

CORONA VIRUS ANALYSIS

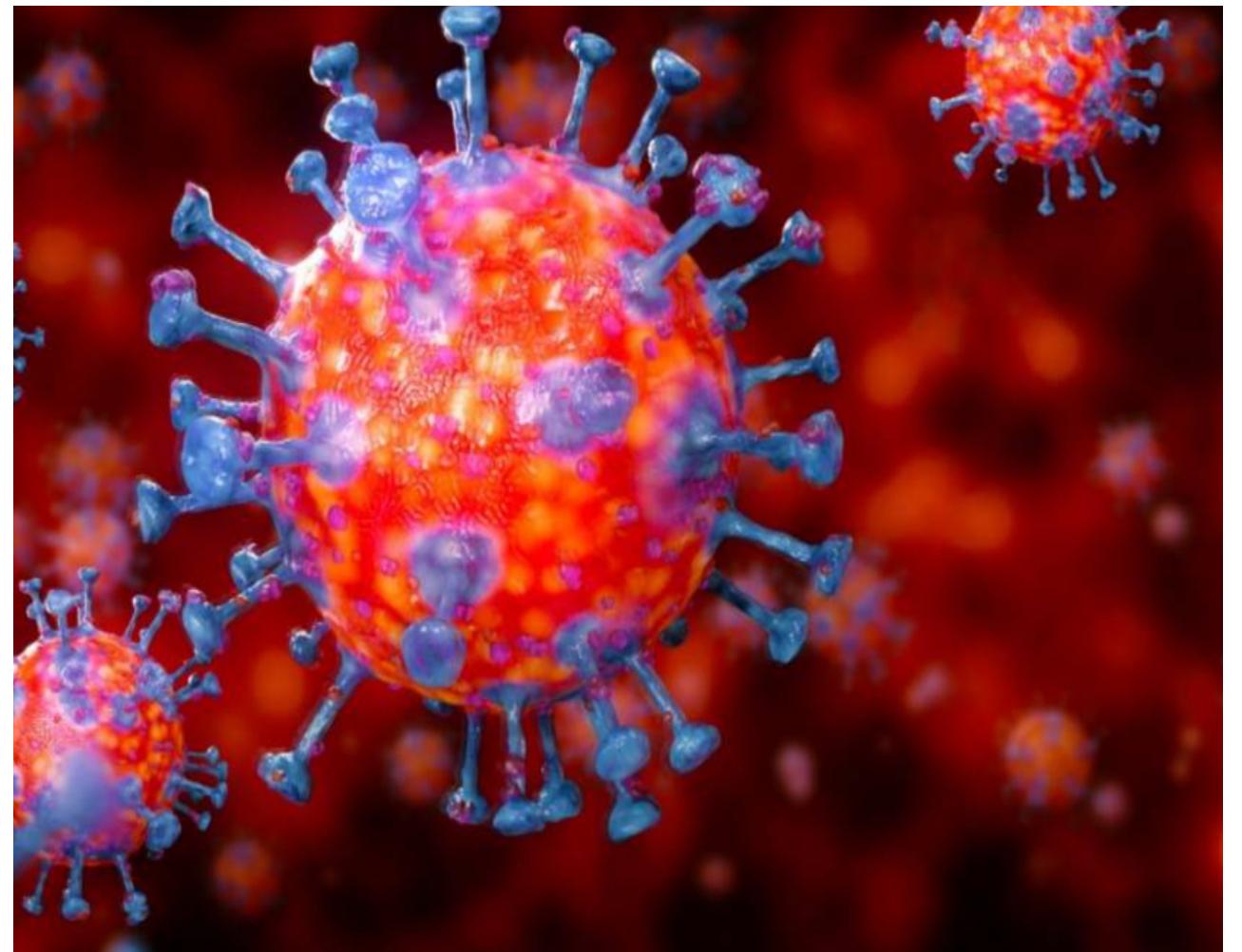
PRESENTING BY
SHRIYA K



Corona Virus Analysis

Overview:

The CORONA VIRUS pandemic has had a significant impact on public health and has created an urgent need for data-driven insights to understand the spread of the virus. As a data analyst, you have been tasked with analyzing a CORONA VIRUS dataset to derive meaningful insights and present your findings.



Dataset:

Description of each column in dataset:

- Province: Geographic subdivision within a country/region.
- Country/Region: Geographic entity where data is recorded.
- Latitude: North-south position on Earth's surface.
- Longitude: East-west position on Earth's surface.
- Date: Recorded date of CORONA VIRUS data.
- Confirmed: Number of diagnosed CORONA VIRUS cases.
- Deaths: Number of CORONA VIRUS related deaths.
- Recovered: Number of recovered CORONA VIRUS cases.



CORONA VIRUS ANALYSIS DATABASE

Province	Country/Region	Latitude	Longitude	Date	Confirmed	Deaths	Recovered
Afghanistan	Afghanistan	33.93911	67.709953	22-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	23-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	24-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	25-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	26-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	27-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	28-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	29-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	30-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	31-01-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	01-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	02-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	03-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	04-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	05-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	06-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	07-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	08-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	09-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	10-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	11-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	12-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	13-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	14-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	15-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	16-02-2020	0	0	0
Afghanistan	Afghanistan	33.93911	67.709953	17-02-2020	0	0	0



TOOL USED FOR ANALYSIS

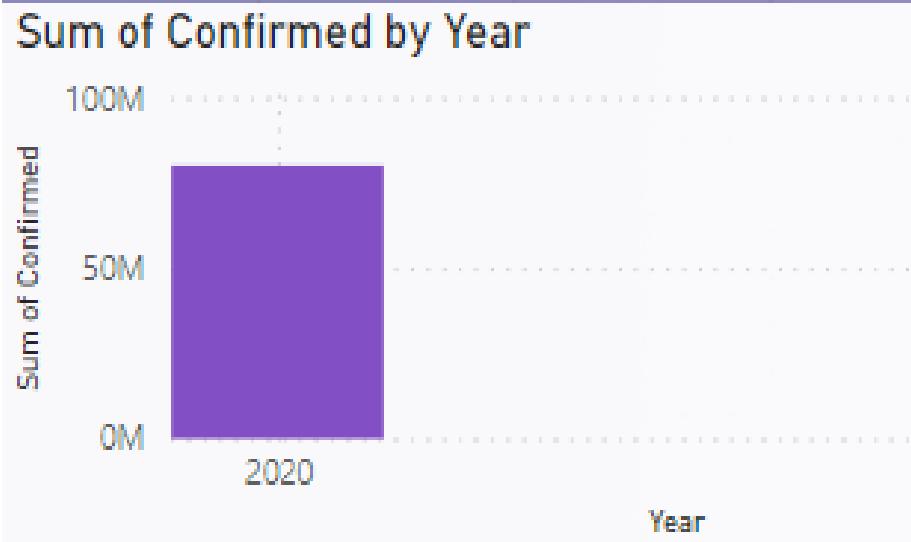
Microsoft SQL Server Management studio

SQL Server Management Studio (SSMS) is an integrated environment for managing any SQL infrastructure. Use SSMS to access, configure, manage, administer, and develop all components of SQL Server, Azure SQL Database, Azure , SQL Server on Azure VM.

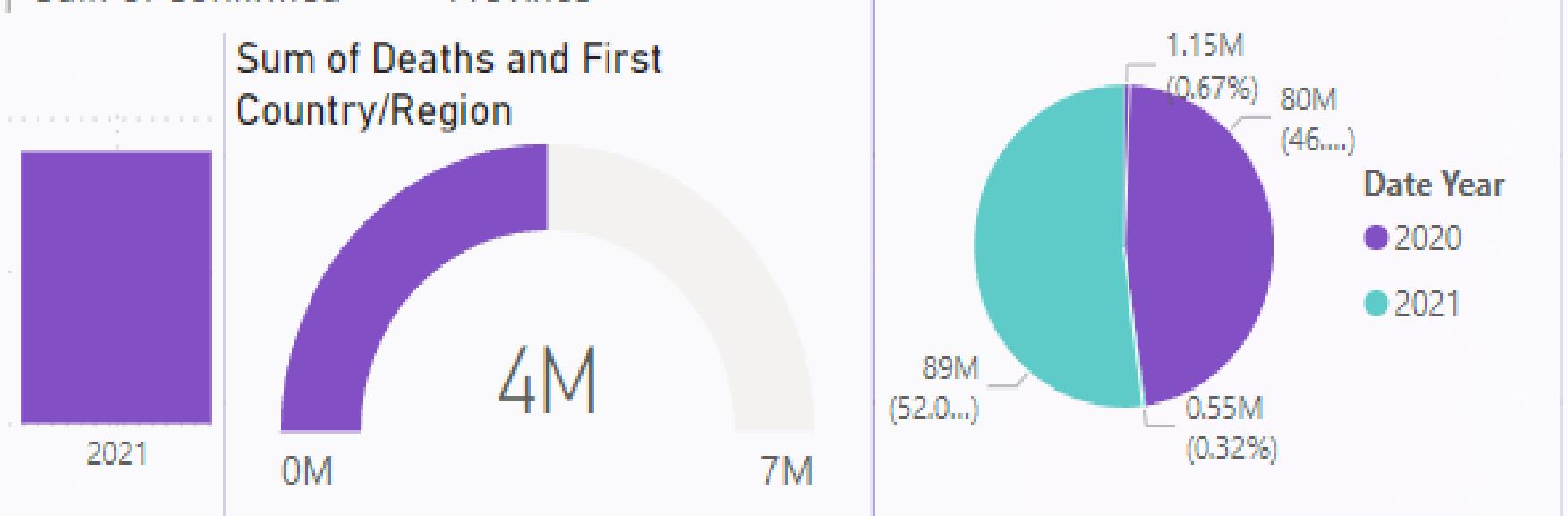
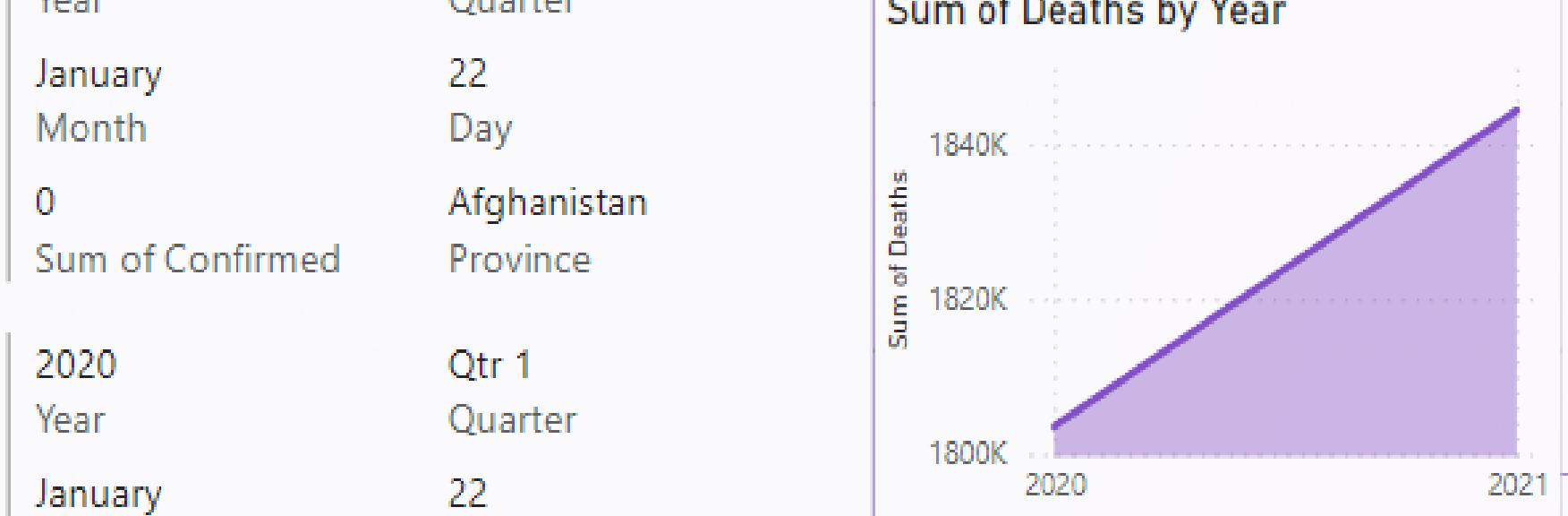


Corona Virus Analysis Dashboard

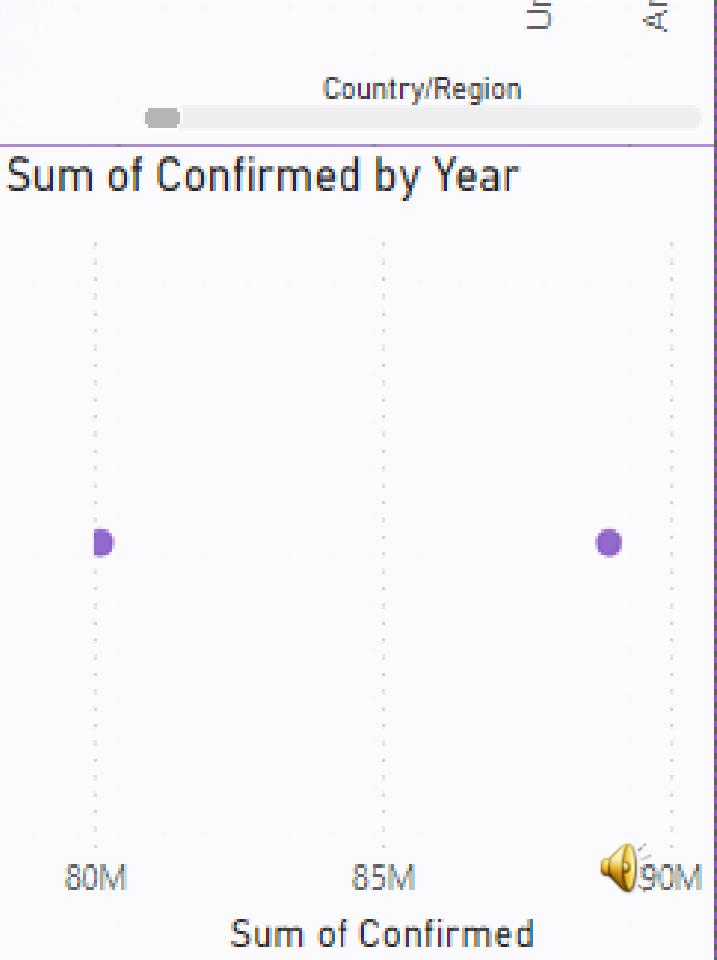
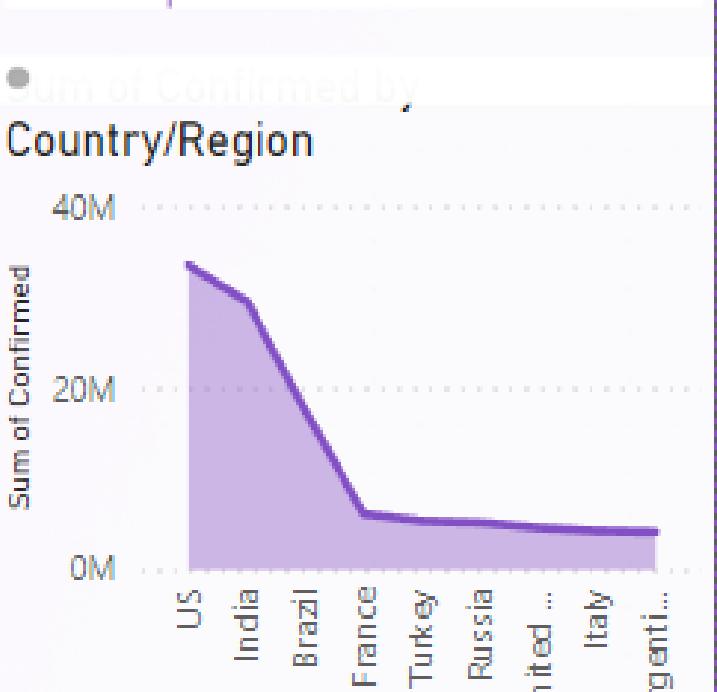
	Year	Quarter	Month	Day	Sum of Deaths	Province
2020	Qtr 1	January	22	0	Afghanistan	
2020	Qtr 1	January	22	0	Algeria	
2020	Qtr 1	January	22	0	Argentina	
2020	Qtr 1	January	22	0	Australia	
2020	Qtr 1	January	22	0	Austria	
2020	Qtr 1	January	22	0	Bangladesh	
2020	Qtr 1	January	22	0	Barbados	
2020	Qtr 1	January	22	0	Beijing	
2020	Qtr 1	January	22	0	Belarus	
2020	Qtr 1	January	22	0	Belgium	
2020	Qtr 1	January	22	0	Bermuda	
2020	Qtr 1	January	22	0	Bhutan	
2020	Qtr 1	January	22	0	Bolivia	
2020	Qtr 1	January	22	0	Bosnia	
Total					3647894	



Year	Afghanistan	Algeria	Argentina	Australia
2020	51526	99610	1625514	
2021	38335	33778	2498676	
Total	89861	133388	4124190	



Year	Afghanistan	Algeria	Argentina
2020	2191	2756	43245
2021	1336	815	42098
Total	3527	3571	85343



Key Questions :

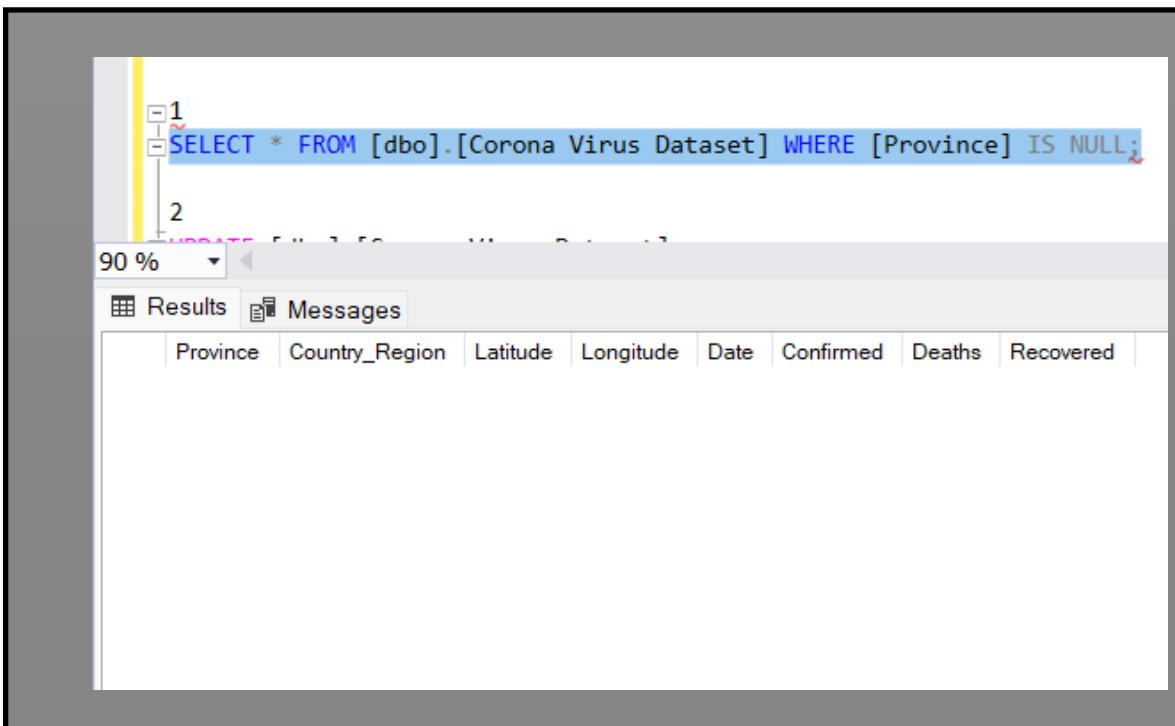
1. Write a code to check NULL values
2. If NULL values are present, update them with zeros for all columns.
3. check total number of rows
4. Check what is start_date and end_date .
5. Number of month present in dataset
6. Find monthly average for confirmed, deaths, recovered
7. Find most frequent value for confirmed, deaths, recovered each month
8. Find minimum values for confirmed, deaths, recovered per year
9. Find maximum values of confirmed, deaths, recovered per year
10. The total number of case of confirmed, deaths, recovered each month
11. Check how corona virus spread out with respect to confirmed case.
(Eg.: total confirmed cases, their average, variance & STDEV)
12. Check how corona virus spread out with respect to death case per month
(Eg.: total confirmed cases, their average, variance & STDEV)
- Q13. Check how corona virus spread out with respect to recovered case
(Eg.: total confirmed cases, their average, variance & STDEV)
14. Find Country having highest number of the Confirmed case.
15. Find Country having lowest number of the death case
16. Find top 5 countries having highest recovered case



Corona Virus Analysis

1. Write code to check null values.

```
SELECT * FROM [dbo].[Corona Virus Dataset] WHERE [Province] IS NULL;
```



The screenshot shows a SQL query window in SSMS. The query is:

```
1
2 SELECT * FROM [dbo].[Corona Virus Dataset] WHERE [Province] IS NULL;
```

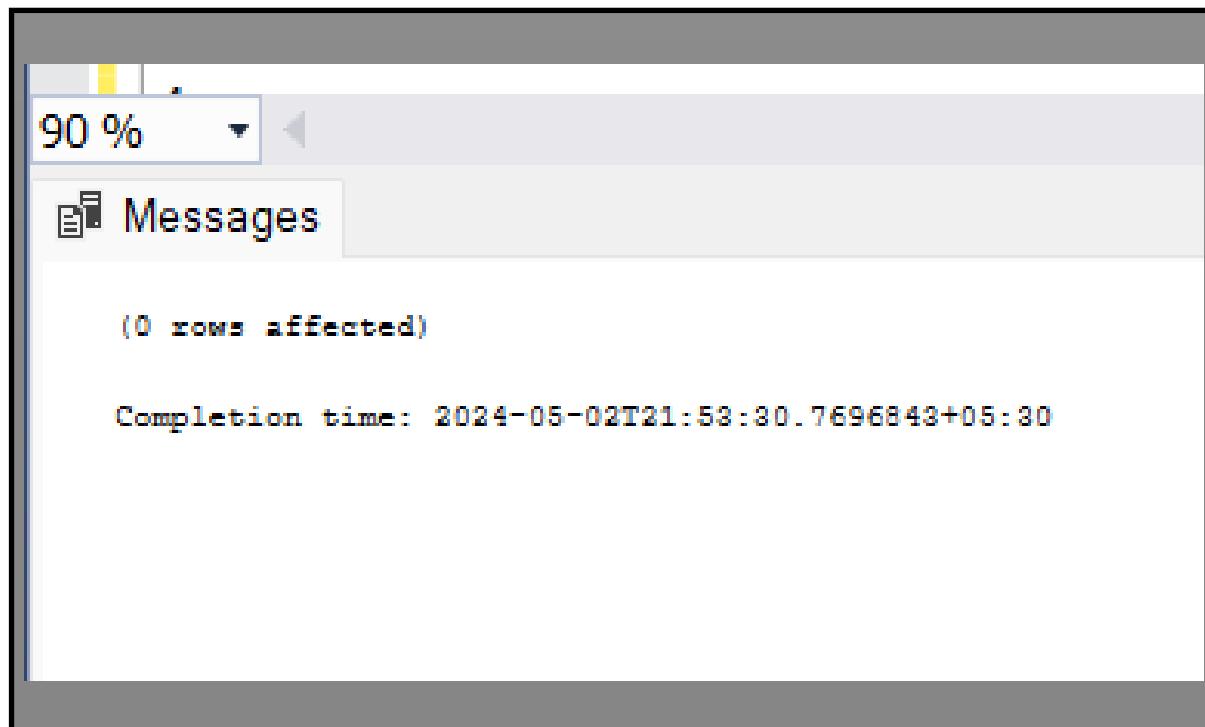
The results pane shows a header row with columns: Province, Country_Region, Latitude, Longitude, Date, Confirmed, Deaths, Recovered. There are no data rows present.

> There are no null values in given database



2. If null values are present, update them to zeros for all columns

```
UPDATE [dbo].[Corona Virus Dataset]
SET Confirmed = ISNULL(Confirmed, 0),
    Deaths = ISNULL(Deaths, 0)
WHERE Confirmed IS NULL OR Deaths IS
NULL;
```

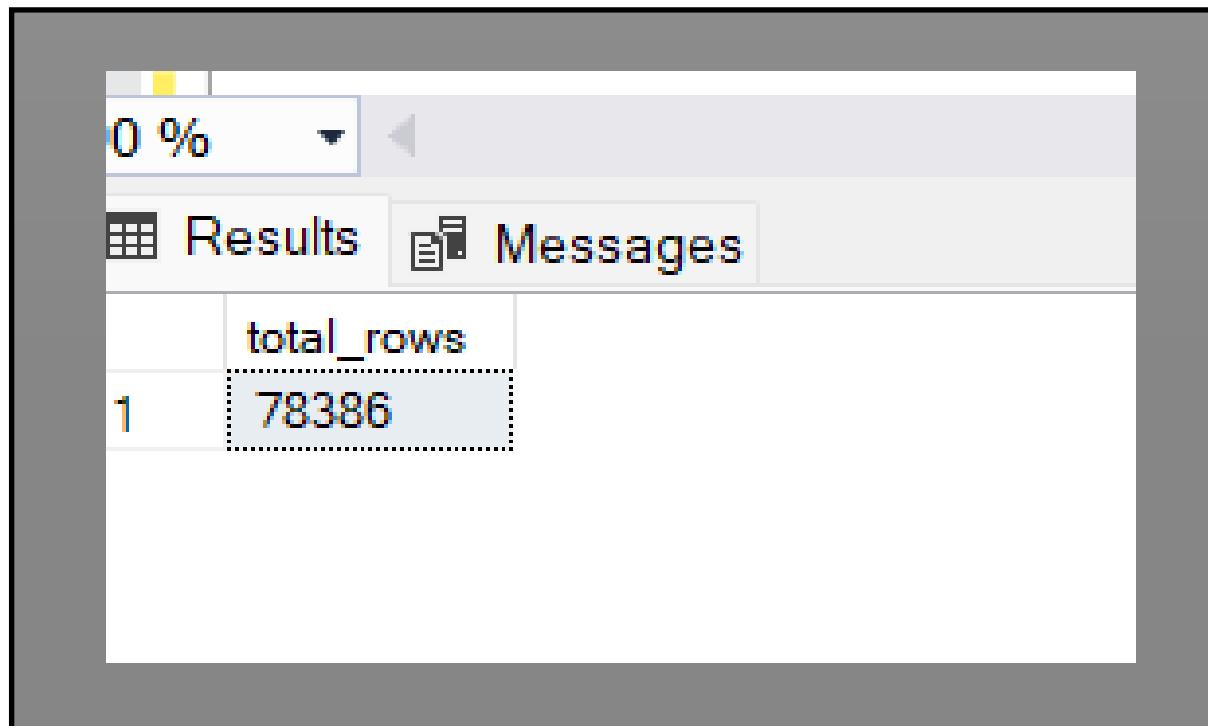


> There are no null values in given database



3. Check total number of rows.

```
SELECT COUNT(*) AS total_rows FROM [dbo].[Corona Virus Dataset];
```



A screenshot of a SQL query results window. The window has a progress bar at the top showing "0 %". Below it are two tabs: "Results" and "Messages", with "Results" selected. The results table contains one row with the column name "total_rows" and the value "78386".

	total_rows
1	78386



4. Check what is start_date and end_date.

```
SELECT  
    MIN(Date) AS start_date,  
    MAX(Date) AS end_date  
FROM [dbo].[Corona Virus Dataset]
```

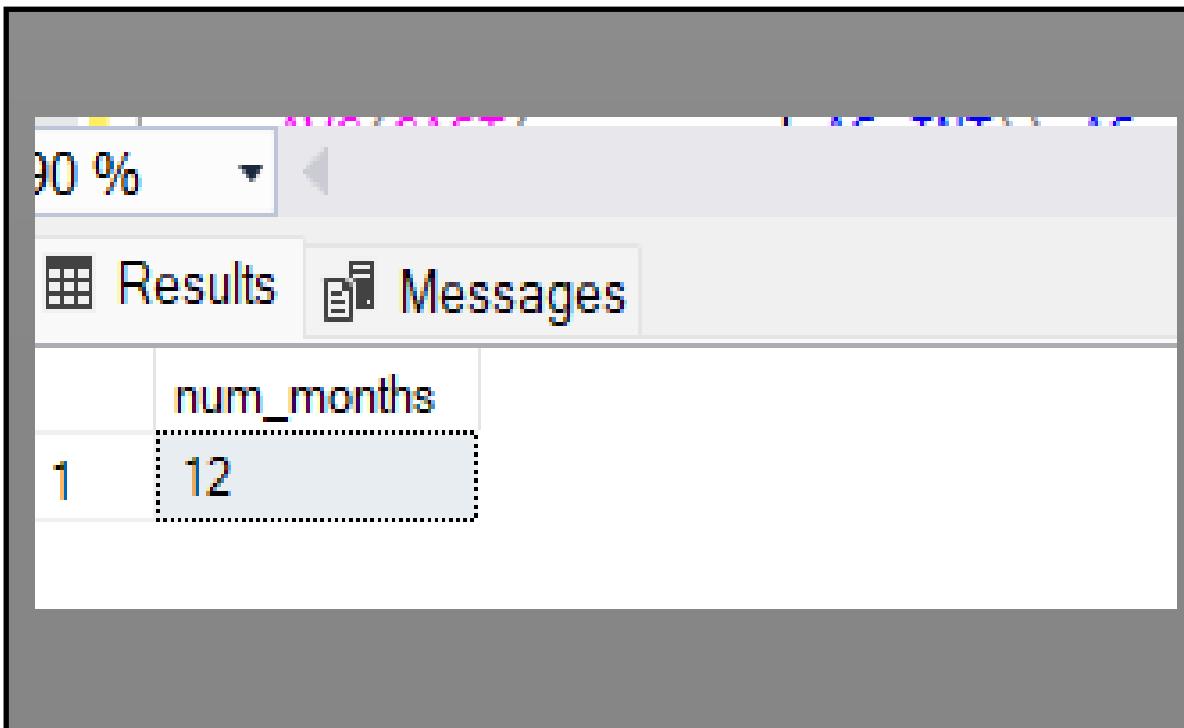
The screenshot shows the SSMS interface with the 'Results' tab selected. The query has been executed, and the results are displayed in a table format:

start_date	end_date
2020-01-22 00:00:00.0000000	2021-06-13 00:00:00.0000000



5. Number of month present in dataset.

```
SELECT COUNT(DISTINCT MONTH(Date)) AS num_months  
FROM [dbo].[Corona Virus Dataset];
```



The screenshot shows a Windows-style application window titled 'Results' with a zoom level of 90%. It displays a single row of data from a query. The column is labeled 'num_months' and contains the value '12'. There is also a 'Messages' tab at the top of the window.

	num_months
1	12



6. Find monthly average for confirmed,deaths,recovered.

```
SELECT
    FORMAT(Date, 'yyyy-MM') AS month,
    AVG(CAST(Confirmed AS INT)) AS
monthly_average_confirmed,
    AVG(CAST(deaths AS INT)) AS
monthly_average_deaths,
    AVG(CAST(recovered AS INT)) AS
monthly_avg_recovered
FROM
    [dbo].[Corona Virus Dataset]
    GROUP BY
        FORMAT(Date, 'yyyy-MM')
ORDER BY
    month;
```

	month	monthly_average_confirmed	monthly_average_deaths	monthly_avg_recovered
1	2020-01	4	0	0
2	2020-02	15	0	7
3	2020-03	161	8	27
4	2020-04	505	41	171
5	2020-05	574	30	318
6	2020-06	859	29	548
7	2020-07	1432	35	983
8	2020-08	1611	37	1299
9	2020-09	1784	34	1438
10	2020-10	2412	36	1420
11	2020-11	3592	56	1985



7. Find most frequent value for confirmed,deaths,recovered each month.

```
WITH MonthlyCounts AS (
    SELECT
        YEAR(Date) AS year,
        MONTH(Date) AS month,
        confirmed,
        deaths,
        recovered,
        ROW_NUMBER() OVER (PARTITION BY YEAR(Date), MONTH(Date), confirmed, deaths, recovered ORDER BY COUNT(*) DESC) AS rn
    FROM [dbo].[Corona Virus Dataset]
    GROUP BY YEAR(Date), MONTH(Date), confirmed, deaths, recovered
)
SELECT
    year,
    month,
    confirmed,
    deaths,
    recovered
FROM MonthlyCounts
WHERE rn = 1;
```



OUTPUT 7:

The screenshot shows a SQL Server Management Studio (SSMS) interface with a results grid. The title bar displays "90 %". The results tab is selected, showing a table with the following data:

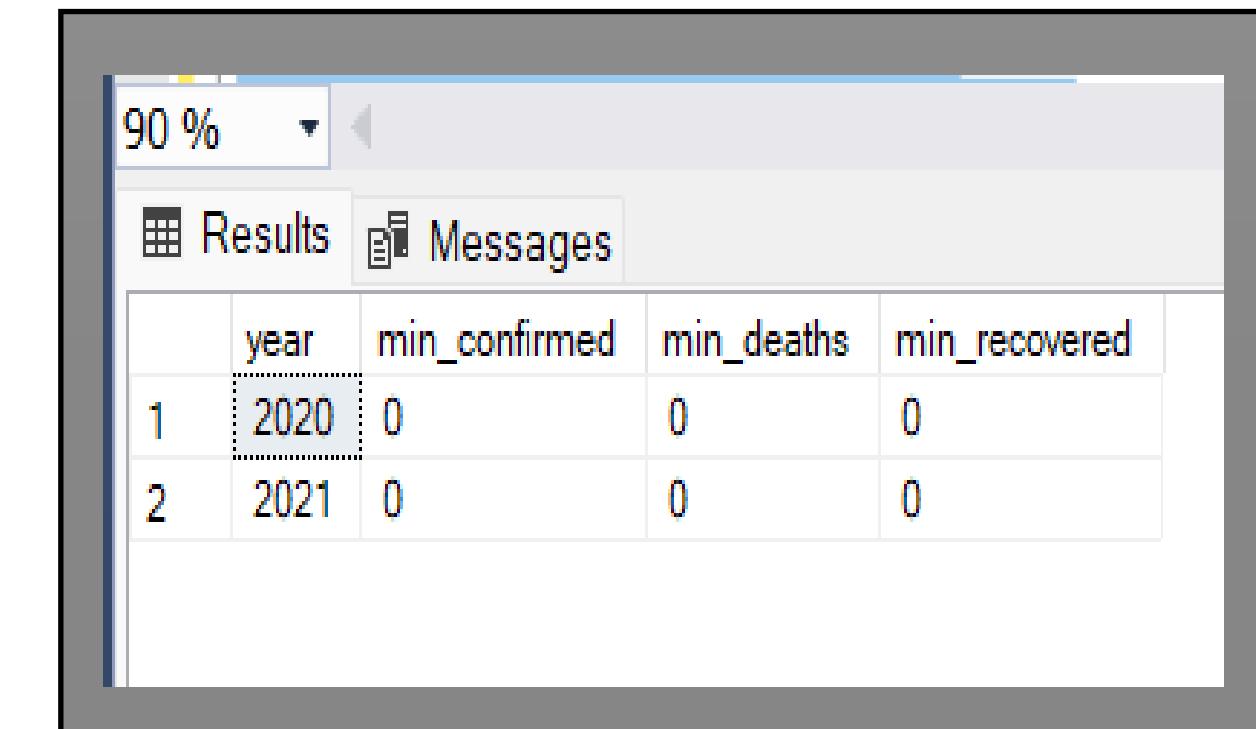
	year	month	confirmed	deaths	recovered
1	2020	1	0	0	0
2	2020	1	0	0	1
3	2020	1	0	0	2
4	2020	1	0	0	8
5	2020	1	0	1	0
6	2020	1	1	0	0
7	2020	1	1	0	1
8	2020	1	10	0	0
9	2020	1	105	7	3
10	2020	1	11	0	0
11	2020	1	11	0	2
12	2020	1	12	0	0
13	2020	1	12	1	0
14	2020	1	13	0	0
15	2020	1	13	0	1
16	2020	1	13	0	2
17	2020	1	1240	27	2

Query executed successfully.



8.Find minimum values for confirmed,deaths,recovered per month.

```
SELECT  
    YEAR(Date) AS year,  
    MIN(confirmed) AS min_confirmed,  
    MIN(deaths) AS min_deaths,  
    MIN(recovered) AS min_recovered  
FROM  
    [dbo].[Corona Virus Dataset]  
GROUP BY  
    YEAR(Date)  
ORDER BY  
    YEAR(Date) ASC;
```



A screenshot of a SQL query results window. The window title bar shows "90 %". Below it, there are two tabs: "Results" and "Messages". The "Results" tab is selected and displays a table with four columns: "year", "min_confirmed", "min_deaths", and "min_recovered". There are two rows of data:

	year	min_confirmed	min_deaths	min_recovered
1	2020	0	0	0
2	2021	0	0	0



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

```
SELECT  
    YEAR(Date) AS year,  
    MAX(confirmed) AS max_confirmed,  
    MAX(deaths) AS max_deaths,  
    MAX(recovered) AS max_recovered  
FROM  
    [dbo].[Corona Virus Dataset]  
GROUP BY  
    YEAR(Date)  
ORDER BY  
    YEAR(Date) ASC;
```

The screenshot shows a SQL query results window with two tabs: 'Results' and 'Messages'. The 'Results' tab is selected and displays a table with four columns: 'year', 'max_confirmed', 'max_deaths', and 'max_recovered'. There are two rows of data: one for the year 2020 with values 9996, 996, and 999 respectively, and another for the year 2021 with values 9997, 999, and 999 respectively. The 'Messages' tab is also visible but contains no text.

	year	max_confirmed	max_deaths	max_recovered
1	2020	9996	996	999
2	2021	9997	999	999



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

```
SELECT  
    MONTH(Date) AS Month,  
    YEAR(Date) AS Year,  
    SUM(CASE WHEN Confirmed = 'Confirmed' THEN 1 ELSE 0 END) AS TotalConfirmedCases,  
    SUM(CASE WHEN Deaths = 'Deaths' THEN 1 ELSE 0 END) AS TotalDeaths,  
    SUM(CASE WHEN Recovered = 'Recovered' THEN 1 ELSE 0 END) AS TotalRecovered  
FROM [dbo].[Corona Virus Dataset]  
GROUP BY YEAR(Date), MONTH(Date)  
ORDER BY YEAR(Date), MONTH(Date);
```

The screenshot shows a SQL Server Management Studio (SSMS) results window. The title bar says "90 %". The results tab is selected, showing a table with six columns: Month, Year, TotalConfirmedCases, TotalDeaths, and TotalRecovered. The data starts at row 1 (Month 1, Year 2020) and ends at row 17 (Month 5, Year 2021). All values are 0 for every row. A yellow status bar at the bottom right says "Query executed successfully." There is also a small speaker icon in the bottom right corner of the window.

	Month	Year	TotalConfirmedCases	TotalDeaths	TotalRecovered
1	1	2020	0	0	0
2	2	2020	0	0	0
3	3	2020	0	0	0
4	4	2020	0	0	0
5	5	2020	0	0	0
6	6	2020	0	0	0
7	7	2020	0	0	0
8	8	2020	0	0	0
9	9	2020	0	0	0
10	10	2020	0	0	0
11	11	2020	0	0	0
12	12	2020	0	0	0
13	1	2021	0	0	0
14	2	2021	0	0	0
15	3	2021	0	0	0
16	4	2021	0	0	0
17	5	2021	0	0	0

11. Check how corona virus spread out with respect to confirmed case (Eg.: total confirmed cases, their average, variance & STDEV).

```
SELECT
    SUM(CAST(Confirmed AS INT)) AS TotalconfirmedCases,
    AVG(CAST(Confirmed AS INT)) AS AverageConfirmedCases,
    VAR(CAST(Confirmed AS INT)) AS confirmedCasesVarience,
    STDEV(CAST(Confirmed AS INT)) AS ConfirmedCasesSTDEV
FROM [dbo].[Corona Virus Dataset]
```

	TotalconfirmedCases	AverageConfirmedCases	confirmedCasesVarience	ConfirmedCasesSTDEV
1	169065144	2156	157290931.698175	12541.5681514783



12. Check how corona virus spread out with respect to death case per month (Eg.: total confirmed cases, their average, variance & STDEV).

```
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    SUM(CAST(Deaths AS INT)) AS TotalDeathCases,
    AVG(CAST(Deaths AS INT)) AS AverageDeathCases,
    VAR(CAST(Deaths AS INT)) AS DeathCasesVariance,
    STDEV(CAST(Deaths AS INT)) AS DeathCasesSTDEV
FROM [dbo].[Corona Virus Dataset]
GROUP BY YEAR(Date), MONTH(Date)
ORDER BY Year, Month;
```

	Year	Month	TotalDeathCases	AverageDeathCases	DeathCasesVariance	DeathCasesSTDEV
1	2020	1	190	0	4.24857598541809	2.06120740960683
2	2020	2	2651	0	68.337150469718	8.26662872455985
3	2020	3	41346	8	3901.60952698687	62.4628651839385
4	2020	4	191833	41	40513.0371733448	201.278506486273
5	2020	5	144561	30	20689.2454049367	143.837566042174
6	2020	6	137757	29	16933.1108854449	130.127287243856
7	2020	7	167613	35	21144.5840570796	145.41177413497
8	2020	8	179200	37	23277.8724251087	152.570876726552
9	2020	9	160671	34	20107.1214145132	141.799581855918
10	2020	10	175484	36	17583.7542527085	132.60374901453



13 Check how corona virus spread out with respect to recovered case (Eg.: total confirmed cases, their average, variance & STDEV).

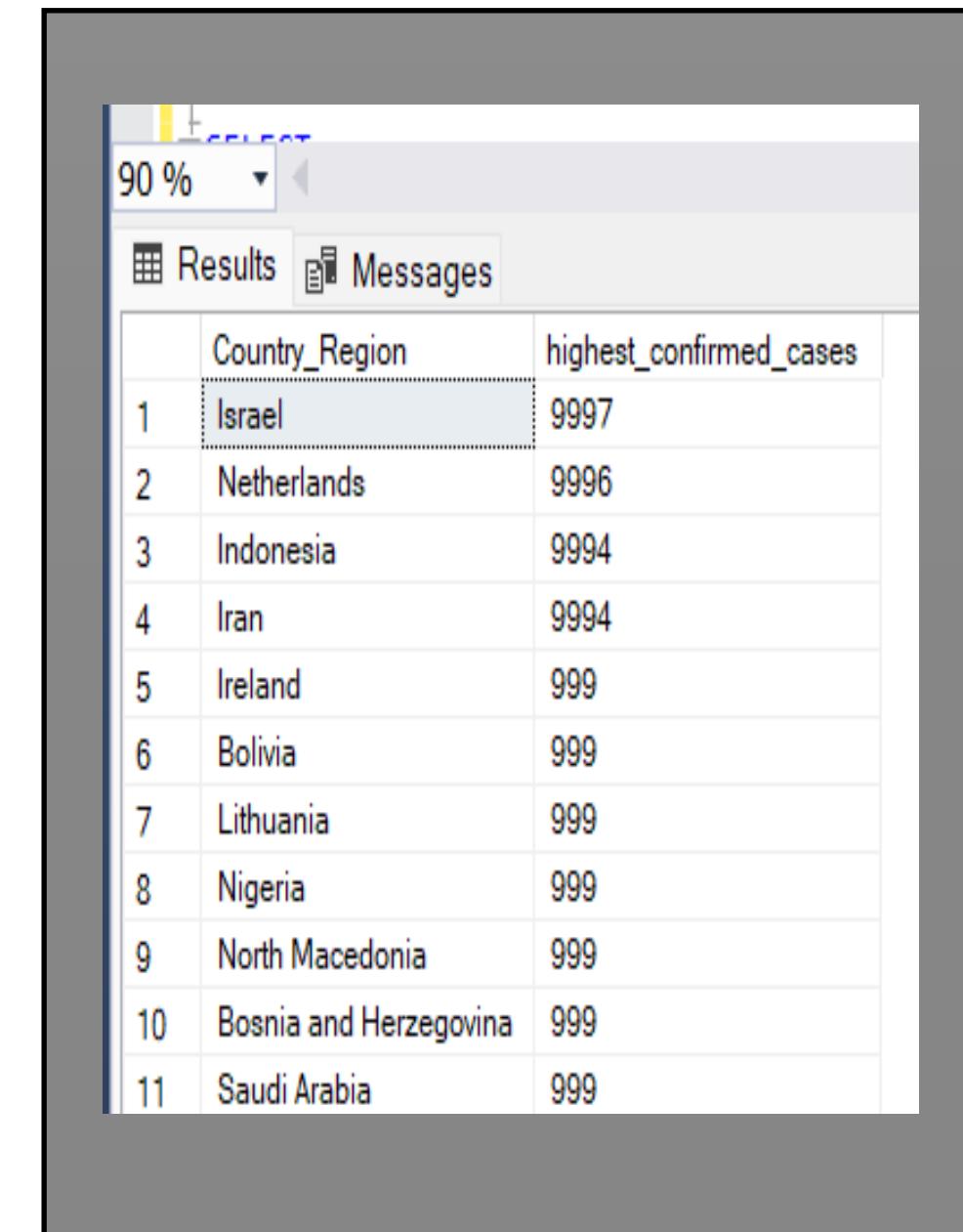
```
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    SUM(CAST(Recovered AS INT)) AS TotalRecoveredCases,
    AVG(CAST(Recovered AS INT)) AS AverageRecoveredCases,
    VAR( CAST(Recovered AS INT)) AS RecoveredCasesVariance,
    STDEV(CAST(Recovered AS INT)) AS RecoveredCasesSTDEV
FROM [dbo].[Corona Virus Dataset]
GROUP BY YEAR(Date), MONTH(Date)
ORDER BY Year, Month;
```

	Year	Month	TotalRecoveredCases	AverageRecoveredCases	RecoveredCasesVariance	RecoveredCasesSTDEV
1	2020	1	143	0	2.63529657477026	1.62335965662889
2	2020	2	31405	7	12449.4495904104	111.577101550499
3	2020	3	133070	27	40121.5939844912	200.303754294549
4	2020	4	792987	171	770059.711532687	877.530461883054
5	2020	5	1519547	318	1978620.87525624	1406.63459194499
6	2020	6	2535417	548	6531586.25639116	2555.69682403668
7	2020	7	4693120	983	24849082.9398306	4984.88544901792
8	2020	8	6202833	1299	40178838.3767708	6338.67796758684
9	2020	9	6647749	1438	57035911.8793661	7552.21238309451
10	2020	10	6782150	1420	73747150.1663075	8587.61609332342



14. Find Country having highest number of the Confirmed case.

```
SELECT  
    [Country_Region],  
    MAX(confirmed) AS highest_confirmed_cases  
FROM  
    [dbo].[Corona Virus Dataset]  
GROUP BY  
    [Country_Region]  
ORDER BY  
    highest_confirmed_cases DESC;
```



The screenshot shows the SQL Server Management Studio interface with the 'Results' tab selected. The results grid displays the top 11 countries with the highest confirmed COVID-19 cases, ordered by highest confirmed cases in descending order. Israel has the highest number of cases at 9997, followed by the Netherlands and Indonesia at 9996 and 9994 respectively. The remaining countries listed have 999 cases each.

	Country_Region	highest_confirmed_cases
1	Israel	9997
2	Netherlands	9996
3	Indonesia	9994
4	Iran	9994
5	Ireland	999
6	Bolivia	999
7	Lithuania	999
8	Nigeria	999
9	North Macedonia	999
10	Bosnia and Herzegovina	999
11	Saudi Arabia	999



15. Find the country having lowest number of death cases.

```
SELECT
    [Country_Region],
    MIN(deaths) AS lowest_death_cases
FROM
    [dbo].[Corona Virus Dataset]
GROUP BY
    [Country_Region]
ORDER BY
    lowest_death_cases ASC;
```

The screenshot shows an SQL query results window with a title bar '90 %' and tabs 'Results' and 'Messages'. The 'Results' tab is selected, displaying a table with two columns: 'Country_Region' and 'lowest_death_cases'. The table contains 17 rows, each representing a country or region with a value of 0 in the 'lowest_death_cases' column. A yellow status bar at the bottom right indicates 'Query executed successfully.'

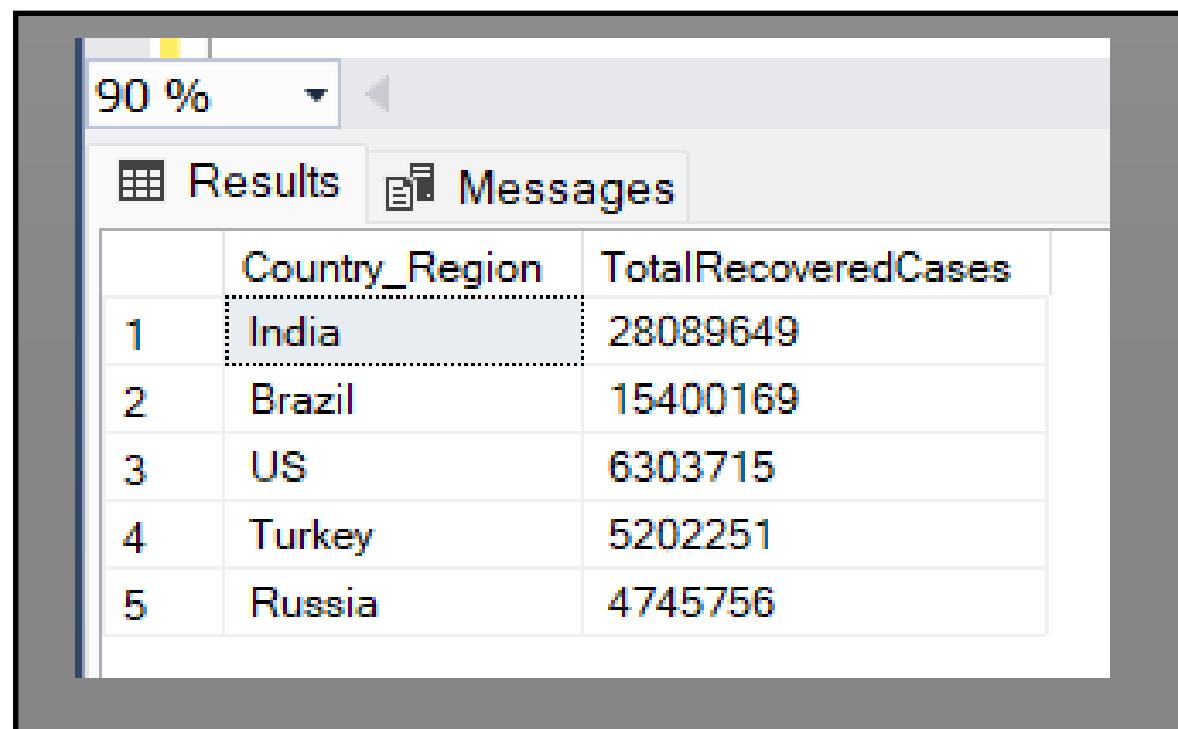
	Country_Region	lowest_death_cases
1	Finland	0
2	Gambia	0
3	West Bank and Gaza	0
4	Vietnam	0
5	New Zealand	0
6	Uganda	0
7	Egypt	0
8	Italy	0
9	Brazil	0
10	Netherlands	0
11	Namibia	0
12	Jordan	0
13	Korea, South	0
14	Belarus	0
15	Madagascar	0
16	Marshall Islands	0
17	Pelvis	0

Query executed successfully.



16. Find the top 5 countries having highest recovered case.

```
SELECT TOP 5 Country_Region, SUM(CAST(Recovered AS INT)) AS TotalRecoveredCases  
FROM [dbo].[Corona Virus Dataset]  
GROUP BY Country_Region  
ORDER BY TotalRecoveredCases DESC
```



	Country_Region	TotalRecoveredCases
1	India	28089649
2	Brazil	15400169
3	US	6303715
4	Turkey	5202251
5	Russia	4745756





THANK YOU

