-On Boarding :

select name from people

SELECT 'SQL'

AS result;

-selecting single column :

select title

from films;

select release\_year

from films;

select name

from people;

-selecting multiple column

select title

from films

SELECT title, release\_year

FROM films;

SELECT title, release\_year, country

FROM films;

SELECT \*

FROM films;

-Select Distinct : find unique value

select distinct country

from films;

select distinct certification

from films

select distinct role

from roles

-Select count : count number of rows of table

select COUNT(\*)

from reviews;

-practice with count :

-Count the number of (non-missing) birth dates in the people table

SELECT COUNT(birthdate)

FROM people;

-Count the number of unique birth dates in the people table.

SELECT COUNT(distinct birthdate)

FROM people;

**#FILTERING ROWS**

-filtering result

For example, you can filter text records such as title. The following code returns all films with the title 'Metropolis':

**SELECT title**

**FROM films**

**WHERE title = 'Metropolis';**

Notice that the WHERE clause always comes after the FROM statement!

-simple filtering for numeric values :

#Get all details for all films released in 2016.

select \*

from films

where release\_year =2016

#Get the number of films released before 2000

select count(\*)

from films

where release\_year < 2000

#Get the title and release year of films released after 2000

select title, release\_year

from films

where release\_year > 2000

-Simple filtering of text :

#Get all details for all French language films

select \*

from films

where language = 'French'

#Get the name and birth date of the person born on November 11th, 1974. Remember to use ISO date format ('1974-11-11')!

select name, birthdate

from people

where birthdate = '1974-11-11'

#Get the number of Hindi language films.

select count(\*)

from films

where language = 'Hindi'

#Get all details for all films with an R certification

select \*

from films

where certification = 'R'

-Where And :

#Get the title and release year for all Spanish language films released before 2000

select title, release\_year

from films

where language ='Spanish' and release\_year <2000

#Get all details for Spanish language films released after 2000

select \*

from films

where language = 'Spanish' and release\_year > 2000

#Get all details for Spanish language films released after 2000, but before 2010

select \*

from films

where language = 'Spanish' and release\_year > 2000 and release\_year < 2010

-Where and OR

When combining AND and OR, be sure to enclose the individual clauses in parentheses, like so:

**SELECT title**

**FROM films**

**WHERE (release\_year = 1994 OR release\_year = 1995)**

**AND (certification = 'PG' OR certification = 'R');**

**-**get the title and release year of films released in the 90s which were in French or Spanish and which took in more than $2M gross :

SELECT \*

from films

where (release\_year >= 1990 and release\_year < 2000) AND

(language = 'French' or language ='Spanish') AND

(gross > 2000000)

-Between : Filter Values in specific range

**SELECT title**

**FROM films**

**WHERE release\_year**

**BETWEEN 1994 AND 2000;**

#get the title and release year of all Spanish language films released between 1990 and 2000 (inclusive) with budgets over $100 million

select title, release\_year

from films

where (release\_year between 1990 and 2000) and

(budget > 100000000) and

(language = 'Spanish' or language = 'French')

-Where In : the function is like OR operator

**SELECT name**

**FROM kids**

**WHERE age IN (2, 4, 6, 8, 10);**

**#**Get the title and release year of all films released in 1990 or 2000 that were longer than two hours. Remember, duration is in minutes

select title, release\_year

from films

where release\_year IN (1990, 2000) and duration >120

**#**Get the title and language of all films which were in English, Spanish, or French

select title, language

from films

where language in ('English', 'Spanish', 'French')

**#**Get the title and certification of all films with an NC-17 or R certification

select title, certification

from films

where certification in ('NC-17', 'R')

-Introduction to Null and is Null

-to count the number of missing birth dates :

**\*SELECT COUNT (\*)**

**\*FROM people**

**\*WHERE birthdate IS NULL**

**-select name of people whose birth dates are not null :**

**\*SELECT name**

**\*FROM people**

**\*WHERE birthdate IS NOT NULL;**

-Get the people name where still alive :

select name

from people

where deathdate is null

-Get the title of film which doesn’t have budget :

select title

from films

where budget is null

-Get the number of film where doesn’t have language :

select count (\*)

from films

where language is null

-LIKE AND NOT LIKE :

\*select name

\*from companies

\*where name like ‘DataC\_mp’;

-Get the name of people whose names begin with ‘B’

select name

from people

where name like 'B%'

-Get the name of people whose names have r as the second later

select name

from people

where name LIKE '\_r%'

-Get the name of people whose names don’t start with A

select name

from people

where name NOT LIKE 'A%'

-Aggregate Function

**\*select avg (budget)**

**\*from films;**

-Combining aggregate functions with where

**\*select sum (budget)**

**\*from films**

**\*where release\_year >= 2020**

-A Note on arithmetic

**-if the integer divided by integer will be result to integer. For example 4/3 will be back to 1**

**-if want more specific result, use comma in the back. For example 4.0/3.0 will be back to 1.3**

-it’s as simple as aliasing

\*select max(budget) as max\_budget

max(duration) as max\_duration

\*from films;

-get the title and net\_profit :

select title, gross - budget AS net\_profit

from films

-get the title and duration hours from films :

select title, duration/60.0 as duration\_hours

from films

-get average duration in hours

select avg (duration)/60.0 as avg\_duration\_hours

from films

-Even more aliasing

-get percentage of dead

-- get the count(deathdate) and multiply by 100.0

-- then divide by count(\*)

select count (deathdate) \* 100.0 / count (\*) as percentage\_dead

from people

-get the number of years between first and last films

select max (release\_year) - min (release\_year) as difference

from films

-get the number between first and last film in decade :

select (max (release\_year) - min (release\_year))/10.0 as number\_of\_decades

from films