SQL Assignment-5

1. **Write SQL query to show all the data in the Movie table. Answer:** SELECT \* FROM movie;

# Write SQL query to show the title of the longest runtime movie.

**Answer:** SELECT title FROM movie WHERE runtime= (SELECT MAX (runtime) FROM movie;

1. **Write SQL query to show the highest revenue generating movie title. Answer:** SELECT title FROM movie WHERE revenue= (SELECT MAX (revenue) FROM movie;

# Write SQL query to show the movie title with maximum value of revenue/budget.

**Answer**: SELECT title FROM movie WHERE revenue= (SELECT MAX (revenue) FROM movie OR budget= (SELECT MAX (budget) FROM movie;

# Write a SQL query to show the movie title and its cast details like name of the person, gender, character name, cast order.

**Answer:** SELECT movie.title, person.person\_name, gender.gender, movie\_cast.cast\_order FROM movie\_cast

INNER JOIN movie

ON movie\_cast.movie\_id= movie. movie\_id INNER JOIN person

ON movie\_cast. person\_id= person. person \_id INNER JOIN gender

ON movie\_cast.gender\_id= gender. gender\_id;

1. **Write a SQL query to show the country name where maximum number of movies has been produced, along with the number of movies produced. Answer:** SELECT country\_name, count(country\_name) AS no\_mov\_prd FROM country

INNER JOIN production\_country

ON country. country\_id= production. country \_id GROUP BY country\_name

ORDER BY count(country.country\_name) desc limit 1;

# Write a SQL query to show all the genre\_id in one column and genre\_name in second column.

**Answer:** SELECT \* FROM genre;

# Write a SQL query to show name of all the languages in one column and number of movies in that particular column in another column.

**Answer:** SELECT language\_name, count (language\_name) AS no\_of\_movie FROM language

INNER JOIN movie\_ language

ON movie\_ language. language\_ id= language.language \_ id INNER JOIN movie

ON movie\_ language.movie\_ id=movie. movie\_ id GROUP BY language\_name;

1. **Write a SQL query to show movie name in first column, no. of crew members in second column and number of cast members in third column. Answer:** SELECT movie.title, count (movie\_crew.job), count(movie\_cast.character\_name)

FROM movie\_crew INNER JOIN movie

ON movie\_crew.movie\_id= movie.movie\_id INNER JOIN movie\_cast

ON movie\_crew.movie\_id=movie\_cast.movie\_id GROUP BY movie.title;

# Write a SQL query to list top 10 movies title according to popularity column in decreasing order.

**Answer:** SELECT title, popularity FROM movie ORDER BY popularity DESC

LIMIT 10;

# Write a SQL query to show the name of the 3rd most revenue generating movie and its revenue.

**Answer:** SELECT title, revenue FROM movie ORDER BY revenue DESC

LIMIT 2,1;

# Write a SQL query to show the names of all the movies which have

**“rumoured” movie status.**

**Answer:** SELECT tiltle FROM movie WHERE movie\_status= “rumoured”;

# Write a SQL query to show the name of the “United States of America”

**produced movie which generated maximum revenue.**

**Answer:** SELECT movie.title, production\_company.company\_name, max(movie.revenue) FROM movie\_company

INNER JOIN movie

ON movie\_company.movie\_id= movie.movie\_id INNER JOIN production\_company

ON movie\_company.company\_id=production\_company.company\_id WHERE production\_company.company\_name= “United States of America” ORDER BY revenue DESC;

# Write a SQL query to print the movie\_id in one column and name of the production company in the second column for all the movies.

**Answer:** SELECT movie.movie\_id, production\_company.company\_name FROM movie\_company.movie\_id=movie.movie\_id

INNER JOIN production\_company

ON movie\_company. company\_id= production\_company. company\_id;

# Write a SQL query to show the title of top 20 movies arranged in decreasing order of their budget.

**Answer:** SELECT title from movie ORDER BY budget DESC LIMIT 10;