

QUESTION:JOIN

Consider the schema for MovieDatabase:

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)

Write SQL queries to

1. List the titles of all movies directed by 'Hitchcock'.
2. Find the movie names where one or more actors acted in two or more movies.
3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).
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.
5. Update rating of all movies directed by 'Steven Spielberg' to 5

PROGRAM

```
CREATE DATABASE MOVIEE;
```

```
USE MOVIEE;
```

```
CREATE TABLE actor(act_id int,act_name char(20),act_gender char(15));
```

```
INSERT INTO ACTOR VALUES(1001,'PRABHAS','MALE');
```

```
INSERT INTO ACTOR VALUES(1002,'RAJKUMAR','MALE');
```

```
INSERT INTO ACTOR VALUES(1003,'SNEHA','FEMALE');
```

```
INSERT INTO ACTOR VALUES(1004,'AISHWARAYA','FEMALE');
```

```
INSERT INTO ACTOR VALUES(1005,'RITHIK','MALE');
```

```
CREATE TABLE director(dir_id int,dir_Name char(20),dir_Phone int);
```

```
INSERT INTO DIRECTOR VALUES(2001,'HITCHCOCK',963963960);
```

```
INSERT INTO DIRECTOR VALUES(2002,'STEVEN SPIELBERG',888777991);
```

```
INSERT INTO DIRECTOR VALUES(2003,'RAJMOULI',999555666);
```

```
INSERT INTO DIRECTOR VALUES(2004,'ATLEE',987456987);
```

```
INSERT INTO DIRECTOR VALUES(2005,'HEMANTH',987455780);
```

```
CREATE TABLE MOVIES(Mov_id INT,Mov_Title CHAR(20), Mov_Year INT,  
Mov_Lang CHAR(20), Dir_id INT);
```

```
INSERT INTO MOVIES VALUES(3001,'BAHUBALI',2014,'HINDI',2003);
```

```
INSERT INTO MOVIES VALUES(3002,'BAHUBALI',2014,'TELUGU',2003);
```

```
INSERT INTO MOVIES VALUES(3003,'BAHUBALI 2',2016,'HINDI',2003);
```

```
INSERT INTO MOVIES VALUES(3004,'BAHUBALI 2',2016,'TELUGU',2003);
```

```
INSERT INTO MOVIES VALUES(3005,'PSYCHO ',1999,'ENGLISH',2001);
```

```
INSERT INTO MOVIES VALUES(3006,'TERMINATOR',2000,'ENGLISH',2002);
```

```
CREATE TABLE MOVIE_CAST(Act_id INT,Mov_id INT,Role CHAR(20) );
```

```
INSERT INTO MOVIE_CAST VALUES(1001,3001,'HERO');
```

```
INSERT INTO MOVIE_CAST VALUES(1001,3002,'HERO');
```

```
INSERT INTO MOVIE_CAST VALUES(1001,3003,'HERO');
```

```
INSERT INTO MOVIE_CAST VALUES(1001,3004,'HERO');
```

```
INSERT INTO MOVIE_CAST VALUES(1003,3001,'HEROINE');
```

```
INSERT INTO MOVIE_CAST VALUES(1003,3002,'HEROINE');
```

```
INSERT INTO MOVIE_CAST VALUES(1004,3001,'HEROINE');
```

```
INSERT INTO MOVIE_CAST VALUES(1004,3002,'HEROINE');
```

```
INSERT INTO MOVIE_CAST VALUES(1001,3005,'HERO');
```

```
INSERT INTO MOVIE_CAST VALUES(1001,3006,'HERO');
```

```
CREATE TABLE RATING(Mov_id INT,Rev_Stars FLOAT(10));
```

```
INSERT INTO RATING VALUES(3001,'8');
```

INSERT INTO RATING VALUES(3002,'7');

INSERT INTO RATING VALUES(3003,'6');

INSERT INTO RATING VALUES(3004,'7');

INSERT INTO RATING VALUES(3005,'4');

INSERT INTO RATING VALUES(3006,'2');

QUERY

1. SELECT MOV_TITLE FROM MOVIES INNER JOIN DIRECTOR ON
MOVIES.Dir_id=DIRECTOR.Dir_id where DIRECTOR.dir_name='HITCHCOCK';

2. 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;

3. SELECT ACT_NAME FROM ACTOR A, MOVIE_CAST MC WHERE
A.ACT_ID=MC.ACT_ID AND MC.MOV_ID IN (SELECT MOV_ID FROM
MOVIES WHERE MOV_YEAR NOT BETWEEN 2000 AND 2015) GROUP BY
ACT_NAME HAVING COUNT(*)>1;

4. SELECT MOV_TITLE,MAX(REV_STARS) FROM MOVIES M INNER JOIN
RATING R ON M.MOV_ID=R.MOV_ID GROUP BY MOV_TITLE ORDER BY
MOV_TITLE;

5. UPDATE RATING SET REV_STARS=5 WHERE MOV_ID IN (SELECT
MOV_ID FROM MOVIES M,DIRECTOR D WHERE D.DIR_ID=M.DIR_ID AND
DIR_NAME='STEVEN SPIELBERG');

OUTPUT

```
45
46
47 • SELECT MOV_TITLE FROM MOVIES INNER JOIN DIRECTOR ON MOVIES.Dir_id=DIRECTOR.Dir_id where
48
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

MOV_TITLE
▶ BABY DRIVER

```
44 • INSERT INTO RATING VALUES(3006,'2');
45
46
47 • SELECT MOV_TITLE FROM MOVIES INNER JOIN DIRECTOR ON MOVIES.Dir_id=DIRECTOR.Dir_id where DIRECTOR.dir_name='HITCHCOCK';
48
49 • SELECT MOV_TITLE
50 FROM MOVIES M,MOVIE_CAST MC
51 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_TI
52
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:




MOV_TITLE
▶ BAHUBALI
BAHUBALI 2

```

49 • SELECT MOV_TITLE
50 FROM MOVIES M,MOVIE_CAST MC
51 WHERE M.MOV_ID=MC.MOV_ID AND ACT_ID IN (SELECT ACT_ID FROM MOVIE_CAST GROUP BY ACT_ID HAVING COUNT(*) > 1)
52
53
54 • SELECT ACT_NAME FROM ACTOR A, MOVIE_CAST MC WHERE A.ACT_ID=MC.ACT_ID AND MC.MOV_ID IN (SELECT MOV_ID FROM MOVIES M WHERE M.REV_STARS > 4)
55

```

<

Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 




ACT_NAME
PRABHAS

```

53
54 • SELECT ACT_NAME FROM ACTOR A, MOVIE_CAST MC WHERE A.ACT_ID=MC.ACT_ID AND MC.MOV_ID IN (SELECT MOV_ID FROM MOVIES M WHERE M.REV_STARS > 4)
55
56 • SELECT MOV_TITLE,MAX(REV_STARS) FROM MOVIES M INNER JOIN MOVIE_CAST MC ON M.MOV_ID=MC.MOV_ID WHERE M.REV_STARS > 4

```

<

Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

MOV_TITLE	MAX(REV_STARS)
AVATAR	4
BABY DRIVER	4
BAHUBALI	8
BAHUBALI 2	7
INCEPTION	4.5
TERMINATOR	2
VERTIGO	3.5

Result 8 x

Output