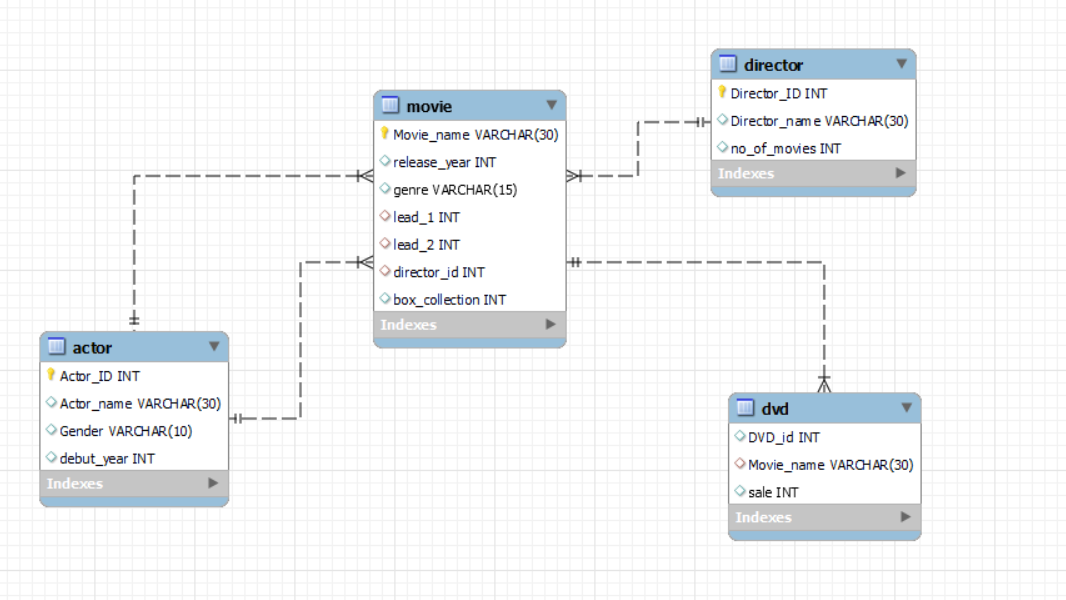
**SQL Group By assignment**

**Database Schema**



Initial queries for database, table creation and inserting values

Create and use database

create database grpby;

use grpby;

Create Tables

CREATE TABLE Actor(Actor\_ID int,Actor\_name varchar(30),Gender varchar(10),debut\_year int);

CREATE TABLE Director(Director\_ID int,Director\_name varchar(30),no\_of\_movies int);

CREATE TABLE Movie(Movie\_name varchar(30),release\_year int,genre varchar(15),lead\_1 int,lead\_2 int,director\_id int,box\_collection int );

CREATE TABLE DVD(DVD\_id int,Movie\_name varchar(30),sale int);

Inserting Values

insert into actor values(1,'Aamir Khan','male',1988);

insert into actor values(2,'Shahrukh Khan','male',1992);

insert into actor values(3,'Kareena Kapoor','female',2000);

insert into actor values(4,'Kajol','female',1992);

insert into actor values(5,'Anushka Sharma','female',2008);

insert into actor values(6,'Ranbir Kapoor','male',2007);

insert into actor values(7,'Alia Bhatt','female',2012);

insert into actor values(8,'Ranveer Singh','male',2010);

insert into actor values(9,'Deepika Padukone','female',2006);

insert into actor values(10,'Shahid Kapoor','male',2003);

insert into actor values(11,'Katrina Kaif','female',2003);

insert into actor values(12,'Varun Dhawan','male',2012);

insert into director values(1,'Rajkumar Hirani',10);

insert into director values(2,'Sanjay Leela Bhansali',8);

insert into director values(3,'Aditya Chopra',6);

insert into director values(4,'Zoya Akhtar',3);

insert into director values(5,'Imtiaz Ali',5);

insert into director values(6,'Ayan Mukerji',4);

insert into director values(7,'Santosh Sivan',9);

insert into director values(8,'Farah Khan',4);

insert into movie values('3 Idiots',2009,'comedy',1,3,1,989600000);

insert into movie values('PK',2014,'satire',1,5,1,789600000);

insert into movie values('Bajirao Mastani',2015,'historic',8,9,2,68960000);

insert into movie values('Rab ne bana di jodi',2008,'romance',2,5,3,48960000);

insert into movie values('Gully boy',2019,'Musical drama',8,7,4,28950000);

insert into movie values('Jab we met',2007,'romance',10,3,5,18980000);

insert into movie values('Padmavat',2018,'historic',8,9,2,38460000);

insert into movie values('Yeh jawani hai deewani',2013,'romance',6,9,6,581600000);

insert into movie values('Ashoka',2001,'historic',2,3,7,8796000);

insert into movie values('Happy new year',2014,'comedy',2,9,8,68960000);

insert into DVD values(1,'3 Idiots',8890);

insert into DVD values(2,'PK',8790);

insert into DVD values(3,'Bajirao Mastani',6810);

insert into DVD values(4,'Rab ne bana di jodi',7920);

insert into DVD values(5,'Gully boy',870);

insert into DVD values(6,'Jab we met',4792);

insert into DVD values(7,'Padmavat',590);

insert into DVD values(8,'Yeh jawani hai deewani',2890);

insert into DVD values(9,'Ashoka',730);

insert into DVD values(10,'Happy new year',1690);

Assignment Questions

Q1. Display the count of actors and actresses.

Output:-

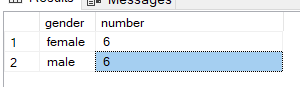
SELECT

Gender, COUNT(Gender) AS Number

FROM

Actor

GROUP BY 1



Q2. Display the count of movies of each genre.

Output:-

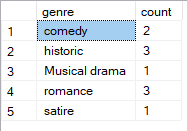
SELECT

genre, COUNT(genre) AS Number

FROM

Movie

GROUP BY 1



Q3. Calculate the oldest released movie of each genre.

SELECT

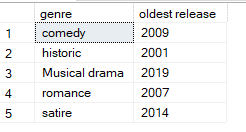
genre, min(release\_year) As Oldest\_Released\_Year

FROM

Movie

GROUP BY 1

Output:-



Q4. Calculate the total box office collection for movies of each genre.

SELECT

genre, sum(box\_collection) As Total\_Collection

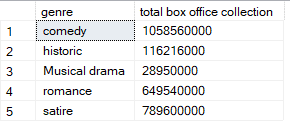
FROM

Movie

GROUP BY 1

order by 2 desc

Output:-



Q5. Display the name of director, current count of movies and total count of

movies directed by him/her.

SELECT

d.Director\_name,

COUNT(m.Movie\_name) AS Current\_Movie\_Count,

d.no\_of\_movies AS Total\_Movies

FROM

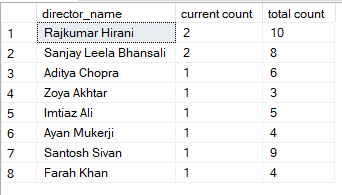
Director d

JOIN

Movie m ON d.Director\_ID = m.Director\_ID

GROUP BY 1 , 3

Output:-



Q6. Display the name of lead\_1 actors, their count of movies and their total DVD sales.

SELECT

m.lead\_1,

COUNT(m.Movie\_name) AS No\_Of\_Movies,

SUM(d.sale) AS Total\_Sales

FROM

Movie m

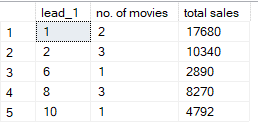
JOIN

DVD d ON m.Movie\_name = d.Movie\_name

GROUP BY 1

ORDER BY 1

Output:-



Q7. Show the release year and number of movies released in each year only if

more than 1 movie is released each year.

SELECT

release\_year, COUNT(\*) AS No\_of\_Movies

FROM

Movie

GROUP BY 1

HAVING No\_of\_Movies > 1

Output:-



Q8. Display the director name and their total box office collection which should be greater than 50 crores.

SELECT

d.Director\_name, SUM(m.box\_collection) AS Total\_Collection

FROM

Director d

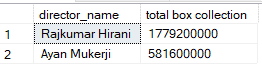
JOIN

Movie m ON d.Director\_ID = m.Director\_ID

GROUP BY 1

HAVING Total\_Collection > 500000000

Output:-



Q9. Give the count of movies of each genre in decreasing order of count.

SELECT

genre, COUNT(Movie\_name) AS Total\_Movie

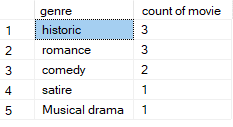
FROM

Movie

GROUP BY 1

ORDER BY 2 DESC

Output:-



Q10. Sort the movie names based on director names.

SELECT

d.Director\_name, m.Movie\_name

FROM

Director d

JOIN

Movie m ON d.Director\_ID = m.Director\_ID

ORDER BY 1

Output:-

