SQL ASSIGNMENT 1 SOLUTION  
SECTION A  
1. create table Customers(

customer\_id int,

store\_id int,

first\_name varchar,

last\_name varchar,

email varchar,

address\_id int,

active int,

create\_data varchar,

last\_update varchar

);

copy Customers from 'C:\Program Files\PostgreSQL\16\data\Import data\SQl Dataset Customer.csv' delimiter ',' csv header;

2. -- Solution 1

SELECT store\_id as Stores,SUM(active) as Active\_Customers FROM Customers GROUP BY store\_id

--Solution 2

CREATE table Active\_Customers AS (SELECT first\_name,last\_name,address\_id,customer\_id from Customers where active=1 AND SUBSTRING(first\_name, 1, 1)=SUBSTRING(last\_name, 1, 1)

)

3. copy Films from 'C:\Program Files\PostgreSQL\16\data\Import data\Film Dataset.csv' delimiter ',' csv header;

--solution 1

select rental\_duration,sum(revenue) as revenue,count(rental\_duration ) as most\_popular from (SELECT rental\_duration,(rental\_duration\*rental\_rate) as revenue from Films ) group by rental\_duration order by count(rental\_duration ) desc

--solution 2

select speacial\_features ,sum(revenue) as rent\_received,sum(replacement\_cost) as replacement\_cost from (SELECT speacial\_features ,CASE WHEN rental\_duration<5 then (rental\_duration\*rental\_rate) else (4\*rental\_rate) end as revenue,replacement\_cost from Films) group by speacial\_features

--solution 3

select rating,rental\_duration,revenue from (select rating,rental\_duration,revenue,row\_number() over (partition by rating order by revenue desc ) as row\_num from( select rating,rental\_duration,sum(revenue) as revenue from (SELECT \*,(rental\_duration\*rental\_rate) as revenue from Films) group by rating,rental\_duration)) where row\_num=1

--solution 4

Create table new\_table as (select title,description,release\_year,rating,rental\_duration from Films)

select rating,sum(rental\_duration) over (partition by rating) as total\_rental\_duration from new\_table

SECTION 2

1. --a)

select CONCAT(name,' ', surname ),concat(StreetAddress,',',City),concat(state,'-',to\_char(zipcode,'99999'))from owners

--b)

select count(\*) from owners where substring(name,1,1)='S' or substring(name,1,1)='s'

--c)

select streetaddress from owners where substring(name,1,1)='M' or substring(name,1,1)='m'

--d)

select CONCAT(name,' ', surname ) from owners where length(CONCAT(name, surname ))>12

2.a) select petid,name,case when age>12 then 'old' else 'young' end as Category from pets

b) select kind,sum(male),sum(female) from (select kind,case when gender='male' then 1 else 0 end male,case when gender='male' then 0 else 1 end female from pets) group by kind

c) SELECT kind,count FROM (select kind,count(\*) from pets group by kind)

UNION

SELECT gender,count FROM (select gender,count(\*) from pets group by gender)

3. –a)

select petid,extract(day from date) as day,extract(month from date) as month,extract(year from date) as year from procedurehist

--b)

select weekday,count(distinct petid) from (SELECT \*,TO\_CHAR(date, 'Day') as weekday

FROM Procedurehist) where weekday not in ('Sunday','Saturday') group by weekday

4) update proceduretype set price=NULL where price<100

--a)

update proceduretype set price=100 where price is null

select proceduretype,max(price) from proceduretype group by proceduretype

select CAST( proceduretype AS text),CAST( proceduresubcode AS text),CAST( description AS text),CAST( price AS text) from proceduretype

--