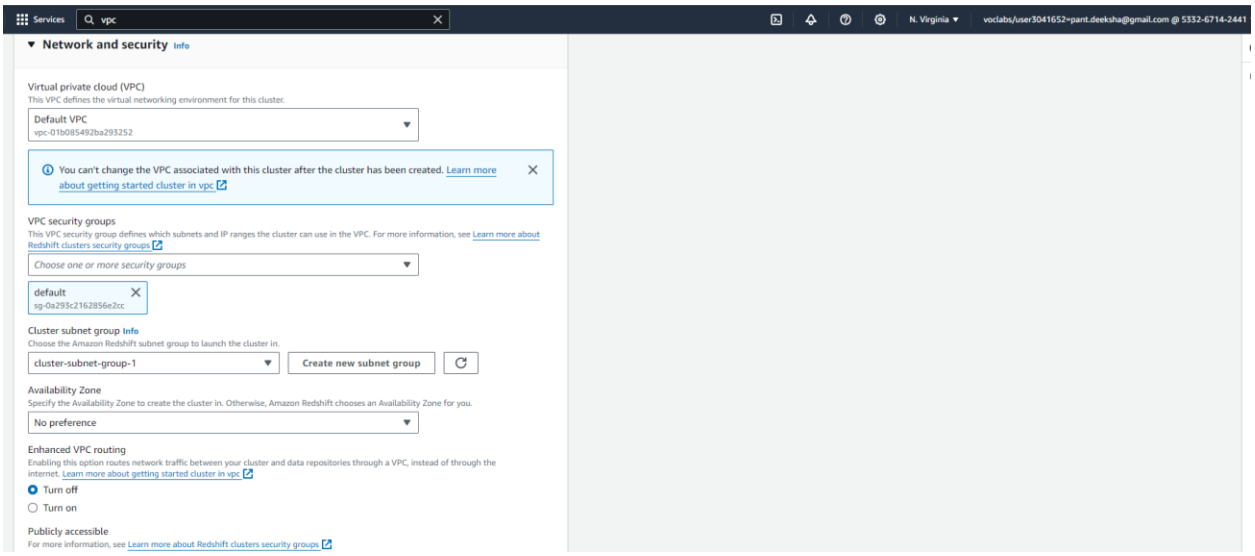
AWS S3 created with different buckets for facts and dimensions:

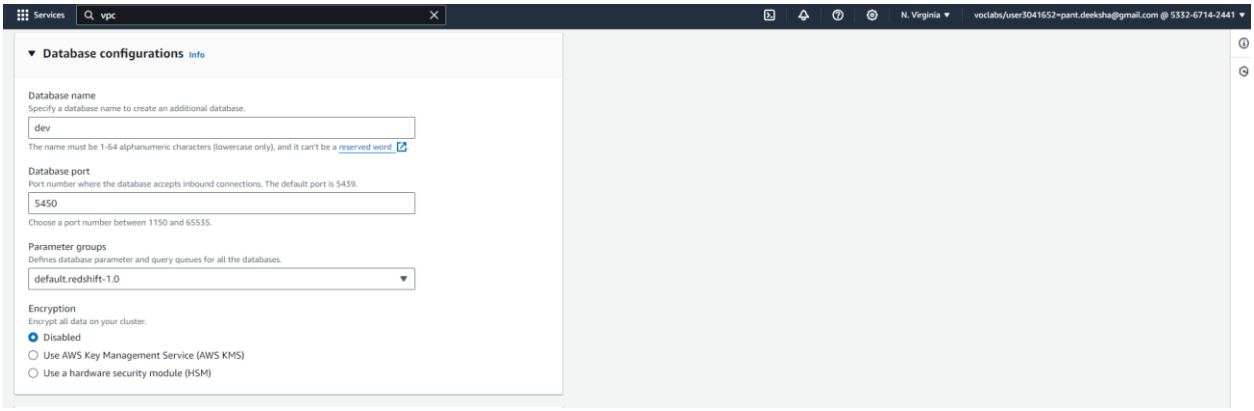



Create cluster
Cluster configuration
Cluster identifier
redshift-cluster-sparnord
Choose the size of the cluster
I'll choose
Help me choose
Node type
dc2.large
Number of nodes
2
Configuration summary
$365.00/month
Estimated on-demand compute price
320 GB
Total compressed storage

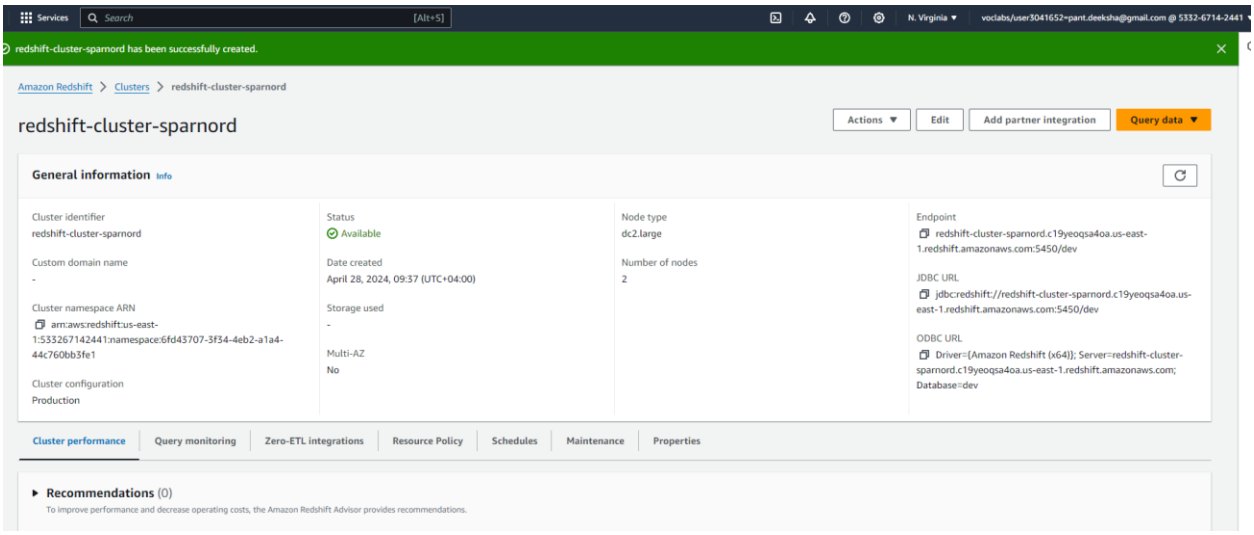

Database configurations
Admin user name
Enter a login ID for the admin user of your DB instance.
awsuser
Admin password
Select an option to manage your admin password.
Manage admin credentials in AWS Secrets Manager
AWS manages a KMS key that encrypts your data.
Generate a password
Amazon Redshift generates an admin password.
Manually add the admin password
Manually enter the admin password.
Admin user password
Show password

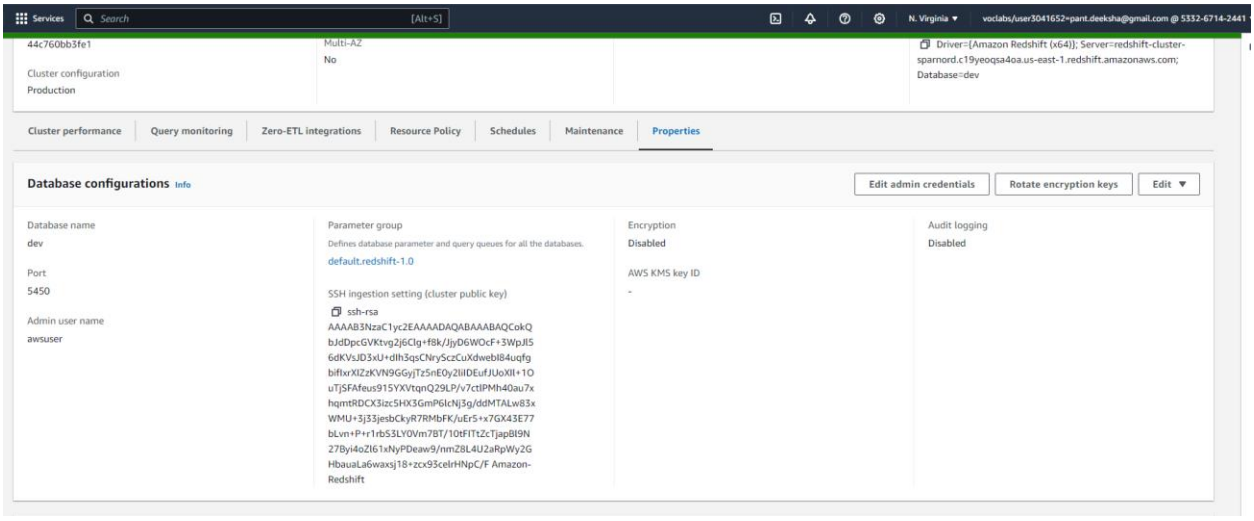

Cluster permissions
Create an IAM role as the default for this cluster that has the AmazonRedshiftAllCommandsFullAccess policy attached. This policy includes permissions to run SQL commands to COPY, UNLOAD, and query data with Amazon Redshift. The policy also grants permissions to run SELECT statements for related services, such as Amazon S3, Amazon CloudWatch logs, Amazon SageMaker, and AWS Glue.
Associated IAM roles (1)
Set default
Manage IAM roles
Create, associate, or remove an IAM role. You can associate up to 50 IAM roles. You can also choose an IAM role and set it as the default for this cluster.
Search for associated IAM role by name, status, or role type
IAM roles
Status
Role type
myRedshiftRole
Not applied

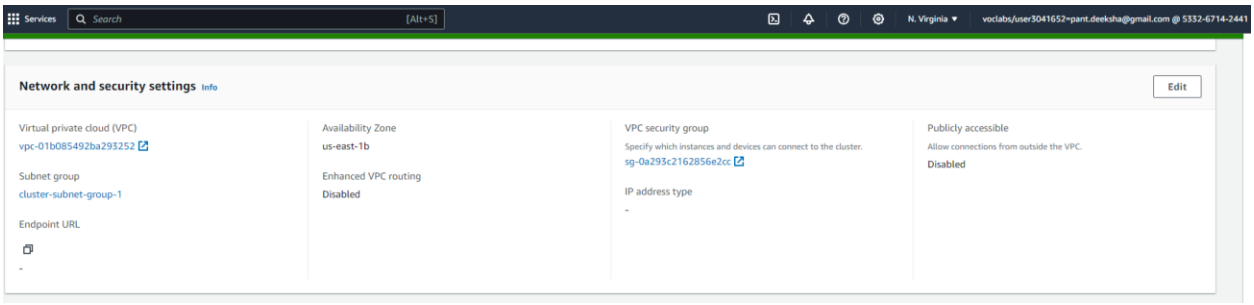

Network and security
Virtual private cloud (VPC)
This VPC defines the virtual networking environment for this cluster.
Default VPC
vpc-01b085492ba293252
You can't change the VPC associated with this cluster after the cluster has been created.
VPC security groups
Choose one or more security groups
default
sg-0a293c2162856e2cc
Cluster subnet group
cluster-subnet-group-1
Create new subnet group
Availability Zone
No preference
Enhanced VPC routing
Turn off
Turn on
Publicly accessible

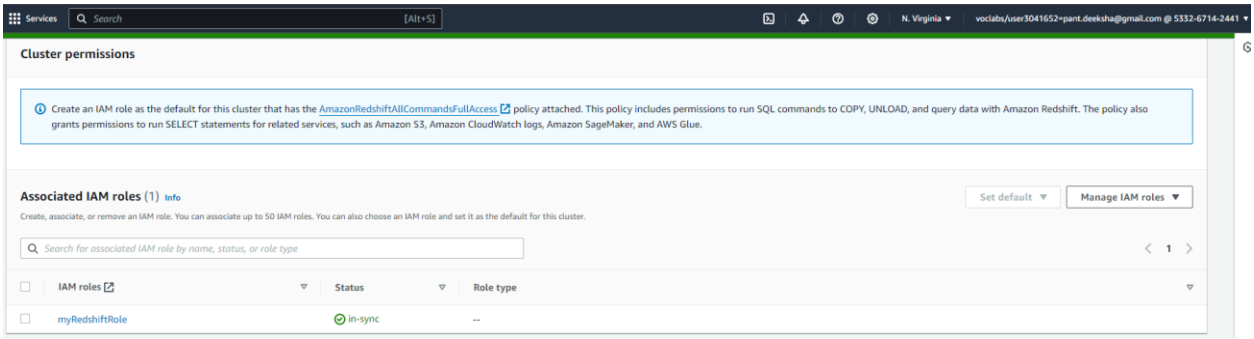

Services
vpc
N. Virginia
voclabs/user3041652=pant.deeksha@gmail.com @ 5332-6714-2441
Database configurations Info
Database name
Specify a database name to create an additional database.
dev
The name must be 1-64 alphanumeric characters (lowercase only), and it can't be a reserved word.
Database port
Port number where the database accepts inbound connections. The default port is 5439.
5450
Choose a port number between 1150 and 65535.
Parameter groups
Defines database parameter and query queues for all the databases.
default.redshift-1.0
Encryption
Encrypt all data on your cluster.
Disabled
Use AWS Key Management Service (AWS KMS)
Use a hardware security module (HSM)

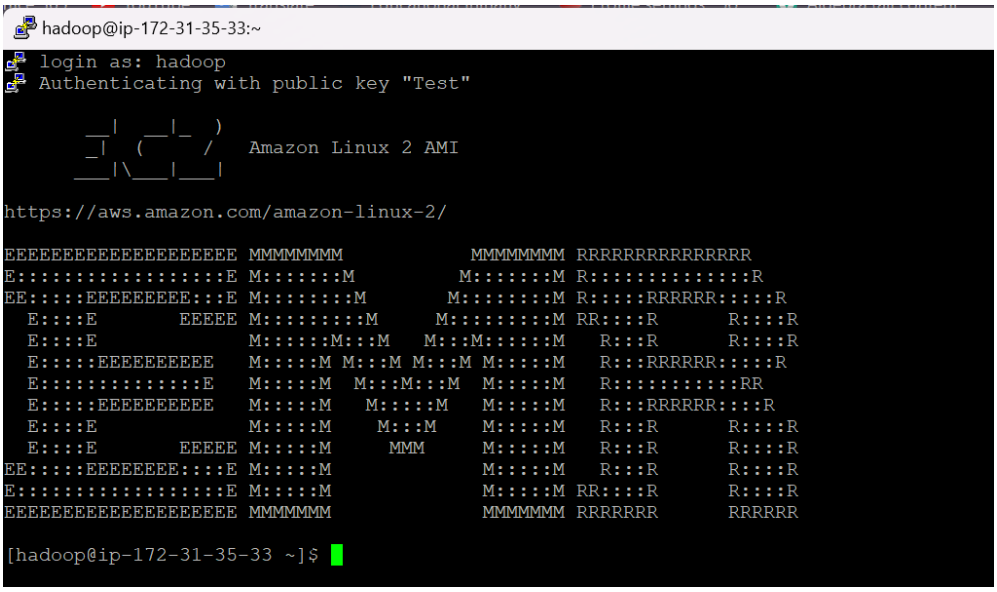
REDSHIFT


Services
Search
[Alt+S]
N. Virginia
voclabs/user3041652=pant.deeksha@gmail.com @ 5332-6714-2441
redshift-cluster-sparnord has been successfully created.
Amazon Redshift > Clusters > redshift-cluster-sparnord
redshift-cluster-sparnord
Actions
Edit
Add partner integration
Query data
General information Info
Cluster identifier
redshift-cluster-sparnord
Custom domain name
-
Cluster namespace ARN
arn:aws:redshift:us-east-1:533267142441:namespace:6fd43707-3f34-4eb2-a1a4-44c760bb3fe1
Cluster configuration
Production
Status
Available
Date created
April 28, 2024, 09:37 (UTC+04:00)
Storage used
-
Multi-AZ
No
Node type
dc2.large
Number of nodes
2
Endpoint
redshift-cluster-sparnord.c19yeoqsa4oa.us-east-1.redshift.amazonaws.com:5450/dev
JDBC URL
jdbc:redshift://redshift-cluster-sparnord.c19yeoqsa4oa.us-east-1.redshift.amazonaws.com:5450/dev
ODBC URL
Driver={Amazon Redshift (x64)}; Server=redshift-cluster-sparnord.c19yeoqsa4oa.us-east-1.redshift.amazonaws.com; Database=dev
Cluster performance
Query monitoring
Zero-ETL integrations
Resource Policy
Schedules
Maintenance
Properties
Recommendations (0)
To improve performance and decrease operating costs, the Amazon Redshift Advisor provides recommendations.


Services
Search
[Alt+S]
N. Virginia
voclabs/user3041652=pant.deeksha@gmail.com @ 5332-6714-2441
44c760bb3fe1
Cluster configuration
Production
Multi-AZ
No
Driver={Amazon Redshift (x64)}; Server=redshift-cluster-sparnord.c19yeoqsa4oa.us-east-1.redshift.amazonaws.com; Database=dev
Cluster performance
Query monitoring
Zero-ETL integrations
Resource Policy
Schedules
Maintenance
Properties
Database configurations Info
Edit admin credentials
Rotate encryption keys
Edit
Database name
dev
Port
5450
Admin user name
awsuser
Parameter group
Defines database parameter and query queues for all the databases.
default.redshift-1.0
SSH ingestion setting (cluster public key)
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCokQ
bJdDpcGVKtvg2j6CIg+f8k/JjyD6WOcF+3WpJI5
6dKVsJD3xU+dIh3qsCNrySczCuXdwebl84uqfg
biflxrXIZzKVN9GGyjTz5nE0y2liIDEufJUoXII+1O
uTjSFAfeus915YXVtqnQ29LP/v7ctIPMh40au7x
hqmtRDCX3izc5HX3GmP6IcNj3g/ddMTALw83x
WMU+3j33jesbCkyR7RMbFK/uEr5+x7GX43E77
bLvn+P+r1rb53LY0Vm7BT/10tFITtZcTjapBI9N
27Byi4oZl61xNyPDeaw9/nmZ8L4U2aRpWy2G
HbauaLa6waxsj18+zcx93celrHNpC/F Amazon-Redshift
Encryption
Disabled
AWS KMS key ID
-
Audit logging
Disabled

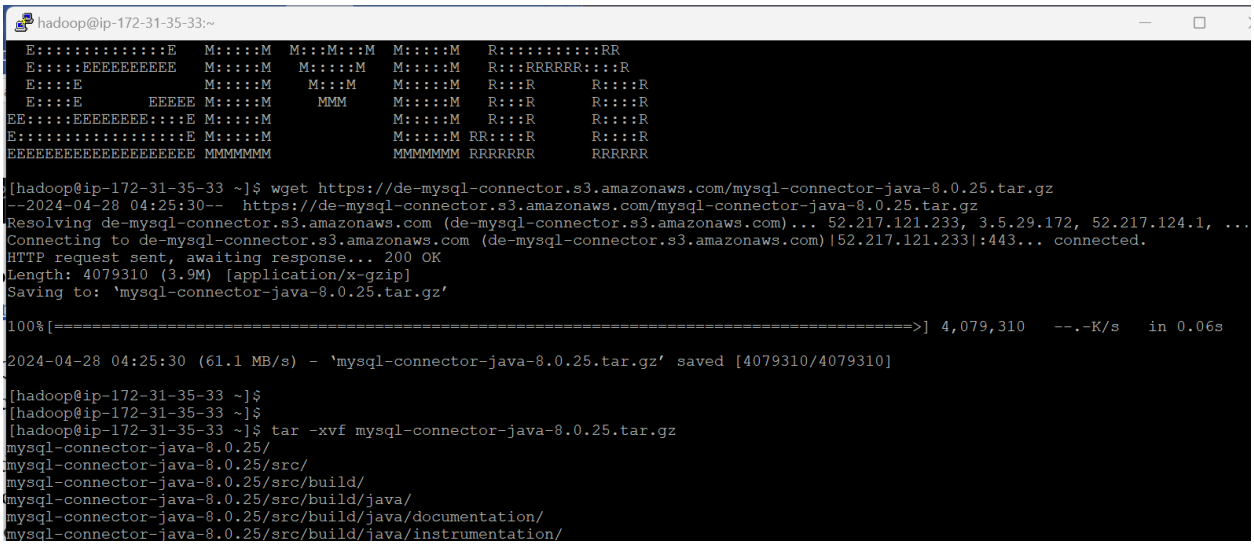






Executed the following command to install MySQL connector jar file
wget https://de-mysql-connector.s3.amazonaws.com/mysql-connector-java-8.0.25.tar.gz

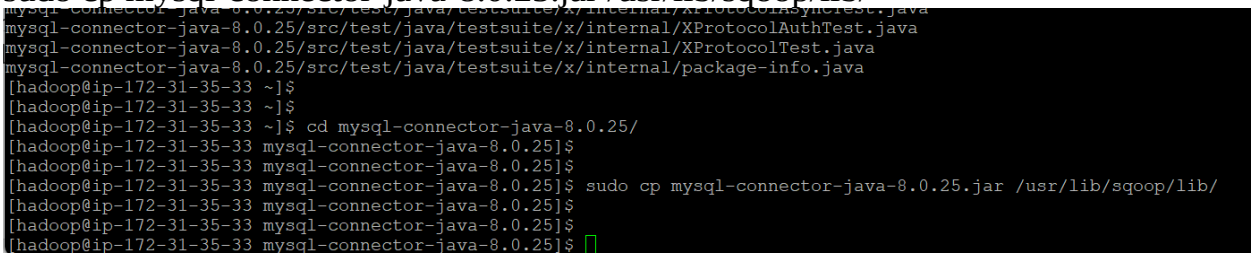
Extracted MySQL connector tar file:
tar -xvf mysql-connector-java-8.0.25.tar.gz



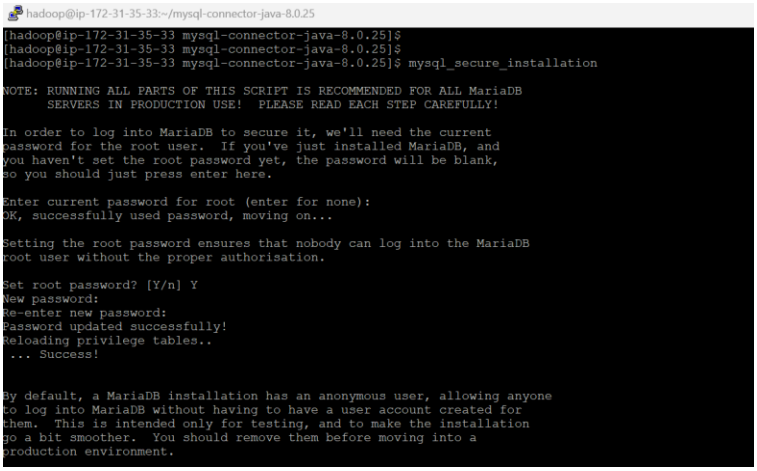
Changed the directory to the MySQL Connector directory and copied to the Sqoop library as follows:

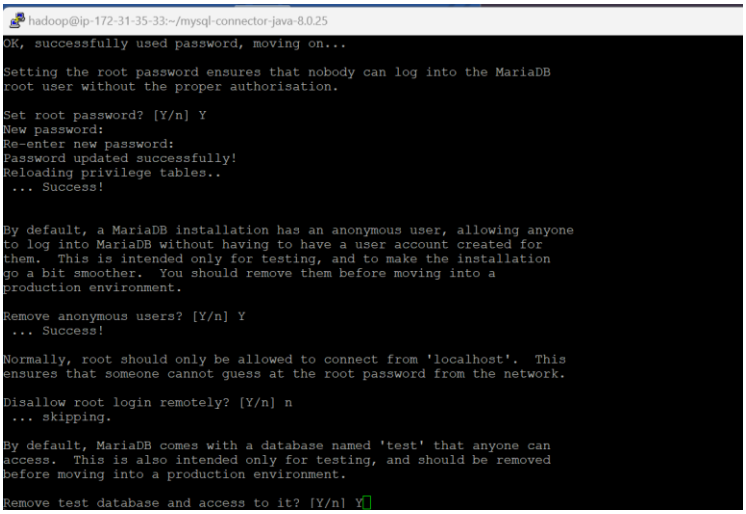
cd mysql-connector-java-8.0.25/

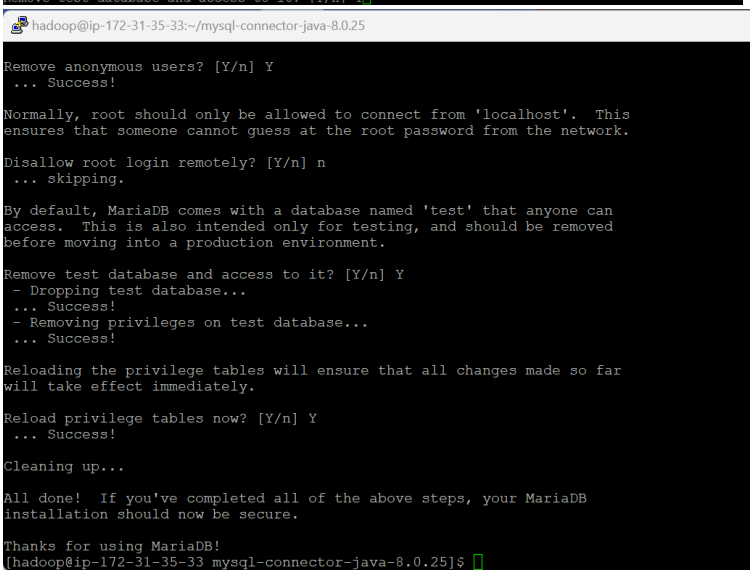
sudo cp mysql-connector-java-8.0.25.jar /usr/lib/sqoop/lib/



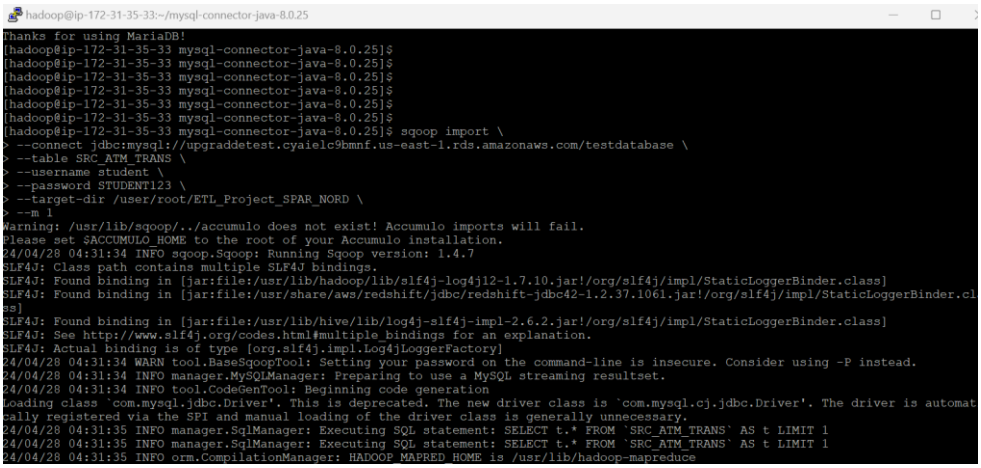
mysql_secure_installation

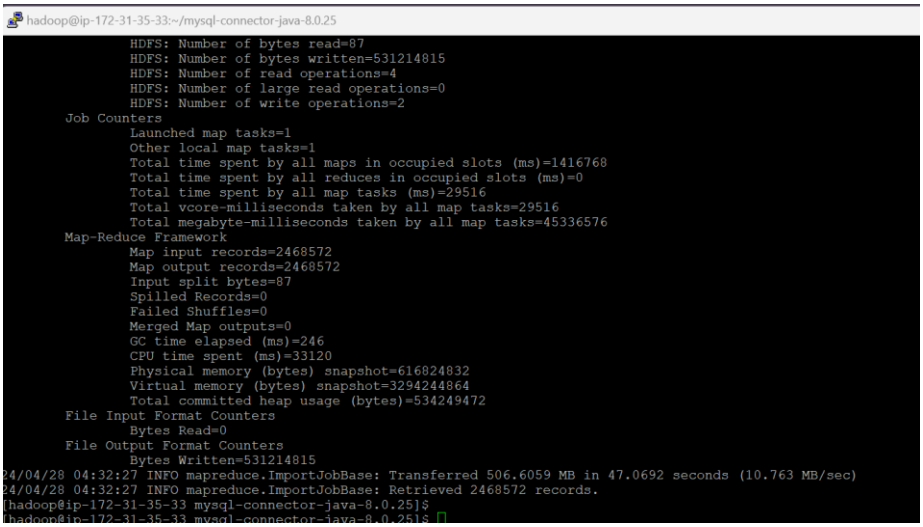




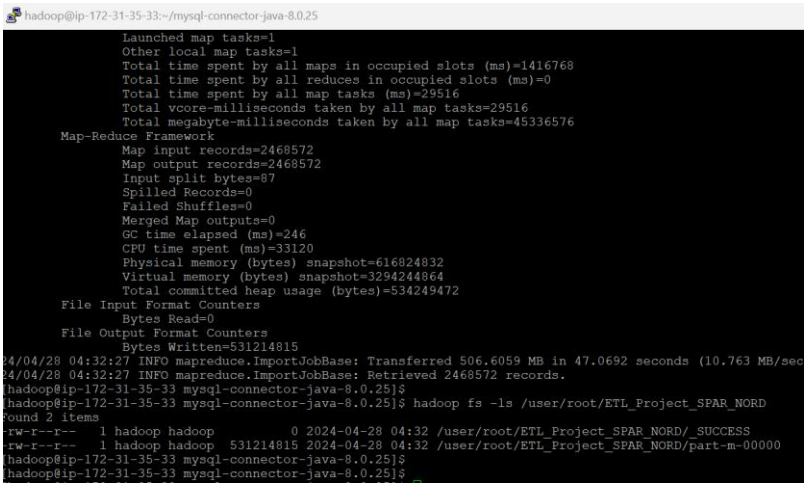



```
sqoop import \
--connect jdbc:mysql://upgraddetest.cyaielc9bmnf.us-east-1.rds.amazonaws.com/testdatabase \
--table SRC_ATM_TRANS \
--username student \
--password STUDENT123 \
--target-dir /user/root/ETL_Project_SPAR_NORD \
--m 1
```

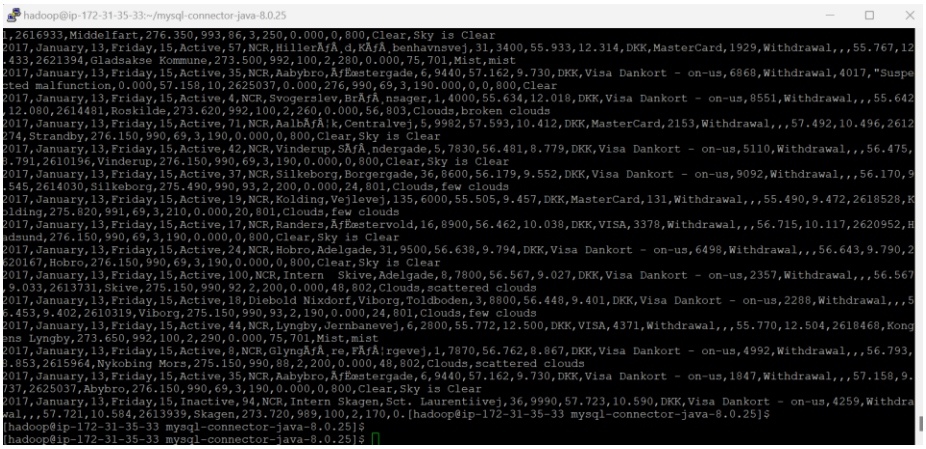




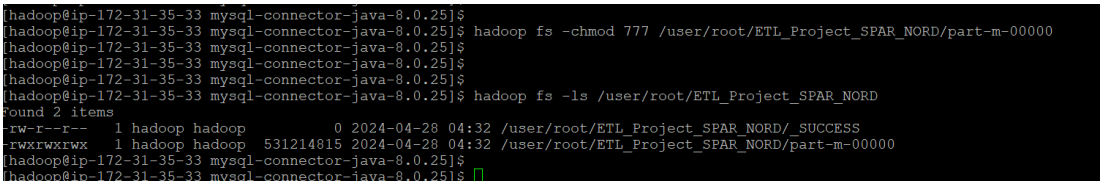
hadoop fs -ls /user/root/ETL_Project_SPAR_NORD



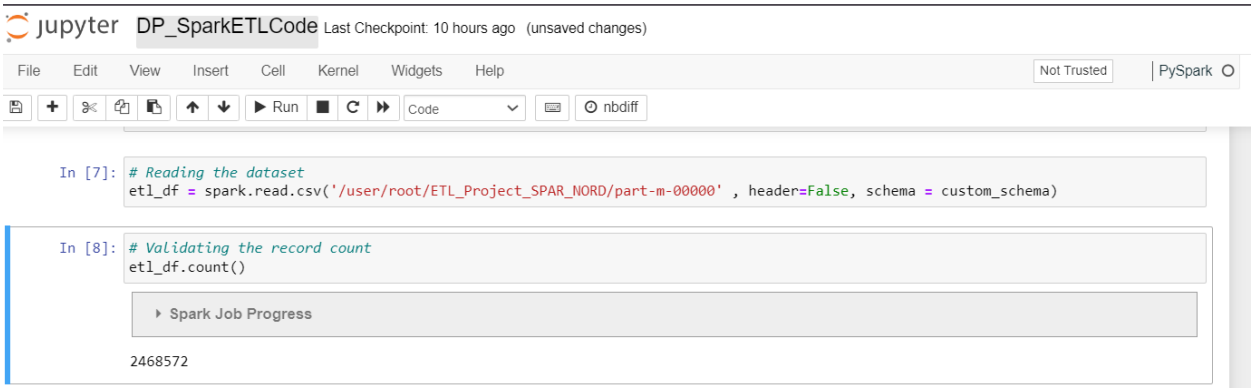
hadoop fs -cat /user/root/ETL_Project_SPAR_NORD/part-m-00000



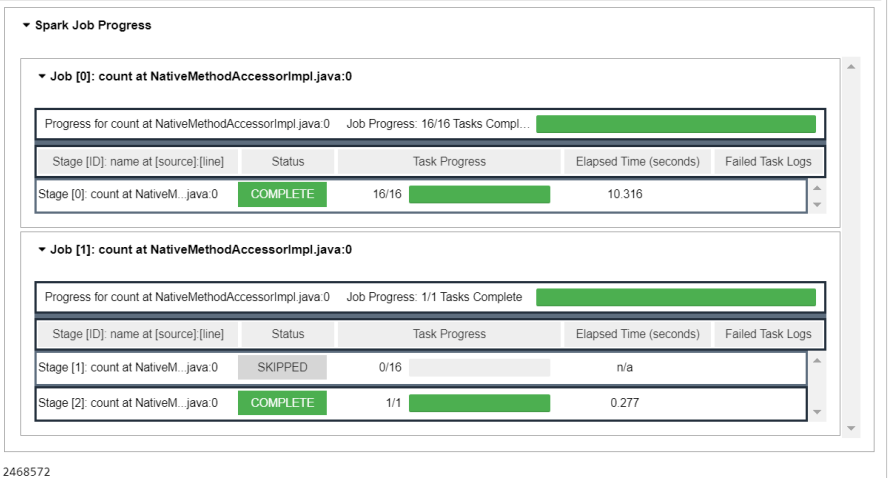
hadoop fs -chmod 777 /user/root/ETL_Project_SPAR_NORD/part-m-00000



Checking count after importing data into a dataframe



```
# Validating the record count
etl_df.count()
```




```
2468572
```

Checking count for the Location Dimension:

```
df_location.orderBy(df_location['location']).show(5)
```

Spark Job Progress

```
+-----------+--------------------+------------+-------------+-------+-------+-------+
|location_id|            location|  streetname|street_number|zipcode|atm_lat|atm_lon|
+-----------+--------------------+------------+-------------+-------+-------+-------+
|          1|             Aabybro|Ã¿Ëæstergade|            6|   9440| 57.162|   9.73|
|          2|      Aalborg Hallen|Europa Plads|            4|   9000| 57.044|  9.913|
|          3|Aalborg Storcente...|    Hobrovej|          452|   9200| 57.005|  9.876|
|          4|Aalborg Storcente...|    Hobrovej|          452|   9200| 57.005|  9.876|
|          5|         Aalborg Syd|    Hobrovej|          440|   9200| 57.005|  9.881|
+-----------+--------------------+------------+-------------+-------+-------+-------+
only showing top 5 rows
```

```
n [18]: # Validating the record count
df_location.count()
```

Spark Job Progress

```
109
```

Checking count for the ATM Dimension

```
In [26]: # Validating the record count
DIM_ATM.count()
```

Spark Job Progress

```
115
```

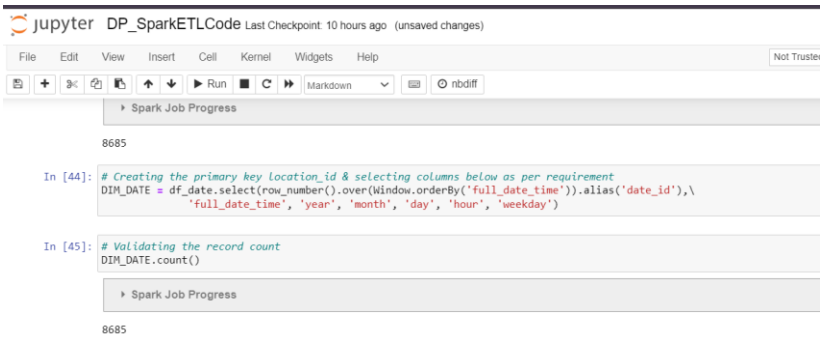
```
In [27]: # Renaming the columns as below in location dimension
DIM_LOCATION = df_location.withColumnRenamed('atm_lat','lat')\
                          .withColumnRenamed('atm_lon','lon')
```

```
In [28]: # Validating the record count
DIM_LOCATION.count()
```

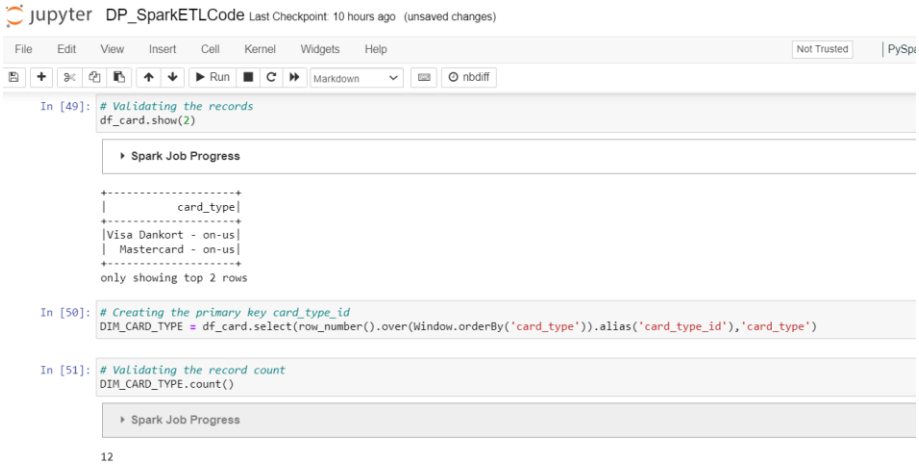
Spark Job Progress

```
109
```

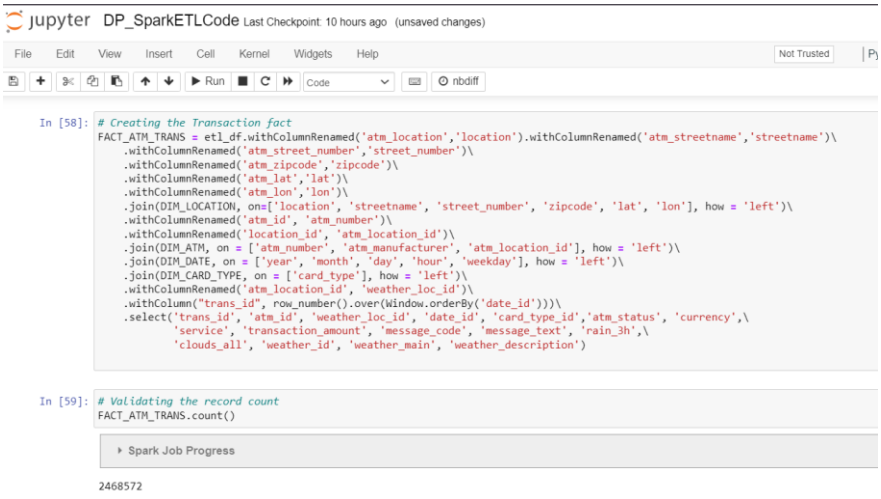
Checking count for the DATE Dimension:



Checking count for the CARD TYPE Dimension:



Checking count for the ATM_TRANS fact



Loading the dimension and fact tables into Amazon S3 bucket:

jupyter DP_SparkETLCode Last Checkpoint: 10 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help Trusted PySpark

Run Code nbdiff

Loading the dimension and fact tables into Amazon S3 bucket ¶

In [62]:
```
DIM_LOCATION.coalesce(1).write.format('csv').option('header','false').save('s3://etl-project/dim_location', mode='overwrite')
```

Spark Job Progress

In [63]:
```
# writing data from pyspark df 'dim_atm' in csv format to dim_atm folder in S3 bucket 'upgrad-bucket-s3'
DIM_ATM.coalesce(1).write.format('csv').option('header','false').save('s3://etl-project/dim_atm', mode='overwrite')
```

Spark Job Progress

In [64]:
```
# writing data from pyspark df 'dim_data' in csv format to dim_data folder in S3 bucket 'upgrad-bucket-s3'
DIM_DATE.coalesce(1).write.format('csv').option('header','false').save('s3://etl-project/dim_date', mode='overwrite')
```

Spark Job Progress

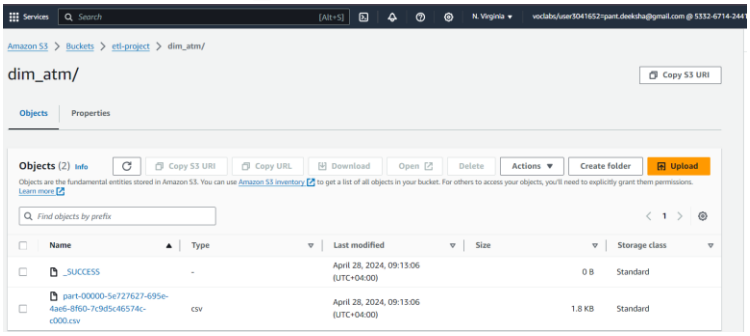
In [65]:
```
# writing data from pyspark df 'dim_card_type' in csv format to dim_card_type folder in S3 bucket 'upgrad-bucket-s3'
DIM_CARD_TYPE.coalesce(1).write.format('csv').option('header','false').save('s3://etl-project/dim_card_type', mode='overwrite')
```

Spark Job Progress

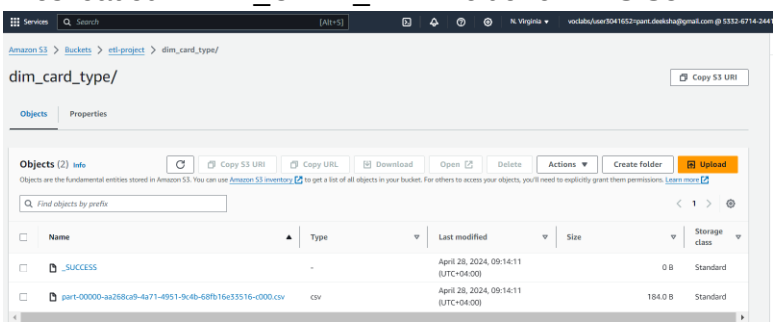
In [66]:
```
# writing data from pyspark df 'fact_atm_trans' in csv format to fact_atm_trans folder in S3 bucket 'upgrad-bucket-s3'
FACT_ATM_TRANS.coalesce(1).write.format('csv').option('header','false').save('s3://etl-project/fact_atm_trans', mode='overwrite')
```

Spark Job Progress

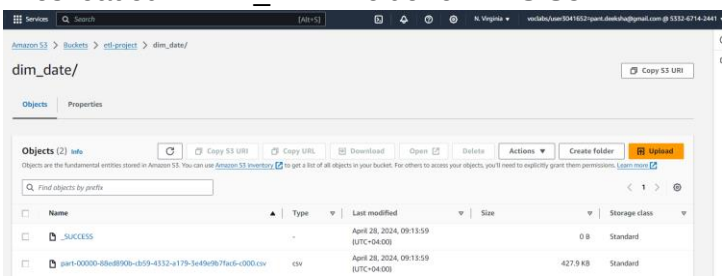
Files loaded in DIM_ATM folder of AWS S3:



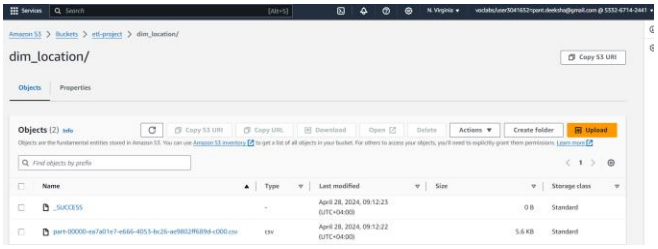
Files loaded in DIM_CARD_TYPE folder of AWS S3:



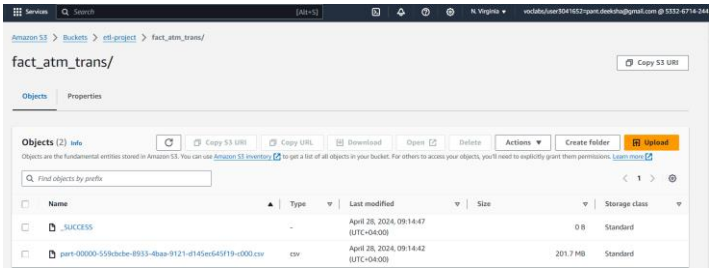
Files loaded in DIM_DATE folder of AWS S3:

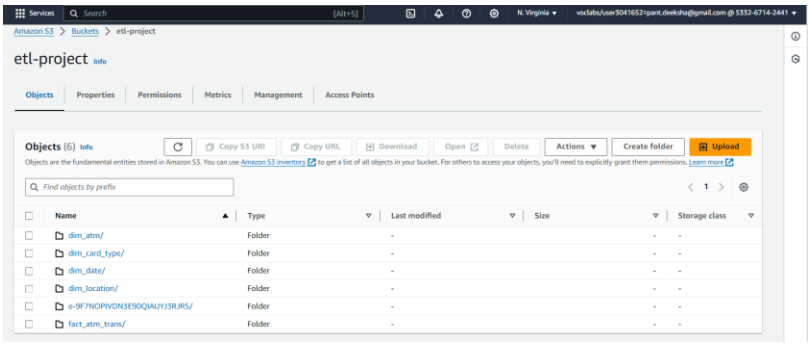


Files loaded in DIM_LOCATION folder of AWS S3:



Files loaded in FACT_ATM_TRANS folder of AWS S3:





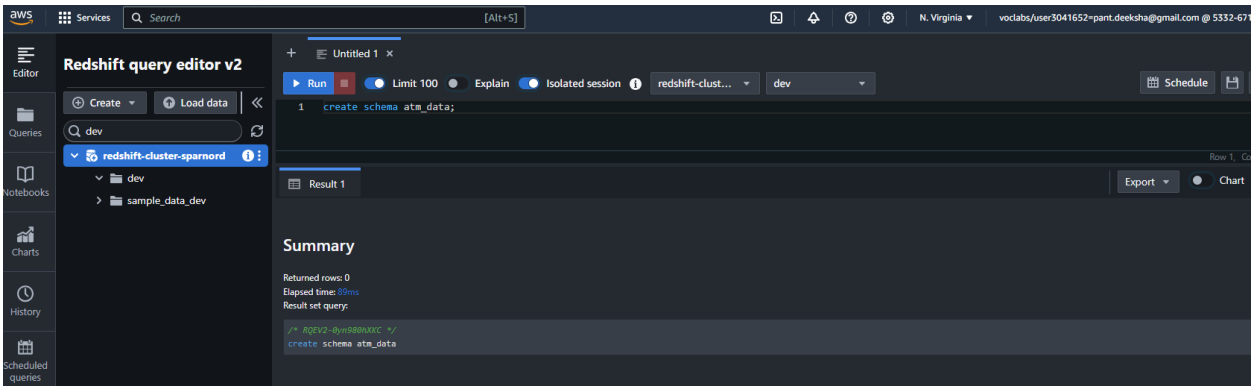
**********\

Setting up a database in the Redshift cluster and running queries to create the dimension and fact tables

Queries to create the various dimension and fact tables with appropriate primary and foreign keys:

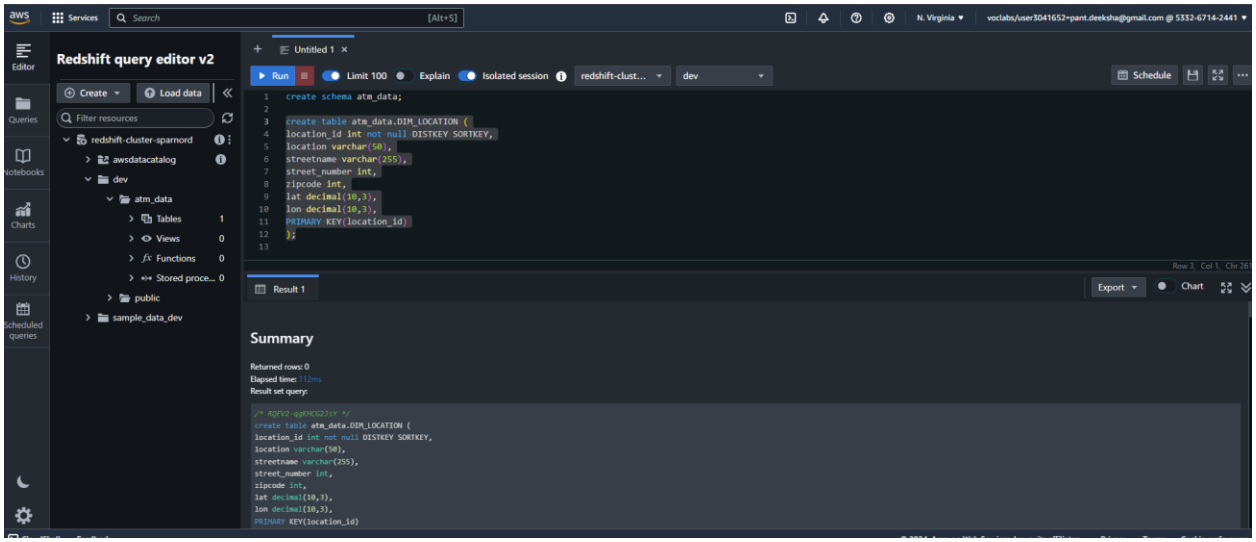
1. Creating schema:

create schema atm_data;



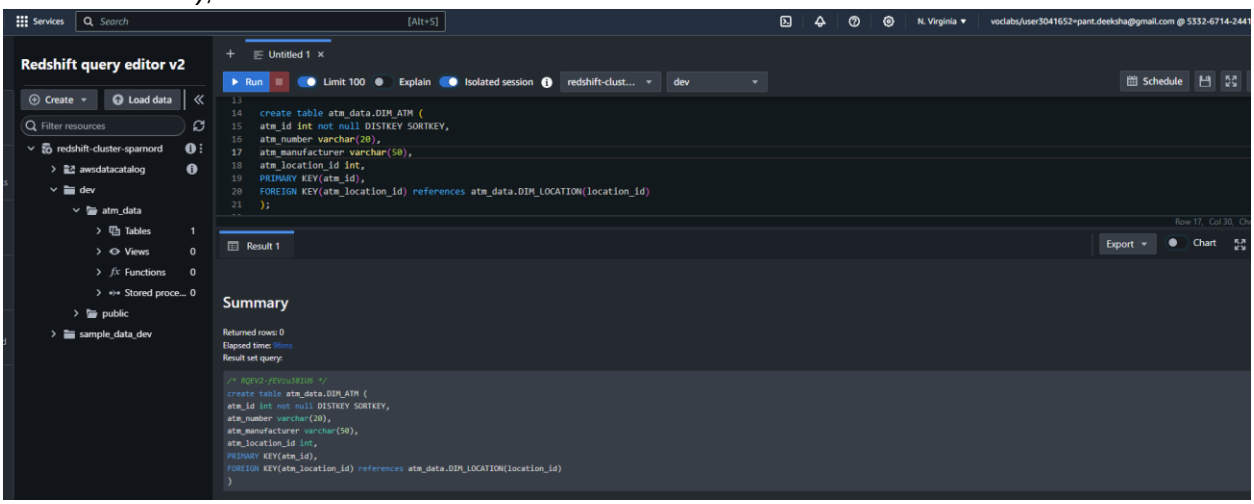
2. DIM_LOCATION

```
create table atm_data.DIM_LOCATION (
        location_id int not null DISTKEY SORTKEY,
        location varchar(50),
        streetname varchar(255),
        street_number int,
        zipcode int,
        lat decimal(10,3),
        lon decimal(10,3),
        PRIMARY KEY(location_id)
        );
```



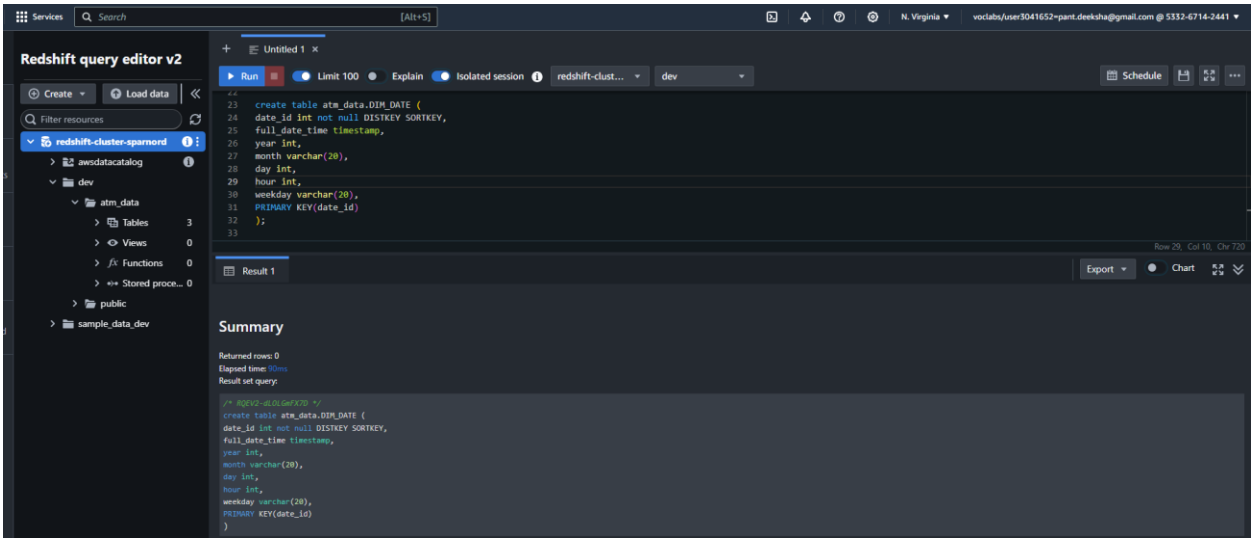
3. DIM_ATM

```
create table atm_data.DIM_ATM (
        atm_id int not null DISTKEY SORTKEY,
        atm_number varchar(20),
        atm_manufacturer varchar(50),
        atm_location_id int,
        PRIMARY KEY(atm_id),
        FOREIGN KEY(atm_location_id) references
        atm_data.DIM_LOCATION(location_id)
        );
```



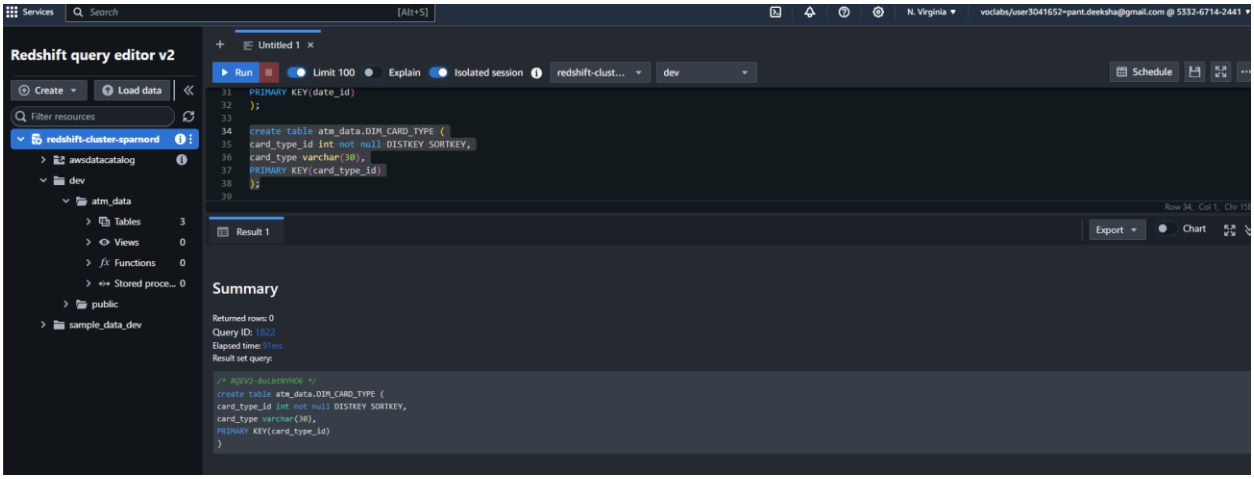
4. DIM_DATE:

```
create table atm_data.DIM_DATE (
        date_id int not null DISTKEY SORTKEY,
        full_date_time timestamp,
        year int,
        month varchar(20),
        day int,
        hour int,
        weekday varchar(20),
        PRIMARY KEY(date_id)
        );
```



5. DIM_CARD_TYPE

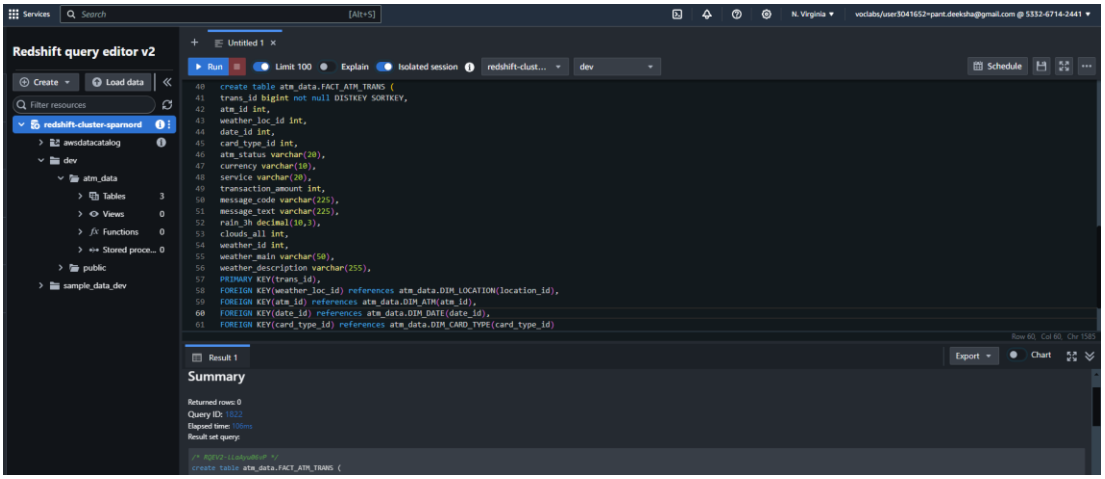
```
create table atm_data.DIM_CARD_TYPE (
        card_type_id int not null DISTKEY SORTKEY,
        card_type varchar(30),
        PRIMARY KEY(card_type_id)
        );
```

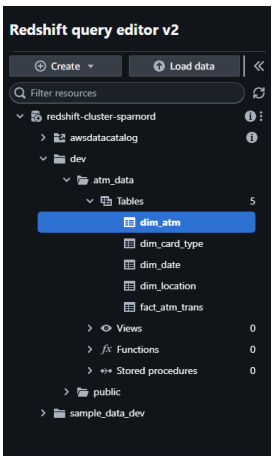


6. FACT_ATM_TRANS

```
create table atm_data.FACT_ATM_TRANS (
        trans_id bigint not null DISTKEY SORTKEY,
        atm_id int,
        weather_loc_id int,
        date_id int,
        card_type_id int,
```

```
atm_status varchar(20),
currency varchar(10),
service varchar(20),
transaction_amount int,
message_code varchar(225),
message_text varchar(225),
rain_3h decimal(10,3),
clouds_all int,
weather_id int,
weather_main varchar(50),
weather_description varchar(255),
PRIMARY KEY(trans_id),
FOREIGN KEY(weather_loc_id) references
atm_data.DIM_LOCATION(location_id),
FOREIGN KEY(atm_id) references atm_data.DIM_ATM(atm_id),
FOREIGN KEY(date_id) references atm_data.DIM_DATE(date_id),
FOREIGN KEY(card_type_id) references
atm_data.DIM_CARD_TYPE(card_type_id)
);
```

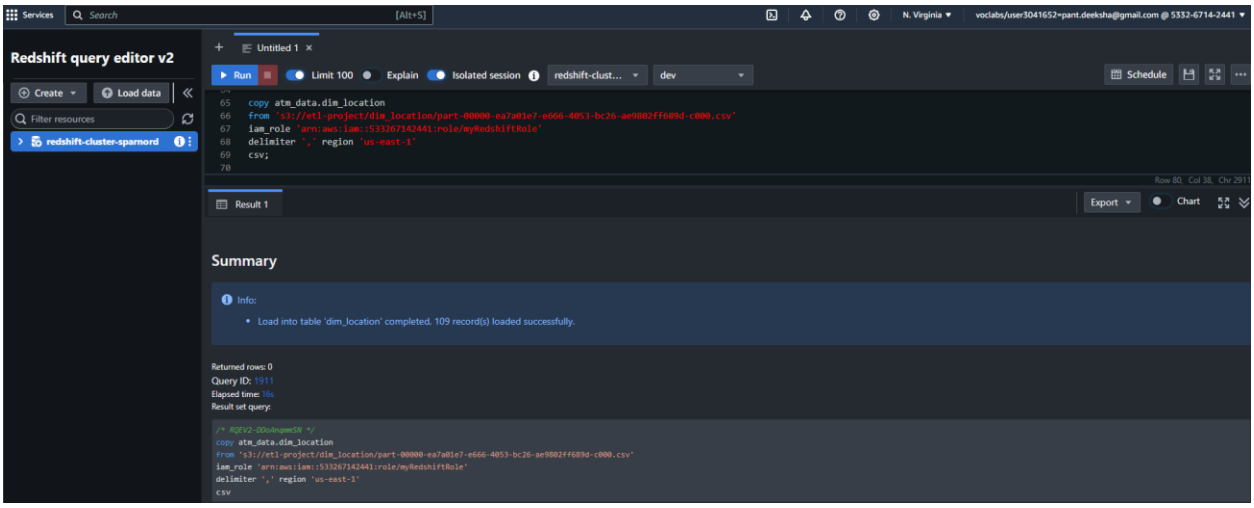




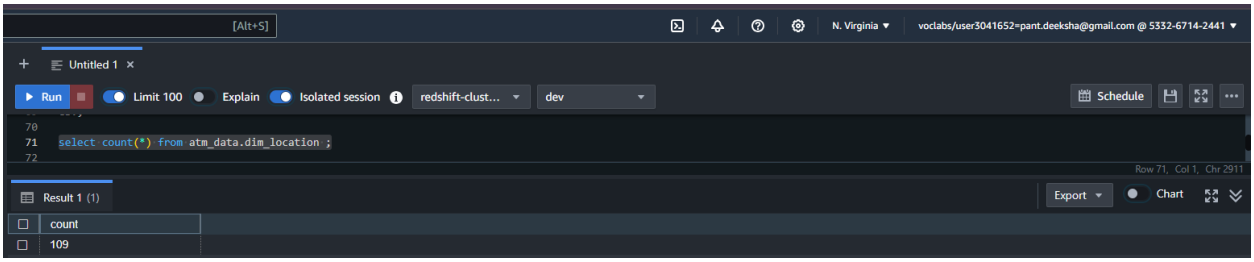
Loading data into a Redshift cluster from Amazon S3 bucket

Queries to copy the data from S3 buckets to the Redshift cluster in the appropriate tables

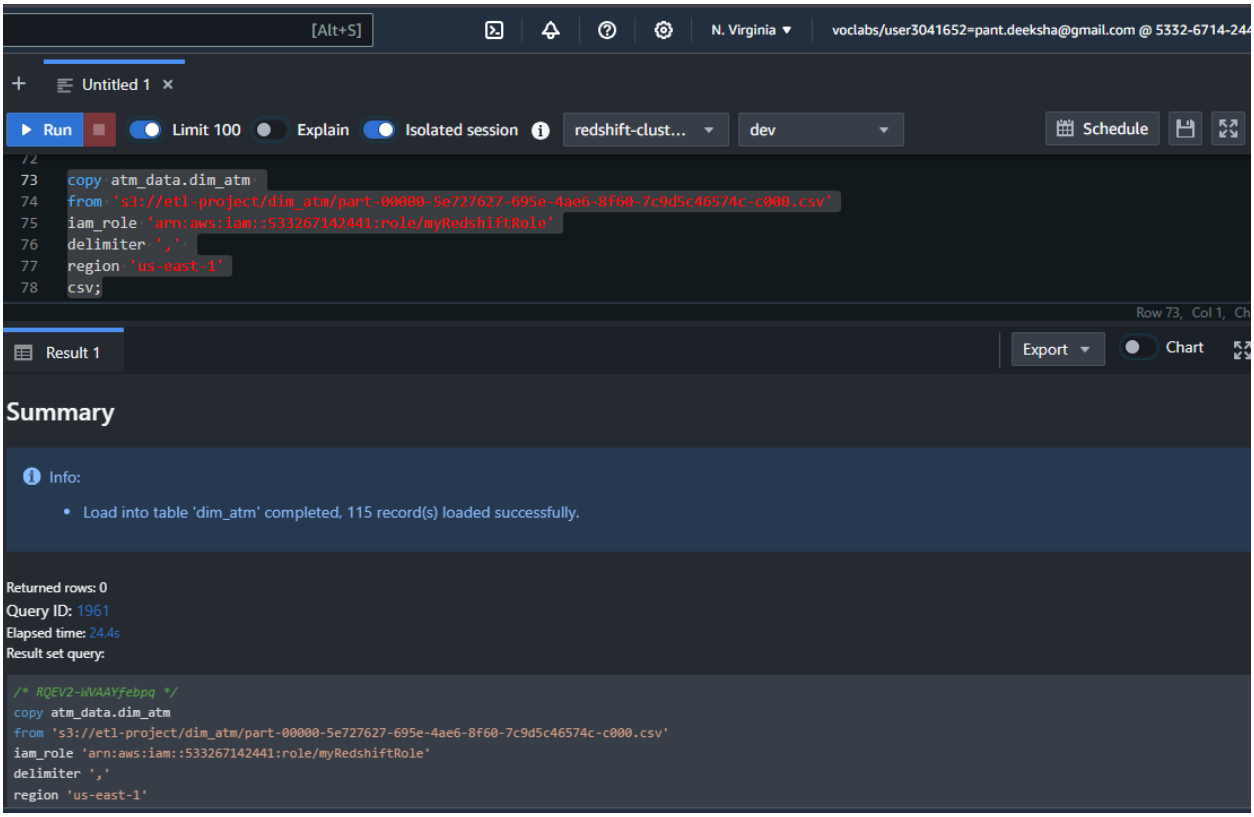
```
copy atm_data.dim_location
from 's3://etl-project/dim_location/part-00000-ea7a01e7-e666-4053-bc26-ae9802ff689d-c000.csv'
iam_role 'arn:aws:iam::533267142441:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
csv;
```



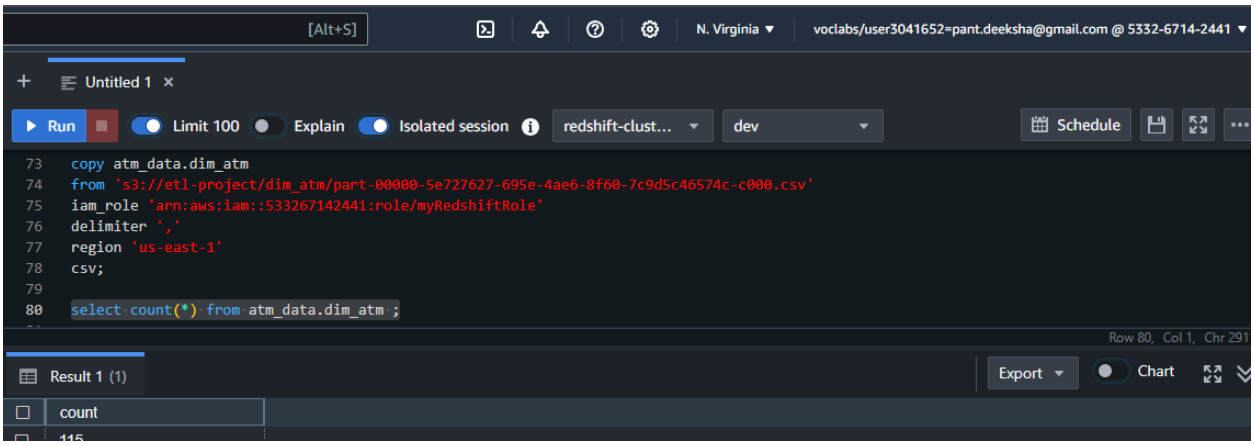

```
select count(*) from atm_data.dim_location ;
```



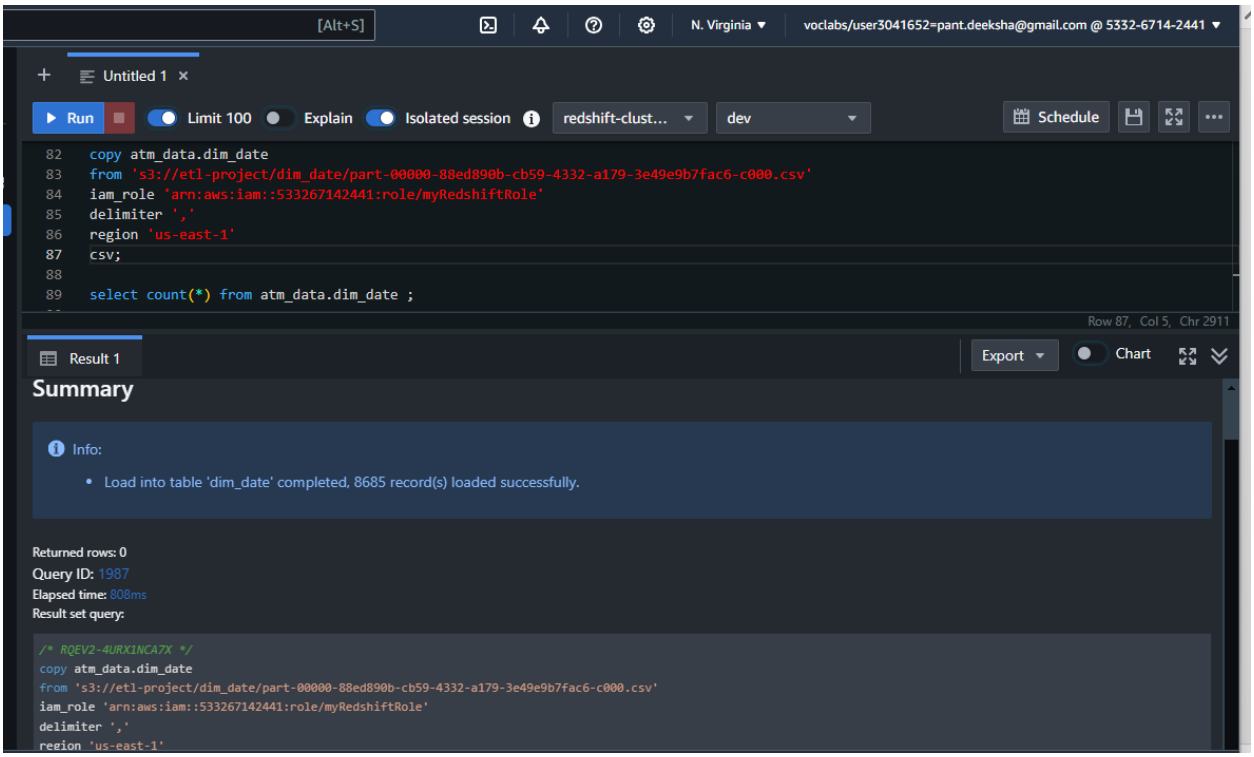

```
copy atm_data.dim_atm
from 's3://etl-project/dim_atm/part-00000-5e727627-695e-4ae6-8f60-7c9d5c46574c-c000.csv'
iam_role 'arn:aws:iam::533267142441:role/myRedshiftRole'
delimiter ','
region 'us-east-1'
csv;
```



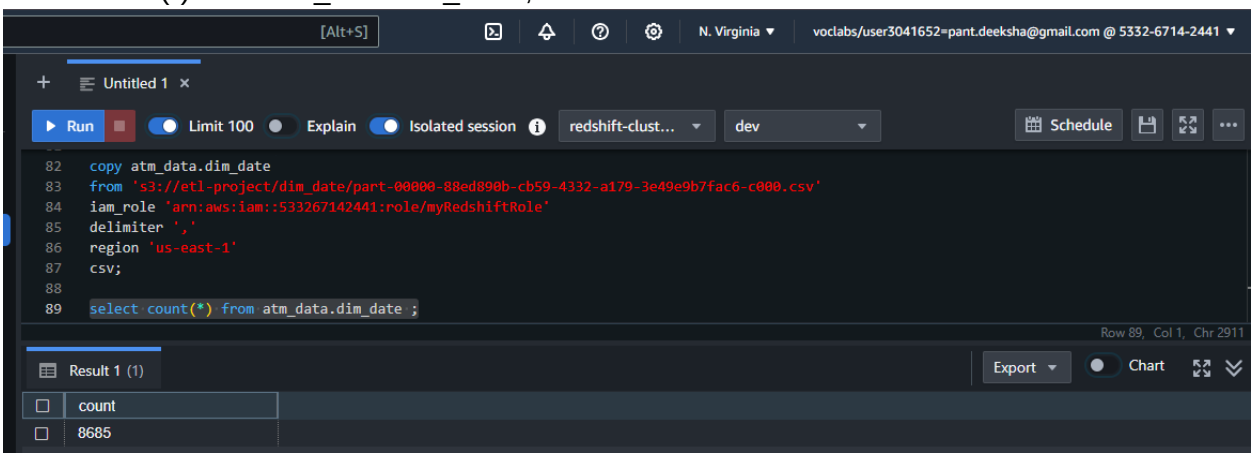
select count(*) from atm_data.dim_atm ;



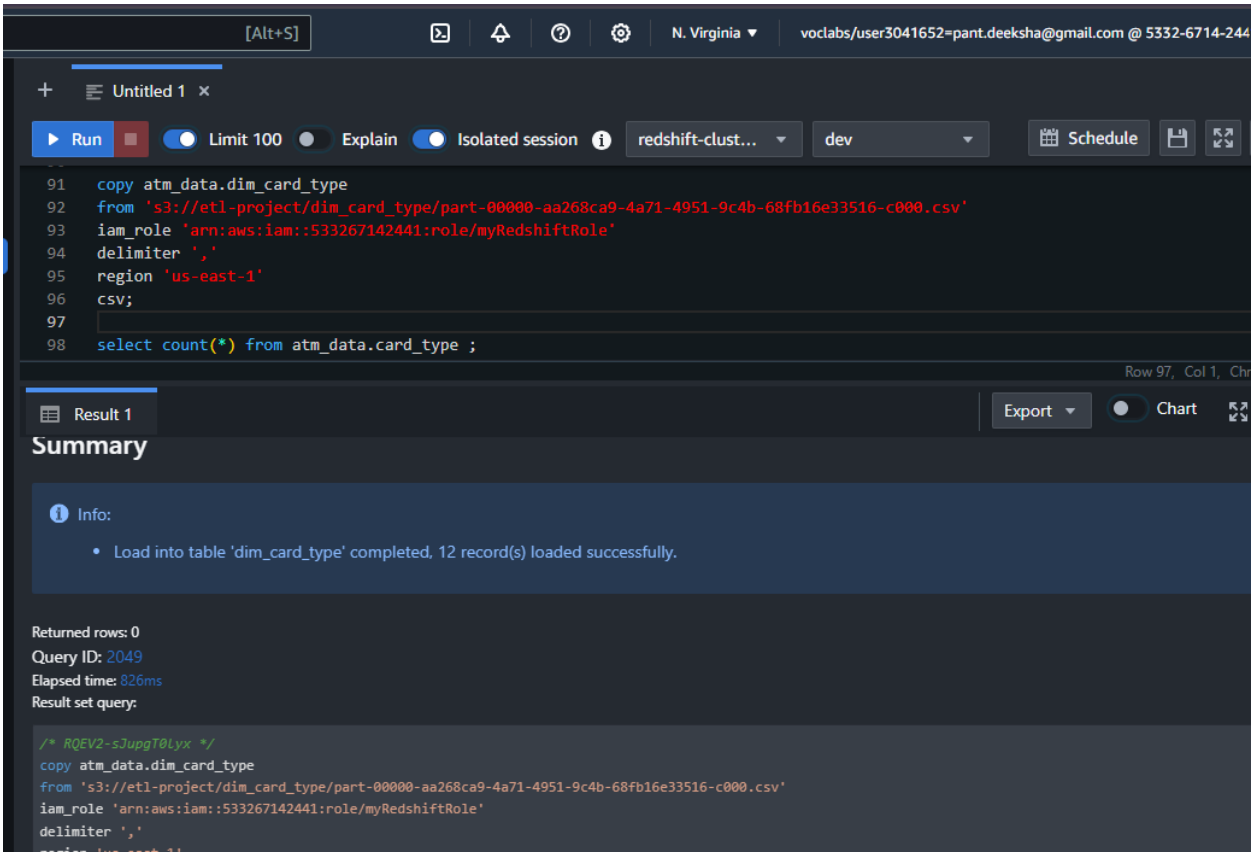
copy atm_data.dim_date
from 's3://etl-project/dim_date/part-00000-88ed890b-cb59-4332-a179-3e49e9b7fac6-c000.csv'
iam_role 'arn:aws:iam::533267142441:role/myRedshiftRole'
delimiter ','
region 'us-east-1'
csv;



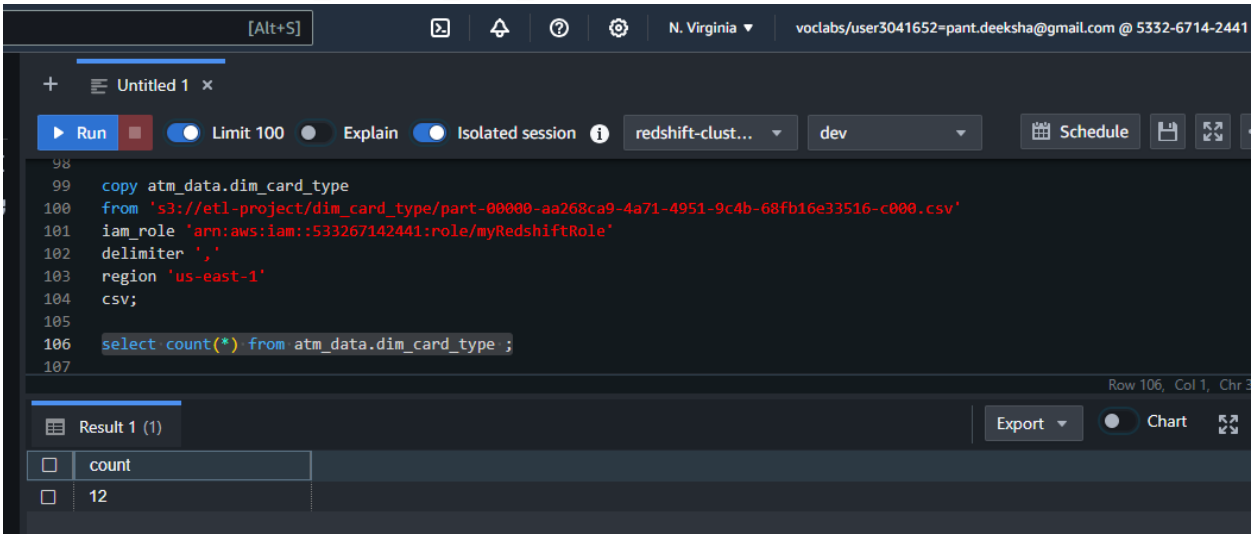
select count(*) from atm_data.dim_date ;



copy atm_data.dim_card_type
from 's3://etl-project/dim_card_type/part-00000-aa268ca9-4a71-4951-9c4b-68fb16e33516-c000.csv'
iam_role 'arn:aws:iam::533267142441:role/myRedshiftRole'
delimiter ','
region 'us-east-1'
csv;

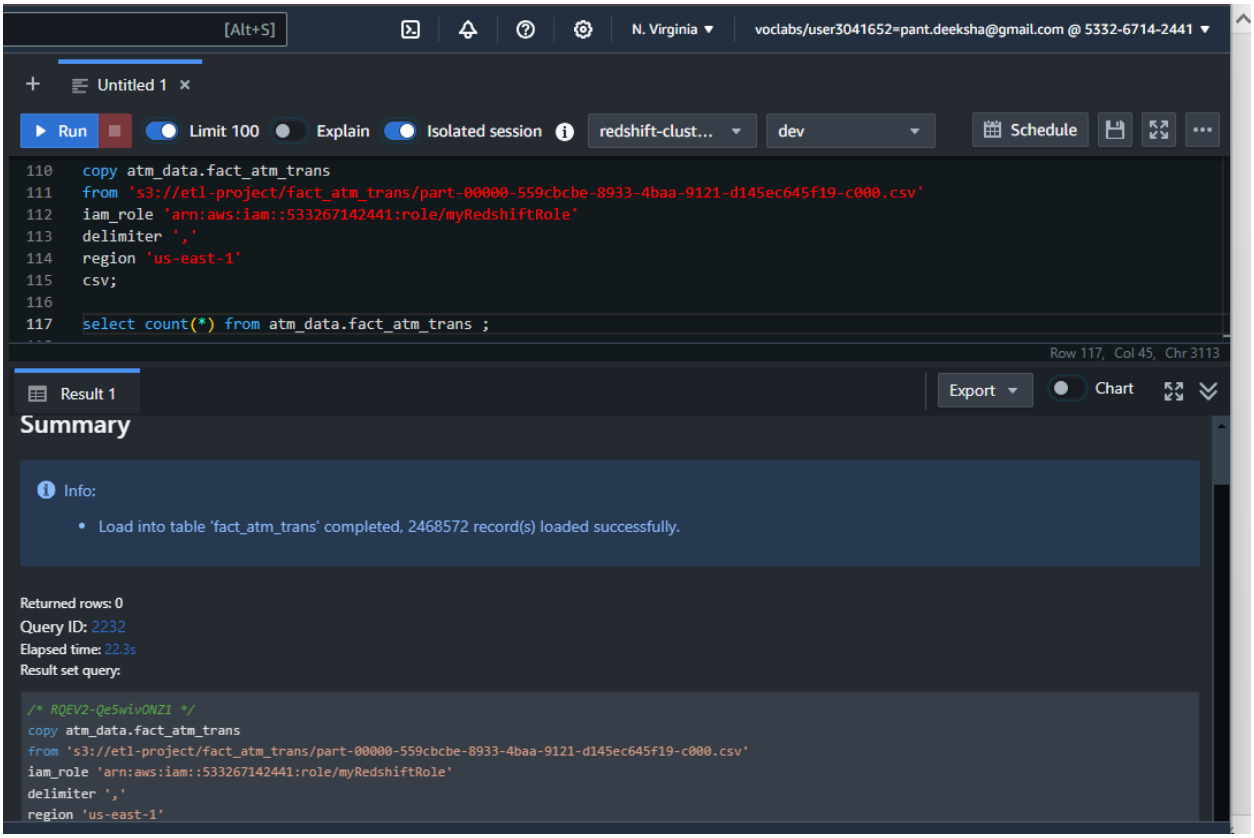


select count(*) from atm_data.dim_card_type ;



copy atm_data.fact_atm_trans
from 's3://etl-project/fact_atm_trans/part-00000-559cbcbe-8933-4baa-9121-d145ec645f19-c000.csv'
iam_role 'arn:aws:iam::533267142441:role/myRedshiftRole'
delimiter ','
region 'us-east-1'

csv;



select count(*) from atm_data.fact_atm_trans ;