-- CREATE TABLE dimDate, DimCustomer, DimMovie, DimStore

--DimDate

CREATE TABLE dimDate

(

date\_key integer NOT NULL PRIMARY KEY,

date date NOT NULL,

year smallint NOT NULL,

quater smallint NOT NULL,

month smallint NOT NULL,

day smallint NOT NULL,

week smallint NOT NULL,

is\_weekend boolean

);

--DimCustomer

CREATE TABLE dimCustomer

(

customer\_key SERIAL PRIMARY KEY,

customer\_id smallint NOT Null,

first\_name varchar(45) NOT Null,

last\_name varchar(45) NOT Null,

email varchar(50),

address varchar(50) NOT Null,

address2 varchar(50),

district varchar(20) NOT Null,

city varchar(50) NOT Null,

country varchar(50) NOT Null,

postal\_code varchar(10),

phone varchar(10) NOT Null,

active smallint NOT Null,

create\_date timestamp NOT Null,

start\_date date NOT Null,

end\_date date NOT Null

);

--DimMovie

CREATE TABLE dimMovie

(

movie\_key SERIAL PRIMARY KEY,

film\_id smallint NOT Null,

title varchar(255) NOT Null,

description text,

release\_year year,

language varchar(50) NOT Null,

original\_language varchar(50) NOT Null,

rental\_duration smallint NOT Null,

length smallint NOT Null,

rating varchar(50) NOT Null,

special\_features varchar(10) NOT Null

);

--DimStore

CREATE TABLE dimStore

(

store\_key SERIAL PRIMARY KEY,

store\_id smallint NOT Null,

address varchar(50) NOT Null,

address2 varchar(50),

district varchar(20) NOT Null,

city varchar(50) NOT Null,

country varchar(50) NOT Null,

postal\_code varchar(10),

manager\_first\_name varchar(45) NOT Null,

manager\_last\_name varchar(45) NOT Null,

start\_date date NOT Null,

end\_date date NOT Null

);

--FactSales

CREATE TABLE factSales

(

sales\_key SERIAL PRIMARY KEY,

date\_key integer REFERENCES dimDate (date\_key),

custom\_key integer REFERENCES dimCustomer (customer\_key),

movie\_key integer REFERENCES dimMovie (movie\_key),

store\_key integer REFERENCES dimStore (store\_key),

sales amount numeric

)

--INSERING data and joining tables to populate the the STAR Schema table with DATA.

--DimDate

INSERT INTO dimdate (date\_key, date, year, quater, month, day, week, is\_weekend)

SELECT DISTINCT (TO\_CHAR (payment\_date :: DATE, 'YYYYMMDD'):: integer) as date\_key,

date (payment\_date) as date,

EXTRACT (year from payment\_date) as year,

EXTRACT (quarter FROM payment\_date) AS quater,

EXTRACT (month FROM payment\_date) AS month,

EXTRACT (day FROM payment\_date) AS day,

EXTRACT (week FROM payment\_date) AS week,

CASE WHEN EXTRACT (ISODOW FROM payment\_date) IN (6, 7) THEN true ELSE false END

FROM payment;

--DimCustomer

INSERT INTO dimCustomer (customer\_key, customer\_id, first\_name, last\_name, email, address,

address2,district,city, country, postal\_code, phone, active, create\_date, start\_date, end\_date)

SELECT c.customer\_id as customer\_key,

c. customer\_id,

c.first\_name,

c.last\_name,

c.email,

a.address,

a.address2,

a.district,

ci.city,

co.country,

a.postal\_code,

a.phone,

c.active,

c.create\_date,

now() as start\_date,

now() as end\_date

FROM customer c

JOIN address a ON (c.address\_id = a address\_id)

JOIN city ci ON (a.city\_id = ci.city\_id)

JOIN country co ON (ci.country\_id = co.country\_id);

--DimMovie

INSERT INTO dimmovie (movie\_key, film\_id, title, description, release\_year, language,

original\_language, rental\_duration, length, rating, special\_features)

SELECT f. film\_id as movie\_key,

f. film\_id,

f.title,

f. description,

f.release\_year,

l.name as language,

orig\_lang.name as original\_language,

f. rental\_duration,

f.length,

f.rating,

f.special\_features

From film f

JOIN language ON (f.language\_id = l.language\_id)

LEFT JOIN language orig\_lang ON (f.language\_id = orig\_lang.language\_id);

--DimStore

INSERT INTO dimstore (store\_key, store\_id, address, address2, district, city, country,

postal\_code, manager\_first\_name, manager\_last\_name ‚start\_date, end\_date)

SELECT s.store\_id as store\_key,

s.store\_id,

a.address,

a.address2,

a.district,

c.city,

co.country,

a.postal\_code,

st.first\_name as manager\_first\_name,

st.last\_name as manager\_last\_name,

now() AS start\_date,

now() AS end\_date

FROM store s

JOIN staff st ON (s.manager\_staff\_id = st.staff\_id)

JOIN address a ON (s.address\_id = a.address\_id)

JOIN city c ON (a.city\_id = c.city\_id)

JOIN country co ON (c.country\_id = co.country\_id)

--FactSales

INSERT INTO factsales (date\_key, customer\_key, movie\_key,

store\_key, sales\_amount)

SELECT

TO\_CHAR (payment\_date :: DATE, 'YYYYMMDD'):: integer AS date\_key,

p. customer\_id as customer\_key,

i.film\_id as movie\_key,

i.store\_id as store\_key,

p.amount as sales\_amount

FROM payment p

JOIN rental r ON (p.rental\_id = r.rental\_id)

JOIN inventory i ON (r.inventory\_id = ¡.inventory\_id);

-- 3NF Schema ( takes longer time to produce the result)

SELECT f.title, EXTRACT(month FROM p.payment\_date) as month, ci.city, sum(p.amount) as revenue

FROM payment p

JOIN rental r ON (p.rental\_id = r.rental\_id)

JOIN inventory i ON (r.inventory\_id = i.inventory\_id)

JOIN film f ON (i.film\_id = f.film\_id)

JOIN customer c ON (p.customer\_id = c.customer\_id)

JOIN address a ON (c.address\_id = a.address\_id)

JOIN city ci ON (a.city\_id = ci.city\_id)

GROUP BY (f.title, month, ci.city)

order BY f.title, month, ci.city, revenue desc;

--STAR Schema ( takes less time to produce the same result)

SELECT dimMovie.title, dimDate.month, dimCustomer.city, sum(sales\_amount) as revenue

FROM factSales

JOIN dimMovie ON (dimMovie.movie\_key = factsales.movie\_key)

JOIN dimDate ON (dimDate.date\_key = factsales.date\_key)

JOIN dimCustomer ON (dimCustomer.customer\_key = factsales.customer\_key)

group by (dimMovie.title, dimDate.month, dimCustomer.city)

order by dimMovie.title, dimDate.month, dimCustomer.city, revenue desc;