Distinct Keyword

select distinct industry from movies;

Like –

* Select all movies where in title Thor letter comes basically 3 movies of Thor

select \* from movies where title like "%Thor%"; (Any letters before or after Thor)

* Gives same answers below 2 queries

select \* from movies where imdb\_rating>=6 and imdb\_rating <=8;

select \* from movies where imdb\_rating between 6 AND 8; -- including 6 and 8

use moviesdb;

select title from movies;

select distinct industry from movies;

select \* from movies where title like "%Thor%";

select \* from movies where imdb\_rating>=6 and imdb\_rating <=8;

select \* from movies where imdb\_rating between 6 AND 8; -- including 6 and 8

select \* from movies where release\_year= 2022 or release\_year= 2019 or release\_year= 2018;

select \* from movies where release\_year IN (2022,2018,2019); -- the , represents OR Condition

select \* from movies where studio IN ("Marvel Studios", "Zee Studios"); -- IN keyword works for strings also

select \* from movies where imdb\_rating is NULL; -- How to fetch null values

select \* from movies where imdb\_rating is NOT NULL; -- How to fetch null values

select \* from movies where industry="bollywood" order by imdb\_rating DESC; -- order by descending value

select \* from movies where industry="hollywood" order by imdb\_rating DESC limit 5; -- order by descending value and only top 2

select \* from movies where industry="hollywood" order by imdb\_rating DESC limit 5 offset 2;

-- order by descending value and only top 5 and showing from the third value so skip first 2 value and then show all

-- It will showing 5 values but the highest rating top 2 values will cut and then it will be measured.

select \* from movies order by release\_year desc;

select title, release\_year from movies where title like '%thor%' order by release\_year asc;

-- select all thor movies by their release year

select \* from movies where studio != 'Marvel studios';

-- select all movies that are not from marvel studios

select AVG(imdb\_rating) as Average\_Rating from movies where studio = "Marvel Studios"; -- returns 7.50000 ,average of all marvel movies

select ROUND(AVG(imdb\_rating)) from movies where studio = "Marvel Studios"; -- returns 8

select ROUND(AVG(imdb\_rating),2) as Average\_Rating from movies where studio = "Marvel Studios"; -- returns 7.50 ,as keyword for rename column name

select count(\*) from movies where industry = "Hollywood";

select industry,count(\*) as cnt from movies group by industry; -- make a list of all different industry type movies

select industry,count(\*) as cnt, avg(imdb\_rating) as avg\_rating from movies group by industry;

-- https://www.w3schools.com/mysql/func\_mysql\_max.asp So much to remember so go through diff funcs from this page

select count(\*) from movies where release\_year between 2015 and 2022; -- count movies release between 2015 to 2022

select release\_year,count(title) as movie\_count from movies group by release\_year order by release\_year desc;

-- print a year and how many movies were released in that year starting with latest year

select release\_year,count(\*) as cnt from movies where cnt>2 group by release\_year ;

-- above will not work when you want further use aggregate function to calculation

select release\_year,count(\*) as cnt from movies group by release\_year having cnt>2;

-- sequence in which SQL engine will perform the actions

-- FROM -> WHERE -> GROUP BY -> HAVING -> ORDER BY

select year(curdate());

-- calculate age from birth\_year of actors

select \*,year(curdate())-birth\_year as actor\_age from actors; -- Similar how you make mew calculated column of Age in Excel or Power BI

select \*,revenue-budget as profit from financials;

select revenue from financials where movie\_id = (select movie\_id from movies where title like "3 Idiots");

select \*,

IF(currency='USD',revenue\*77,revenue) as revenue\_inr

from financials;

-- convert USD revenue into INR (1 usd = 77 inr)

select distinct unit from financials; -- it will gives billions, million, thousands <- these 3 records.

-- Show Revenue in Million in new column

select \*,

case

when unit="thousands" then revenue/1000

when unit="billions" then revenue\*1000

else revenue

end as revenue\_million

from financials;

select \* from actors;