EXP-3

Consider the schema for Movie Database:

ACTOR(Act_id, Act_Name, Act_Gender)

DIRECTOR(Dir_id, Dir_Name, Dir_Phone)

MOVIES(Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)

MOVIE_CAST(Act_id, Mov_id, Role)

RATING(Mov_id, Rev_Stars)

create database Movie;

Query OK, 1 row affected (0.05 sec)

use Movie;

Database changed

create table actor(act_id int primary key, act_name varchar(30),gender char(10));

Query OK, 0 rows affected (0.52 sec)

create table director(dir_id int primary key, dir_name varchar(20), dir_phone bigint);

Query OK, 0 rows affected (0.35 sec)

create table movies (mov_id int primary key, mov_title varchar(20), mov_year year, mov_lang varchar(20), dir_id int references director(dir_id));

Query OK, 0 rows affected (0.28 sec)

create table movie_cast(act_id int references actor(act_id), mov_id int references movies(mov_id), role varchar(20),primary key(act_id,mov_id));

Query OK, 0 rows affected (0.34 sec)

create table rating (mov_id int primary key references movies(mov_id),rev_stars varchar(20));

Query OK, 0 rows affected (0.29 sec)

insert into actor values(301,"anshika",'f');

Query OK, 1 row affected (0.09 sec)

insert into actor values(302,"prabhas",'m');

Query OK, 1 row affected (0.07 sec)

insert into actor values(303,"punith",'m');

Query OK, 1 row affected (0.07 sec)

insert into actor values(304,"jermy",'m');

Query OK, 1 row affected (0.07 sec)

select * from actor;

```
+--------+----------+--------+
| act_id | act_name | gender |
+--------+----------+--------+
|    301 | anshika  | f      |
|    302 | prabhas  | m      |
|    303 | punith   | m      |
|    304 | jermy    | m      |
```

```
+--------+----------+--------+
```

4 rows in set (0.04 sec)

insert into director values(60,"rajamouli",987653245);

Query OK, 1 row affected (0.06 sec)

insert into director values(61,"hitchcock",887663245);

Query OK, 1 row affected (0.07 sec)

 insert into director values(62,"faran",997663245);

Query OK, 1 row affected (0.07 sec)

insert into director values(63,"steven spielberg",897663245);

Query OK, 1 row affected (0.07 sec)

select * from director;

```
+--------+-----------------+-----------+
| dir_id | dir_name        | dir_phone |
+--------+-----------------+-----------+
|     60 | rajamouli       | 987653245 |
|     61 | hitchcock       | 887663245 |
|     62 | faran           | 997663245 |
|     63 | steven spielberg | 897663245 |
+--------+-----------------+-----------+
```

4 rows in set (0.00 sec)

insert into movies values(1001,"bahubali-2",2017,"telagu",60);

Query OK, 1 row affected (0.08 sec)

insert into movies values(1002,"bahubali-1",2015,"telagu",60);

Query OK, 1 row affected (0.07 sec)

insert into movies values(1003,"akash",2008,"kannada",61);

Query OK, 1 row affected (0.10 sec)

insert into movies values(1004,"war horse",2011,"english",63);

Query OK, 1 row affected (0.06 sec)

select * from movies;

```
+--------+------------+----------+----------+--------+
| mov_id | mov_title  | mov_year | mov_lang | dir_id |
+--------+------------+----------+----------+--------+
|   1001 | bahubali-2 |     2017 | telagu   |     60 |
|   1002 | bahubali-1 |     2015 | telagu   |     60 |
|   1003 | akash      |     2008 | kannada  |     61 |
|   1004 | war horse  |     2011 | english  |     63 |
+--------+------------+----------+----------+--------+
```
4 rows in set (0.00 sec)

insert into movie_cast values(301,1002,"heroine");

Query OK, 1 row affected (0.08 sec)

insert into movie_cast values(301,1001,"heroine");

Query OK, 1 row affected (0.05 sec)

insert into movie_cast values(303,1003,"hero");

Query OK, 1 row affected (0.07 sec)

 insert into movie_cast values(303,1002,"guest");

Query OK, 1 row affected (0.06 sec)

insert into movie_cast values(304,1004,"hero");

Query OK, 1 row affected (0.03 sec)

select * from movie_cast;

```
+--------+--------+---------+
| act_id | mov_id | role    |
+--------+--------+---------+
|   301 |   1001 | heroine |
|   301 |   1002 | heroine |
|   303 |   1002 | guest   |
|   303 |   1003 | hero    |
|   304 |   1004 | hero    |
+--------+--------+---------+
```

5 rows in set (0.00 sec)

insert into rating values(1001,4);

Query OK, 1 row affected (0.04 sec)

insert into rating values(1002,2);

Query OK, 1 row affected (0.07 sec)

insert into rating values(1003,5);

Query OK, 1 row affected (0.05 sec)

insert into rating values(1004,4);

Query OK, 1 row affected (0.07 sec)

select * from rating;

+--------+-----------+

| mov_id | rev_stars |

+--------+-----------+

| 1001 | 4 |

| 1002 | 2 |

| 1003 | 5 |

| 1004 | 4 |

+--------+-----------+

4 rows in set (0.00 sec)

Write SQL queries to
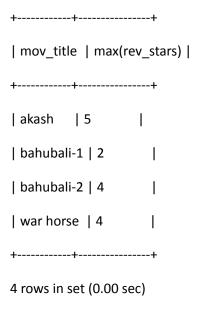
1. List the titles of all movies directed by 'Hitchcock'.

select mov_title from movies where dir_id =(select dir_id from director where dir_name="hitchcock");

```
+-----------+
| mov_title |
+-----------+
| akash     |
+-----------+
```
1 row in set (0.04 sec)

2. Find the movie names where one or more actors acted in two or more movies.

select mov_title from movies m,movie_cast mc where m.mov_id = mc.mov_id and act_id in(select act_id from movie_cast group by act_id having count(act_id)>1)group by mov_title having count(*)>1;

```
+------------+
| mov_title  |
+------------+
| bahubali-1 |
+------------+
```
1 row in set (0.05 sec)

3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

select act_name from actor a join movie_cast c on a.act_id = c.act_id join movies m on c.mov_id = m.mov_id where m.mov_year not between 2000 and 2015;

```
+----------+
| act_name |
+----------+
| anshika  |
+----------+
```

1 row in set (0.04 sec)

4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.

select mov_title, max(rev_stars)from movies inner join rating using (mov_id) group by mov_title having max(rev_stars) >0 order by mov_title;

```
+------------+----------------+
| mov_title  | max(rev_stars) |
+------------+----------------+
| akash      | 5              |
| bahubali-1 | 2              |
| bahubali-2 | 4              |
| war horse  | 4              |
+------------+----------------+
```

4 rows in set (0.00 sec)

5. Update rating of all movies directed by 'Steven Spielberg' to 5.

update rating set rev_stars = 5 where mov_id in (select mov_id from movies where dir_id in (select dir_id from director where dir_name ='steven spielberg'));

Query OK, 1 row affected (0.08 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from rating;

```
+--------+-----------+
| mov_id | rev_stars |
+--------+-----------+
|   1001 | 4         |
```

```
|  1002 | 2      |
|  1003 | 5      |
|  1004 | 5      |
+--------+-----------+
```

4 rows in set (0.00 sec)