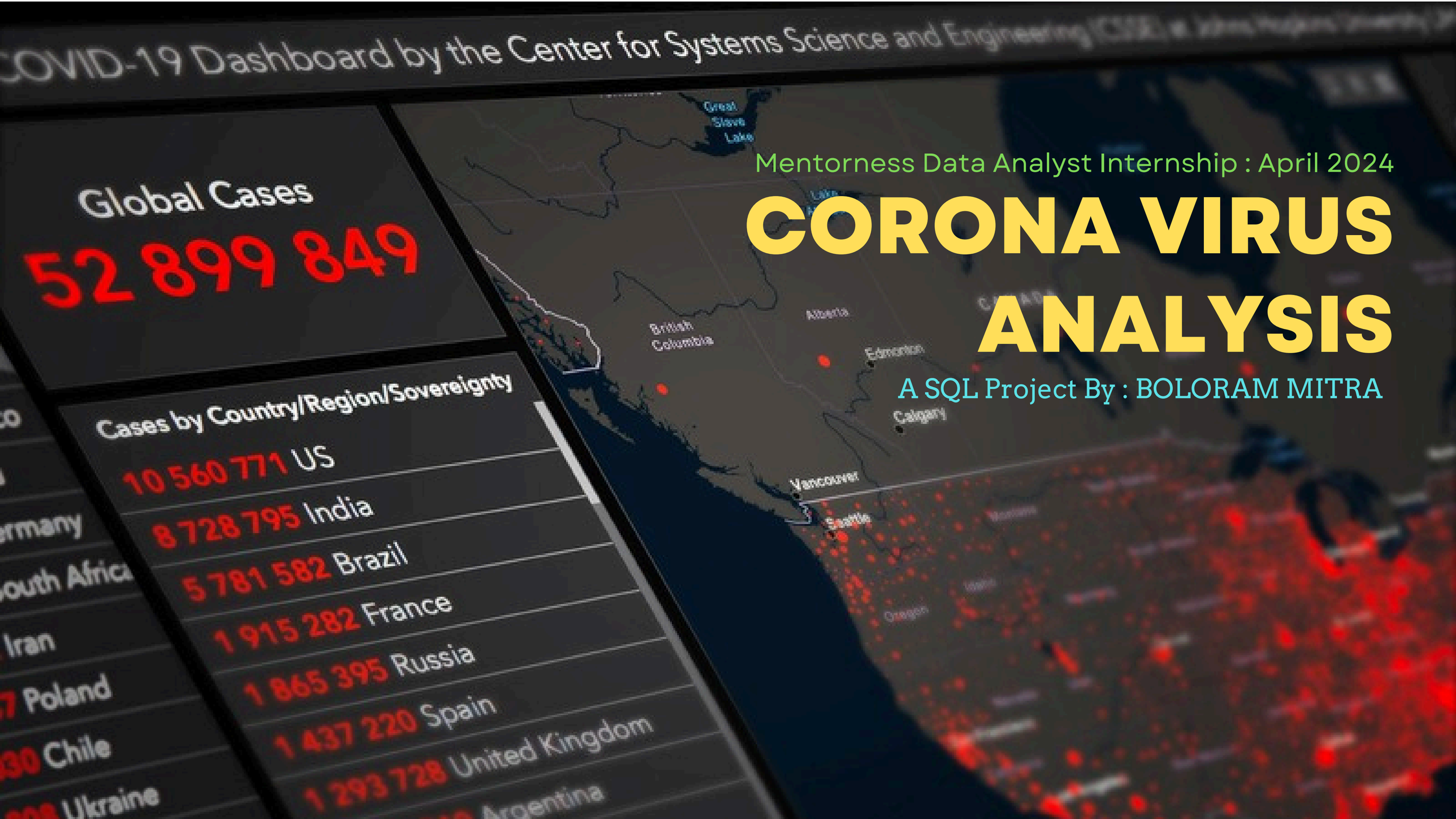


Mentorness Data Analyst Internship : April 2024

CORONA VIRUS ANALYSIS

A SQL Project By : BOLORAM MITRA

TABLE OF CONTENTS

INTRODUCTION TO THE PROJECT

- The COVID-19 pandemic, caused by the novel coronavirus SARS-CoV-2, has significantly impacted the world since its emergence in late 2019. With its rapid spread across continents, the pandemic has posed unprecedented challenges to public health systems, economies, and societies worldwide. Analyzing COVID-19 data has become crucial for understanding the dynamics of the pandemic, tracking its progression, and informing effective responses.
- Our project aims to analyze COVID-19 data to gain insights into the spread and impact of the virus. By leveraging data analytics techniques, we seek to extract meaningful patterns, trends, and correlations from various datasets related to COVID-19 cases, deaths, recoveries, and other relevant variables.

ABOUT THE DATASET

Table: corona

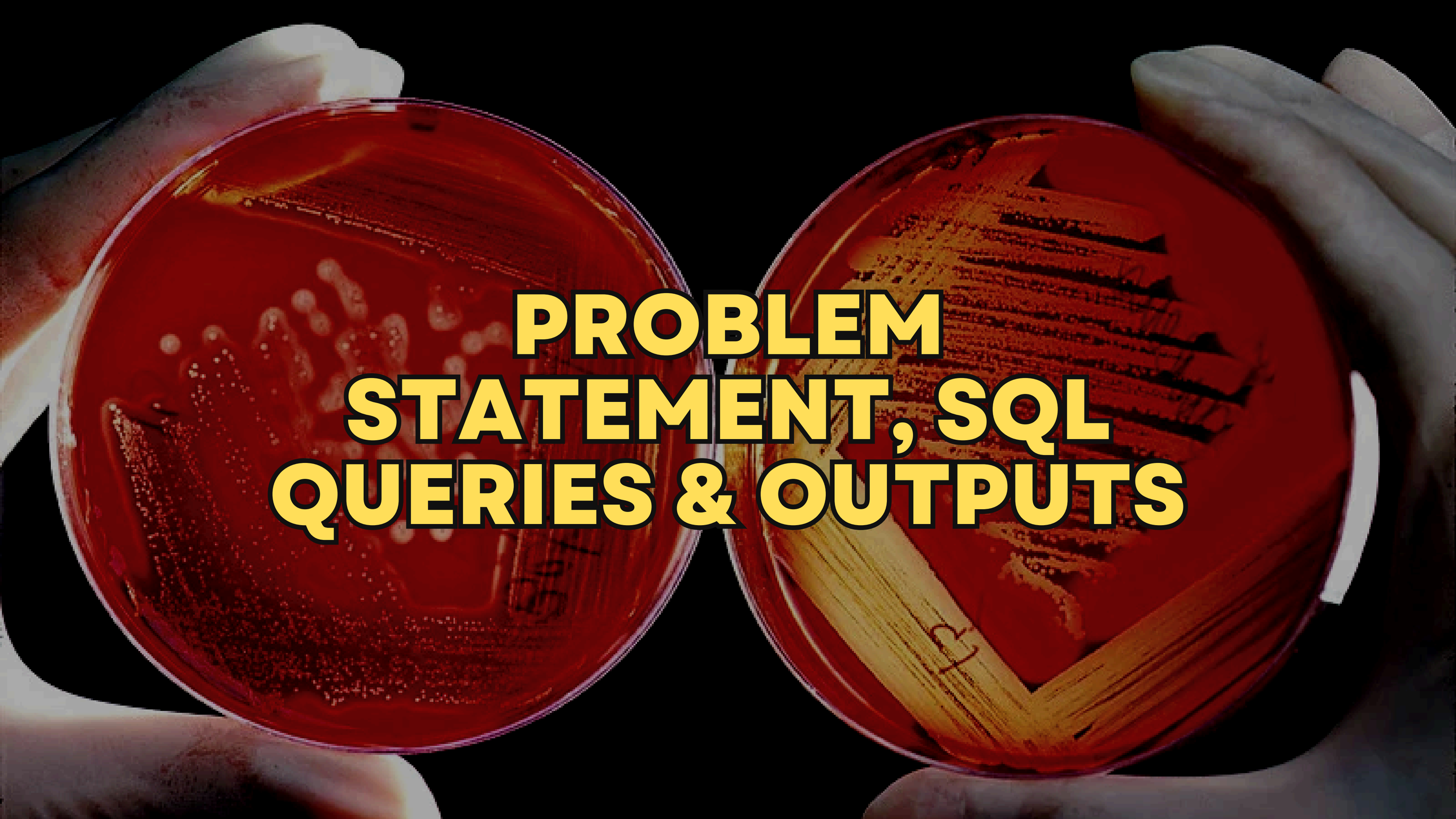
Columns:

Province	text
Country	text
Latitude	double
Longitude	double
Confirmed	int
Deaths	int
Recovered	int
Date	date

- The Dataset has 1 table, 8 Columns and 78368 Rows.

- Sample Database

Province	Country	Latitude	Longitude	Confirmed	Deaths	Recovered	Date
United Kingdom	United Kingdom	55.3781	-3.436	5584	1	0	2021-06-07
Philippines	Philippines	12.879721	121.774017	11008	2	41195	2021-04-04
Spain	Spain	40.463667	-3.74922	4088	1	0	2020-08-06
Turkey	Turkey	38.9637	35.2433	823225	220	5232	2020-12-10
Sweden	Sweden	60.128161	18.643501	7158	2	0	2021-04-29
France	France	46.2276	2.2137	6908	2	0	2020-09-06


PROBLEM
STATEMENT, SQL
QUERIES & OUTPUTS

Q1. Write a code to check NULL value

```
SELECT *
FROM Corona
WHERE Province IS NULL OR
Country IS NULL OR
Latitude IS NULL OR
Longitude IS NULL OR
Date IS NULL OR
Confirmed IS NULL OR
Deaths IS NULL OR
Recovered IS NULL;
```

- There is no NULL value found in the Database.

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

```
SELECT
    COALESCE(Province, 0) AS Province,
    COALESCE(Country, 0) AS Country,
    COALESCE(Latitude, 0) AS Latitude,
    COALESCE(Longitude, 0) AS Longitude,
    COALESCE(Date, 0) AS Date,
    COALESCE(Confirmed, 0) AS Confirmed,
    COALESCE(Deaths, 0) AS Deaths,
    COALESCE(Recovered, 0) AS Recovered
FROM Corona;
```

- In this query, the COALESCE function is used to replace each column value with 0 if it is null.

Q3. check total number of rows

```
Select count(*) as total_number_of_rows
from corona;
```

- The total no. of rows found in the database is 78368.

	total_number_of_rows
▶	78386

Q4. Check what is start_date and end_date

```
Select Min(Date) as Start_date,
       Max(Date) as End_date
from corona;
```

Output :

	Start_date	End_date
▶	2020-01-22	2021-06-13

Q5. Number of month present in dataset

```
Select count(distinct(Month(date))) as count_of_month
from corona;
-- ----------------------------------------------------------OR
Select distinct(Month(date)) as No_of_Mon
from corona;
```

Output :

	count_of_month
▶	12

	No_of_Mon
▶	1
	2
	3
	4
	5
	6
	7
	8
	9
	10
	11
	12

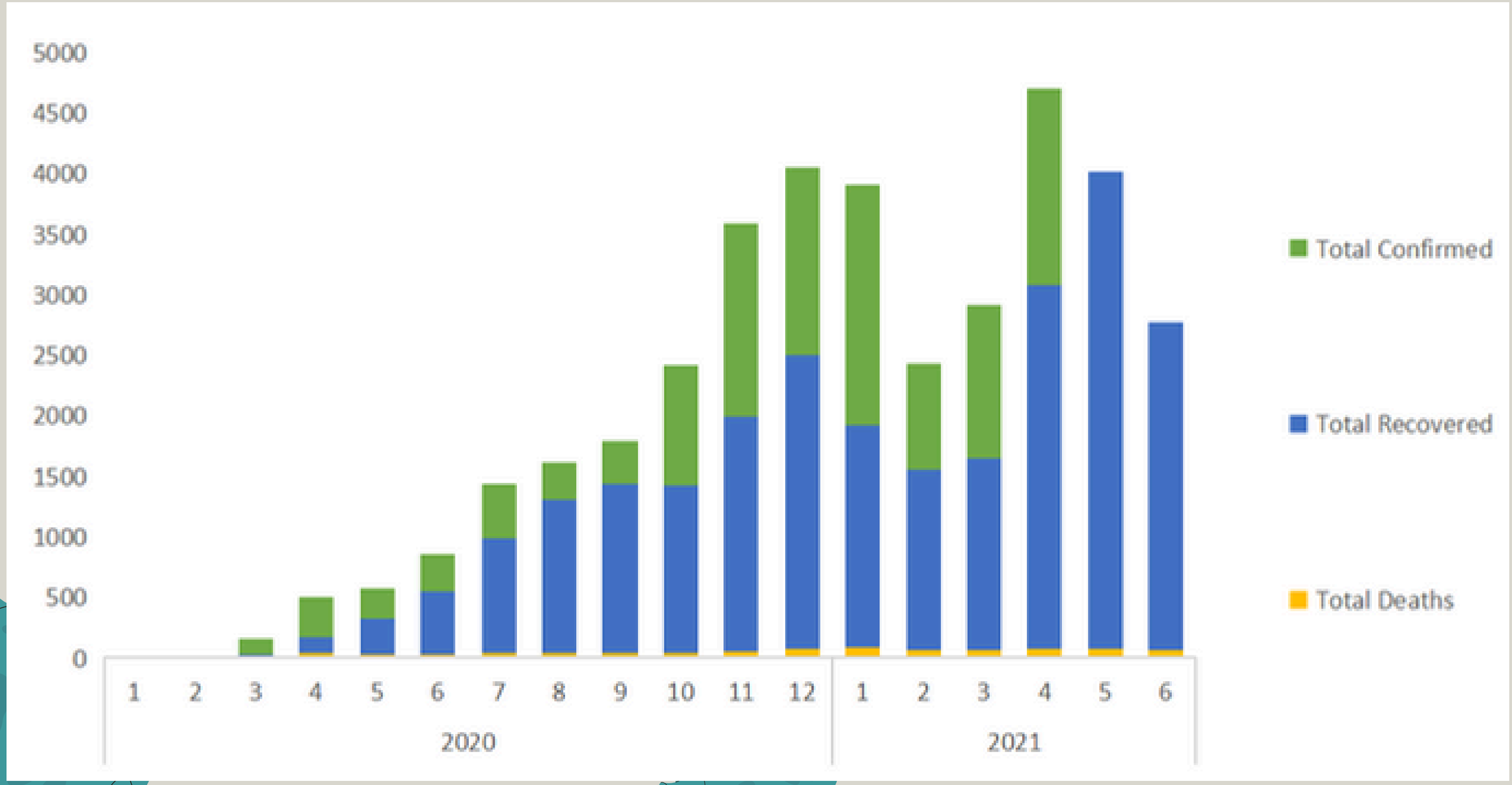
Q6. Find monthly average for confirmed, deaths, recovered

```
SELECT
    YEAR(Date) AS Years,
    Month(Date) AS Months,
    Round(AVG(Confirmed),2) AS AvgConfirmed,
    Round(AVG(Deaths),2) AS AvgDeaths,
    Round(AVG(Recovered),2) AS AvgRecovered
FROM Corona
GROUP BY YEAR(Date), Month(Date)
ORDER BY Years,Months;
```

Output :

Years	Months	AvgConfirmed	AvgDeaths	AvgRecovered
2020	1	4.15	0.12	0.09
2020	2	15.30	0.59	7.03
2020	3	161.13	8.66	27.87
2020	4	505.80	41.52	171.64
2020	5	574.85	30.28	318.30
2020	6	859.23	29.82	548.79
2020	7	1432.36	35.11	983.06
2020	8	1611.84	37.54	1299.29
2020	9	1784.59	34.78	1438.91
2020	10	2412.20	36.76	1420.64
2020	11	3592.19	56.76	1985.34
2020	12	4050.44	71.22	2497.89
2021	1	3911.23	84.18	1919.64
2021	2	2433.36	69.16	1558.39

- **Output of monthly average for confirmed, deaths, recovered**



Q7. Find most frequent value for confirmed, deaths, recovered each month

```sql
SELECT ConfirmedCount, Confirmed, DeathsCount, Deaths, RecoveredCount, Recovered, Months, Years
FROM (SELECT
    COUNT(Confirmed) AS ConfirmedCount, Confirmed,
    COUNT(Deaths) AS DeathsCount, Deaths,
    COUNT(Recovered) AS RecoveredCount, Recovered,
    MONTH(Date) AS Months, YEAR(Date) AS Years,
    DENSE_RANK() OVER (PARTITION BY MONTH(Date), YEAR(Date) ORDER BY COUNT(Confirmed) DESC) AS RankedConfirmed,
    DENSE_RANK() OVER (PARTITION BY MONTH(Date), YEAR(Date) ORDER BY COUNT(Deaths) DESC) AS RankedDeaths,
    DENSE_RANK() OVER (PARTITION BY MONTH(Date), YEAR(Date) ORDER BY COUNT(Recovered) DESC) AS RankedRecovered
    FROM Corona
    GROUP BY Confirmed, Deaths, Recovered, MONTH(Date), YEAR(Date)
) AS subquery
WHERE RankedConfirmed = 1 AND RankedDeaths = 1 AND RankedRecovered = 1
ORDER BY Years, Months;
```

Output : The most frequent value is 0

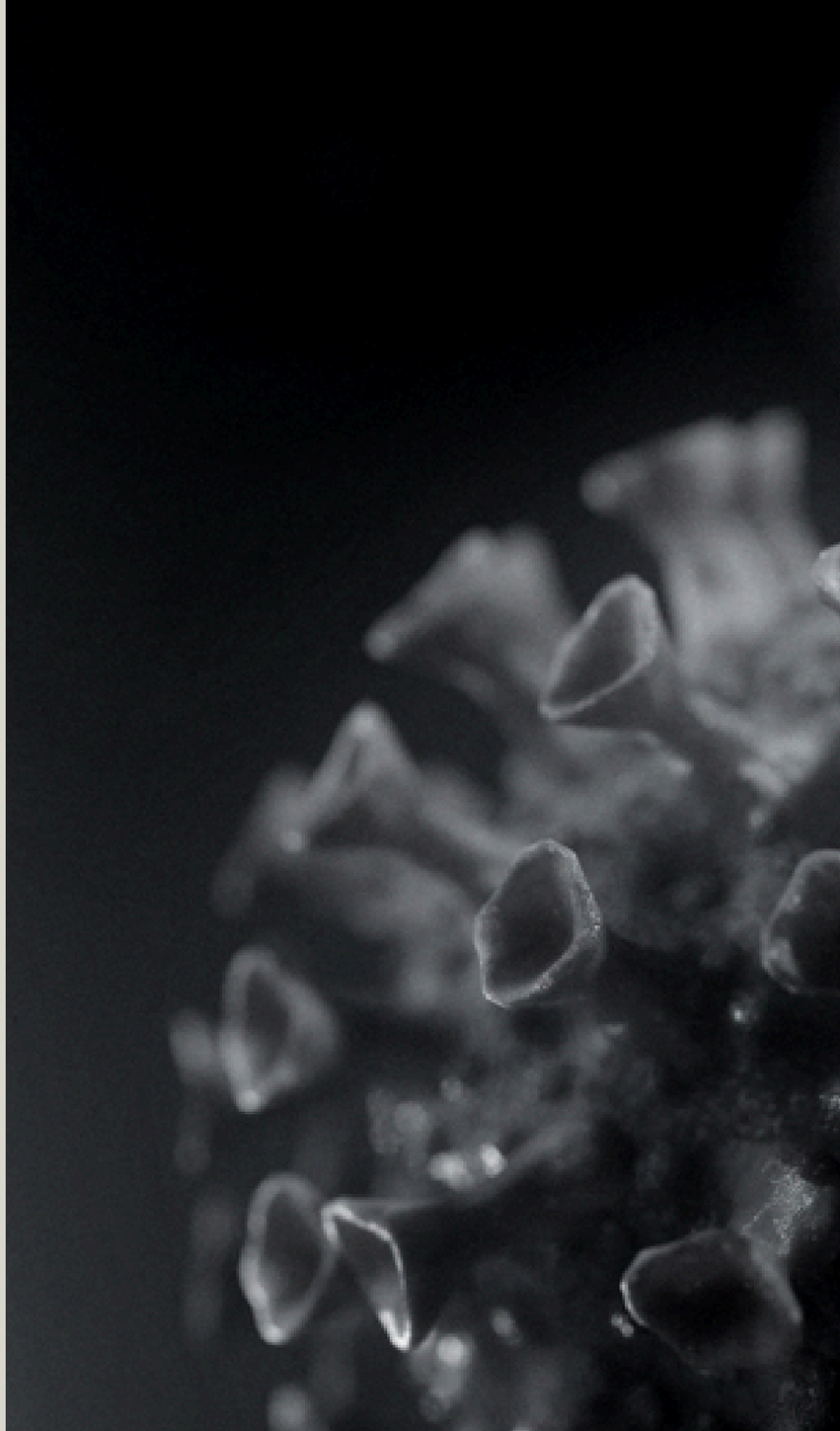
ConfirmedCount	Confirmed	DeathsCount	Deaths	RecoveredCount	Recovered	Months	Years
1368	0	1368	0	1368	0	1	2020
3794	0	3794	0	3794	0	2	2020
1994	0	1994	0	1994	0	3	2020
1064	0	1064	0	1064	0	4	2020
1253	0	1253	0	1253	0	5	2020
1249	0	1249	0	1249	0	6	2020
1245	0	1245	0	1245	0	7	2020
1116	0	1116	0	1116	0	8	2020
1135	0	1135	0	1135	0	9	2020
1112	0	1112	0	1112	0	10	2020
1033	0	1033	0	1033	0	11	2020
1087	0	1087	0	1087	0	12	2020
1048	0	1048	0	1048	0	1	2021
987	0	987	0	987	0	2	2021
1144	0	1144	0	1144	0	3	2021
1009	0	1009	0	1009	0	4	2021
1142	0	1142	0	1142	0	5	2021
510	0	510	0	510	0	6	2021

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

```
Select Min(confirmed), Min(deaths), Min(recovered), Year(date) as years
from corona
group by Year(date);
```

Output :

Min(confirmed)	Min(deaths)	Min(recovered)	years
0	0	0	2020
0	0	0	2021



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

```
Select Max(confirmed), Max(deaths), Max(recovered), Year(date) as years
from corona
group by Year(date);
```

Output :

Max(confirmed)	Max(deaths)	Max(recovered)	years
823225	3752	1123456	2020
414188	7374	422436	2021

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

```
Select Sum(confirmed) as total_confirmed_cases,
       Sum(deaths) as total_deaths,
       Sum(Recovered) as total_recovered,
       Year(date) as Years, Month(Date) as Months
From Corona
group by Year(date),Month(Date)
order by Years;
```

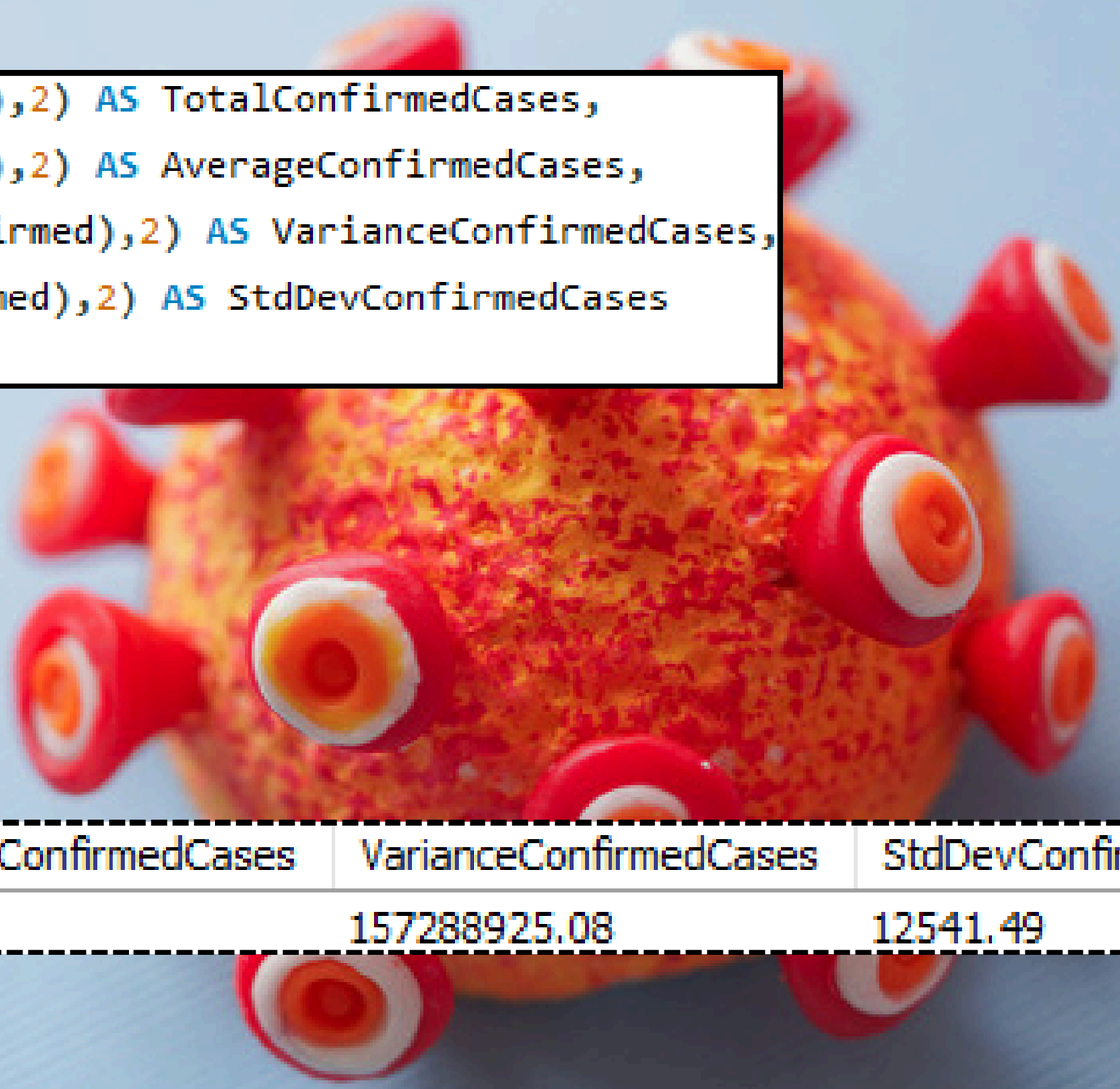
Output :

total_confirmed_cases	total_deaths	total_recovered	Years	Months
6384	190	143	2020	1
68312	2651	31405	2020	2
769236	41346	133070	2020	3
2336798	191833	792987	2020	4
2744333	144561	1519547	2020	5
3969634	137757	2535417	2020	6
6838092	167613	4693120	2020	7
7694938	179200	6202833	2020	8
8244794	160671	6647749	2020	9
11515841	175484	6782150	2020	10
16595938	262247	9172292	2020	11
19336799	339996	11924903	2020	12
18672205	401893	9164347	2021	1
10492664	298239	6719785	2021	2
13924790	282620	7888013	2021	3
21711021	362387	14205507	2021	4
19121083	366549	19131842	2021	5
5022282	132657	5544438	2021	6

- **Output of total number of case of confirmed, deaths, recovered each month**

Q11. Check how corona virus spread out with respect to confirmed case (Eg.: total confirmed cases, their average, variance & STDEV)

```
SELECT Round(SUM(Confirmed),2) AS TotalConfirmedCases,
       Round(AVG(Confirmed),2) AS AverageConfirmedCases,
       Round(VARIANCE(Confirmed),2) AS VarianceConfirmedCases,
       Round(STDDEV(Confirmed),2) AS StdDevConfirmedCases
FROM Corona;
```

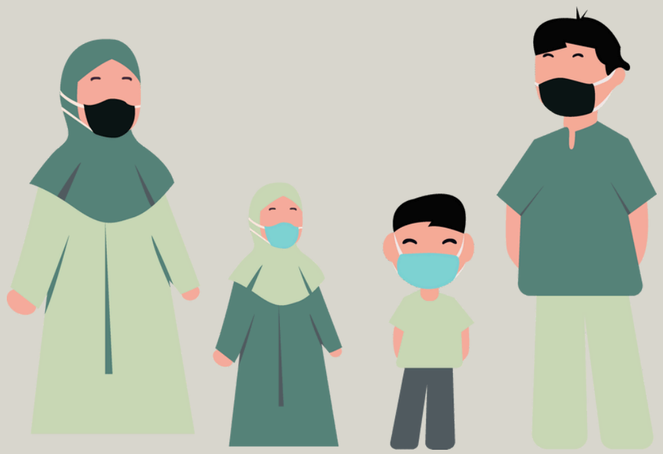


Output :

TotalConfirmedCases	AverageConfirmedCases	VarianceConfirmedCases	StdDevConfirmedCases
169065144	2156.83	157288925.08	12541.49

Q13. Check how corona virus spread out with respect to recovered case (Eg.: total confirmed cases, their average, variance & STDEV)

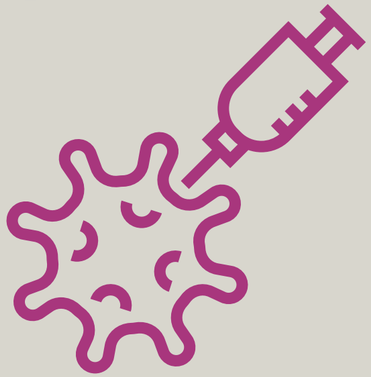

```sql
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    ROUND(SUM(Recovered), 2) AS TotalRecoveredCases,
    ROUND(AVG(Recovered), 2) AS AverageRecoveredCases,
    ROUND(VARIANCE(Recovered), 2) AS VarianceRecoveredCases,
    ROUND(STDDEV(Recovered), 2) AS StdDevRecoveredCases
FROM Corona
GROUP BY YEAR(Date), MONTH(Date)
ORDER BY Year, Month;
```



Output :

Year	Month	TotalRecoveredCases	AverageRecoveredCases	VarianceRecoveredCases	StdDevRecoveredCases
2020	1	143	0.09	2.63	1.62
2020	2	31405	7.03	12446.66	111.56
2020	3	133070	27.87	40113.19	200.28
2020	4	792987	171.64	769893.03	877.44
2020	5	1519547	318.30	1978206.42	1406.49
2020	6	2535417	548.79	6530172.49	2555.42
2020	7	4693120	983.06	24843877.85	4984.36
2020	8	6202833	1299.29	40170422.2	6338.01
2020	9	6647749	1438.91	57023566.44	7551.39
2020	10	6782150	1420.64	73731702.5	8586.72
2020	11	9172292	1985.34	50727618.87	7122.33
2020	12	11924903	2497.89	326694724.1	18074.7
2021	1	9164347	1919.64	31493700.12	5611.92
2021	2	6719785	1558.39	24427411.6	4942.41
2021	3	7888013	1652.29	34897391.64	5907.4
2021	4	14205507	3074.79	224419585.15	14980.64
2021	5	19131842	4007.51	755175531.76	27480.46
2021	6	5544438	2769.45	233034407.39	15265.46

Q12. Check how corona virus spread out with respect to death case per month (Eg.: total confirmed cases, their average, variance & STDEV)



```sql
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    Round(SUM(Deaths),2) AS TotalDeathCases,
    Round(AVG(Deaths),2) AS AverageDeathCases,
    Round(VARIANCE(Deaths),2) AS VarianceDeathCases,
    Round(STDDEV(Deaths),2) AS StdDevDeathCases
FROM Corona
GROUP BY YEAR(Date), MONTH(Date)
ORDER BY Year, Month;
```

Output :

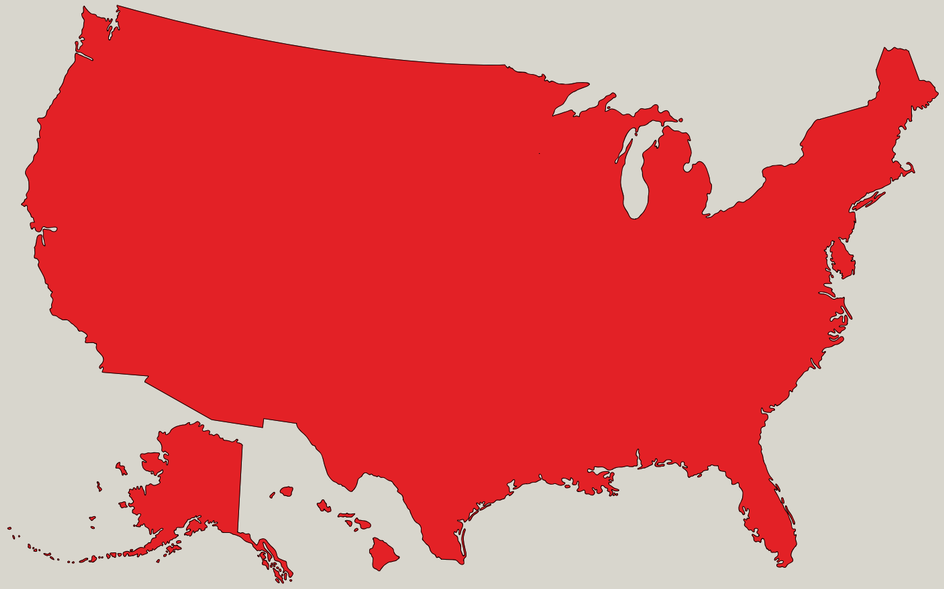
Year	Month	TotalDeathCases	AverageDeathCases	VarianceDeathCases	StdDevDeathCases
2020	1	190	0.12	4.25	2.06
2020	2	2651	0.59	68.32	8.27
2020	3	41346	8.66	3900.79	62.46
2020	4	191833	41.52	40504.27	201.26
2020	5	144561	30.28	20684.91	143.82
2020	6	137757	29.82	16929.45	130.11
2020	7	167613	35.11	21140.15	145.4
2020	8	179200	37.54	23273	152.55
2020	9	160671	34.78	20102.77	141.78
2020	10	175484	36.76	17580.07	132.59
2020	11	262247	56.76	27773.79	166.65
2020	12	339996	71.22	65345.37	255.63
2021	1	401893	84.18	102758.43	320.56
2021	2	298239	69.16	68478.87	261.68
2021	3	282620	59.20	54385.97	233.21
2021	4	362387	78.44	94611.47	307.59
2021	5	366549	76.78	131769.47	363
2021	6	132657	66.26	112963.67	336.1

Q14. Find Country having highest number of the Confirmed case

```
Select Sum(Confirmed) as TotalConfirmedCases, Country
From Corona
Group by Country
Order by TotalConfirmedCases DESC Limit 1;
```

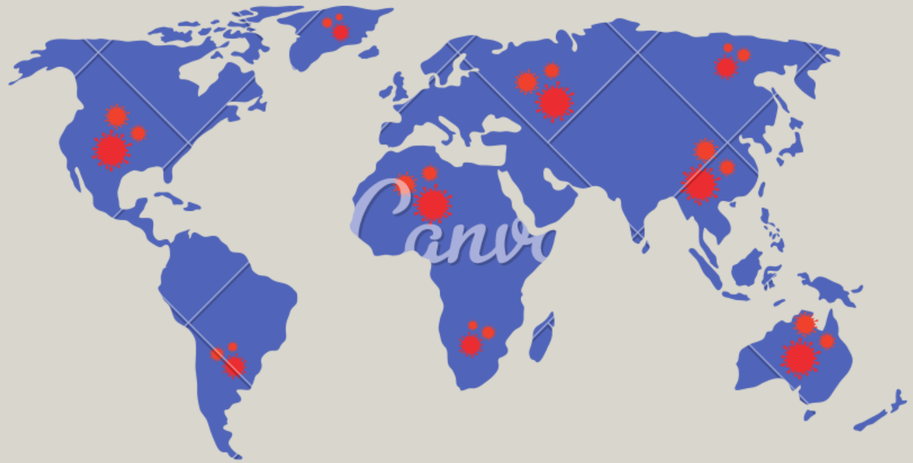
Output :

TotalConfirmedCases	Country
33461982	US



Q15. Find Country having lowest number of the death case

```sql
WITH CountryDeaths AS (
SELECT Country,SUM(Deaths) AS TotalDeaths,
DENSE_RANK() OVER (ORDER BY SUM(Deaths)) AS DeathRank
FROM Corona
GROUP BY Country
)
SELECT Country,TotalDeaths
FROM CountryDeaths
WHERE DeathRank = 1;
```



Output :

Country	TotalDeaths
Dominica	0
Kiribati	0
Marshall Islands	0
Samoa	0

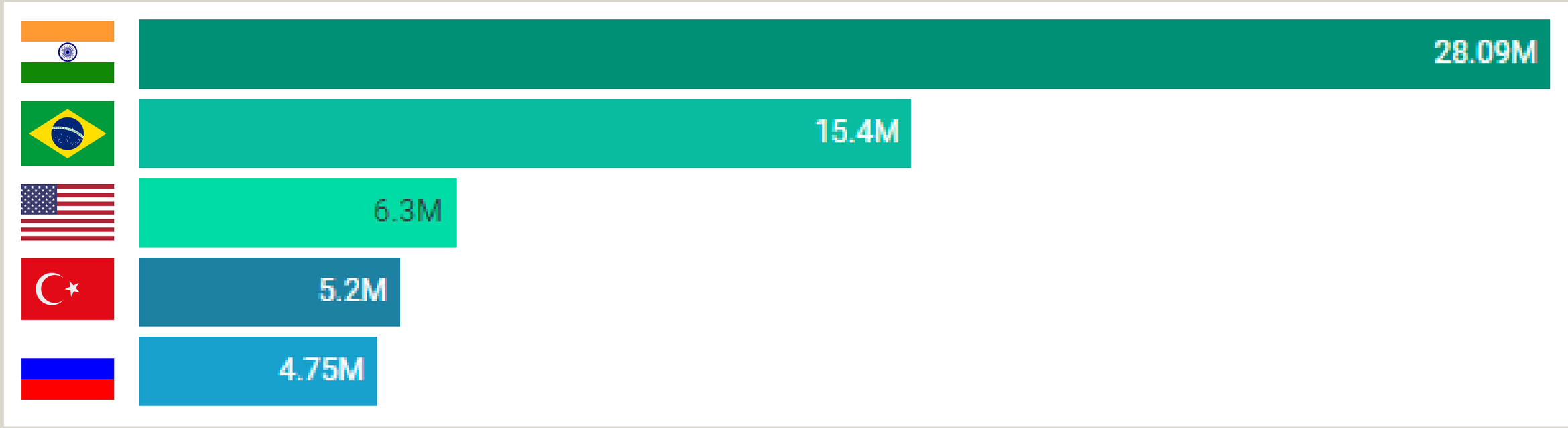
There are several countries with lowest number of total deaths = 0, Such as Dominica, Kiribati, Marshall Islands, Samoa

Q16. Find top 5 countries having highest recovered case

```
Select Country, SUM(Recovered) as TotalRecoveredCases
From Corona
Group by Country
Order by TotalRecoveredCases DESC Limit 5;
```

Output :

Country	TotalRecovered
India	28089649
Brazil	15400169
US	6303715
Turkey	5202251
Russia	4745756



CONCLUSION

In summary, our project on analyzing COVID-19 data has yielded valuable insights into the pandemic's spread and impact. Through thorough data analysis, we've uncovered trends and patterns, revealing variations in transmission rates, mortality rates, and recovery rates across regions and demographics. These findings underscore the pandemic's complexity and the need for data-driven decision-making in response efforts.

Our analysis has informed public health policies, resource allocation, and interventions, highlighting the importance of adaptability and evidence-based approaches. Looking ahead, our project emphasizes the ongoing need for data collection, analysis, and collaboration to address emerging challenges and strengthen global resilience against future health crises.