



**MENTORNESS**

# **CORONA VIRUS ANALYSIS**

**Presented By: MD. SHAKIB**

# PROJECT OVERVIEW

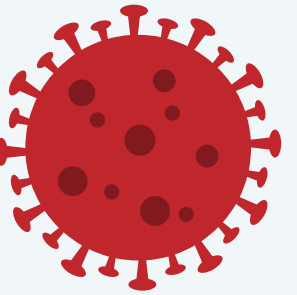
1. The Corona Virus pandemic had a significant impact on public health and has created an urgent need for data-driven insights to understand the spread of the virus.
2. As a data analysts, we are going to analyze the Corona Virus dataset to derive meaningful insights and present the findings.



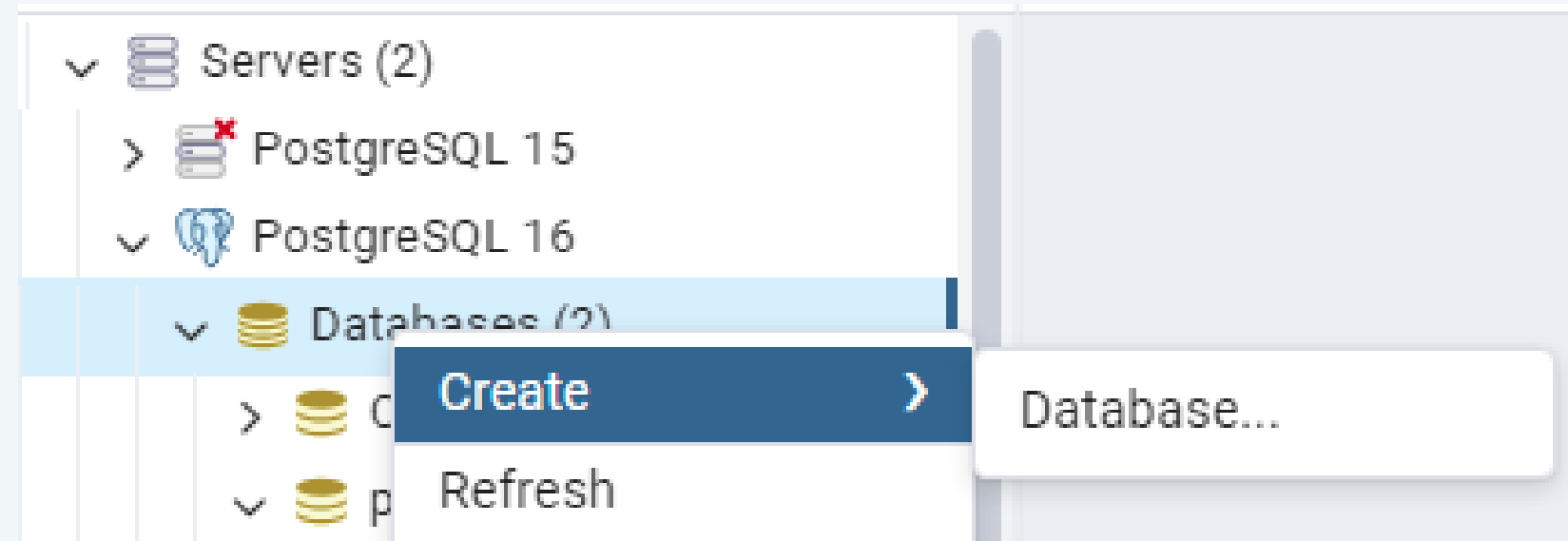
# DATASET

Description of each column in dataset:

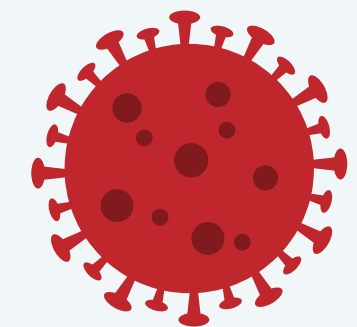
1. **Province:** Geographic subdivision within a country/region.
2. **Country/Region:** Geographic entity where data is recorded.
3. **Latitude:** North-South position on Earth's surface.
4. **Longitude:** East-West position on Earth's surface.
5. **Date:** Recorded date of Corona Virus data.
6. **Confirmed:** Number of diagnosed Corona Virus cases.
7. **Deaths:** Number of Corona Virus related deaths.
8. **Recovered:** Number of recovered Corona Virus cases.



# CREATING DATABASE



A screenshot of the 'Create - Database' dialog box. The 'General' tab is active, showing fields for 'Database' (corona\_database), 'OID', 'Owner' (postgres), and 'Comment'. The 'Save' button is highlighted.



# CREATING TABLE IN SQL

Create - Table

General

Columns

Advanced

Constraints

Partitions

Parameters

Security

SQL

Name

Corona\_Virus

Owner

postgres

Schema

public

Tablespace

Select an item...

Partitioned table?

Comment

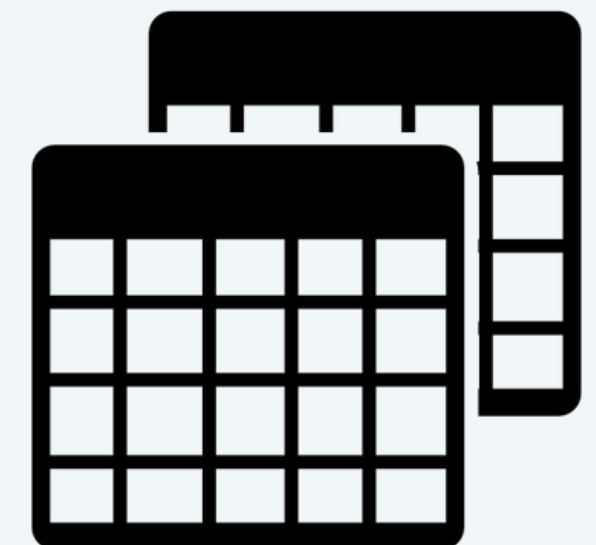
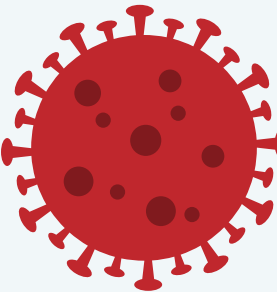
i

?

Close

Reset

Save



# IMPORTING DATA

Import/Export data - table 'corona\_virus' ✕

General Options Columns

Import/Export ☒ Import ☐ Export

Filename  📁

Format  ▾

Encoding  ✕ ▾

ℹ ? ✕ Close ↺ Reset ✓ OK

Import/Export data - table 'corona\_virus' ✕

General Options Columns

Columns to import  ▾

NOT NULL columns  ▾

An optional list of columns to be copied. If no column list is specified, all columns of the table will be copied.

Do not match the specified column values against the null string. In the default case where the null string is empty, this means that empty values will be read as zero-length strings rather than nulls, even when they are not quoted. This option is allowed only in import, and only when using CSV format.

ℹ ? ✕ Close ↺ Reset ✓ OK

# QUERY ANALYSIS

Q1. Write a code to check null.

```
select * from Corona_Virus
where Province is null or
Country is null or
Latitude is null or
Longitude is null or
Confirmed is null or
Recovered is null or
Deaths is null;
```



Result

province character varying (100)	country character varying (100)	latitude double precision	longitude double precision	date date	confirmed integer	recovered integer	deaths integer

# QUERY ANALYSIS

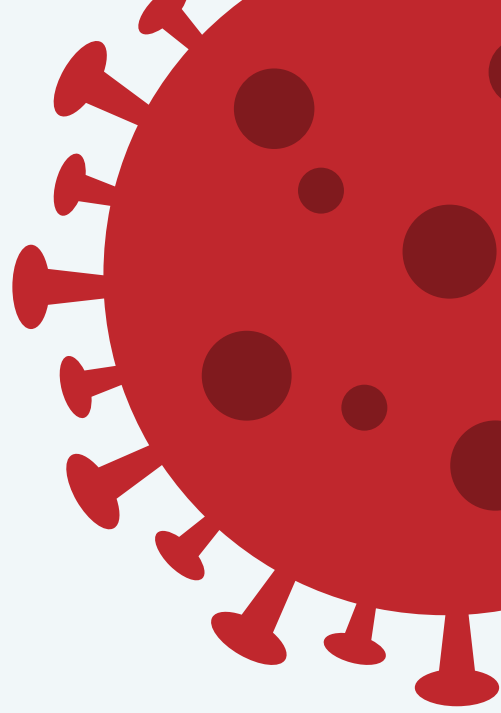
Q2. If Null values are present, update them with zero in all column.

```
update Corona_Virus
Set Province = COALESCE(Province, ''),
Country = COALESCE(Country, ''),
Latitude = COALESCE(Latitude, 0),
Longitude = COALESCE(Longitude, 0),
Confirmed = COALESCE(Confirmed, 0),
Recovered = COALESCE(Recovered, 0),
Deaths = COALESCE(Deaths, 0);
```

## Result

UPDATE 78386

Query returned successfully in 488 msec.






# QUERY ANALYSIS

Q3. Check total numbers of rows.

```
SELECT COUNT (*) FROM Corona_Virus;
```

Result

	count bigint 
1	78386




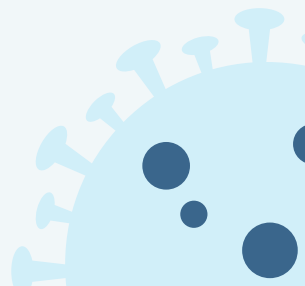
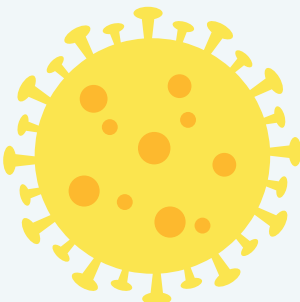
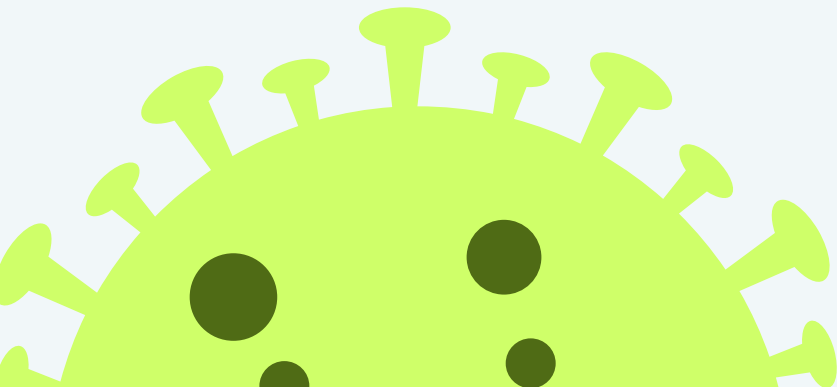
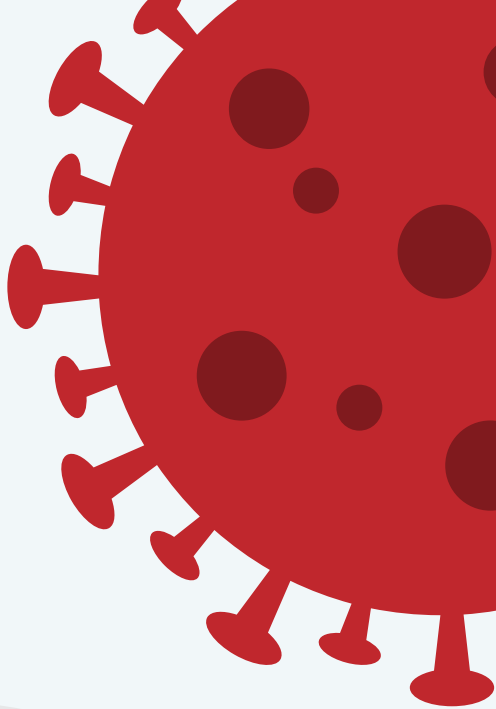
# QUERY ANALYSIS

Q4.1 what is start\_date and end\_date.

```
SELECT date FROM Corona_Virus
ORDER BY date
limit 1;
```

Result

	date 
1	2020-01-22




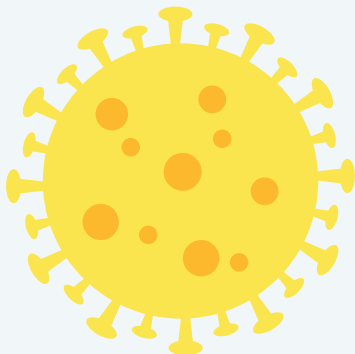
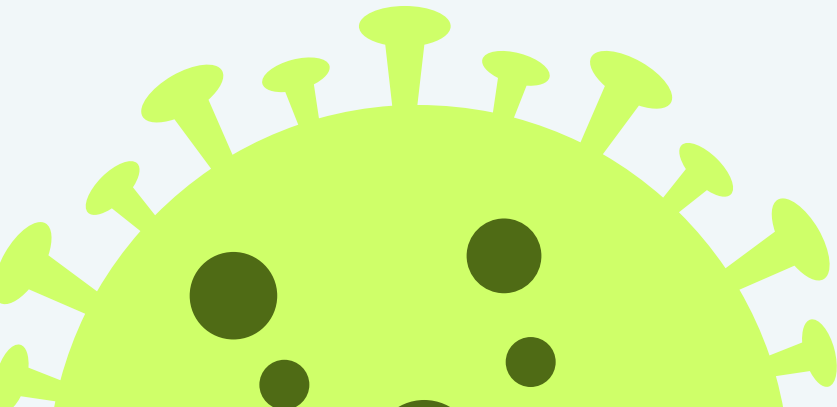
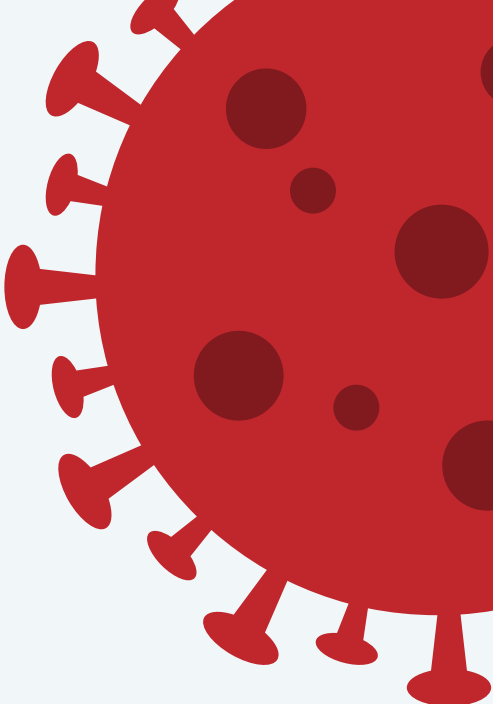
# QUERY ANALYSIS

Q4.2 end\_date

```
SELECT date FROM Corona_Virus
ORDER BY date desc
limit 1;
```

Result

	date 
1	2021-06-13



# QUERY ANALYSIS

Q5. Number of month present in dataset.

```
SELECT Extract(month from date) as number_of_months
FROM Corona_Virus
GROUP BY Extract(month from date)
ORDER BY number_of_months;
```

Result

	number_of_months numeric
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12

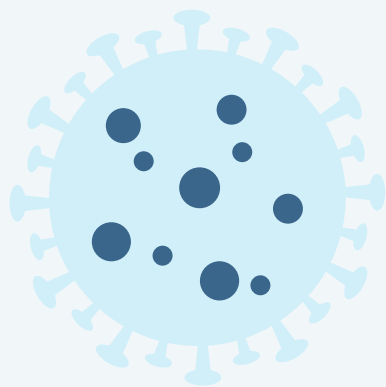
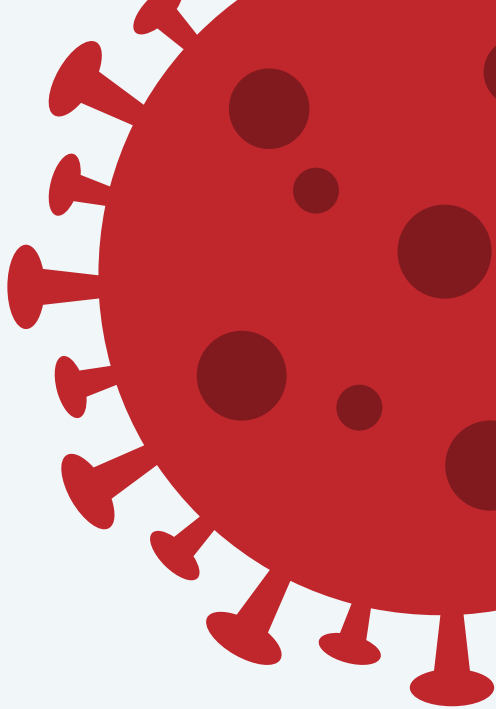
# QUERY ANALYSIS

Q6. Find monthly average for confirmed, death, recovered.

```
SELECT Extract(month from date) as months,
AVG(Confirmed) as avg_confirmed,
AVG(Deaths) as avg_deaths,
AVG(Recovered) as avg_recovered
FROM Corona_Virus
GROUP BY Extract(month from date)
ORDER BY months;
```

## Result

	months numeric	avg_confirmed numeric	avg_deaths numeric	avg_recovered numeric
1	1	2958.2814380741210010	1451.4554957237884067	63.6811846689895470
2	2	1203.1187058555479608	769.1034404192298929	34.2777398040555935
3	3	1538.9637620444072057	840.0799120234604106	33.9302471721826561
4	4	2602.5778138528138528	1623.2136363636363636	59.9805194805194805
5	5	2290.0519480519480519	2162.9020737327188940	53.5305823209049016
6	6	1357.8852310480217457	1220.1532769556025370	40.8356991845363938
7	7	1432.3611227482195224	983.0582320904901550	35.1095517385839966
8	8	1611.8428990364474235	1299.2947214076246334	37.5366568914956012
9	9	1784.5874458874458874	1438.9067099567099567	34.7772727272727273
10	10	2412.1996229576874738	1420.6430666108085463	36.7582739840804357
11	11	3592.1943722943722944	1985.3445887445887446	56.7634199134199134
12	12	4050.4396732299958106	2497.8850020946795140	71.2182656053623796



# QUERY ANALYSIS

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

```
SELECT Extract (month from date) as months,
MAX(Confirmed) as most_frequent_confirmed,
MAX(Deaths) as most_frequent_deaths,
MAX(Recovered) as most_frequent_recovered
FROM Corona_Virus
GROUP BY Extract(month from date)
ORDER BY months;
```

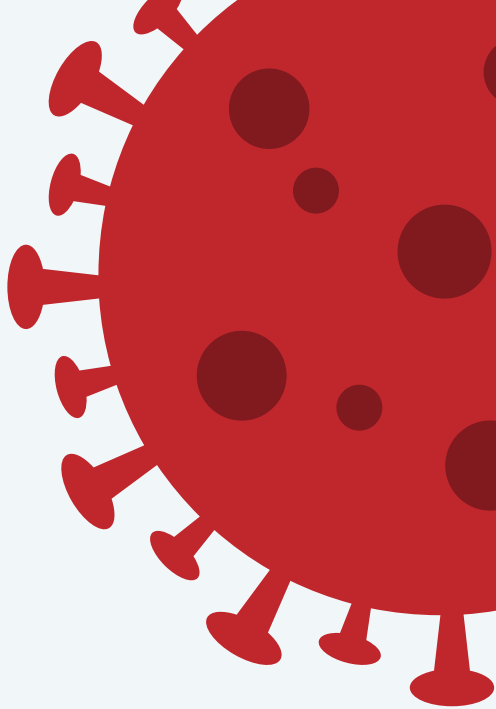
## Result

	months numeric	most_frequent_confirmed integer	most_frequent_deaths integer	most_frequent_recovered integer
1	1	300462	87090	4475
2	2	134975	98389	3907
3	3	100158	102138	3869
4	4	401993	299988	4249
5	5	414188	422436	4529
6	6	134154	231456	7374
7	7	75866	140050	1595
8	8	85687	95881	1505
9	9	97894	101468	1703
10	10	99264	388340	3351
11	11	207933	139292	2259
12	12	823225	1123456	3752





# QUERY ANALYSIS

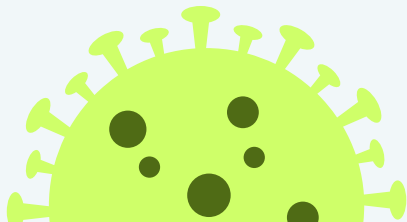
Q8. Find minimum value for confirmed, deaths, recovered per year.

```
select Extract(year from date) as years,  
Min(Confirmed) as min_confirmed,  
Min(Deaths) as min_deaths,  
Min(Recovered) as min_recovered  
from Corona_virus  
Group by Extract(year from date)  
Order by years;
```



Result

	years numeric 	min_confirmed integer 	min_deaths integer 	min_recovered integer 
1	2020	0	0	0
2	2021	0	0	0



# QUERY ANALYSIS

Q9. Find maximum value for confirmed, deaths, recovered per year.

```
SELECT Extract (year from date) as years,  
MAX(Confirmed) as max_confirmed,  
MAX(Deaths) as max_deaths,  
MAX(Recovered) as max_recovered  
FROM Corona_Virus  
GROUP BY Extract(year from date)  
ORDER BY years;
```



## Result

	years numeric	max_confirmed integer	max_deaths integer	max_recovered integer
1	2020	823225	1123456	3752
2	2021	414188	422436	7374



# QUERY ANALYSIS

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

```
SELECT Extract (month from date) as months,  
COUNT(Confirmed) as total_confirmed,  
COUNT(Deaths) as total_deaths,  
COUNT(Recovered) as total_recovered  
FROM Corona_Virus  
GROUP BY Extract(month from date)  
ORDER BY months;
```

## Result

	months numeric	total_confirmed bigint	total_deaths bigint	total_recovered bigint
1	1	6314	6314	6314
2	2	8778	8778	8778
3	3	9548	9548	9548
4	4	9240	9240	9240
5	5	9548	9548	9548
6	6	6622	6622	6622
7	7	4774	4774	4774
8	8	4774	4774	4774
9	9	4620	4620	4620
10	10	4774	4774	4774
11	11	4620	4620	4620
12	12	4774	4774	4774





# QUERY ANALYSIS

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

```
SELECT COUNT (confirmed) as total_confirmed_cases,  
AVG(Confirmed) as avg_confirmed_cases,  
variance(Confirmed) as variance_confirmed,  
stddev(Confirmed) as stddev_confirmed  
FROM Corona_Virus;
```



## Result

	total_confirmed_cases 	avg_confirmed_cases 	variance_confirmed 	stddev_confirmed 
	78386	2156.8283111780164825	157290931.69817455	12541.56815148

# QUERY ANALYSIS

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

```
SELECT Extract (month from date) as months,
COUNT(Deaths) as total_death_cases,
AVG (Deaths) as avg_death_cases,
variance(Deaths) as variance_deaths,
stddev(Deaths) as stddev_deaths
FROM Corona_Virus
GROUP BY Extract (month from date)
ORDER BY months;
```

## Result

	months numeric	total_death_cases bigint	avg_death_cases numeric	variance_deaths numeric	stddev_deaths numeric
1	1	6314	1451.4554957237884067	24495691.201646491111	4949.312194805102
2	2	8778	769.1034404192298929	12608693.194722125936	3550.872173807743
3	3	9548	840.0799120234604106	18130329.882479011847	4257.972508422173
4	4	9240	1623.2136363636363636	114714213.66168220	10710.47214934
5	5	9548	2162.9020737327188940	382019449.60484330	19545.31784354
6	6	6622	1220.1532769556025370	76059821.134484804420	8721.228189566238
7	7	4774	983.0582320904901550	24849082.939830605247	4984.885449017922
8	8	4774	1299.2947214076246334	40178838.376770789749	6338.677967586837
9	9	4620	1438.9067099567099567	57035911.879366094683	7552.212383094512
10	10	4774	1420.6430666108085463	73747150.166307459631	8587.616093323424
11	11	4620	1985.3445887445887446	50738601.254690348260	7123.103344378091
12	12	4774	2497.8850020946795140	326763170.51579010	18076.59178373

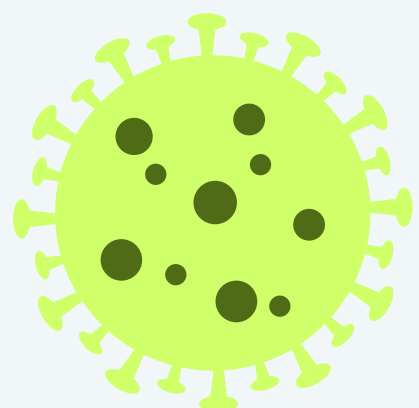
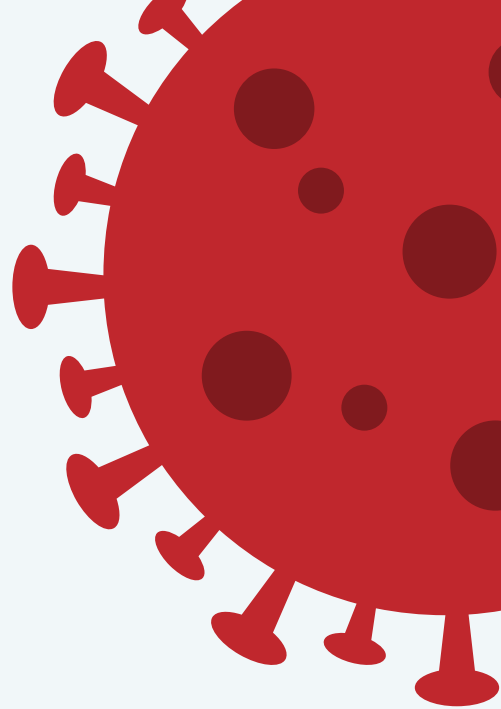
# QUERY ANALYSIS

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

```
SELECT COUNT(Recovered) as total_recovered_cases,  
AVG(recovered) as avg_recovered_cases,  
variance(recovered) as variance_recovered,  
stddev(Recovered) as stddev_recovered  
FROM Corona_Virus;
```

Result

	total_recovered_cases 	avg_recovered_cases 	variance_recovered 	stddev_recovered 
1	78386	46.5375704845252979	45892.604322956217	214.225592128850



# QUERY ANALYSIS

Q14. Find country highest number of confirmed cases.

```
SELECT Country, MAX(Confirmed) as highest_confirmed_cases
FROM Corona_Virus
GROUP BY Country
ORDER BY highest_confirmed_cases desc
limit 1;
```



## Result

	country character varying (100) 	highest_confirmed_cases integer 
1	Turkey	823225

# QUERY ANALYSIS

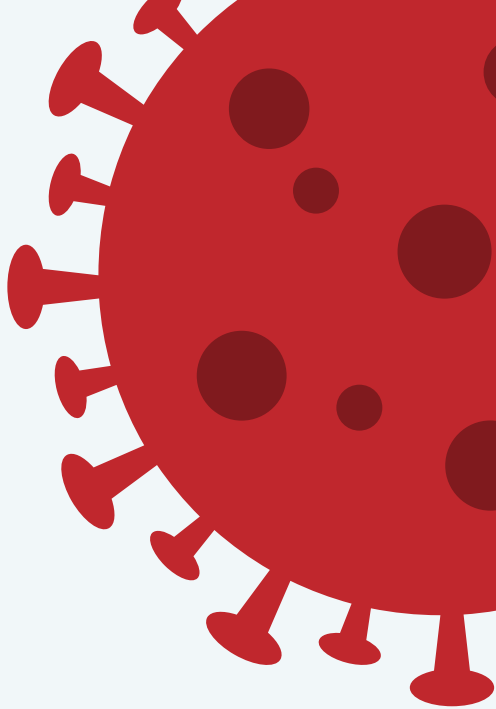
Q15. Find country having lowest number of death cases.

```
SELECT Country, MIN(Deaths) as lowest_death_cases
FROM Corona_Virus
GROUP BY Country
ORDER BY lowest_death_cases;
```

Result

	country character varying (100)	lowest_death_cases integer
104	Panama	0
105	Yemen	0
106	South Sudan	0
107	Lithuania	0
108	Bulgaria	0
109	Croatia	0
110	Tunisia	0
111	North Macedonia	0
112	Morocco	0
113	Mexico	0
114	Nepal	0
115	Tanzania	0
116	Poland	0
117	Lebanon	0
118	Costa Rica	0
119	Haiti	0
120	Samoa	0
121	Somalia	0

# QUERY ANALYSIS

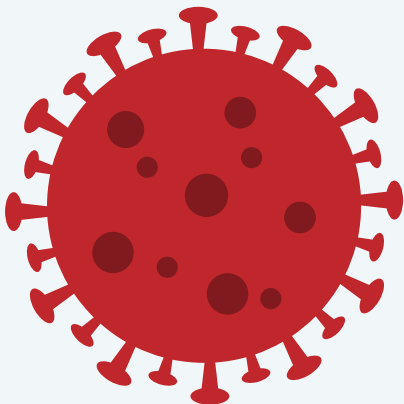


Q16. Find top 5 countries having highest recovered cases.

```
Select Country, Max(Recovered) as highest_recovered_cases
from Corona_Virus
Group by Country
Order by highest_recovered_cases desc
limit 5;
```

## Result

	country character varying (100)	highest_recovered_cases integer
1	India	7374
2	US	4475
3	Mexico	4272
4	Brazil	4249
5	Argentina	3351



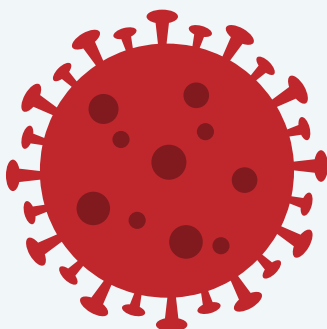
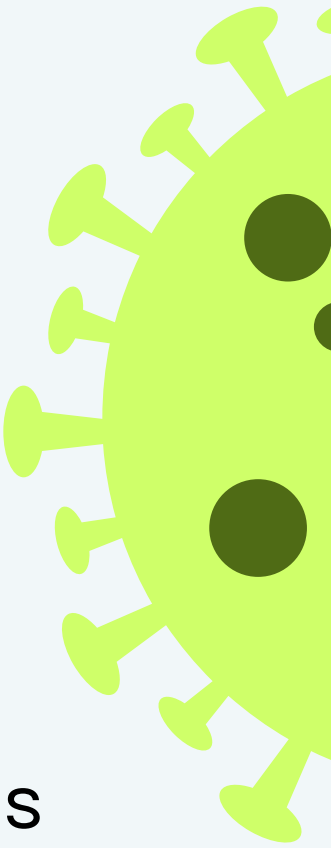
# SUMMARY

**Data Gathering:** First, we collect data from various sources like hospitals, health departments and research institutes. This includes information on confirmed cases, deaths, recoveries and demographic details.

**Data cleaning:** Next, We clean the data to remove any inconsistencies, errors or missing values, this ensures that our analysis is based on accurate information.

**Exploratory Analysis:** We use SQL queries to explore the data, looking for patterns, trends and correlations. This helps us to understand how the virus is spreading, which regions are most affected and how different factors like age and gender influence outcomes.

**Aggregation:** We aggregate the data to summarize key metrics such as total cases, deaths and allow us to compare the impact of the virus recovery rates for different countries, regions and time periods. This allows us to compare the impact of the virus across different areas and track its progression over time.





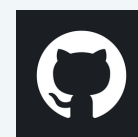
# THANK YOU

For joining me on this journey of exploring Corona Virus Analysis.  
Our knowledge and skills will continue to evolve with practice and experimentation.

## **FOLLOW ME**



<https://www.linkedin.com/in/md-shakib-6283a7239/>



<https://github.com/shaky1405>

