**DBMS LAB 9**

**PROGRAM 9: MOVIE DATABASE**

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;**

**Use movie;**

**create table Actor**

**(**

**act\_id integer primary key,**

**act\_name varchar(100),**

**act\_gender varchar(10)**

**);**

**create table Director**

**(**

**dir\_id integer primary key,**

**dir\_name varchar(200),**

**dir\_phone varchar(100)**

**);**

**create table Movies**

**(**

**mov\_id integer primary key,**

**mov\_title varchar(255),**

**mov\_year year,**

**mov\_lang varchar(100),**

**dir\_id int,**

**foreign key (dir\_id) references Director(dir\_id)**

**);**

**create table Movie\_cast**

**( act\_id int,**

**foreign key (act\_id) references Actor(act\_id),**

**mov\_id int,**

**foreign key(mov\_id) references Movies(mov\_id),**

**role varchar(100),**

**primary key(act\_id,mov\_id)**

**);**

**create table Rating**

**(**

**mov\_id integer primary key ,**

**foreign key(mov\_id) references Movies(mov\_id),**

**rev\_stars integer**

**);**

**insert into Actor values**

**(1001, 'Tom Crusie','M'),**

**(1002, 'Chris Hemsworth','M'),**

**(1003, 'Angelina Jolie','F'),**

**(1004, 'Margot Robbie','F'),**

**(1005, 'Kate Winslet','F'),**

**(1006, 'Robert Downey','M');**

**insert into Director values**

**(9001, 'Hitchcock',9874562154),**

**(9002, 'Steven Spielberg',9874560054),**

**(9003, 'Joseph Levitan',9874562178),**

**(9004, 'Christopher Loyd',9874564454),**

**(9005, 'Yash Chopra',9874562994),**

**(9006, 'Tom Jones',9874503154);**

**insert into Movies values**

**(101,'Iron Man',2014,'English',9001), (102,'Prosperity',2001,'Spanish',9001), (103,'Spiderman',1998,'English',9002), (104,'Star Wars',1999,'English',9003), (105,'Thor',2017,'English',9002),(106,'Captain America',1994,'English',9004);**

**insert into Movie\_cast values**

**(1001,101,'Joey'),**

**(1001,102,'Conor'),**

**(1002,102,'Tim'),**

**(1003,103,'Kate'),**

**(1004,104,'Claire'),**

**(1006,105,'Sally'),**

**(1005,106,'Jo'),**

**(1002,106,'Craft'),**

**(1002,104,'Josh'),**

**(1005,105,'Roy');**

**insert into Rating values (101,4),**

**(102,3),**

**(103,5),**

**(104,2),**

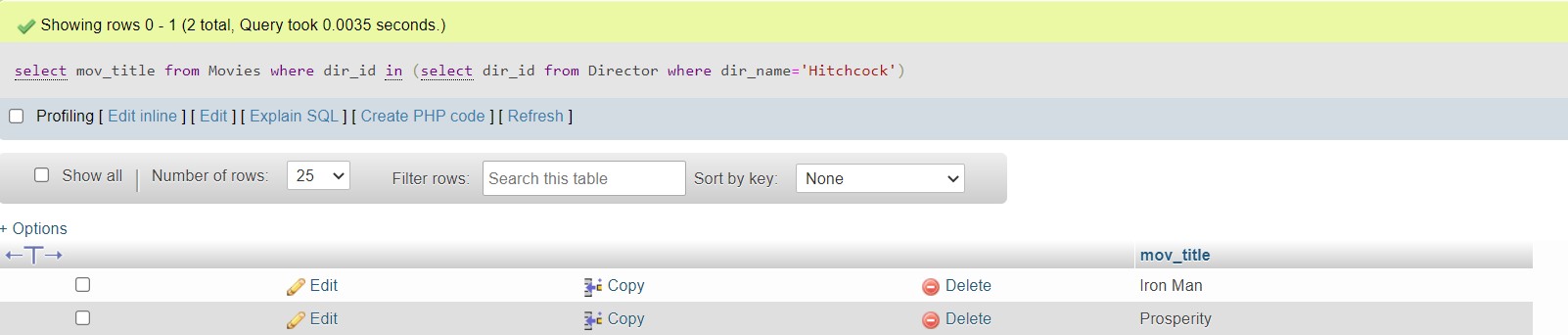
**(105,4),**

**(106,3);**

Write SQL queries to

i.List the titles of all movies directed by ‘Hitchcock’.

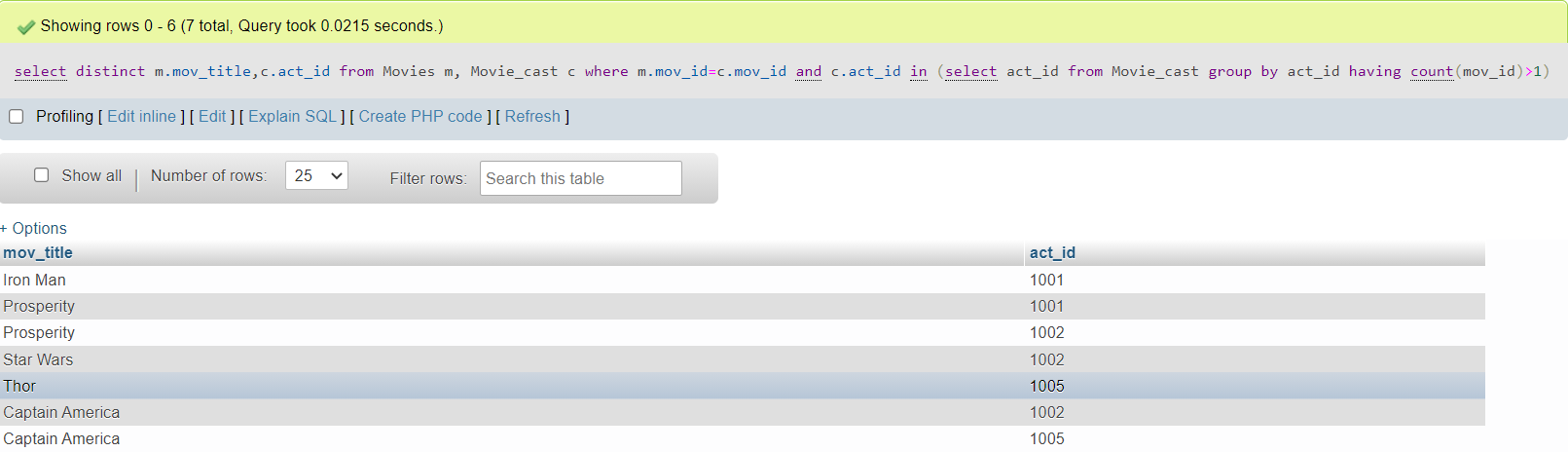
**select mov\_title from Movies where dir\_id in (select dir\_id from Director where dir\_name='Hitchcock');**



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

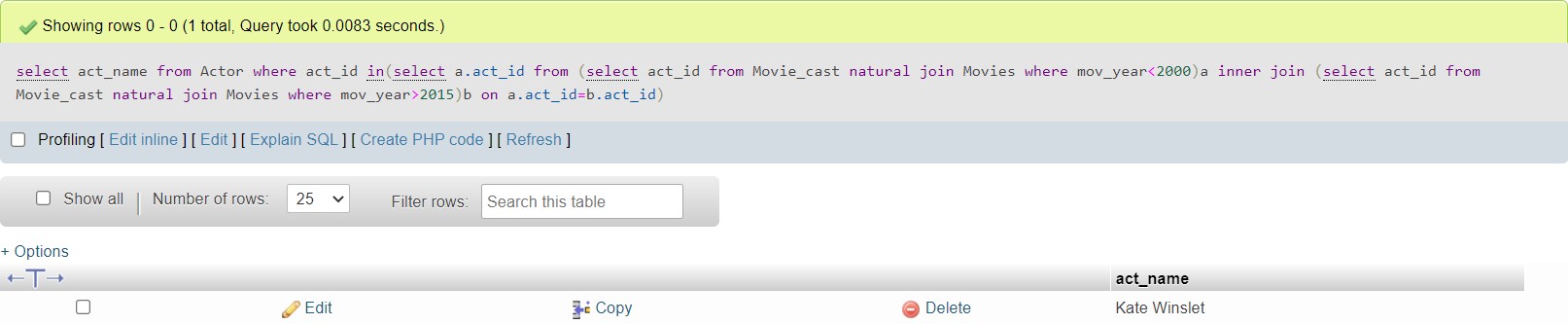
**select distinct m.mov\_title,c.act\_id from Movies m, Movie\_cast c where m.mov\_id=c.mov\_id and**

**c.act\_id in (select act\_id from Movie\_cast group by act\_id having count(mov\_id)>1);**



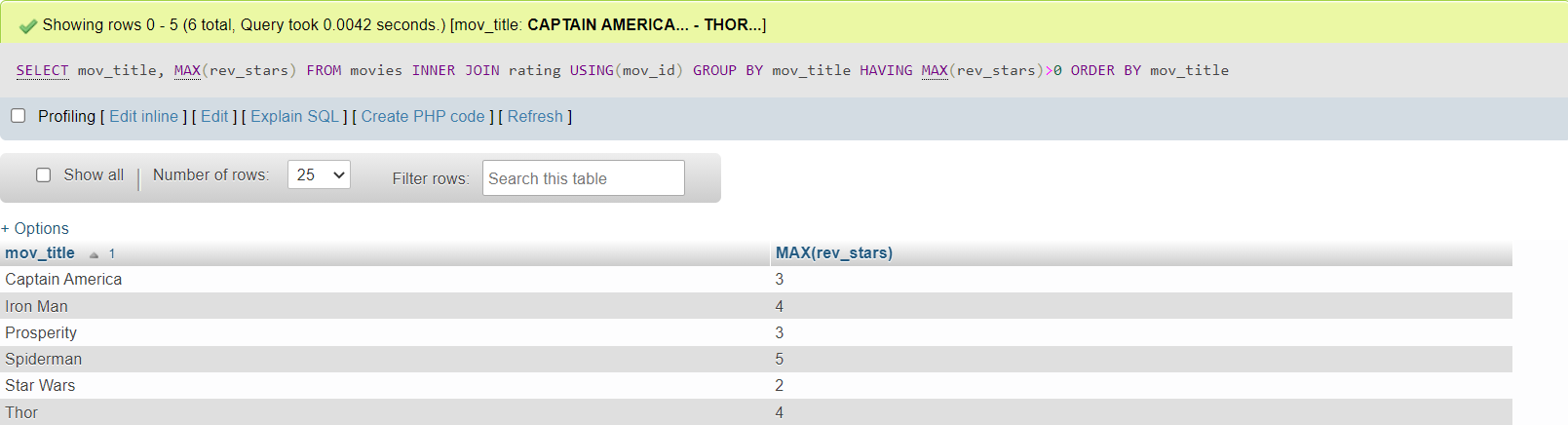
iii.List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

**select act\_name from Actor whereact\_id in(select a.act\_id from (select act\_id from Movie\_cast natural join Movies where mov\_year<2000)a inner join (select act\_id from Movie\_cast natural join Movies where mov\_year>2015)b on a.act\_id=b.act\_id);**



iv.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 movie inner join rating using(mov\_id) group by mov\_title having max(rev\_stars)>0 order by mov\_title;**



v.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 inner join Director on Movies.dir\_id=Director.dir\_id and Director.dir\_name='Steven Spielberg');**

