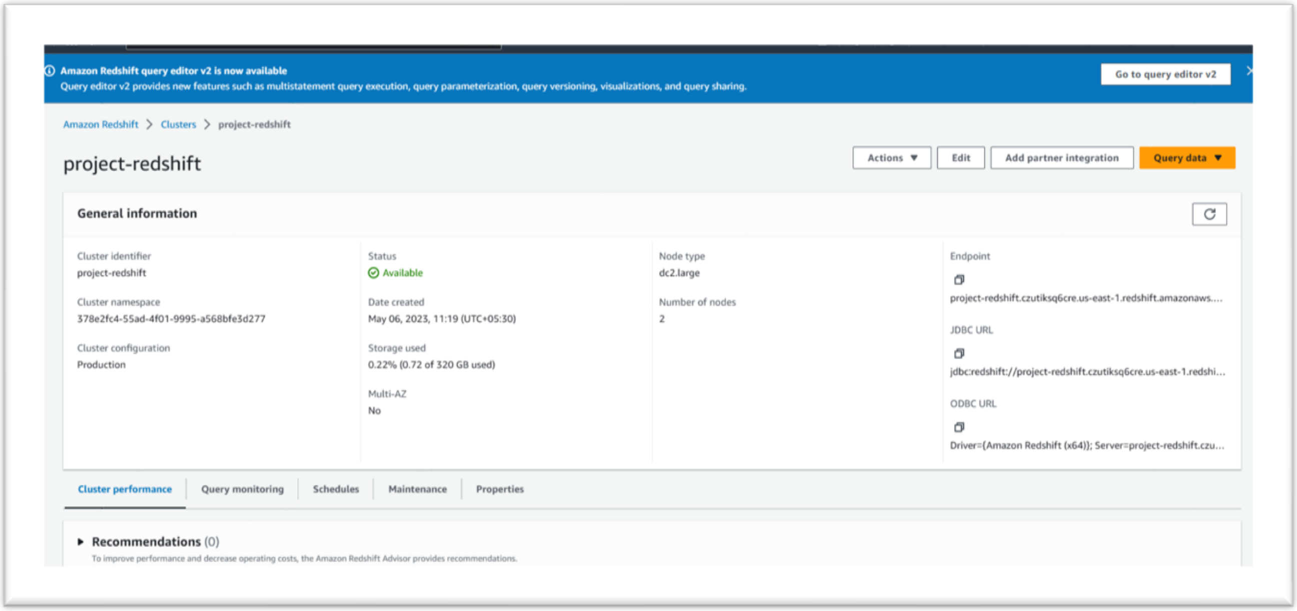
Creation of a Redshift Cluster

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



Amazon Redshift query editor v2 is now available
Query editor v2 provides new features such as multistatement query execution, query parameterization, query versioning, visualizations, and query sharing.

project-redshift

General information

Cluster identifier
project-redshift

Cluster namespace
378e2fc4-55ad-4f01-9995-a568bfe3d277

Cluster configuration
Production

Status
Available

Date created
May 06, 2023, 11:19 (UTC+05:30)

Storage used
0.22% (0.72 of 320 GB used)

Multi-AZ
No

Node type
dc2.large

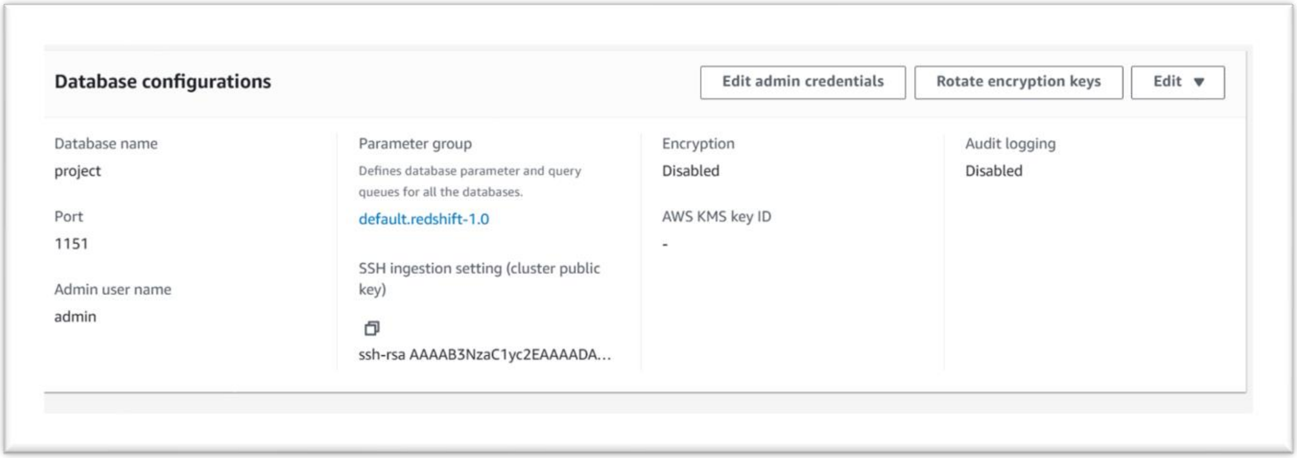
Number of nodes
2

Endpoint
project-redshift.czutiksq6cre.us-east-1.redshift.amazonaws....

JDBC URL
jdbc:redshift://project-redshift.czutiksq6cre.us-east-1.redshi...

ODBC URL
Driver={Amazon Redshift (x64)}; Server=project-redshift.czu...

Recommendations (0)
To improve performance and decrease operating costs, the Amazon Redshift Advisor provides recommendations.



Database configurations

Database name
project

Port
1151

Admin user name
admin

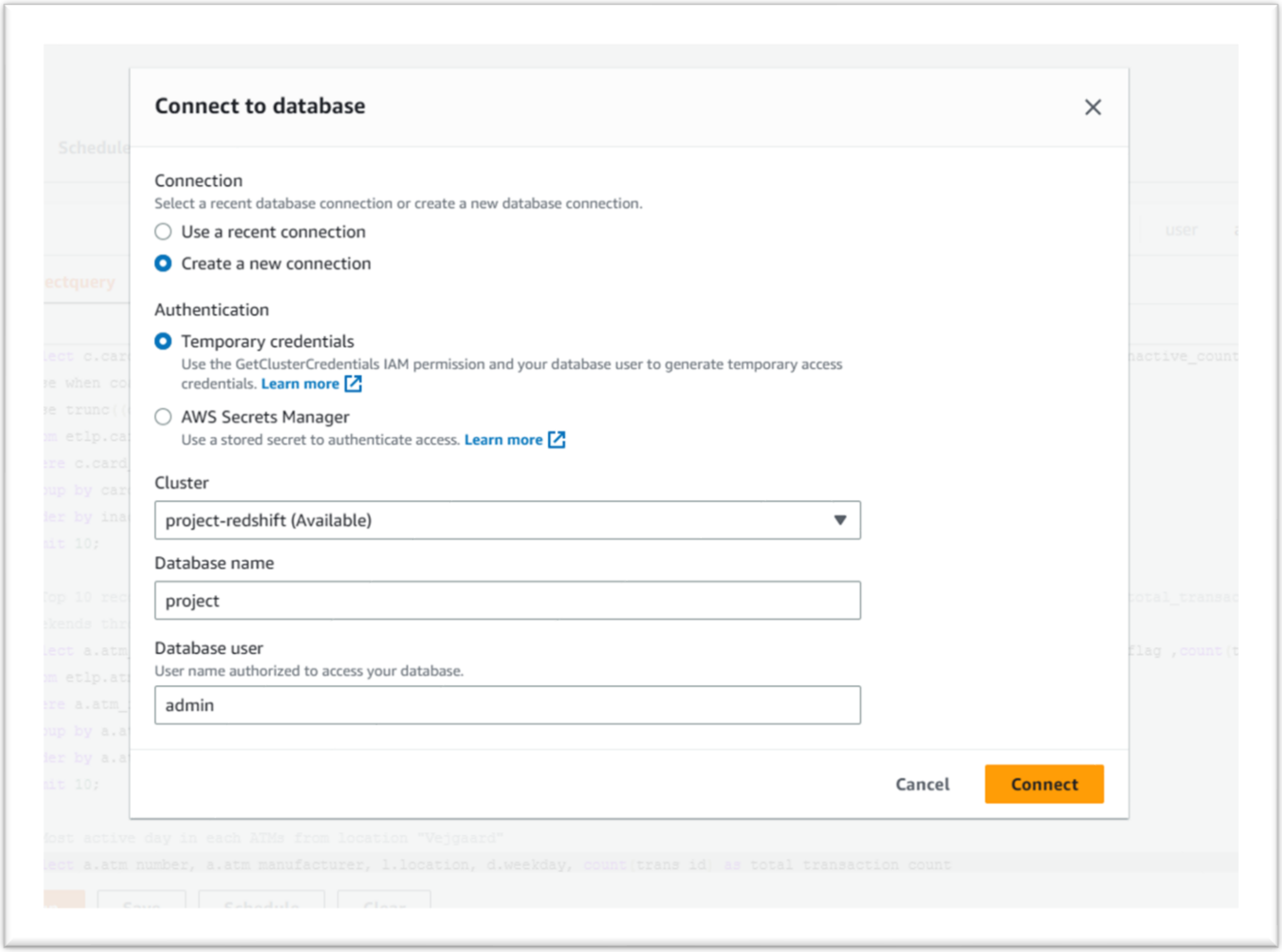
Parameter group
Defines database parameter and query queues for all the databases.
default.redshift-1.0

SSH ingestion setting (cluster public key)
ssh-rsa AAAAB3NzaC1yc2EAAAADA...

Encryption
Disabled

AWS KMS key ID
-

Audit logging
Disabled



Connect to database

Connection
Select a recent database connection or create a new database connection.

○ Use a recent connection
● Create a new connection

Authentication

● Temporary credentials
Use the GetClusterCredentials IAM permission and your database user to generate temporary access credentials. Learn more

○ AWS Secrets Manager
Use a stored secret to authenticate access. Learn more

Cluster: project-redshift (Available)

Database name: project

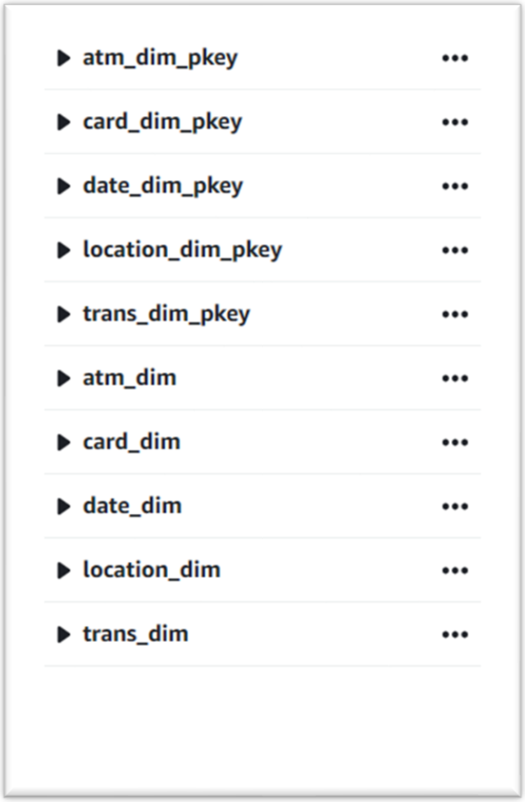
Database user
User name authorized to access your database.
admin

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 the schema**

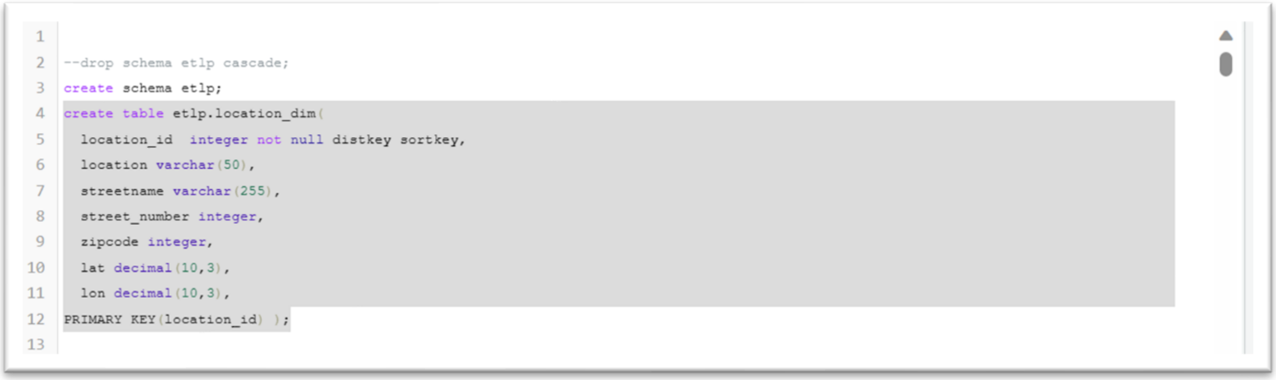
```
create schema etlp;
```



- ▶ atm_dim_pkey •••
- ▶ card_dim_pkey •••
- ▶ date_dim_pkey •••
- ▶ location_dim_pkey •••
- ▶ trans_dim_pkey •••
- ▶ atm_dim •••
- ▶ card_dim •••
- ▶ date_dim •••
- ▶ location_dim •••
- ▶ trans_dim •••

2) Creating the location dimension table

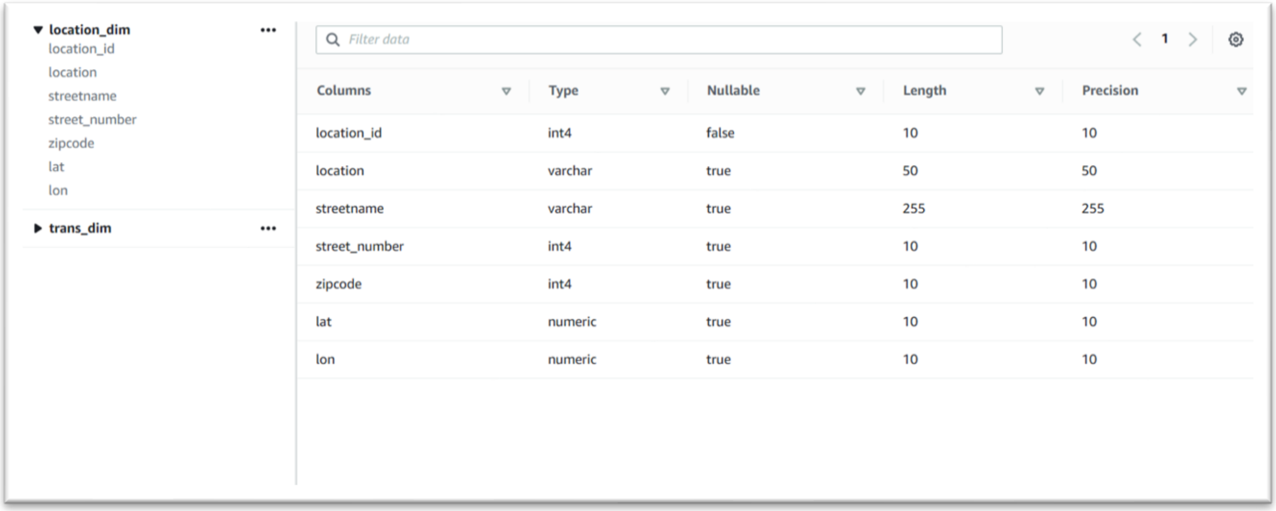
```
 create table etlp.location_dim(
  location_id  integer not null distkey sortkey,
  location varchar(50),
  streetname varchar(255),
  street_number integer,
  zipcode integer,
  lat decimal(10,3),
  lon decimal(10,3),
PRIMARY KEY(location_id) );
```



```

--drop schema etlp cascade;
create schema etlp;
create table etlp.location_dim(
  location_id  integer not null distkey sortkey,
  location varchar(50),
  streetname varchar(255),
  street_number integer,
  zipcode integer,
  lat decimal(10,3),
  lon decimal(10,3),
PRIMARY KEY(location_id) );

```

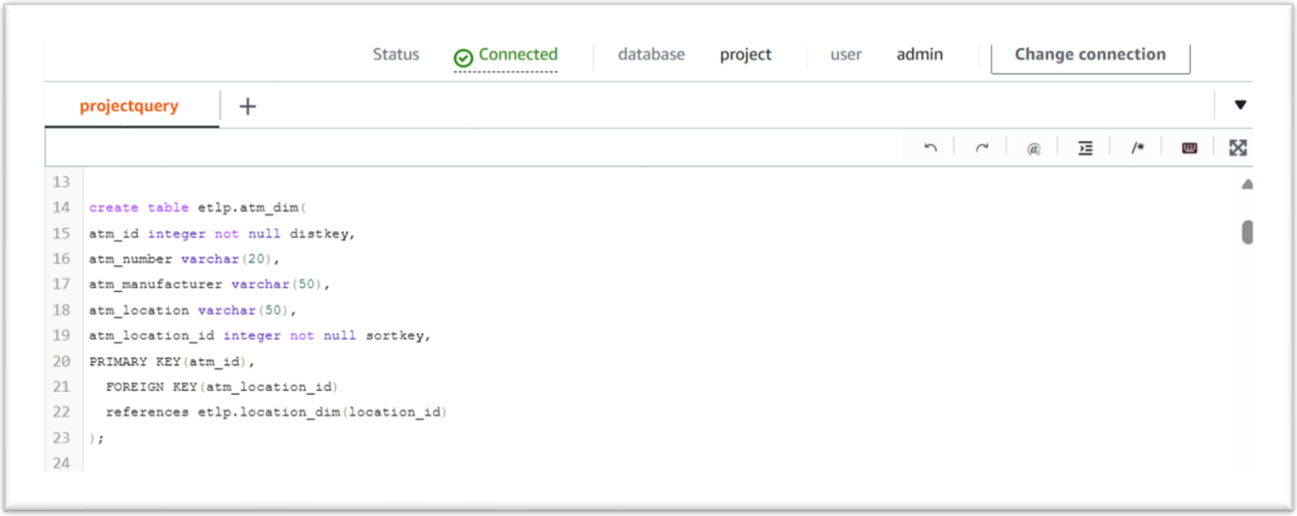


- location_dim
  - location_id
  - location
  - streetname
  - street_number
  - zipcode
  - lat
  - lon
- trans_dim

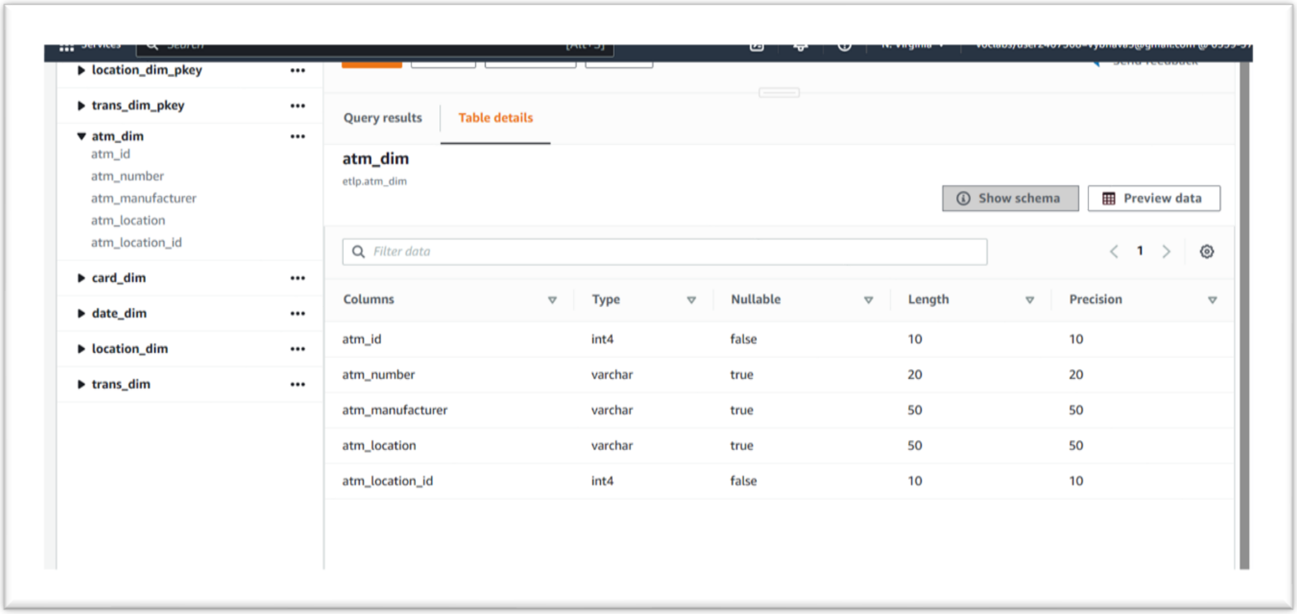
Columns	Type	Nullable	Length	Precision
location_id	int4	false	10	10
location	varchar	true	50	50
streetname	varchar	true	255	255
street_number	int4	true	10	10
zipcode	int4	true	10	10
lat	numeric	true	10	10
lon	numeric	true	10	10

3) Creating atm dimension table

```
create table etlp.atm_dim(
atm_id integer not null distkey,
atm_number varchar(20),
atm_manufacturer varchar(50),
atm_location varchar(50),
atm_location_id integer not null sortkey,
PRIMARY KEY(atm_id),
  FOREIGN KEY(atm_location_id)
  references etlp.location_dim(location_id)
);
```



```
create table etlp.atm_dim(
atm_id integer not null distkey,
atm_number varchar(20),
atm_manufacturer varchar(50),
atm_location varchar(50),
atm_location_id integer not null sortkey,
PRIMARY KEY(atm_id),
  FOREIGN KEY(atm_location_id)
  references etlp.location_dim(location_id)
);
```



- location_dim_pkey
- trans_dim_pkey
- atm_dim
  - atm_id
  - atm_number
  - atm_manufacturer
  - atm_location
  - atm_location_id
- card_dim
- date_dim
- location_dim
- trans_dim

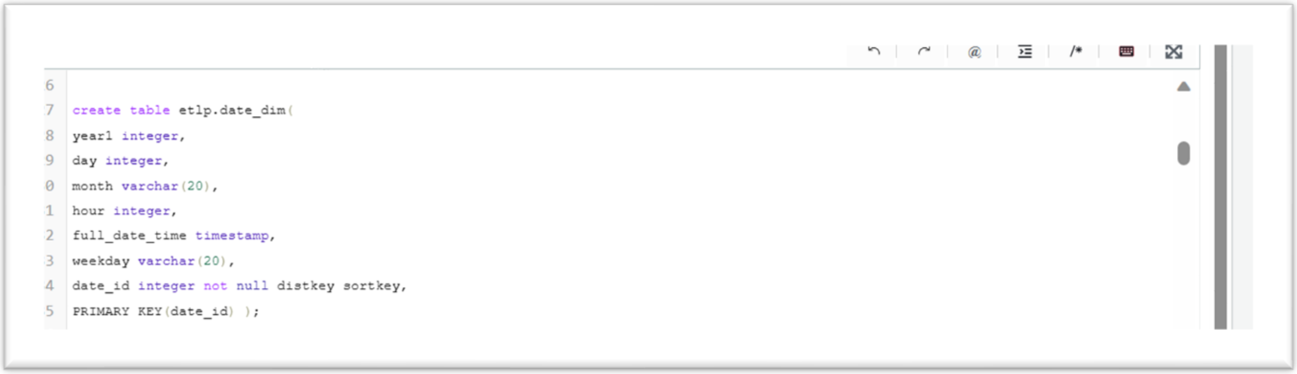
atm_dim

etlp.atm_dim

Columns	Type	Nullable	Length	Precision
atm_id	int4	false	10	10
atm_number	varchar	true	20	20
atm_manufacturer	varchar	true	50	50
atm_location	varchar	true	50	50
atm_location_id	int4	false	10	10

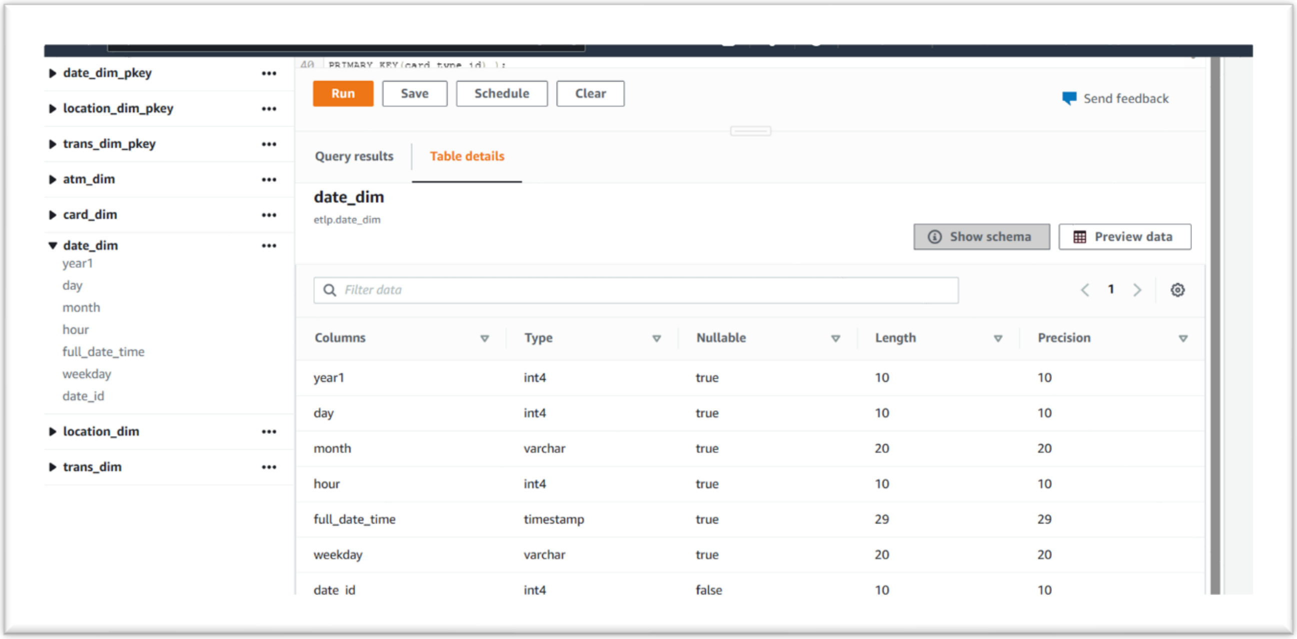
4) Creating Date dimension table

```
create table etlp.date_dim(
year1 integer,
day integer,
month varchar(20),
hour integer,
full_date_time timestamp,
weekday varchar(20),
date_id integer not null distkey sortkey,
PRIMARY KEY(date_id) );
```



```

create table etlp.date_dim(
year1 integer,
day integer,
month varchar(20),
hour integer,
full_date_time timestamp,
weekday varchar(20),
date_id integer not null distkey sortkey,
PRIMARY KEY(date_id) );
```



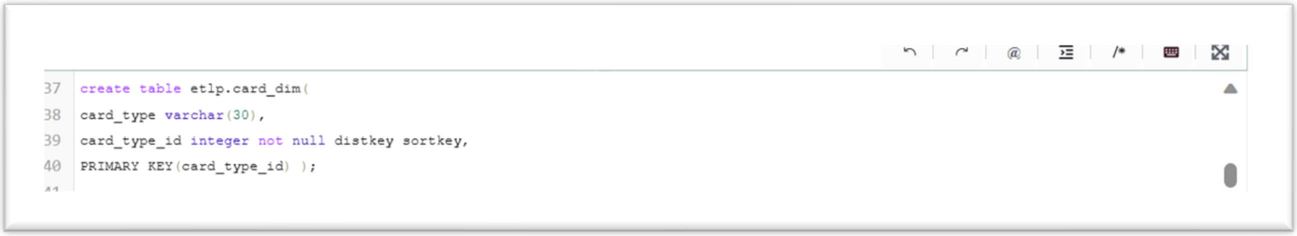
```
PRIMARY KEY(card_type_id) );
```

date_dim
etlp.date_dim

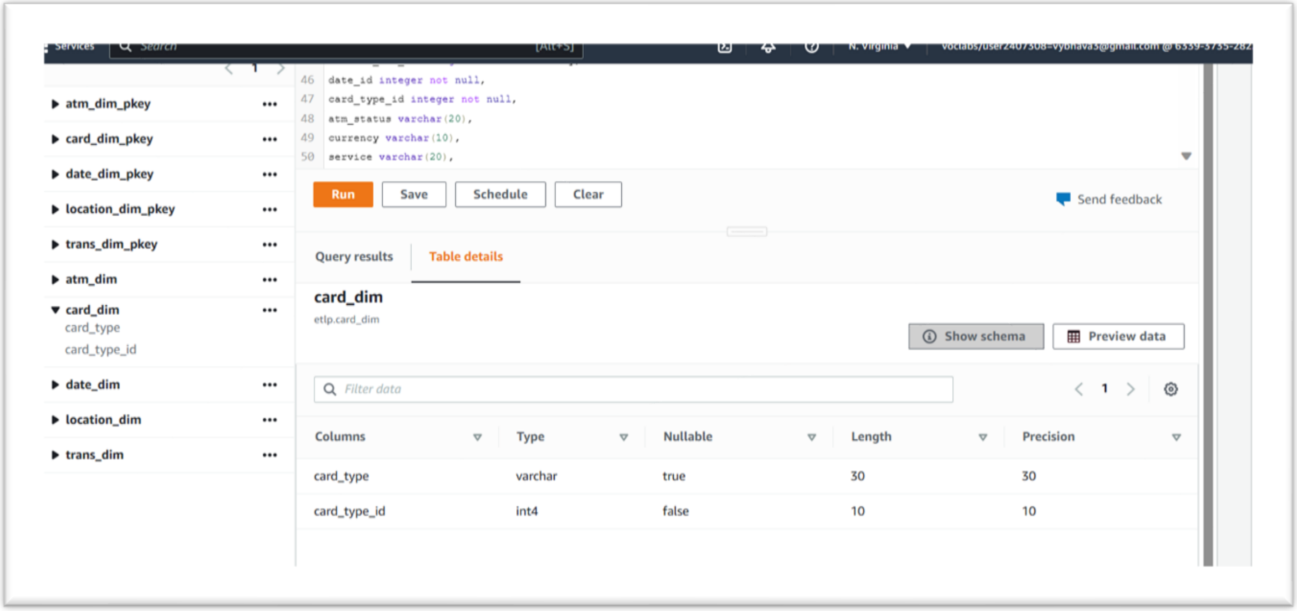
Columns	Type	Nullable	Length	Precision
year1	int4	true	10	10
day	int4	true	10	10
month	varchar	true	20	20
hour	int4	true	10	10
full_date_time	timestamp	true	29	29
weekday	varchar	true	20	20
date_id	int4	false	10	10

5) Creating card dimension table

create table etlp.card_dim(
card_type varchar(30),
card_type_id integer not null distkey sortkey,
PRIMARY KEY(card_type_id));



```
create table etlp.card_dim(
card_type varchar(30),
card_type_id integer not null distkey sortkey,
PRIMARY KEY(card_type_id) );
```



```
date_id integer not null,
card_type_id integer not null,
atm_status varchar(20),
currency varchar(10),
service varchar(20),
```

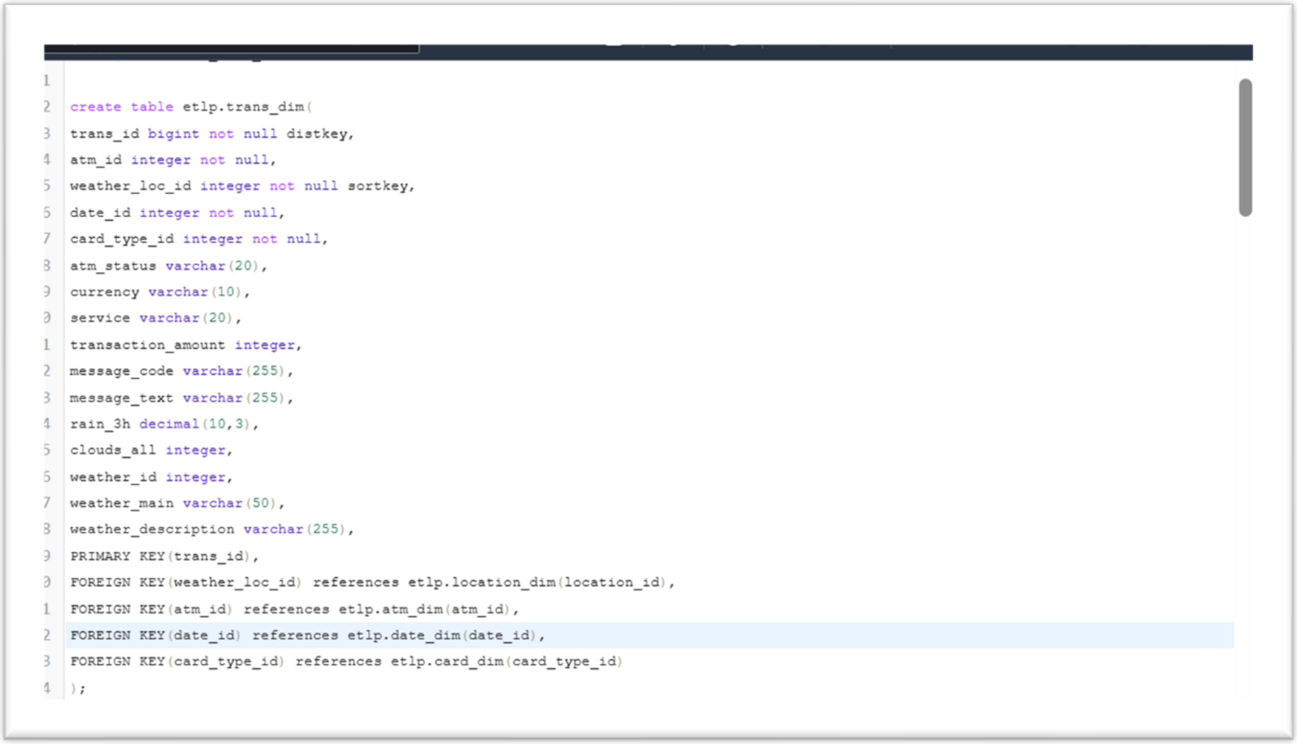
card_dim

etlp.card_dim

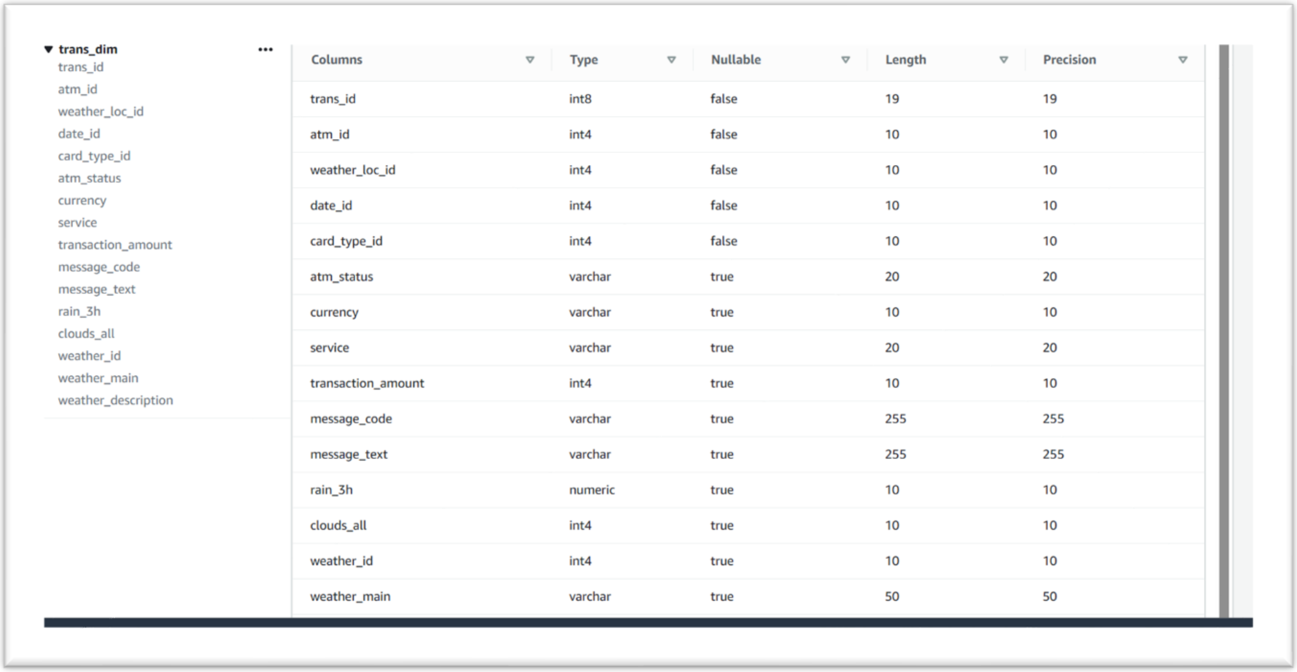
Columns	Type	Nullable	Length	Precision
card_type	varchar	true	30	30
card_type_id	int4	false	10	10

6) Fact table creation

```
create table etlp.trans_dim(
trans_id bigint not null distkey,
atm_id integer not null,
weather_loc_id integer not null sortkey,
date_id integer not null,
card_type_id integer not null,
atm_status varchar(20),
currency varchar(10),
service varchar(20),
transaction_amount integer,
message_code varchar(255),
message_text varchar(255),
rain_3h decimal(10,3),
clouds_all integer,
weather_id integer,
weather_main varchar(50),
weather_description varchar(255),
PRIMARY KEY(trans_id),
FOREIGN KEY(weather_loc_id) references etlp.location_dim(location_id),
FOREIGN KEY(atm_id) references etlp.atm_dim(atm_id),
FOREIGN KEY(date_id) references etlp.date_dim(date_id),
FOREIGN KEY(card_type_id) references etlp.card_dim(card_type_id)
);
```



```
create table etlp.trans_dim(
trans_id bigint not null distkey,
atm_id integer not null,
weather_loc_id integer not null sortkey,
date_id integer not null,
card_type_id integer not null,
atm_status varchar(20),
currency varchar(10),
service varchar(20),
transaction_amount integer,
message_code varchar(255),
message_text varchar(255),
rain_3h decimal(10,3),
clouds_all integer,
weather_id integer,
weather_main varchar(50),
weather_description varchar(255),
PRIMARY KEY(trans_id),
FOREIGN KEY(weather_loc_id) references etlp.location_dim(location_id),
FOREIGN KEY(atm_id) references etlp.atm_dim(atm_id),
FOREIGN KEY(date_id) references etlp.date_dim(date_id),
FOREIGN KEY(card_type_id) references etlp.card_dim(card_type_id)
);
```



▼ trans_dim

- 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

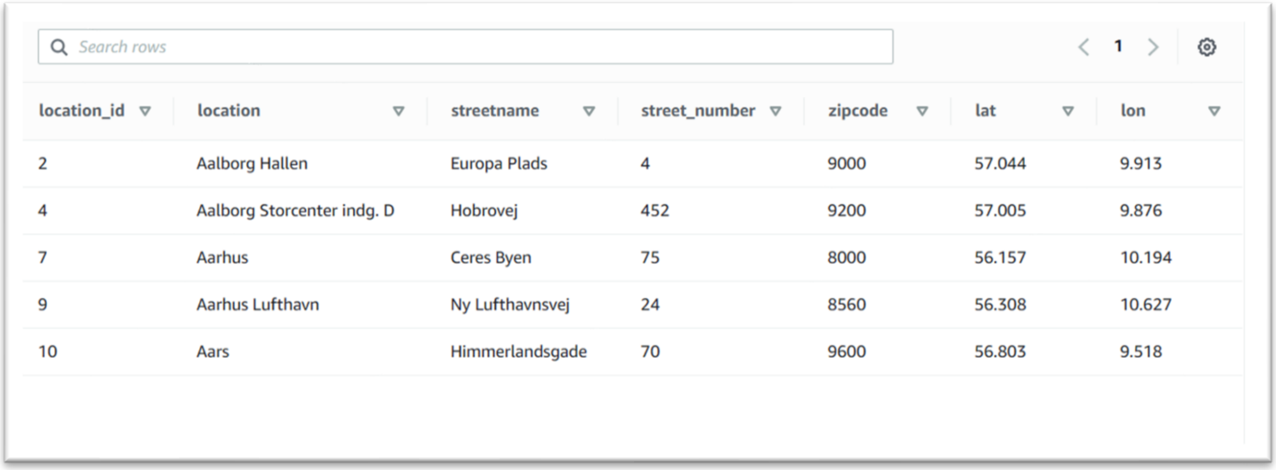
Columns	Type	Nullable	Length	Precision
trans_id	int8	false	19	19
atm_id	int4	false	10	10
weather_loc_id	int4	false	10	10
date_id	int4	false	10	10
card_type_id	int4	false	10	10
atm_status	varchar	true	20	20
currency	varchar	true	10	10
service	varchar	true	20	20
transaction_amount	int4	true	10	10
message_code	varchar	true	255	255
message_text	varchar	true	255	255
rain_3h	numeric	true	10	10
clouds_all	int4	true	10	10
weather_id	int4	true	10	10
weather_main	varchar	true	50	50

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

1) Loading data to location table

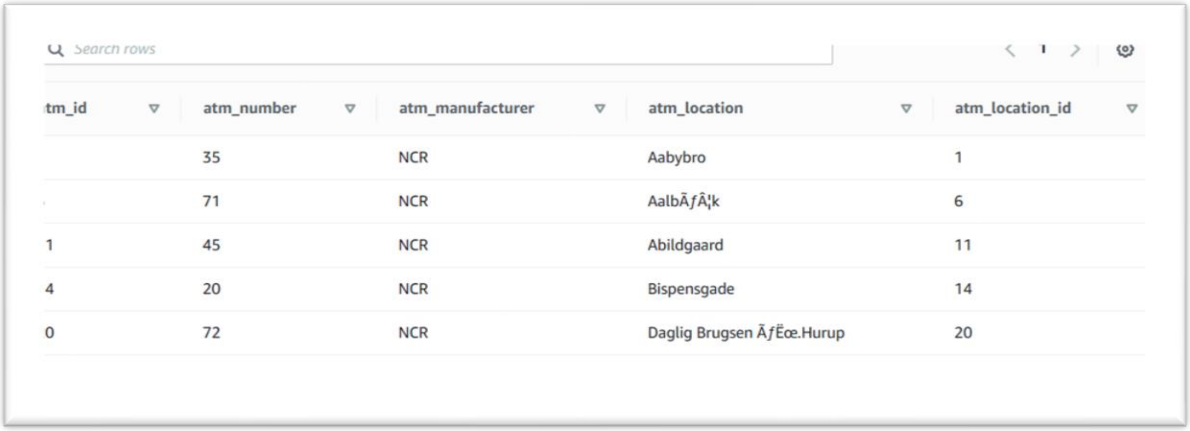
```
copy etlp.location_dim from
's3://elt-projectbucket/dim_location/part-00000-c3bd2d17-176b-4fd1-8593-8fc2bc18514e-c000.csv'
iam_role 'arn:aws:iam::633937352822:role/myRedshiftRole'
delimiter ',' region 'us-east-1' CSV;
```



location_id	location	streetname	street_number	zipcode	lat	lon
2	Aalborg Hallen	Europa Plads	4	9000	57.044	9.913
4	Aalborg Storcenter indg. D	Hobrovej	452	9200	57.005	9.876
7	Aarhus	Ceres Byen	75	8000	56.157	10.194
9	Aarhus Lufthavn	Ny Lufthavnsvej	24	8560	56.308	10.627
10	Aars	Himmerlandsgade	70	9600	56.803	9.518

2) Loading data to atm table

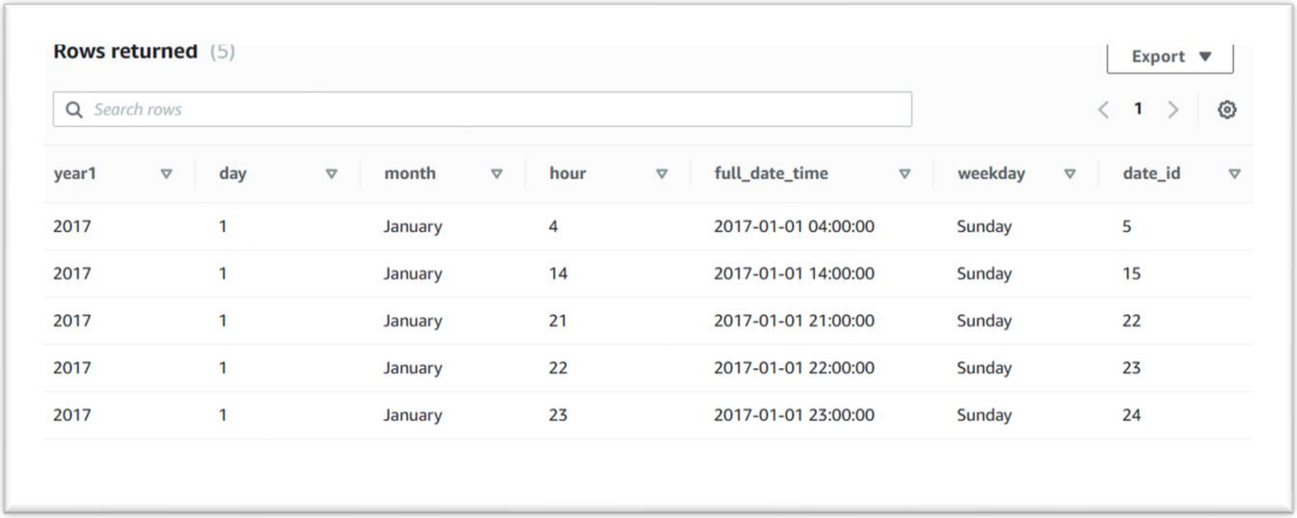
```
copy etlp.atm_dim from
's3://elt-projectbucket/dim_atm/part-00000-36e75bbf-0850-4ca2-94e8-6083d7a79bb5-c000.csv'
iam_role 'arn:aws:iam::633937352822:role/myRedshiftRole'
delimiter ',' region 'us-east-1' CSV;
```



tm_id	atm_number	atm_manufacturer	atm_location	atm_location_id
	35	NCR	Aabybro	1
	71	NCR	AalbÃƒÂ¦k	6
1	45	NCR	Abildgaard	11
4	20	NCR	Bispensgade	14
0	72	NCR	Daglig Brugsen ÃƒËœ.Hurup	20

3) Loading data to date table

```
copy etlp.date_dim from
's3://elt-projectbucket/dim_date/part-00000-f8c997a6-e205-4d99-8d77-9b8f0c6e851a-c000.csv'
iam_role 'arn:aws:iam::633937352822:role/myRedshiftRole'
delimiter ',' region 'us-east-1' CSV  timeformat 'auto';
```

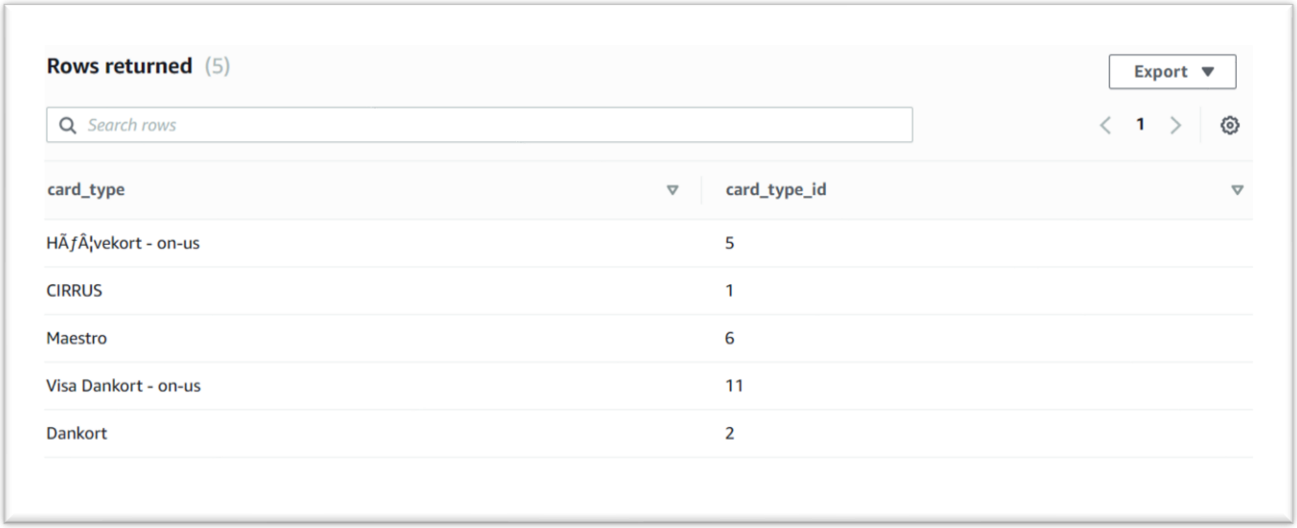


Rows returned (5)

year1	day	month	hour	full_date_time	weekday	date_id
2017	1	January	4	2017-01-01 04:00:00	Sunday	5
2017	1	January	14	2017-01-01 14:00:00	Sunday	15
2017	1	January	21	2017-01-01 21:00:00	Sunday	22
2017	1	January	22	2017-01-01 22:00:00	Sunday	23
2017	1	January	23	2017-01-01 23:00:00	Sunday	24

4) Loading data to card table

```
copy etlp.card_dim from
's3://elt-projectbucket/dim_card_type/part-00000-c169559d-37ee-4cd1-b650-714da6d6c3cb-c000.csv'
iam_role 'arn:aws:iam::633937352822:role/myRedshiftRole'
delimiter ',' region 'us-east-1' CSV;
```

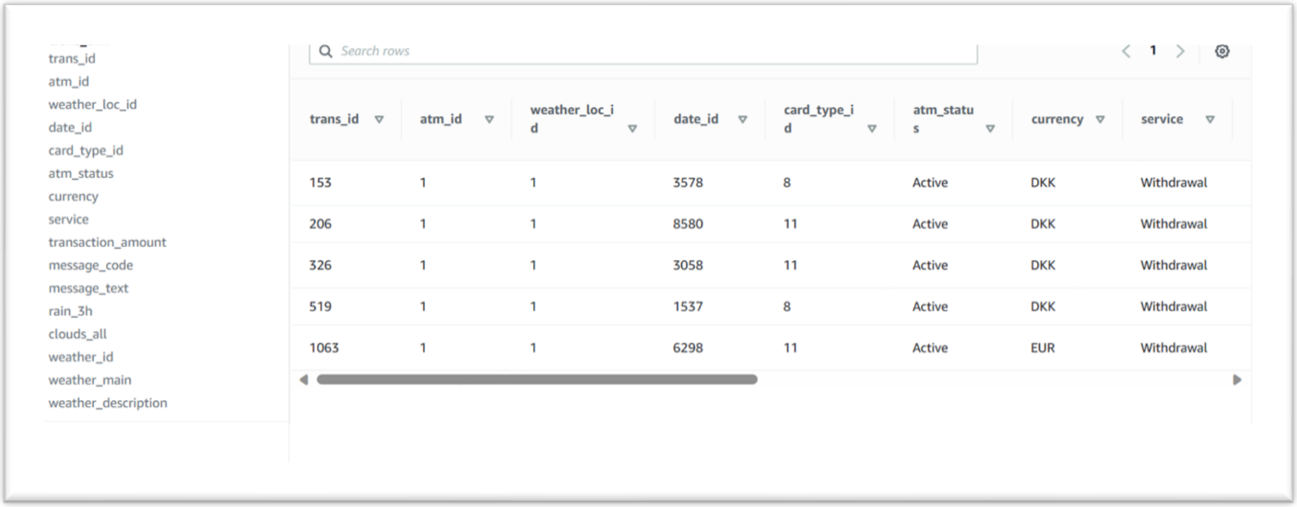


Rows returned (5)

card_type	card_type_id
HÃƒÂ¦vekort - on-us	5
CIRRUS	1
Maestro	6
Visa Dankort - on-us	11
Dankort	2

5) Loading data to fact table

```
copy etlp.trans_dim from
's3://elt-projectbucket/dim_fact_trans/part-00000-50562b6c-fad2-4dc4-995a-a06ded0a95f1-c000.csv'
iam_role 'arn:aws:iam::633937352822:role/myRedshiftRole'
delimiter ',' region 'us-east-1' CSV;
```



trans_id	atm_id	weather_loc_id	date_id	card_type_id	atm_status	currency	service
153	1	1	3578	8	Active	DKK	Withdrawal
206	1	1	8580	11	Active	DKK	Withdrawal
326	1	1	3058	11	Active	DKK	Withdrawal
519	1	1	1537	8	Active	DKK	Withdrawal
1063	1	1	6298	11	Active	EUR	Withdrawal