create database project;

use database project;

CREATE OR REPLACE TABLE ACCOUNT(

account\_id INT PRIMARY KEY,

district\_id INT,

frequency VARCHAR(40),

Date DATE ,

Account\_Type VARCHAR(100) ,

Card\_Assigned VARCHAR(20),

FOREIGN KEY (district\_id) references DISTRICT(District\_Code)

);

-----------------------------------------------------------

create or replace table CARD(

card\_id INT PRIMARY KEY,

disp\_id INT,

`type` CHAR(10) ,

issued DATE,

FOREIGN KEY (disp\_id) references DISPOSITION(disp\_id)

);

------------------------------------------

create or replace table Client(

client\_id int primary key,

Sex char(10),

Birth\_date Date,

district\_id int,

FOREIGN KEY (district\_id) references DISTRICT(District\_Code)

);

-------------------------------------------------

CREATE OR REPLACE TABLE DISPOSITION(

disp\_id INT PRIMARY KEY,

client\_id INT,

account\_id INT,

type CHAR(15),

FOREIGN KEY (account\_id) references ACCOUNT(account\_id),

FOREIGN KEY (client\_id) references CLIENT(client\_id)

);

------------------------------------------------------

CREATE or replace TABLE DISTRICT(

District\_Code INT PRIMARY KEY ,

District\_Name VARCHAR(100) ,

Region VARCHAR(100) ,

No\_of\_inhabitants INT,

No\_of\_municipalities\_with\_inhabitants\_less\_499 INT,

No\_of\_municipalities\_with\_inhabitants\_500\_btw\_1999 INT,

No\_of\_municipalities\_with\_inhabitants\_2000\_btw\_9999 INT,

No\_of\_municipalities\_with\_inhabitants\_less\_10000 INT,

No\_of\_cities INT,

Ratio\_of\_urban\_inhabitants FLOAT,

Average\_salary INT,

No\_of\_entrepreneurs\_per\_1000\_inhabitants INT,

No\_committed\_crime\_2017 INT,

No\_committed\_crime\_2018 INT

) ;

-----------------------------------------------

CREATE TABLE LOAN(

loan\_id int,

account\_id int,

`date` date,

amount int,

duration int,

payments int,

`status` varchar(35)

);

alter table loan add constraint fore\_KEY

FOREIGN KEY (ACCOUNT\_ID) REFERENCES ACCOUNT(ACCOUNT\_ID);

-----------------------------------------------------

create or replace table transaction(

trans\_id int ,

account\_id int,

`date` date,

`type` varchar(30),

operation varchar(40),

amount int,

balance float,

purpose varchar(40),

bank varchar(40),

account\_pattern\_id int,

FOREIGN KEY (account\_id) references ACCOUNT(account\_id)

);

--------------------------------------------------------------

CREATE TABLE `ORDER`(

order\_id INT PRIMARY KEY,

account\_id INT,

bank\_to VARCHAR(45),

account\_to INT,

amount FLOAT,

FOREIGN KEY (account\_id) references ACCOUNT(account\_id)

);

------------------------------------------------------------------

create or replace STORAGE integration s3\_int

type=external\_stage

storage\_provider=s3

enabled=true

storage\_aws\_role\_arn='arn:aws:iam::798227111673:role/Bankrolepolicy'

storage\_allowed\_locations=('s3://czechobank/');

desc integration s3\_int;

create or replace stage Bank

URL='s3://czechobank'

file\_format=CSV

storage\_integration=s3\_int;

list@Bank;

show stages;

--CREATE SNOWPIPE THAT RECOGNISES CSV THAT ARE INGESTED FROM EXTERNAL STAGE AND COPIES THE DATA INTO EXISTING TABLE

--The AUTO\_INGEST=true parameter specifies to read

--- event notifications sent from an S3 bucket to an SQS queue when new data is ready to load.

create or replace pipe BANK\_SNOWPIPE\_DISTRICT AUTO\_INGEST=TRUE AS

COPY INTO DISTRICT

FROM '@Bank/district/' --s3 bucket subfolde4r name

FILE\_FORMAT = CSV;

create or replace pipe BANK\_SNOWPIPE\_ACCOUNT AUTO\_INGEST=TRUE AS

COPY INTO ACCOUNT

FROM '@Bank/Account/' --s3 bucket subfolde4r name

FILE\_FORMAT = CSV;

CREATE OR REPLACE PIPE BANK\_SNOWPIPE\_TXNS AUTO\_INGEST = TRUE AS

COPY INTO PROJECT.PUBLIC.TRANSACTION

FROM '@BANK/tranx/'

FILE\_FORMAT = CSV;

CREATE OR REPLACE PIPE BANK\_SNOWPIPE\_DISP AUTO\_INGEST = TRUE AS

COPY INTO DISPOSITION

FROM '@BANK/Disposition/'

FILE\_FORMAT = CSV;

CREATE OR REPLACE PIPE BANK\_SNOWPIPE\_CARD AUTO\_INGEST = TRUE AS

COPY INTO CARD

FROM '@BANK/CARD/'

FILE\_FORMAT = CSV;

CREATE OR REPLACE PIPE BANK\_SNOWPIPE\_ORDER\_LIST AUTO\_INGEST = TRUE AS

COPY INTO PROJECT.PUBLIC."`ORDER`"

FROM '@BANK/order/'

FILE\_FORMAT = CSV;

CREATE OR REPLACE PIPE BANK\_SNOWPIPE\_LOAN AUTO\_INGEST = TRUE AS

COPY INTO PROJECT.PUBLIC.LOAN

FROM '@BANK/loan/'

FILE\_FORMAT = CSV;

CREATE OR REPLACE PIPE BANK\_SNOWPIPE\_CLIENT AUTO\_INGEST = TRUE AS

COPY INTO PROJECT.PUBLIC.CLIENT

FROM '@BANK/client/'

FILE\_FORMAT = CSV;

SHOW PIPES;

SELECT COUNT(\*) FROM account;

SELECT COUNT(\*) FROM district;

SELECT count(\*) FROM DISPOSITION;

SELECT count(\*) FROM CARD;

SELECT count(\*) FROM "`ORDER`";

SELECT count(\*) FROM LOAN;

SELECT count(\*) FROM CLIENT;

select count(\*) from transaction;

LIST@BANK;

ALTER PIPE BANK\_SNOWPIPE\_DISTRICT REFRESH;

ALTER PIPE BANK\_SNOWPIPE\_ACCOUNT REFRESH;

alter pipe BANK\_SNOWPIPE\_DISP refresh;

ALTER PIPE BANK\_SNOWPIPE\_CARD REFRESH;

ALTER PIPE BANK\_SNOWPIPE\_ORDER\_LIST REFRESH;

ALTER PIPE BANK\_SNOWPIPE\_LOAN REFRESH;

ALTER PIPE BANK\_SNOWPIPE\_CLIENT REFRESH;

ALTER PIPE BANK\_SNOWPIPE\_TXNS REFRESH;

select SYSTEM$PIPE\_STATUS('BANK\_SNOWPIPE\_DISTRICT');