/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* File: Assignment3.sql

\*\* Desc: Assignment 3

\*\* Auth: Janmejay

\*\* Date: 21.6.2018

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# a) Show the list of databases.

show databases;

# b) Select sakila database.

USE sakila;

# c) Show all tables in the sakila database.

show tables;

# d) Show each of the columns along with their data types for the actor table.

SHOW COLUMNS FROM sakila.actor;

describe sakila.actor;

# e) Show the total number of records in the actor table.

Select Count(\*) FROM sakila.actor;

# f) What is the first name and last name of all the actors in the actor table ?

select first\_name,last\_name from sakila.actor;

# g) Insert your first name and middle initial ( in the last name column ) into the actors table.

INSERT INTO actor VALUES (201,'Janmejay','V','2018-06-19 12:36:30');

# h) Update your middle initial with your last name in the actors table.

update sakila.actor set last\_name='Dwivedi' where last\_name='V';

# i) Delete the record from the actor table where the first name matches your first name.

SET SQL\_SAFE\_UPDATES = 0;

delete from actor where first\_name='Janmejay'

# j) Create a table payment\_type with the following specifications and appropriate data types

# Table Name : “Payment\_type”

# Primary Key: "payment\_type\_id”

# Column: “Type”

# Insert following rows in to the table: 1, “Credit Card” ; 2, “Cash”; 3, “Paypal” ; 4 , “Cheque”

CREATE TABLE Payment\_type (

payment\_type\_id smallint(10) unsigned NOT NULL AUTO\_INCREMENT,

Type varchar(45) NOT NULL,

PRIMARY KEY (payment\_type\_id)

);

insert into Payment\_type values (1,'Credit Card'),(2,'Cash'),(3,'Paypal'),(4,'Cheque');

# k) Rename table payment\_type to payment\_types.

rename table payment\_type to payment\_types;

# l) Drop the table payment\_types.

drop table payment\_types;

################################## QUESTION 2 ################################

# a) List all the movies ( title & description ) that are rated PG-13.

select title, description from film where rating='PG-13';

# b) List all movies that are either PG OR PG-13 using IN operator.

select title, description from film where rating in ('PG','PG-13');

# c) Report all payments greater than and equal to 2$ and Less than equal to 7$.

# Note : write 2 separate queries conditional operator and BETWEEN keyword

select \* from payment where (amount>=2 and amount<=7);

select \* from payment where amount between 2 and 7;

# d) List all addresses that have phone number that contain digits 589, start with 140 or end with 589

# Note : write 3 different queries

select \* from address where phone like '%589%';

select \* from address where phone like '140%';

select \* from address where phone like '%589';

# e) List all staff members ( first name, last name, email ) who have no password.

select first\_name,last\_name,email from staff where ifnull(password,'')='';

# f) Select all films that have title names like ZOO and rental duration greater than or equal to 4

select \* from film where title like '%ZOO%' and rental\_duration>=4;

# g) What is the cost of renting the movie ACADEMY DINOSAUR for 2 weeks ?

# Note : use of column alias

select (rental\_rate\*14) TwoWeekRentalRate from film where title='ACADEMY DINOSAUR';

#h) List all unique districts where the customers, staff, and stores are located

# Note : check for NOT NULL values

select distinct a.district from customer c

join address a on c.address\_id=a.address\_id

where ifnull(a.district,'')<>'';

select distinct a.district from staff c

join address a on c.address\_id=a.address\_id

where ifnull(a.district,'')<>'';

select distinct a.district from store c

join address a on c.address\_id=a.address\_id

where ifnull(a.district,'')<>'';

# i) List the top 10 newest customers across all stores

select \* from customer order by create\_date desc limit 10;

################################## QUESTION 3 ################################

# a) Show total number of movies

select count(film\_id) from film;

# b) What is the minimum payment received and max payment received across all transactions ?

select min(amount) MinAmount, max(amount) MaxAmount from payment;

# c) Number of customers that rented movies between Feb-2005 and May-2005 ( based on payment date ).

select distinct customer\_id from payment where payment\_date between '2005-02-01' and '2005-05-31';

# d) List all movies where replacement\_cost is greater than 15$ or rental\_duration is between 6 and 10 days

select title, description from film where (replacement\_cost>15 or rental\_duration between 6 and 10);

# e) What is the total amount spent by customers for movies in the year 2005 ?

select sum(amount) from payment where year(payment\_date)=2005;

# f) What is the average replacement cost across all movies ?

select avg(replacement\_cost) from film;

# g) What is the standard deviation of rental rate across all movies ?

select stddev(rental\_rate) from film;

# h) What is the midrange of the rental duration for all movies

set @rowindex:=-1;

SELECT AVG(g.grade) FROM (SELECT @rowindex:=@rowindex + 1 AS rowindex,film.rental\_duration AS grade FROM film ORDER BY film.rental\_duration) AS g WHERE

g.rowindex IN (FLOOR(@rowindex / 2) , CEIL(@rowindex / 2));

################################## QUESTION 4 (Optional) ################################

# a) Customers sorted by first name and last name in ascending order.

use sakila

SELECT first\_name,last\_name FROM customer ORDER BY first\_name ASC,last\_name ASC;

# b) Group distinct addresses by district.

SELECT distinct address,district from address order by district

#table has 3 blank district and 2 abu dhabi district.results displayed by above query are different as it does not group the data by district.

# it looks like we can use the below query for the question.

select distinct district,address from address order by district;

# c) Count of movies that are either G/NC-17/PG-13/PG/R grouped by rating.

Select count(title) movies,rating from film where rating IN('G','NC-17','PG-13','PG','R') group by rating

# d) Number of addresses in each district.

SELECT district,COUNT(ADDRESS) from address GROUP BY district;

# e) Find the movies where rental rate is greater than 1$ and order result set by descending order.

SELECT TITLE,description,rental\_rate FROM film WHERE rental\_rate>1 ORDER BY rental\_rate DESC,TITLE DESC,description DESC;

SELECT TITLE,description,rental\_rate FROM film WHERE rental\_rate>1 ORDER BY TITLE DESC,description DESC;

# f) Top 2 movies that are rated R with the highest replacement cost.

SELECT \* FROM FILM WHERE replacement\_cost = (SELECT MAX(replacement\_cost) FROM film WHERE rating='R')

AND RATING='R'

ORDER BY title

limit 2;

# g) Find the most frequently occurring (mode) rental rate across products.

SELECT rental\_rate,COUNT(rental\_rate) COUNTRR

FROM FILM

GROUP BY rental\_rate

ORDER BY COUNTRR DESC

limit 1;

# h) Find the top 2 movies with movie length greater than 50mins and which has commentaries as a special features.

SELECT title, description FROM film WHERE LENGTH>50 AND special\_features LIKE '%Commentaries%' limit 2;

# i) List the years with more than 2 movies released.

SELECT release\_year,COUNT(release\_year) count\_ry FROM film GROUP BY release\_year HAVING count\_ry>2;