

Creation of a Redshift Cluster

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

[Alt+S]

N. Virginia ▼

voclabs/user2631029=abhijeet044@gmail.com @ 9713-78

Create cluster [Info](#)

Looking for free trial? Try Redshift Serverless. First-time Redshift Serverless customers receive a \$300 credit to use in their account.

Launch Redshift Serverless

×

Cluster configuration

Cluster identifier

This is the unique key that identifies a cluster.

etlredshiftcluster

The identifier must be from 1-63 characters. Valid characters are a-z (lowercase only) and - (hyphen).

Choose the size of the cluster

☒ I'll choose

☐ Help me choose

Node type [Info](#)

Choose a node type that meets your CPU, RAM, storage capacity, and drive type requirements.

dc2.large ▼

Database configurations

Admin user name

Enter a login ID for the admin user of your DB instance.

awsuser

The name must be 1-128 alphanumeric characters, and it can't be a [reserved word](#) [↗](#).

Admin password

Select an option to manage your admin password.

☐ Manage admin credentials in AWS Secrets Manager [Info](#)
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

.....

Must be 8-64 characters long. Must contain at least one uppercase letter, one lowercase letter and one number. Can be any printable ASCII character except "/", "", or "@".

☐ Show password

Associated IAM roles (1) [Info](#)

[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.

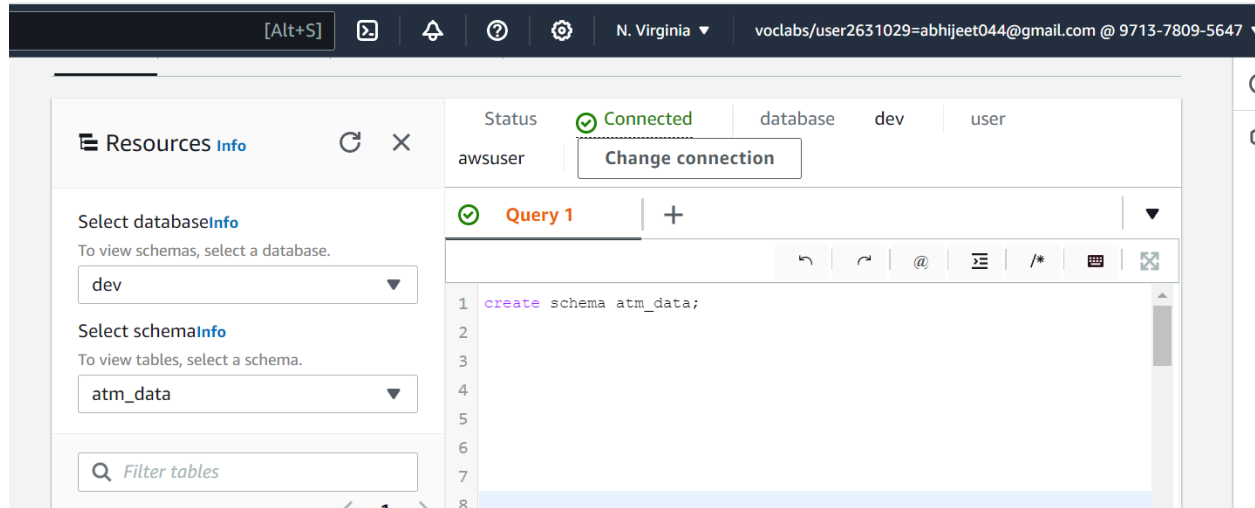
[<](#) 1 [>](#)

<input type="checkbox"/>	IAM roles ↗	▼	Status	▼	Role type	▼
<input type="checkbox"/>	myRedshiftRole		Not applied		--	

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

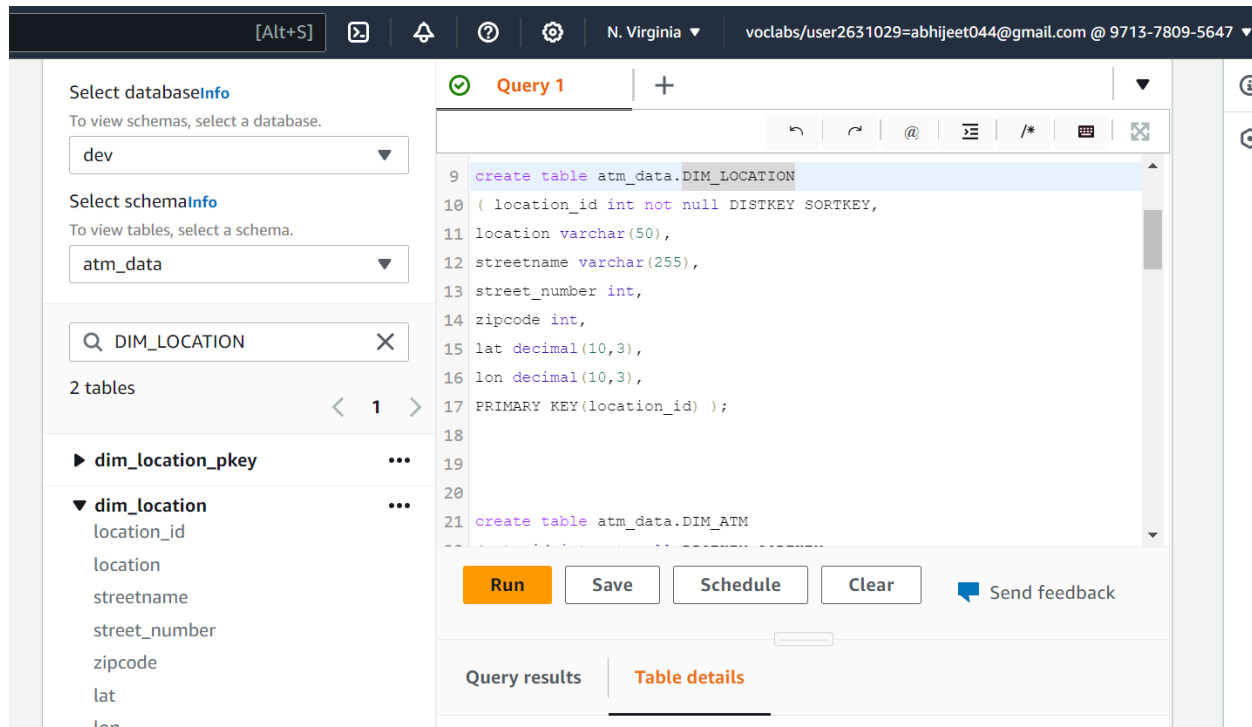
Query to create a schema for the dimension and fact tables:

- create schema atm_data;



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

- **Creating location dimension table**
 - 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));



The screenshot shows a SQL IDE interface with a dark theme. On the left, there's a sidebar with 'Select database' (dev) and 'Select schema' (atm_data). Below that, a search bar contains 'DIM_LOCATION' and shows '2 tables'. The first table, 'dim_location_pkey', is expanded, showing columns: location_id, location, streetname, street_number, zipcode, lat, and lon. The main editor area shows a SQL query for 'Query 1' with line numbers 9 to 21. The query creates a table 'atm_data.DIM_LOCATION' with columns: 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)), and a PRIMARY KEY on location_id. Below the query editor are buttons for 'Run', 'Save', 'Schedule', and 'Clear', along with a 'Send feedback' link. At the bottom, there are tabs for 'Query results' and 'Table details'.

- **Creating atm dimension table**
 - 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));

[Alt+S]

N. Virginia

voclabs/user2631029=abhijeet044@gmail.com @ 9713-7809-5647

Select databaseInfo

To view schemas, select a database.

dev

Select schemaInfo

To view tables, select a schema.

atm_data

Q DIM_ATM

2 tables

dim_atm_pkey

dim_atm

atm_id

atm_number

atm_manufacturer

atm_location_id

Query 1

21 create table atm_data.DIM_ATM

22 (atm_id int not null DISTKEY SORTKEY,

23 atm_number varchar(20),

24 atm_manufacturer varchar(50),

25 atm_location_id int,

26 PRIMARY KEY(atm_id),

27 FOREIGN KEY(atm_location_id) references

28 atm_data.DIM_LOCATION(location_id));

29

30

31

32

33 create table atm_data.DIM DATE

Run

Save

Schedule

Clear

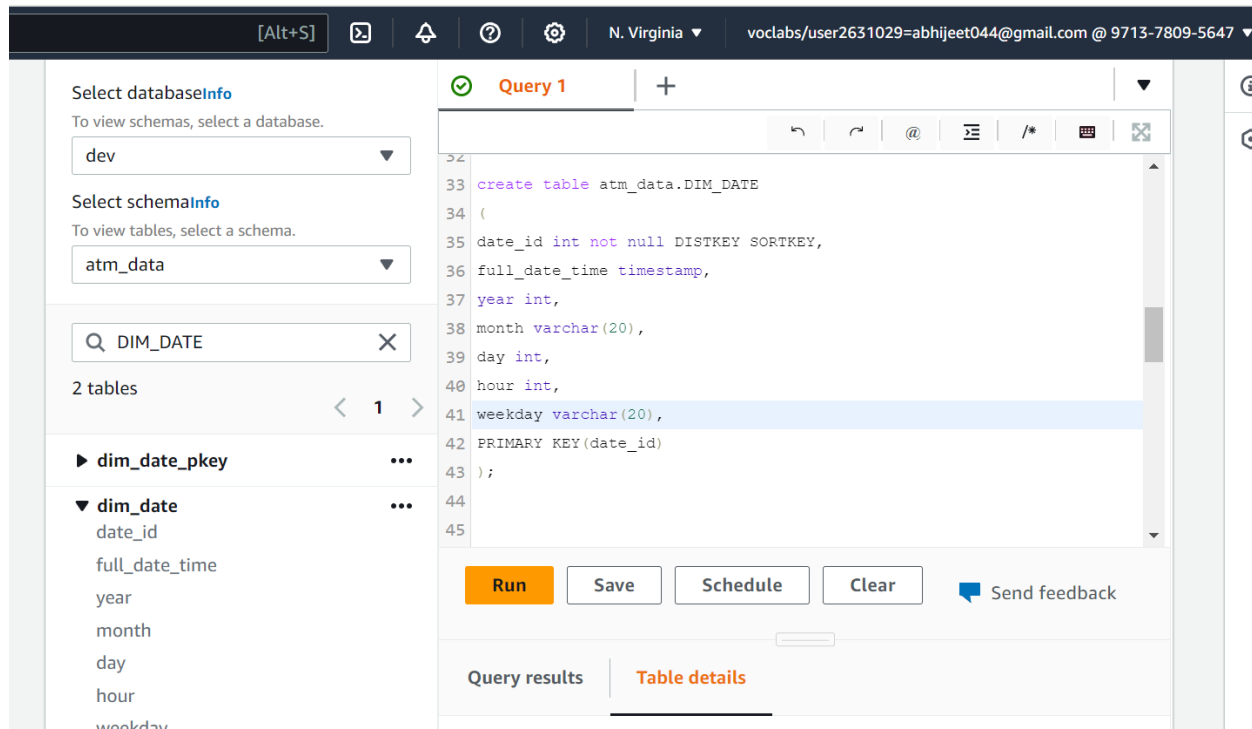
Send feedback

Query results

Table details

- Creating date dimension table**
 - 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)
);

© Copyright. upGrad Education Pvt. Ltd. All rights reserved



The screenshot shows the upGrad SQL IDE interface. On the left, there is a sidebar with a search bar and a list of tables. The search bar contains "DIM_DATE". Below it, a list of tables is shown, including "dim_date_pkey" and "dim_date". The "dim_date" table is expanded, showing its columns: "date_id", "full_date_time", "year", "month", "day", "hour", and "weekday". The main area displays a SQL query in a code editor. The query is as follows:

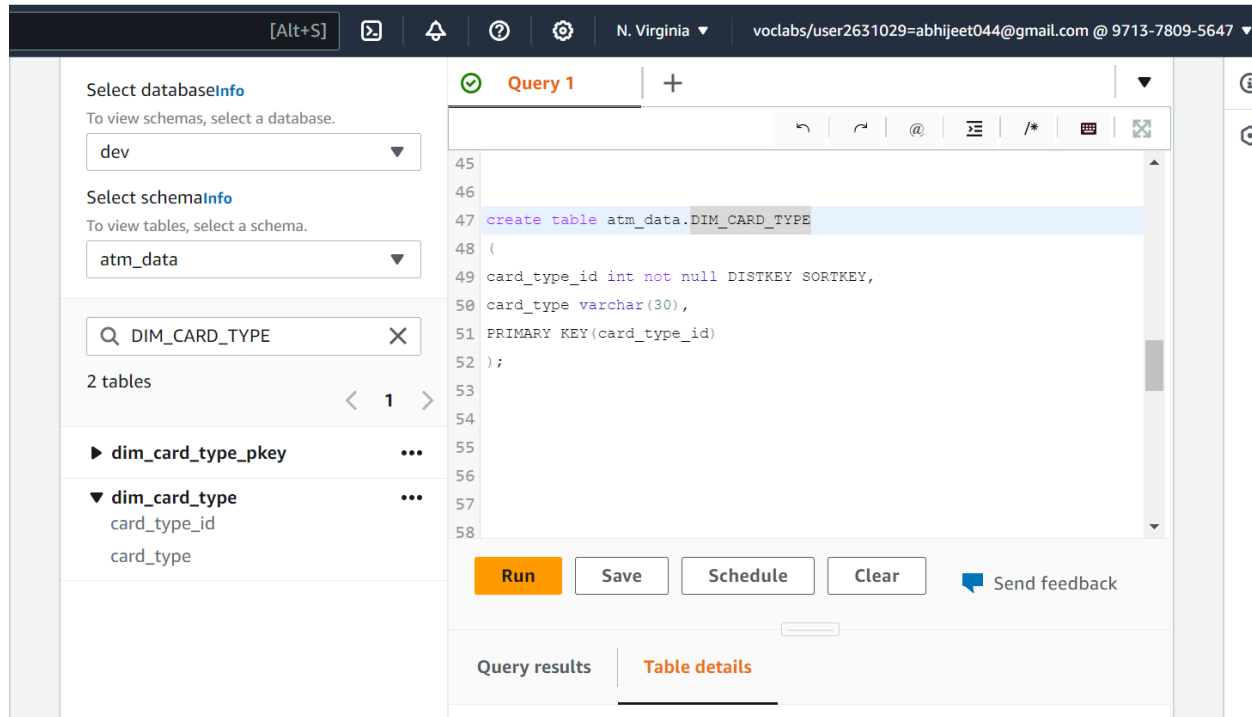
```

32
33 create table atm_data.DIM_DATE
34 (
35 date_id int not null DISTKEY SORTKEY,
36 full_date_time timestamp,
37 year int,
38 month varchar(20),
39 day int,
40 hour int,
41 weekday varchar(20),
42 PRIMARY KEY(date_id)
43 );
44
45

```

Below the query editor, there are buttons for "Run", "Save", "Schedule", and "Clear". To the right of these buttons is a "Send feedback" link. At the bottom, there are tabs for "Query results" and "Table details".

- **Creating card type dimension table**
 - create table atm_data.DIM_CARD_TYPE
 (
 card_type_id int not null DISTKEY SORTKEY,
 card_type varchar(30) ,
 PRIMARY KEY(card_type_id)
);



The screenshot shows a SQL IDE interface. On the left, there's a sidebar with a 'Select database' dropdown set to 'dev' and a 'Select schema' dropdown set to 'atm_data'. Below these, a search bar contains 'DIM_CARD_TYPE' and a list of 2 tables is shown: 'dim_card_type_pkey' and 'dim_card_type'. The 'dim_card_type' table is expanded, showing columns 'card_type_id' and 'card_type'. The main area is a query editor titled 'Query 1' containing the following SQL code:

```

45
46
47 create table atm_data.DIM_CARD_TYPE
48 (
49   card_type_id int not null DISTKEY SORTKEY,
50   card_type varchar(30),
51   PRIMARY KEY(card_type_id)
52 );
53
54
55
56
57
58

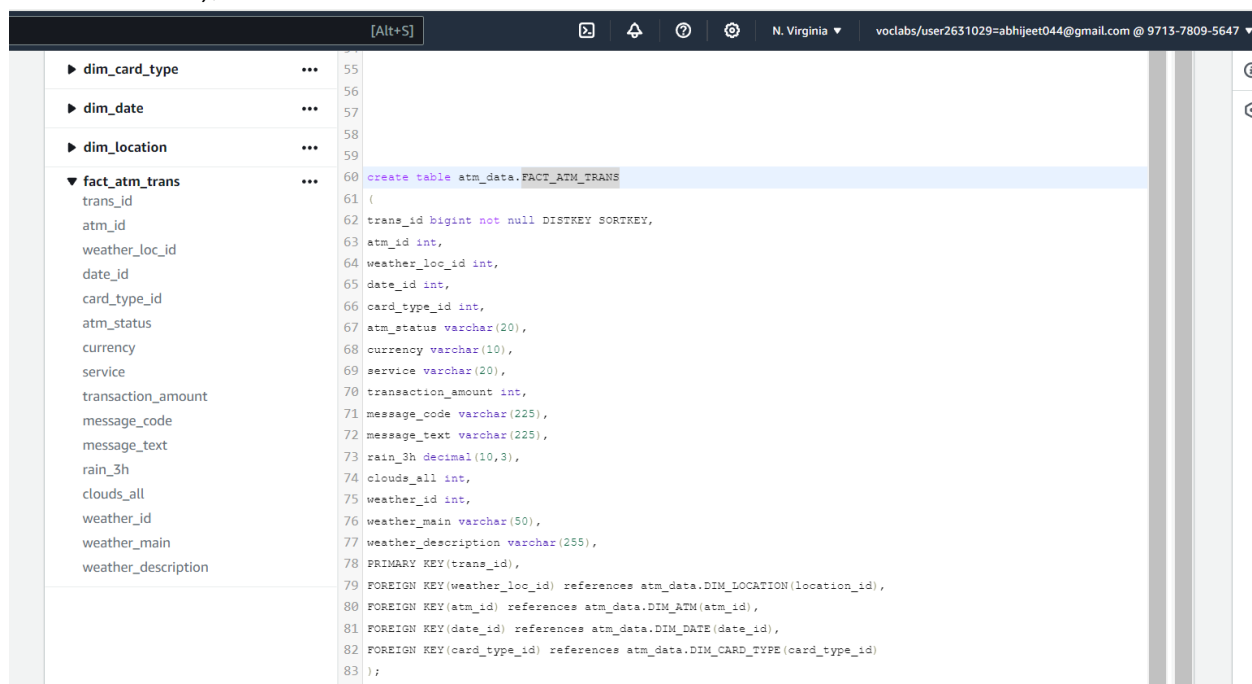
```

Below the query editor are buttons for 'Run', 'Save', 'Schedule', and 'Clear', along with a 'Send feedback' link. At the bottom, there are tabs for 'Query results' and 'Table details'.

- **Creating atm transactions fact table**

- 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_DATA(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)
);
```



The screenshot shows a SQL IDE interface. On the left, a table schema is displayed with the following tables and columns:

- dim_card_type** (3 columns)
- dim_date** (3 columns)
- dim_location** (3 columns)
- fact_atm_trans** (15 columns): trans_id, atm_id, weather_loc_id, date_id, card_type_id, atm_status, currency, service, transaction_amount, message_code, message_text, rain_3h, clouds_all, weather_id, weather_main, weather_description

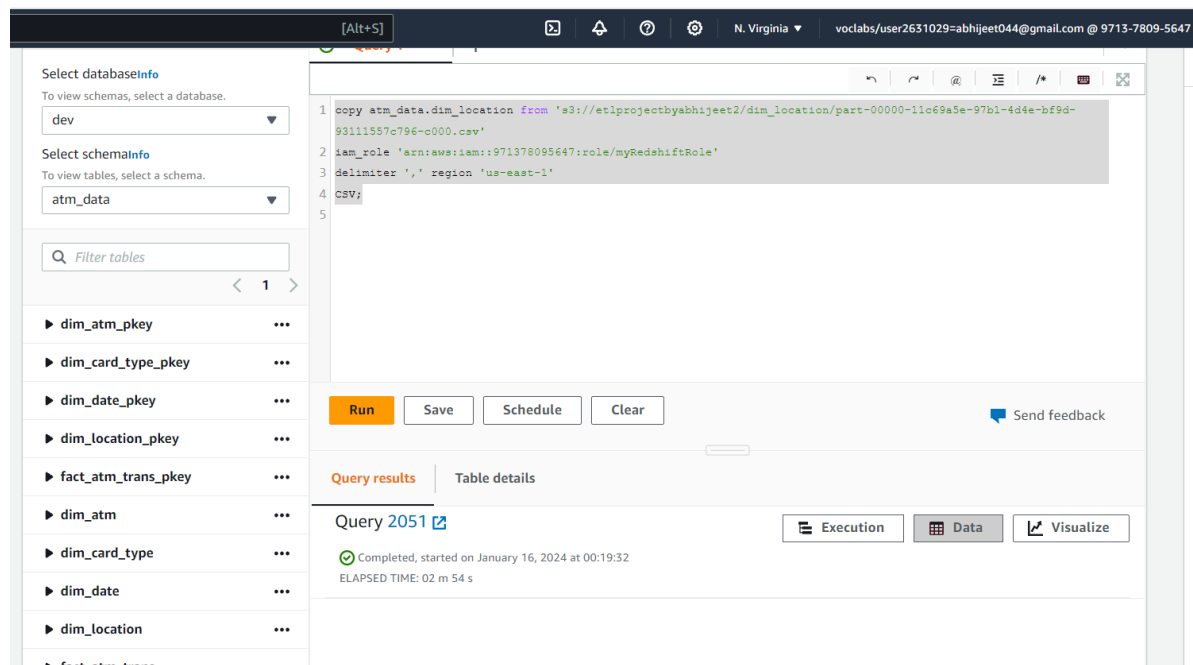
The main editor shows the following SQL code:

```
55
56
57
58
59
60 create table atm_data.FACT_ATM_TRANS
61 (
62 trans_id bigint not null DISTKEY SORTKEY,
63 atm_id int,
64 weather_loc_id int,
65 date_id int,
66 card_type_id int,
67 atm_status varchar(20),
68 currency varchar(10),
69 service varchar(20),
70 transaction_amount int,
71 message_code varchar(225),
72 message_text varchar(225),
73 rain_3h decimal(10,3),
74 clouds_all int,
75 weather_id int,
76 weather_main varchar(50),
77 weather_description varchar(255),
78 PRIMARY KEY(trans_id),
79 FOREIGN KEY(weather_loc_id) references atm_data.DIM_LOCATION(location_id),
80 FOREIGN KEY(atm_id) references atm_data.DIM_ATM(atm_id),
81 FOREIGN KEY(date_id) references atm_data.DIM_DATE(date_id),
82 FOREIGN KEY(card_type_id) references atm_data.DIM_CARD_TYPE(card_type_id)
83 );
```

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

- **Copying the data to dim_location table**
 - copy atm_data.dim_location from 's3://etlprojectbyabhijeet2/dim_location/part-00000-11c69a5e-97b1-4d4e-bf9d-93111557c796-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
CSV;



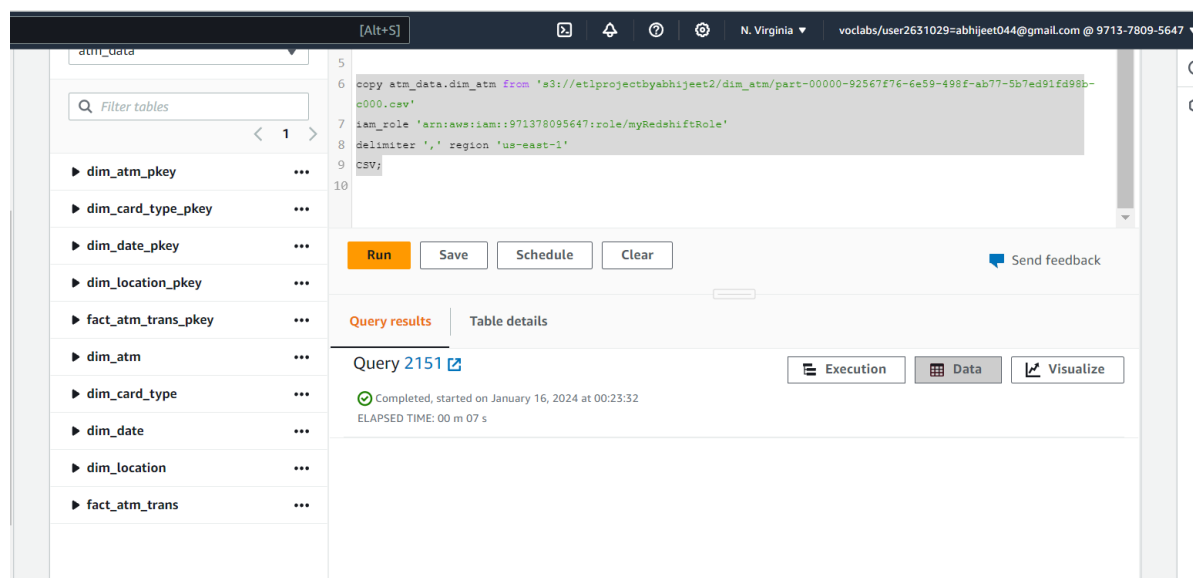
The screenshot shows the AWS Redshift console interface. On the left, the 'Select database' dropdown is set to 'dev' and the 'Select schema' dropdown is set to 'atm_data'. Below these, a list of tables is visible, including 'dim_atm_pkey', 'dim_card_type_pkey', 'dim_date_pkey', 'dim_location_pkey', 'fact_atm_trans_pkey', 'dim_atm', 'dim_card_type', 'dim_date', 'dim_location', and 'fact_atm_trans'. The main query editor displays the following SQL code:

```
1 copy atm_data.dim_location from 's3://etlprojectbyabhijeet2/dim_location/part-00000-11c69a5e-97b1-4d4e-bf9d-
2 93111557c796-c000.csv'
3 iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
4 delimiter ',' region 'us-east-1'
5 CSV;
```

Below the query editor, there are buttons for 'Run', 'Save', 'Schedule', and 'Clear'. The 'Run' button is highlighted. To the right of the buttons is a 'Send feedback' link. Below the buttons, the 'Query results' tab is selected, showing 'Query 2051'. The status indicates 'Completed, started on January 16, 2024 at 00:19:32' and 'ELAPSED TIME: 02 m 54 s'. There are also buttons for 'Execution', 'Data', and 'Visualize'.

- **Copying the data to dim_atm table**

- copy atm_data.dim_atm from 's3://etlprojectbyabhijeet2/dim_atm/part-00000-92567f76-6e59-498f-ab77-5b7ed91fd98b-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
CSV;



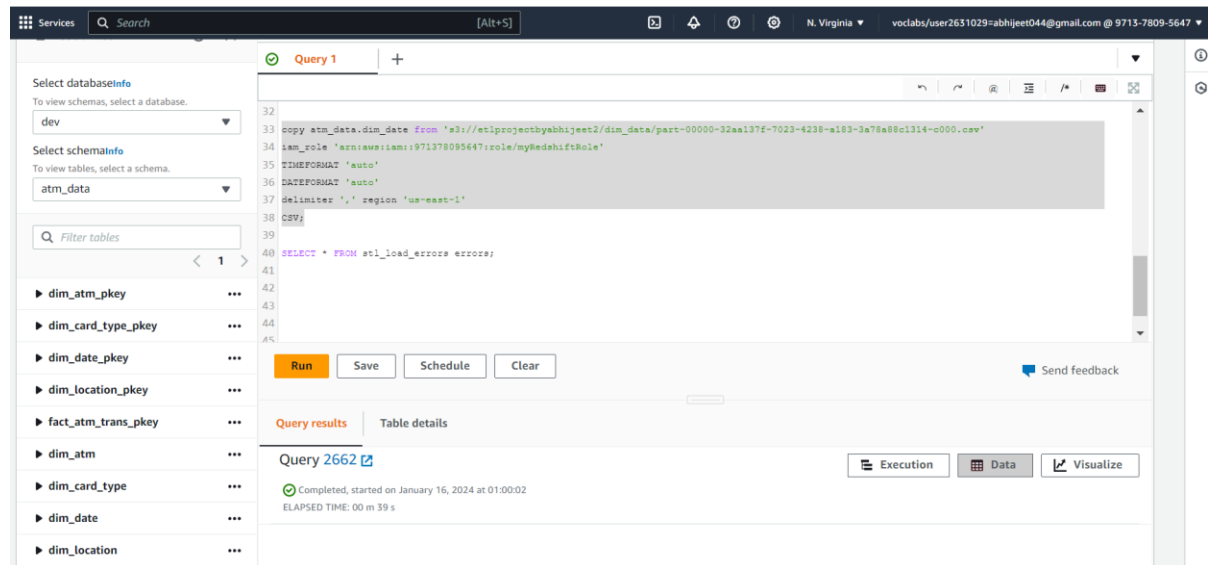
The screenshot shows the AWS Redshift console interface. On the left, the 'Select database' dropdown is set to 'dev' and the 'Select schema' dropdown is set to 'atm_data'. Below these, a list of tables is visible, including 'dim_atm_pkey', 'dim_card_type_pkey', 'dim_date_pkey', 'dim_location_pkey', 'fact_atm_trans_pkey', 'dim_atm', 'dim_card_type', 'dim_date', 'dim_location', and 'fact_atm_trans'. The main query editor displays the following SQL code:

```
5
6 copy atm_data.dim_atm from 's3://etlprojectbyabhijeet2/dim_atm/part-00000-92567f76-6e59-498f-ab77-5b7ed91fd98b-
7 c000.csv'
8 iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
9 delimiter ',' region 'us-east-1'
10 CSV;
```

Below the query editor, there are buttons for 'Run', 'Save', 'Schedule', and 'Clear'. The 'Run' button is highlighted. To the right of the buttons is a 'Send feedback' link. Below the buttons, the 'Query results' tab is selected, showing 'Query 2151'. The status indicates 'Completed, started on January 16, 2024 at 00:23:32' and 'ELAPSED TIME: 00 m 07 s'. There are also buttons for 'Execution', 'Data', and 'Visualize'.

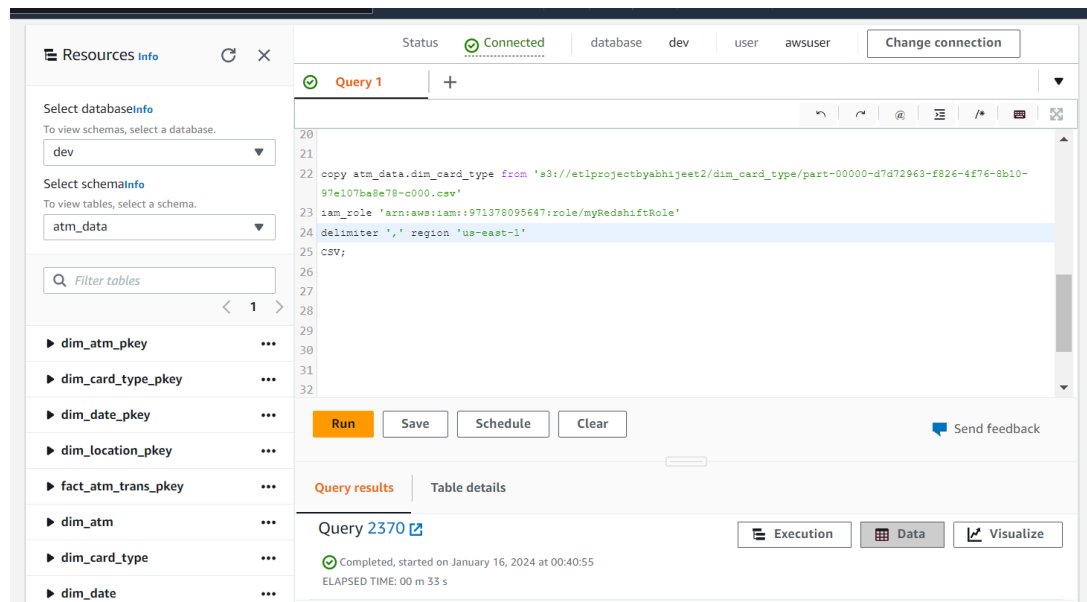
- **Copying the data to dim_date table**

- copy atm_data.dim_date from 's3://etlprojectbyabhijeet2/dim_data/part-00000-32aa137f-7023-4238-a183-3a78a88c1314-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
TIMEFORMAT 'auto'
DATEFORMAT 'auto'
delimiter ',' region 'us-east-1'
CSV;



• Copying the data to dim_card_type table

- copy atm_data.dim_card_type from
's3://etlprojectbyabhijeet2/dim_card_type/part-00000-d7d72963-f826-4f76-8b10-97e107ba8e78-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
CSV;



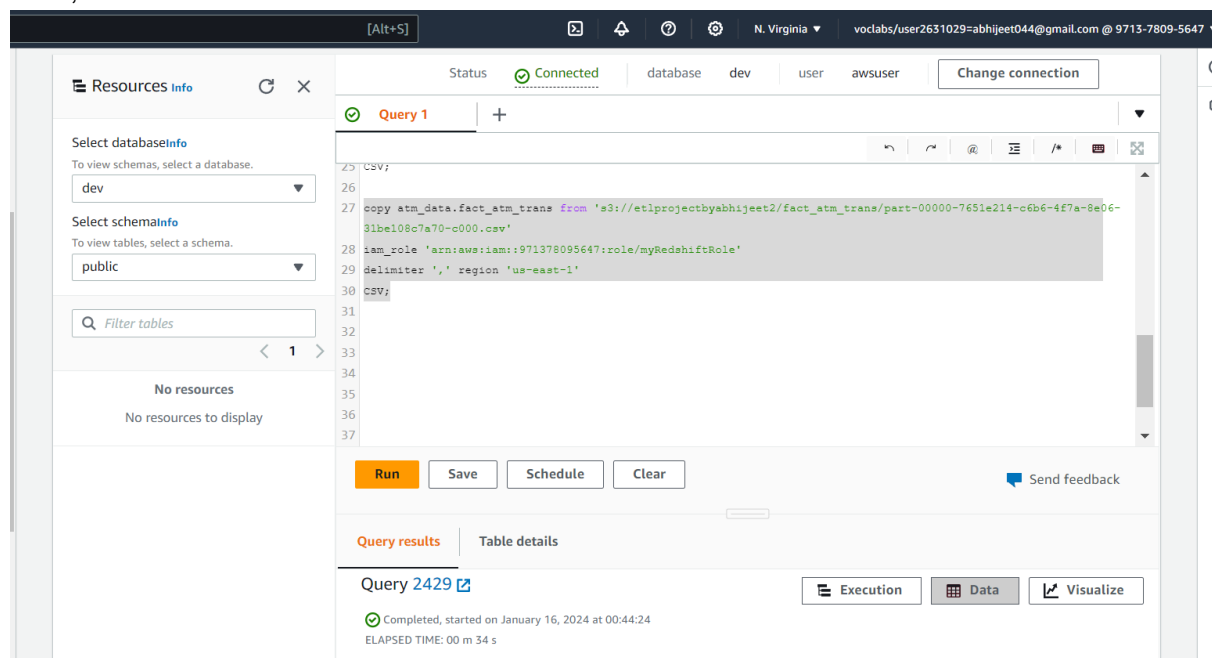
The screenshot shows the AWS Redshift console interface. On the left, the 'Resources' panel is open, showing the 'dev' database and 'atm_data' schema. The 'fact_atm_trans' table is highlighted. The main panel displays a SQL query (Query 1) that copies data from an S3 bucket to the 'fact_atm_trans' table. The query is as follows:

```
copy atm_data.fact_atm_trans from 's3://etlprojectbyabhijeet2/fact_atm_trans/part-00000-7651e214-c6b6-4f7a-8e06-31be108c7a70-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
CSV;
```

Below the query, the 'Run' button is highlighted. The 'Query results' tab is active, showing the query status as 'Completed' and the elapsed time as '00 m 33 s'.

- Copying the data to fact_atm_trans table

- copy atm_data.fact_atm_trans from 's3://etlprojectbyabhijeet2/fact_atm_trans/part-00000-7651e214-c6b6-4f7a-8e06-31be108c7a70-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
CSV;



The screenshot shows the AWS Redshift console interface. On the left, the 'Resources' panel is open, showing the 'dev' database and 'public' schema. The 'fact_atm_trans' table is highlighted. The main panel displays a SQL query (Query 1) that copies data from an S3 bucket to the 'fact_atm_trans' table. The query is as follows:

```
copy atm_data.fact_atm_trans from 's3://etlprojectbyabhijeet2/fact_atm_trans/part-00000-7651e214-c6b6-4f7a-8e06-31be108c7a70-c000.csv'
iam_role 'arn:aws:iam::971378095647:role/myRedshiftRole'
delimiter ',' region 'us-east-1'
CSV;
```

Below the query, the 'Run' button is highlighted. The 'Query results' tab is active, showing the query status as 'Completed' and the elapsed time as '00 m 34 s'.