**SQL CODE FOR CORONA VIRUS PROBLEM DATASET**

Q1. Write a code to check NULL values.

Ans-There is no null values in the dataset. Here is the code to check NULL values.

SELECT \* FROM CORONA **WHERE** COLUMN\_NAME **IS NULL**;

Eg- SELECT \* FROM CORONA WHERE Recovered IS NULL;

Q2. If NULL values are present, update them with zeros for all columns.

Ans. There is no null values present. Here is the code to update them with zero

**UPDATE** corona

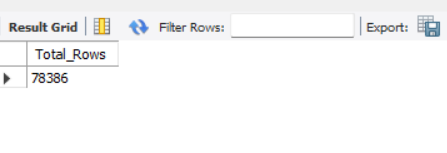
**SET** column1 = COALESCE(column1, 0),

column2 = COALESCE(column2, 0),

column3 = COALESCE(column3, 0),

Q3. check total number of rows.

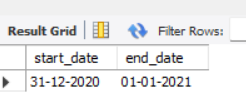
* **select Count(\*)** as Total\_Rows from corona;



Q4. Check what is start\_date and end\_date.

> SELECT **Max(date)** AS start\_date, **Min(date)** AS end\_date

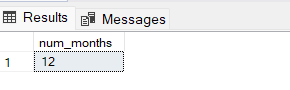
FROM corona;



Q5. Number of month present in dataset

SELECT COUNT(DISTINCT Month(date)) AS num\_months

FROM Corona;



Q6. Find monthly average for confirmed, deaths, recovered

SELECT

distinct(datename(Month,date)) AS month\_year,

AVG(confirmed) AS avg\_confirmed,

AVG(deaths) AS avg\_deaths,

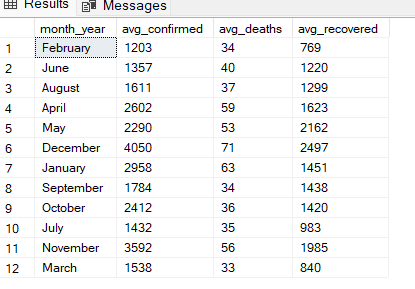
AVG(recovered) AS avg\_recovered

FROM

Corona

GROUP BY

datename(Month,date);



Q8. Find minimum values for confirmed, deaths, recovered per year

SELECT

YEAR(date) AS year,

MIN(confirmed) AS min\_confirmed,

MIN(deaths) AS min\_deaths,

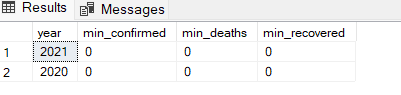
MIN(recovered) AS min\_recovered

FROM

corona

GROUP BY

YEAR(date);



Q9. Find maximum values of confirmed, deaths, recovered per year

SELECT

YEAR(date) AS year,

Max(confirmed) AS min\_confirmed,

Max(deaths) AS min\_deaths,

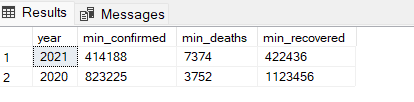
Max(recovered) AS min\_recovered

FROM

corona

GROUP BY

YEAR(date);



Q10. The total number of case of confirmed, deaths, recovered each month

SELECT

datename(month,date) AS month\_name,

Sum(confirmed) AS Total\_confirmed,

Sum(deaths) AS Total\_deaths,

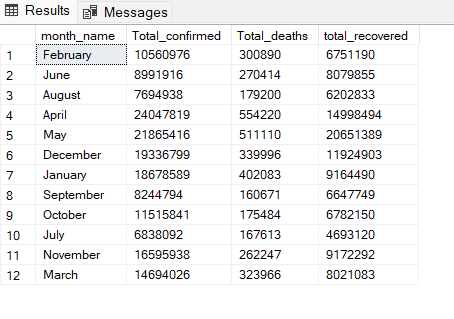
Sum(recovered) AS total\_recovered

FROM

corona

GROUP BY

datename(month,date) ;



Q11. Check how corona virus spread out with respect to confirmed case

(Eg.: total confirmed cases, their average, variance & STDEV )

SELECT SUM(confirmed) AS total\_confirmed\_cases

FROM corona;

-- Average confirmed cases

SELECT AVG(confirmed) AS average\_confirmed\_cases

FROM corona;

-- Variance of confirmed cases

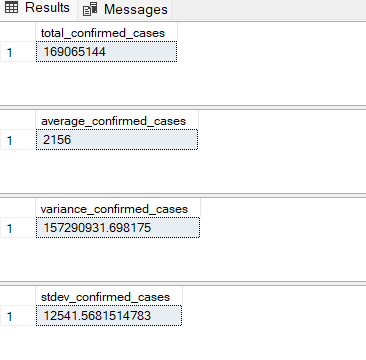
SELECT VAR(confirmed) AS variance\_confirmed\_cases

FROM corona;

-- Standard deviation of confirmed cases

SELECT STDEV(confirmed) AS stdev\_confirmed\_cases

FROM corona;



Q12. Check how corona virus spread out with respect to death case per month

(Eg.: total confirmed cases, their average, variance & STDEV )

SELECT SUM(deaths) AS total\_deaths\_cases

FROM corona;

-- Average confirmed cases

SELECT AVG(deaths) AS average\_deaths\_cases

FROM corona;

-- Variance of confirmed cases

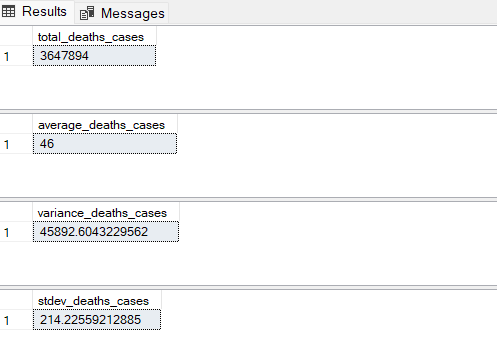
SELECT VAR(deaths) AS variance\_deaths\_cases

FROM corona;

-- Standard deviation of confirmed cases

SELECT STDEV(deaths) AS stdev\_deaths\_cases

FROM corona;



Q13. Check how corona virus spread out with respect to recovered case

-- (Eg.: total confirmed cases, their average, variance & STDEV )

SELECT SUM(Recovered) AS total\_recovered\_cases

FROM corona;

-- Average confirmed cases

SELECT AVG(Recovered) AS average\_recovered\_cases

FROM corona;

-- Variance of confirmed cases

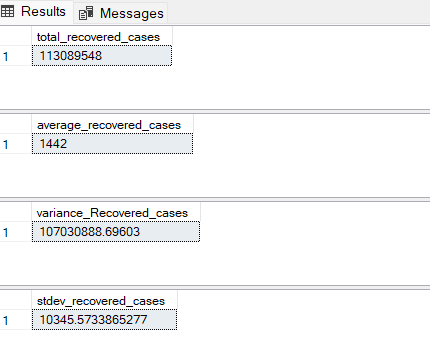
SELECT VAR(Recovered) AS variance\_Recovered\_cases

FROM corona;

-- Standard deviation of confirmed cases

SELECT STDEV(Recovered) AS stdev\_recovered\_cases

FROM corona;



Q14. Find Country having highest number of the Confirmed case

SELECT Country\_Region

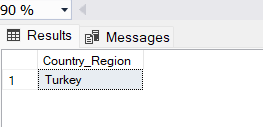
FROM corona

WHERE confirmed = (

SELECT MAX(confirmed)

FROM corona

);



Q15. Find Country having lowest number of the death case

SELECT distinct Country\_Region

FROM corona

WHERE deaths = (

SELECT min(deaths)

FROM corona

);



Q16. Find top 5 countries having highest recovered case

ELECT TOP 5 Country\_Region, SUM(recovered) AS total\_recovered\_cases

FROM corona

GROUP BY Country\_Region

ORDER BY total\_recovered\_cases DESC;

