

SQL WORKSHEET 5

1. Write SQL query to show all the data in the Movie table.

```
SELECT * FROM Movie;
```

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

```
SELECT title FROM movie ORDER BY runtime DESC LIMIT 1;
```

3. Write SQL query to show the highest revenue generating movie title.

```
SELECT title FROM movie ORDER BY revenue DESC LIMIT 1;
```

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

```
SELECT title FROM movie ORDER BY budget DESC LIMIT 1;
```

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

```
SELECT title, gender, character_name, cast_order, person_name FROM movie a INNER JOIN  
movie_cast b on a.movie_id=b.movie_id INNER JOIN gender c on c.gender_id=b.gender_id  
INNER JOIN person d on d.person_id= b.person_id;
```

6. Write a SQL query to show the country name where maximum number of movies has been produced, along with the number of movies produced.

```
SELECT country_name, count(country_name) as count FROM country as a INNER JOIN  
production_country as b on b.country_id=a.country_id group by country_name ORDER BY  
COUNT DESC LIMIT 1;
```

7. Write a SQL query to show all the genre_id in one column and genre_name in second column.

```
SELECT * FROM genre;
```

8. 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.

```
SELECT language_name, movie_id, COUNT(language_name) FROM movie_languages as a  
JOIN language as b on a.language_id=b.language_id GROUP BY language_name ORDER BY  
COUNT(language_name) DESC;
```

9. 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.

```
SELECT m.title as movie_name, COUNT(cr.person_id) as no_of_crews, COUNT(ca.person_id)
as no_of_cast from movie as m INNER JOIN movie_crew as cr on cr.movie_id=m.movie_id
INNER JOIN movie_cast ca on ca.person_id=cr_person_id;
```

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

```
SELECT title FROM movie ORDER BY popularity DESC LIMIT 10;
```

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

```
SELECT title FROM movie ORDER BY revenue DESC offset 3 LIMIT 1;
```

12. Write a SQL query to show the names of all the movies which have “rumoured” movie status.

```
SELECT title FROM movie WHERE movie_status LIKE 'rumored';
```

13. Write a SQL query to show the name of the “United States of America” produced movie which generated maximum revenue.

```
SELECT title, revenue FROM movie a INNER JOIN production_country b on b.movie_id =
a.movie_id INNER JOIN country c on c.country_id = b. country_id WHERE country_name=
'United State of America';
```

14. 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.

```
SELECT m.movie_id, pc.company_name FROM movie m INNER JOIN
movie_company mc on mc.movie_id = m.movie_id INNER JOIN production_company
pc on pc.company_id
=mc.company_id;
```

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

```
SELECT title FROM movie ORDER BY budget DESC LIMIT 20;
```