

SQL WORKSHEET – 5

QUESTIONS:

Q1. Write SQL query to show all the data in the Movie table.

Answer:

```
SELECT * FROM Movie;
```

Q2. Write SQL query to show the title of the longest runtime movie

Answer:

```
SELECT title
```

```
FROM Movie
```

```
ORDER BY runtime DESC
```

```
LIMIT 1;
```

Q3. Write SQL query to show the highest revenue generating movie title

Answer:

```
SELECT title
```

```
FROM Movie
```

```
ORDER BY revenue DESC
```

```
LIMIT 1;
```

Q3. Write SQL query to show the movie title with maximum value of revenue/budget.

Answer:

```
SELECT title
```

```
FROM Movie
```

```
WHERE revenue/budget = (SELECT MAX(revenue/budget) FROM Movie);
```

Q4. Write a SQL query to show the movie title and its cast details like name of the person, gender, charactername, cast order.

Answer:

```
SELECT      Movie.title,      Person.name,      Gender.gender,      Movie_cast.character,  
Movie_cast.cast_order
```

```
FROM Movie
```

```
INNER JOIN Movie_cast ON Movie_cast.movie_id = Movie.id
```

```
INNER JOIN Person ON Person.id = Movie_cast.person_id
```

```
LEFT JOIN Gender ON Gender.id = Movie_cast.gender_id;
```

Q5. 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 c.name AS country_name, COUNT(*) AS num_movies
```

```
FROM movie_country mc
```

```
JOIN country c ON c.id = mc.country_id
```

```
GROUP BY country_name
```

```
ORDER BY num_movies DESC
```

```
LIMIT 1;
```

Q6. Write a SQL query to show all the genre_id in one column and genre_name in second column.

Answer:

```
SELECT id AS genre_id, name AS genre_name
```

```
FROM genre;
```

Q7. 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 l.name AS language_name, COUNT(ml.movie_id) AS num_movies  
FROM language l  
LEFT JOIN movie_languages ml ON l.id = ml.language_id  
GROUP BY l.name  
ORDER BY num_movies DESC;
```

Q8. 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 m.title AS movie_name, COUNT(DISTINCT mc.person_id) AS crew_count,  
COUNT(DISTINCT mcc.person_id) AS cast_count  
FROM movie AS m  
LEFT JOIN movie_crew AS mc ON m.id = mc.movie_id  
LEFT JOIN movie_cast AS mcc ON m.id = mcc.movie_id  
GROUP BY m.title  
ORDER BY m.title;
```

Q9. 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;
```

Q10. Write a SQL query to show the name of the 3rd most revenue generating movie and its revenue.

Answer:

```
SELECT title, revenue  
FROM movie  
WHERE revenue IS NOT NULL  
ORDER BY revenue DESC  
LIMIT 1 OFFSET 2;
```

Q11. Write a SQL query to show the names of all the movies which have “rumoured” movie status.

Answer:

```
SELECT title  
FROM movie  
WHERE status = "Rumoured";
```

Q12. Write a SQL query to show the name of the “United States of America” produced movie which generate maximum revenue.

Answer:

```
SELECT m.title, m.revenue  
FROM movie m  
JOIN movie_country mc ON m.id = mc.movie_id  
JOIN country c ON mc.country_id = c.id  
WHERE c.name = 'United States of America'  
ORDER BY m.revenue DESC  
LIMIT 1;
```

Q13. . 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.id AS movie_id, production_company.name AS company_name  
FROM movie  
INNER JOIN movie_company ON movie.id = movie_company.movie_id  
INNER JOIN production_company ON movie_company.company_id =  
production_company.id;
```

Q14. Write a SQL query to show the title of top 20 movies arranged in decreasing order of their budget.

Answer:

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