

```
CREATE TABLE ACTOR (ACT_ID NUMBER (3), ACT_NAME VARCHAR (20),  
ACT_GENDER CHAR (1), PRIMARY KEY (ACT_ID));
```

```
CREATE TABLE DIRECTOR (DIR_ID NUMBER (3), DIR_NAME VARCHAR (20),  
DIR_PHONE NUMBER (10), PRIMARY KEY (DIR_ID));
```

```
CREATE TABLE MOVIES ( MOV_ID NUMBER (4), MOV_TITLE VARCHAR (25),  
MOV_YEAR NUMBER (4), MOV_LANG VARCHAR (12), DIR_ID NUMBER (3), PRIMARY  
KEY (MOV_ID), FOREIGN KEY (DIR_ID) REFERENCES DIRECTOR (DIR_ID));
```

```
CREATE TABLE MOVIE_CAST (ACT_ID NUMBER (3), MOV_ID NUMBER (4), ROLE  
VARCHAR (10), PRIMARY KEY (ACT_ID, MOV_ID), FOREIGN KEY (ACT_ID)  
REFERENCES ACTOR (ACT_ID), FOREIGN KEY (MOV_ID) REFERENCES MOVIES  
(MOV_ID));
```

```
CREATE TABLE RATING ( MOV_ID NUMBER (4), REV_STARS VARCHAR (25), PRIMARY  
KEY (MOV_ID), FOREIGN KEY (MOV_ID) REFERENCES MOVIES (MOV_ID));
```

Insertion of Values to Tables

```
INSERT INTO ACTOR VALUES (301,'ANUSHKA','F');
```

Insert as many as required rows

```
INSERT INTO DIRECTOR VALUES (60,'RAJAMOULI', 1111122222);
```

Insert as many as required rows

```
INSERT INTO MOVIES VALUES (1001,'BAHUBALI-2', 2017, 'TELAGU', 60);
```

Insert as many as required rows

```
INSERT INTO MOVIE_CAST VALUES (301, 1002, 'HEROINE');
```

Insert as many as required rows

```
INSERT INTO RATING VALUES (1001, 4);
```

Insert as many as required rows

Queries:

1. List the titles of all movies directed by 'Hitchcock'.

```
SELECT MOV_TITLE FROM MOVIES  
WHERE DIR_ID IN (SELECT DIR_ID FROM DIRECTOR  
WHERE DIR_NAME = 'HITCHCOCK');
```

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

```
SELECT MOV_TITLE FROM MOVIES M, MOVIE_CAST MV  
WHERE M.MOV_ID=MV.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. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

```
SELECT ACT_NAME, MOV_TITLE, MOV_YEAR 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;  
OR  
SELECT A.ACT_NAME, A.ACT_NAME, C.MOV_TITLE, C.MOV_YEAR  
FROM ACTOR A, MOVIE_CAST B, MOVIES C  
WHERE A.ACT_ID=B.ACT_ID  
AND B.MOV_ID=C.MOV_ID  
AND C.MOV_YEAR NOT BETWEEN 2000 AND 2015;
```

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;
```

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'));
```