1. Count the total number of records in each table of the database.

delimiter $$

Create procedure NO\_OF\_Records()

begin

select count(\*) as Number\_Of\_Records\_director\_mapping from director\_mapping;

select count(\*) as Number\_Of\_Records\_genre from genre;

select count(\*) as Number\_Of\_Records\_movie from movie;

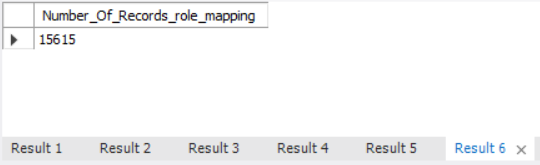
select count(\*) as Number\_Of\_Records\_names from names;

select count(\*) as Number\_Of\_Records\_ratings from ratings;

select count(\*) as Number\_Of\_Records\_role\_mapping from role\_mapping;

end$$

delimiter ;



Notes : Like above all the tables result excecuted in separate result sheets.

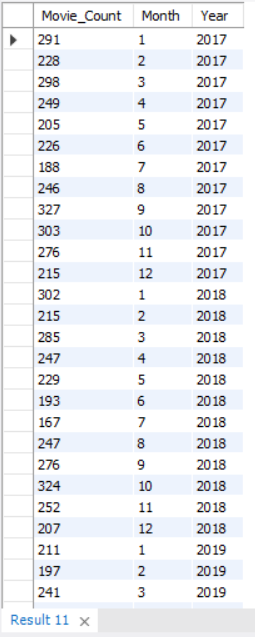
1. Identify which columns in the movie table contain null values.

select \* from movie where id is null or title is null or year is null or date\_published is null or duration is null or country is null or worlwide\_gross\_income is null or languages is null or production\_company is null;



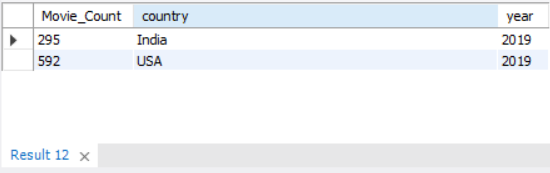
1. Determine the total number of movies released each year, and analyze how the trend changes month-wise.

select count(title) as Movie\_Count,month(date\_published) as Month, Year from movie group by month(date\_published),year order by year,Month asc;



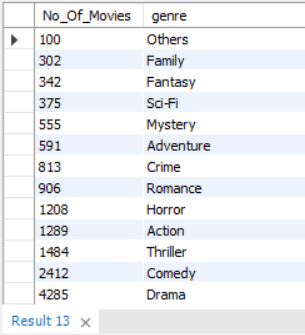
1. How many movies were produced in either the USA or India in the year 2019?

select count(title) as Movie\_Count, country, year from movie where year = 2019 group by country having (country = "USA" or Country = "India");



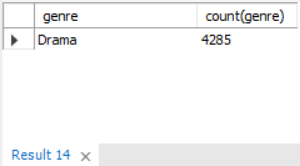
1. List the unique genres in the dataset, and count how many movies belong exclusively to one genre.

select count(genre) as No\_Of\_Movies,genre from genre group by genre order by count(genre);



1. Which genre has the highest total number of movies produced?

select distinct genre,count(genre) from genre group by genre order by count(genre) desc limit 1;



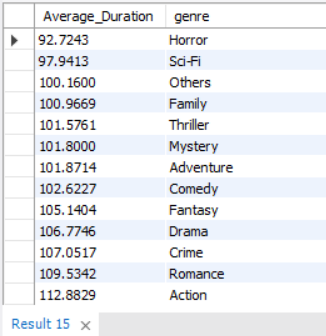
1. Calculate the average movie duration for each genre.

select avg(m.duration) as Average\_Duration,g.genre

from movie m inner join genre g

on m.id = g.movie\_id

group by g.genre order by avg(m.duration) asc;



8. Identify actors or actresses who have appeared in more than three movies with an average rating below 5.

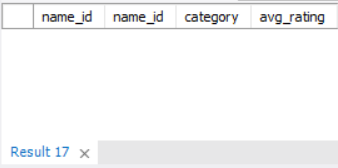
select count(rm.name\_id),rm.name\_id,rm.category,r.avg\_rating

from role\_mapping rm inner join ratings r

on rm.movie\_id = r.movie\_id

where r.avg\_rating <5

group by rm.name\_id,rm.category,r.avg\_rating having rm.name\_id >3;



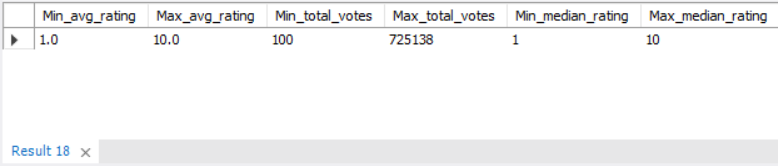
1. Find the minimum and maximum values for each column in the ratings table, excluding the movie\_id column.

select min(avg\_rating) as Min\_avg\_rating,max(avg\_rating) as Max\_avg\_rating,

min(total\_votes) as Min\_total\_votes, max(total\_votes) as Max\_total\_votes,

min(median\_rating) as Min\_median\_rating,max(median\_rating) as

Max\_median\_rating from ratings;

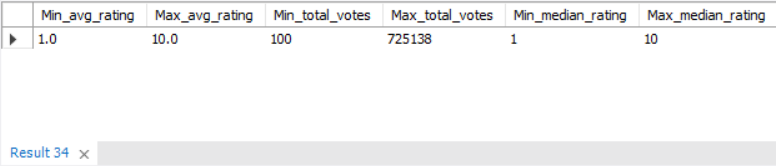


1. Find the minimum and maximum values for each column in the ratings table, excluding the movie\_id column.

select min(avg\_rating) as Min\_avg\_rating,max(avg\_rating) as Max\_avg\_rating,

min(total\_votes) as Min\_total\_votes, max(total\_votes) as Max\_total\_votes,

min(median\_rating) as Min\_median\_rating,max(median\_rating) as Max\_median\_rating from ratings



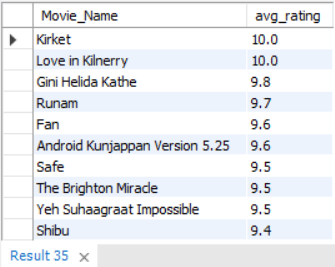
1. Which are the top 10 movies based on their average rating?

select m.title as Movie\_Name,r.avg\_rating

from ratings r inner join movie m

on r.movie\_id = m.id

order by r.avg\_rating desc limit 10;



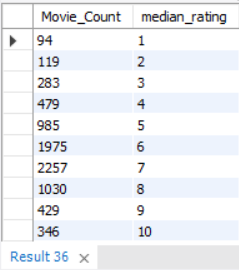
1. Summarize the ratings table by grouping movies based on their median ratings.

select count(m.title) as Movie\_Count,r.median\_rating

from ratings r inner join movie m

on r.movie\_id = m.id

group by r.median\_rating order by r.median\_rating asc;



1. How many movies, released in March 2017 in the USA within a specific genre, had more than 1,000 votes?

select m.title as

Movie\_Name,m.date\_published,m.year,m.country,r.total\_votes,group\_concat(g.genre) as Genre

from movie m inner join genre g

on m.id = g.movie\_id

inner join ratings r

on m.id = r.movie\_id

where month(m.date\_published) = 3

and year = 2017

and r.total\_votes > 1000 and m.country like '%USA%' group by

m.title,m.date\_published,m.year,m.country,r.total\_votes order by r.total\_votes asc;



1. Find movies from each genre that begin with the word “The” and have an average rating greater than 8.

select m.title,group\_concat(g.genre) as Genre,r.avg\_rating

from movie m inner join genre g

on m.id = g.movie\_id inner join ratings r

on m.id = r.movie\_id

where r.avg\_rating > 8

group by m.title,r.avg\_rating having m.title like 'the%';



1. Of the movies released between April 1, 2018, and April 1, 2019, how many received a median rating of 8?

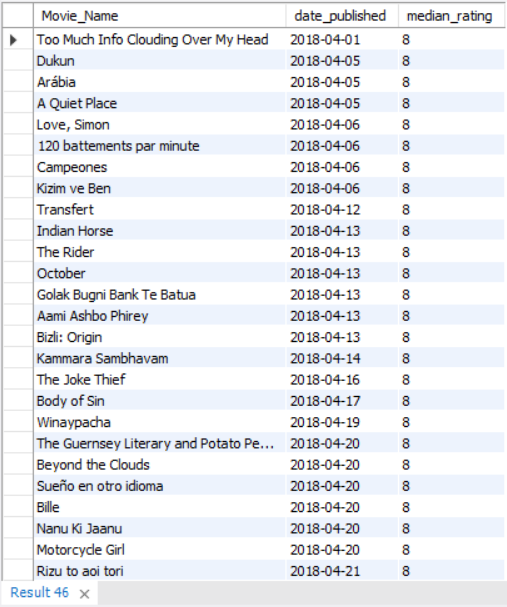
select m.title as Movie\_Name,m.date\_published,r.median\_rating

from movie m inner join ratings r

on m.id = r.movie\_id

where m.date\_published between '2018-04-01' and '2019-04-01'

and r.median\_rating = 8 order by m.date\_published asc;



1. Do German movies receive more votes on average than Italian movies?

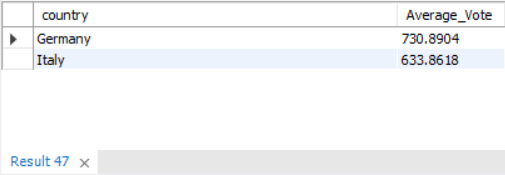
select m.country,avg(r.total\_votes) as Average\_Vote

from movie m inner join ratings r

on m.id = r.movie\_id

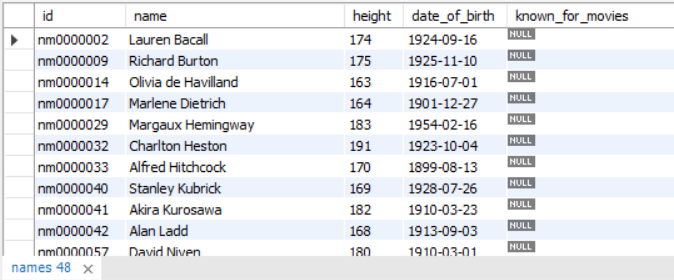
where m.country = "Germany" or m.country = "Italy"

group by m.country;



1. Identify the columns in the names table that contain null values.

select \* from names where id is null or name is null or height is null or date\_of\_birth is null or known\_for\_movies is null;



1. Who are the top two actors whose movies have a median rating of 8 or higher?

select rm.name\_id,n.name,rm.category,m.title as Movie\_Name,r.median\_rating

from role\_mapping rm

inner join movie m

on rm.movie\_id = m.id

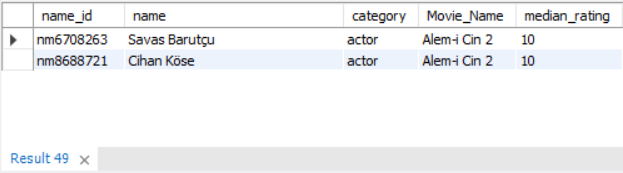
inner join ratings r

on rm.movie\_id = r.movie\_id

inner join names n

on rm.name\_id = n.id

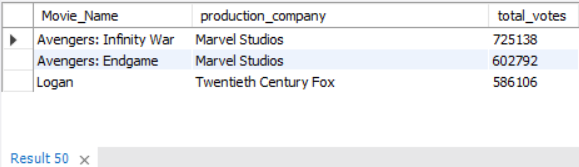
where r.median\_rating>=8 order by r.median\_rating desc limit 2;



1. Which are the top three production companies based on the total number of votes their movies received?

select m.title as Movie\_Name,m.production\_company,r.total\_votes

from movie m inner join ratings r

on m.id = r.movie\_id order by r.total\_votes desc limit 3;

1. How many directors have worked on more than three movies?

select count(dm.name\_id) as Movie\_count, dm.name\_id,n.name as Director\_Name

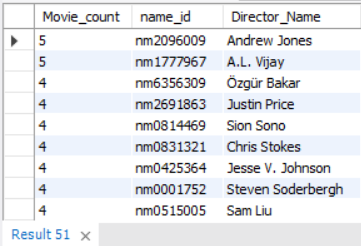
from director\_mapping dm inner join names n

on dm.name\_id = n.id

inner join movie m

on dm.movie\_id = m.id

group by dm.name\_id having Movie\_count>3 order by Movie\_count desc;



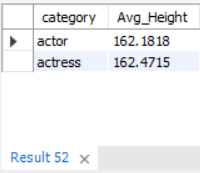
1. Calculate the average height of actors and actresses separately.

select rm.category,avg(n.height) as Avg\_Height

from role\_mapping rm inner join names n

on rm.name\_id = n.id

group by rm.category;



1. List the 10 oldest movies in the dataset along with their title, country, and director.

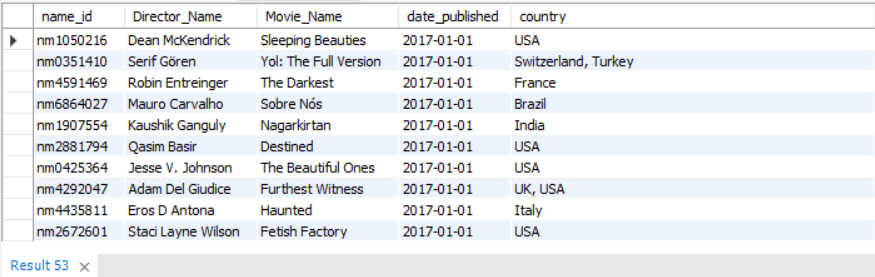
select dm.name\_id,n.name as Director\_Name,m.title as Movie\_Name,m.date\_published,m.country

from director\_mapping dm inner join names n

on dm.name\_id = n.id

inner join movie m

on dm.movie\_id = m.id order by m.date\_published asc limit 10;



1. List the top 5 movies with the highest total votes, along with their genres.

select m.title,r.total\_votes,group\_concat(distinct g.genre) as Genre

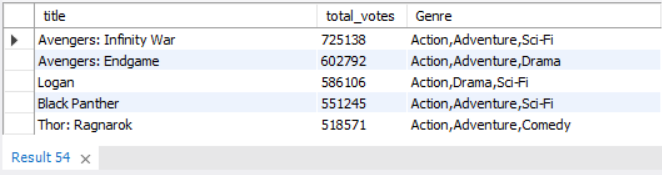
from movie m inner join ratings r

on m.id = r.movie\_id

inner join genre g

on m.id = g.movie\_id

group by m.title,r.total\_votes

order by r.total\_votes desc limit 5;

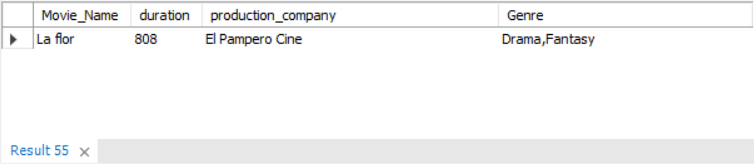
1. Identify the movie with the longest duration, along with its genre and production company.

select m.title as Movie\_Name,m.duration,m.production\_company,group\_concat(g.genre) as Genre

from movie m inner join genre g

on m.id = g.movie\_id

group by m.title,m.duration,m.production\_company

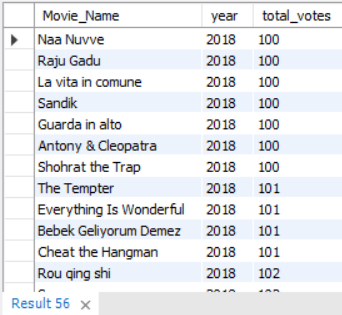
having m.duration = (select max(m.duration) from movie m);

1. Determine the total number of votes for each movie released in 2018.

select m.title as Movie\_Name,m.year,r.total\_votes

from movie m inner join ratings r

on m.id = r.movie\_id where m.year = 2018 order by r.total\_votes asc;



1. What is the most common language in which movies were produced?

select

Languages,

COUNT(\*) as Movie\_count

from (

select

TRIM(SUBSTRING\_INDEX(SUBSTRING\_INDEX(languages, ',', n.n), ',', -1)) as Languages

from movie

inner join (

SELECT 1 AS n UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT 5

UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION ALL SELECT 9 UNION ALL SELECT 10

) n on CHAR\_LENGTH(languages) - CHAR\_LENGTH(replace(languages, ',', '')) >= n.n - 1

) sub

group by languages

order by movie\_count desc;

