WORKSHEET 5 SQL

WORKSHEET ***- 5 (Vivek Kumar Sahu (Internship 35))***

QUESTION & ANSWER:

1. Write SQL query to show all the data in the Movie table.
ANS: SELECT * FROM MOVIE;

2. Write SQL query to show the title of the longest runtime movie.
ANS: SELECT TITLE FROM MOVIE ORDER BY RUNTIME DESC LIMIT 1;

3. Write SQL query to show the highest revenue generating movie title.
ANS: 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.
ANS: SELECT TITLE FROM MOVIE ORDER BY REVENUE/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.
ANS: SELECT TITLE, CHARACTER_NAME, GENDER, CAST_ORDER, PERSON_NAME FROM MOVIE INNER JOIN MOVIE_CAST ON MOVIE.MOVIE_ID=MOVIE_CAST.MOVIE_ID INNER JOIN GENDER ON MOVIE_CAST.GENDER_ID=GENDER.GENDER_ID INNER JOIN PERSON ON MOVIE_CAST.PERSON_ID=PERSON.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.
ANS: SELECT COUNTRY_NAME, COUNT(TITLE) FROM MOVIE INNER JOIN PRODUCTION_COUNTRY ON MOVIE.MOVIE_ID = PRODUCTION_COUNTRY.MOVIE_ID INNER JOIN COUNTRY ON PRODUCTION_COUNTRY.COUNTRY_ID=COUNTRY.COUNTRY_ID GROUP BY COUNTRY_NAME ORDER BY COUNT(TITLE) DESC LIMIT 1;

7. Write a SQL query to show all the genre_id in one column and genre_name in second column.
ANS: 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.
ANS: SELECT LANGUAGE_NAME, COUNT(TITLE) FROM MOVIE INNER JOIN MOVIE_LANGUAGES ON MOVIE.MOVIE_ID = MOVIE_LANGUAGES.MOVIE_ID INNER JOIN LANGUAGE ON MOVIE_LANGUAGES.LANGUAGE_ID=LANGUAGE. LANGUAGE_ID GROUP BY LANGUAGE_NAME;

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.
ANS: SELECT TITLE, COUNT (MOVIE_CAST.PERSON_ID), COUNT(MOVIE_CREW.PERSON_ID) FROM MOVIE INNER JOIN MOVIE_CAST ON MOVIE.MOVIE_ID=MOVIE_CAST.MOVIE_ID INNER JOIN MOVIE_CREW ON MOVIE.MOVIE_ID=MOVIE_CREW.MOVIE_ID;

10. Write a SQL query to list top 10 movies title according to popularity column in decreasing order.
ANS: SELECT TITLE, POPULARITY 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.
ANS: SELECT TITLE, REVENUE FROM MOVIE ORDER BY REVENUE DESC LIMIT 1 OFFSET 2;

12. Write a SQL query to show the names of all the movies which have “rumoured” movie status.
ANS: SELECT TITLE FROM MOVIE WHERE MOVIE_STATUS = ‘rumoured’;

13. Write a SQL query to show the name of the “United States of America” produced movie which generated maximum revenue.
ANS: SELECT TITLE FROM MOVIE INNER JOIN PRODUCTION_COUNTRY ON MOVIE.MOVIE_ID= PRODUCTION_COUNTRY.MOVIE_ID INNER JOIN COUNTRY ON PRODUCTION_COUNTRY.COUNTRY_ID = COUNTRY.COUNTRY_ID WHERE COUNTRY_NAME='United States of America' ORDER BY REVENUE DESC LIMIT 1;

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.
ANS: SELECT MOVIE_COMPANY.MOVIE_ID, COMPANY_NAME FROM MOVIE INNER JOIN MOVIE_COMPANY ON MOVIE.MOVIE_ID = MOVIE_COMPANY.MOVIE_ID INNER JOIN PRODUCTION_COMPANY ON MOVIE_COMPANY.COMPANY_ID = PRODUCTION_ COMPANY.COMPANY_ID;

15. Write a SQL query to show the title of top 20 movies arranged in decreasing order of their budget
ANS: SELECT TITLE FROM MOVIE ORDER BY BUDGET DESC LIMIT 20;