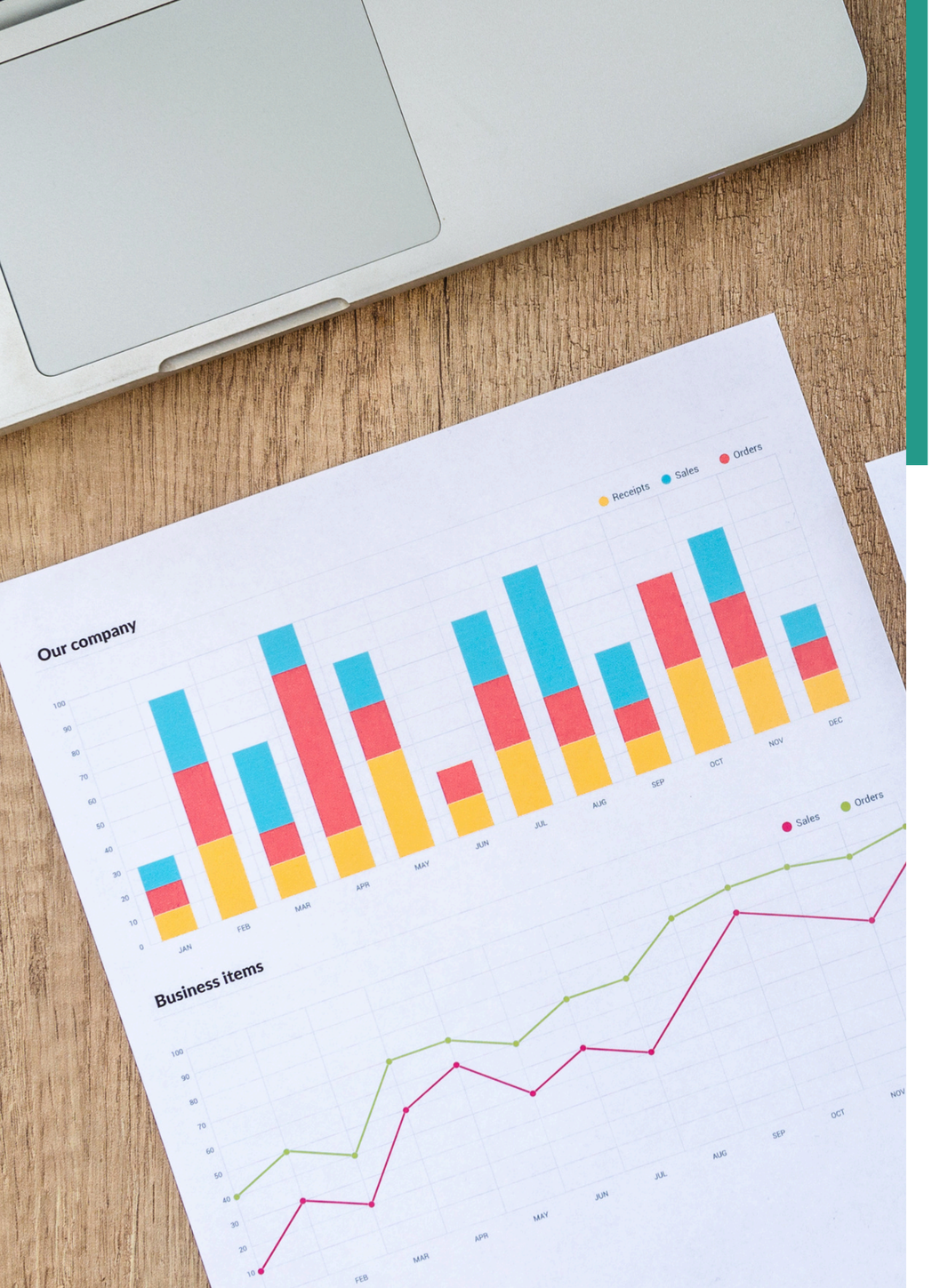
CORONA VIRUS ANALYSIS

Presented By
Prashant Pal

Data Analyst Intern- Mentorness



OVERVIEW

The Corona Virus Analysis project challenges to demonstrate SQL and data analysis skills in a real-world scenario. The project emphasizes the significant impact of the pandemic on public health, highlighting the urgent need for data-driven insights to comprehend the virus's spread. I am tasked with deriving meaningful insights from the data and presenting those findings. We are encouraged to be creative, seek guidance as needed, and utilize a reference file containing detailed questions to guide our analysis. This project offers an opportunity to delve into the world of data analysis and contribute valuable insights into the ongoing pandemic.

DATASET INFORMATION

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.

CREATING A TABLE CORONA_VIRUS AND IMPORTING DATASET

```sql
-- Creating a Table corona_virus and inserting columns in it.
CREATE TABLE corona_virus (
    Province VARCHAR(100), CountryRegion VARCHAR(100),
    Latitude FLOAT,
    Longitude FLOAT,
    Date DATE,
    Confirmed INT,
    Deaths INT,
    Recovered INT
);
SELECT * FROM corona_virus

-- Importing the datset directly in PGADMIN4
```

01. WRITE A CODE TO CHECK FOR NULL VALUES

```sql
-- 01. Write a code to check for NULL values
select * from corona_virus
where province is null
or countryregion 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;
```

Data Output | Messages | Notifications

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

02. IF NULL VALUES ARE PRESENT, UPDATE THEM WITH ZEROS FOR ALL COLUMNS

```sql
-- 02. If NULL values are present, update them with zeros for all columns
update corona_virus set
    Province = coalesce(Province, 'Not Available'),
    Countryregion = coalesce(Countryregion, 'Not Available'),
    Latitude = coalesce(Latitude, 0.0),
    Longitude = coalesce(Longitude, 0.0),
    Date = coalesce(Date, '1970-01-01'::Date),
    Confirmed = coalesce(Confirmed, 0),
    Deaths = coalesce(Deaths, 0),
    Recovered = coalesce(Recovered, 0);
```

Data Output | Messages | Notifications

```
UPDATE 78386

Query returned successfully in 850 msec.
```

03. CHECK TOTAL NUMBER OF ROWS

```
40  -- 03. Check Total number of rows
41  select count(*) as total_rows from corona_virus;
42
```

Data Output | Messages | Notifications

	total_rows bigint
1	78386

04. CHECK WHAT IS START_DATE AND WHAT IS THE END_DATE.

```
44  -- 04. Check what is start_date and what is the end_date
45  select min(date) as start_date, max(date) as end_date from corona_virus;
46
47
```

Data Output | Messages | Notifications

	start_date date	end_date date
1	2020-01-22	2021-06-13

05. NUMBER OF MONTH PRESENT IN DATASET

```
48  -- 05. Number of month present in Dataset
49  select extract(month from date) as month_number,
50         count(*) as month_count from corona_virus group by month_number order by month_number;
```

Data Output | Messages | Notifications

	month_number numeric	month_count bigint
1	1	6314
2	2	8778
3	3	9548
4	4	9240
5	5	9548
6	6	6622
7	7	4774
8	8	4774
9	9	4620
10	10	4774
11	11	4620
12	12	4774

06. FIND MONTHLY AVERAGE FOR CONFIRMED, DEATHS AND RECOVERED

```sql
-- 06. Find monthly average for confirmed, deaths and recovered
select extract (year from date) as year,
extract (month from date) as month,
round(avg(confirmed), 2) as avg_confirmed,
round(avg(deaths), 2) as avg_deaths,
round(avg(recovered), 2) as avg_recovered
from corona_virus
group by year, month
order by year, month;
```

Data Output | Messages | Notifications

	year numeric	month numeric	avg_confirmed numeric	avg_deaths numeric	avg_recovered numeric
1	2020	1	4.15	0.12	0.09
2	2020	2	15.30	0.59	7.03
3	2020	3	161.13	8.66	27.87
4	2020	4	505.80	41.52	171.64
5	2020	5	574.85	30.28	318.30
6	2020	6	859.23	29.82	548.79
7	2020	7	1432.36	35.11	983.06
8	2020	8	1611.84	37.54	1299.29
9	2020	9	1784.59	34.78	1438.91
10	2020	10	2412.20	36.76	1420.64
11	2020	11	3592.19	56.76	1985.34
12	2020	12	4050.44	71.22	2497.89
13	2021	1	3911.23	84.18	1919.64
14	2021	2	2433.36	69.16	1558.39
15	2021	3	2916.80	59.20	1652.29
16	2021	4	4699.36	78.44	3074.79
17	2021	5	4005.25	76.78	4007.51
18	2021	6	2508.63	66.26	2769.45

07. FIND MOST FREQUENT VALUE FOR CONFIRMED, DEATHS, RECOVERED EACH MONTH

```sql
-- 07. Find most frequent value for confirmed, deaths, recovered each month
select extract (year from date) as year,
extract (month from date) as month,
max(confirmed) as most_confirmed,
max(deaths) as most_deaths,
max(recovered) as most_recovered
from corona_virus
group by year, month
order by year, month;
```

Data Output | Messages | Notifications

	year numeric	month numeric	most_confirmed integer	most_deaths integer	most_recovered integer
1	2020	1	2131	49	51
2	2020	2	14840	242	3418
3	2020	3	26314	1085	4289
4	2020	4	50740	2607	33227
5	2020	5	34907	2309	51717
6	2020	6	54771	2003	94305
7	2020	7	75866	1595	140050
8	2020	8	85687	1505	95881
9	2020	9	97894	1703	101468
10	2020	10	99264	3351	388340
11	2020	11	207933	2259	139292
12	2020	12	823225	3752	1123456
13	2021	1	300462	4475	87090
14	2021	2	134975	3907	98389
15	2021	3	100158	3869	102138
16	2021	4	401993	4249	299988
17	2021	5	414188	4529	422436
18	2021	6	134154	7374	231456

08. FIND MINIMUM VALUES FOR CONFIRMED, DEATHS, RECOVERED PER YEAR

```
75  -- 08. Find minimum values for confirmed, deaths, recovered per year
76  select extract (year from date) as year,
77  min (confirmed) as min_confirmed,
78  min (deaths) as min_deaths,
79  min (recovered) as min_recovered
80  from corona_virus
81  group by year
82  order by year;
83
```

Data Output | Messages | Notifications

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

09. FIND MAXIMUM VALUES FOR CONFIRMED, DEATHS, RECOVERED PER YEAR

```
85  -- 09. Find maximum values for confirmed, deaths, recovered per year
86  select extract (year from date) as year,
87  max (confirmed) as max_confirmed,
88  max (deaths) as max_deaths,
89  max (recovered) as max_recovered
90  from corona_virus
91  group by year
92  order by year;
93
```

Data Output | Messages | Notifications

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

10. FIND TOTAL NUMBER OF CASES OF CONFIRMED, DEATHS, RECOVERED EACH MONTH

```
select extract (year from date) as year,
extract (month from date) as month,
sum(confirmed) as total_confirmed,
sum(deaths) as total_deaths,
sum(recovered) as total_recovered
from corona_virus
group by year, month
order by year, month;
```

Data Output | Messages | Notifications

	year numeric	month numeric	total_confirmed bigint	total_deaths bigint	total_recovered bigint
1	2020	1	6384	190	143
2	2020	2	68312	2651	31405
3	2020	3	769236	41346	133070
4	2020	4	2336798	191833	792987
5	2020	5	2744333	144561	1519547
6	2020	6	3969634	137757	2535417
7	2020	7	6838092	167613	4693120
8	2020	8	7694938	179200	6202833
9	2020	9	8244794	160671	6647749
10	2020	10	11515841	175484	6782150
11	2020	11	16595938	262247	9172292
12	2020	12	19336799	339996	11924903
13	2021	1	18672205	401893	9164347
14	2021	2	10492664	298239	6719785
15	2021	3	13924790	282620	7888013
16	2021	4	21711021	362387	14205507
17	2021	5	19121083	366549	19131842
18	2021	6	5022282	132657	5544438

11. CHECK HOW CORONA VIRUS SPREAD OUT WITH RESPECT TO CONFIRMED CASES

```sql
select extract(year from date) as year,
extract(month from date) as month,
sum(confirmed) as total_confirmed,
round(avg(confirmed), 2) as avg_confirmed,
round(variance(confirmed), 2) as variance_confirmed,
round(STDDEV(confirmed), 2) as standard_dev_confirmed
from corona_virus
group by year, month
order by year, month;
```

Data Output | Messages | Notifications

	year numeric	month numeric	total_confirmed bigint	avg_confirmed numeric	variance_confirmed numeric	standard_dev_confirmed numeric
1	2020	1	6384	4.15	4836.05	69.54
2	2020	2	68312	15.30	78507.03	280.19
3	2020	3	769236	161.13	1026629.22	1013.23
4	2020	4	2336798	505.80	7013581.36	2648.32
5	2020	5	2744333	574.85	6064850.73	2462.69
6	2020	6	3969634	859.23	13782194.73	3712.44
7	2020	7	6838092	1432.36	46923851.93	6850.10
8	2020	8	7694938	1611.84	54419982.40	7376.99
9	2020	9	8244794	1784.59	69329705.03	8326.45
10	2020	10	11515841	2412.20	69002612.88	8306.78
11	2020	11	16595938	3592.19	195858271.38	13994.94
12	2020	12	19336799	4050.44	459981798.11	21447.19
13	2021	1	18672205	3911.23	316370963.72	17786.82
14	2021	2	10492664	2433.36	79606383.04	8922.24
15	2021	3	13924790	2916.80	83742806.92	9151.11
16	2021	4	21711021	4699.36	501121674.28	22385.75
17	2021	5	19121083	4005.25	628779318.45	25075.47
18	2021	6	5022282	2508.63	110988215.34	10535.09

12. CHECK HOW CORONA VIRUS SPREAD OUT WITH RESPECT TO DEATH CASE PER MONTH

```sql
select extract (year from date) as year,
extract (month from date) as month,
sum(deaths) as total_deaths,
round(avg(deaths), 2) as avg_deaths,
round(variance(deaths), 2) as variance_deaths,
round(STDDEV(deaths), 2) as standard_dev_deaths
from corona_virus
group by year, month
order by year, month;
```

Data Output | Messages | Notifications

	year numeric	month numeric	total_deaths bigint	avg_deaths numeric	variance_deaths numeric	standard_dev_deaths numeric
1	2020	1	190	0.12	4.25	2.06
2	2020	2	2651	0.59	68.34	8.27
3	2020	3	41346	8.66	3901.61	62.46
4	2020	4	191833	41.52	40513.04	201.28
5	2020	5	144561	30.28	20689.25	143.84
6	2020	6	137757	29.82	16933.11	130.13
7	2020	7	167613	35.11	21144.58	145.41
8	2020	8	179200	37.54	23277.87	152.57
9	2020	9	160671	34.78	20107.12	141.80
10	2020	10	175484	36.76	17583.75	132.60
11	2020	11	262247	56.76	27779.81	166.67
12	2020	12	339996	71.22	65359.06	255.65
13	2021	1	401893	84.18	102779.96	320.59
14	2021	2	298239	69.16	68494.76	261.72
15	2021	3	282620	59.20	54397.36	233.23
16	2021	4	362387	78.44	94631.95	307.62
17	2021	5	366549	76.78	131797.08	363.04
18	2021	6	132657	66.26	113020.13	336.18

13. CHECK HOW CORONA VIRUS SPREAD OUT WITH RESPECT TO RECOVERED CASES

```
select extract(year from date) as year,
extract(month from date) as month,
sum(recovered) as total_recovered,
round(avg(recovered), 2) as avg_recovered,
round(variance(recovered), 2) variance_recovered,
round(STDDEV(recovered), 2) as standard_dev_recovered
from corona_virus
group by year, month
order by year, month;
```

Data Output | Messages | Notifications

	year numeric	month numeric	total_recovered bigint	avg_recovered numeric	variance_recovered numeric	standard_dev_recovered numeric
1	2020	1	143	0.09	2.64	1.62
2	2020	2	31405	7.03	12449.45	111.58
3	2020	3	133070	27.87	40121.59	200.30
4	2020	4	792987	171.64	770059.71	877.53
5	2020	5	1519547	318.30	1978620.88	1406.63
6	2020	6	2535417	548.79	6531586.26	2555.70
7	2020	7	4693120	983.06	24849082.94	4984.89
8	2020	8	6202833	1299.29	40178838.38	6338.68
9	2020	9	6647749	1438.91	57035911.88	7552.21
10	2020	10	6782150	1420.64	73747150.17	8587.62
11	2020	11	9172292	1985.34	50738601.25	7123.10
12	2020	12	11924903	2497.89	326763170.52	18076.59
13	2021	1	9164347	1919.64	31500298.42	5612.51
14	2021	2	6719785	1558.39	24433077.90	4942.98
15	2021	3	7888013	1652.29	34904703.06	5908.02
16	2021	4	14205507	3074.79	224468171.33	14982.26
17	2021	5	19131842	4007.51	755333749.97	27483.34
18	2021	6	5544438	2769.45	233150866.36	15269.28

14. FIND THE COUNTRY HAVE HIGHEST NUMBER OF CONFIRMED CASES

```
142  -- 14. Find the Country have highest number of confirmed cases
143  select countryregion,
144  sum(confirmed) as total_confirmed_cases from corona_virus
145  group by countryregion
146  order by total_confirmed_cases desc
147  limit 1;
148
```

Data Output | Messages | Notifications

	countryregion character varying (100)	total_confirmed_cases bigint
1	US	33461982

15. FIND THE COUNTRY HAVE LOWEST NUMBER OF DEATH CASES

```
150  -- 15. Find the Country have lowest number of death cases
151
152  select countryregion,
153  sum(deaths) as lowest_deaths_cases from corona_virus
154  group by countryregion
155  order by lowest_deaths_cases Asc
156  limit 7;
```

Data Output | Messages | Notifications

	countryregion character varying (100)	lowest_deaths_cases bigint
1	Marshall Islands	0
2	Dominica	0
3	Kiribati	0
4	Samoa	0
5	Bhutan	1
6	Mauritius	18
7	Tanzania	21

16. FIND TOP 5 COUNTRIES HAVE HIGHEST RECOVERED CASES

```
159  -- 16. Find top 5 countries have highest recovered cases
160  select countryregion,
161  sum(recovered) as total_recovered_cases
162  from corona_virus
163  group by countryregion
164  order by total_recovered_cases desc
165  limit 5;
166
```

Data Output | Messages | Notifications

	countryregion character varying (100)	total_recovered_cases bigint
1	India	28089649
2	Brazil	15400169
3	US	6303715
4	Turkey	5202251
5	Russia	4745756

INSIGHTS

- COVID-19 Pandemic started from 22 January 2020 stayed till 13 June 2021.
- India has the Highest number of recovered cases highlighting the extensive recovery efforts and medical interventions undertaken.
- US reported the highest number of confirmed cases , showing it was heavily affected.
- Samoa, Kiribati, Dominica, and The Marshall Islands have the lowest number of Deaths.
- Peak confirmed cases recorded in April 2021.
- Peak death recorded in January 2021.
- The variance and standard deviation in confirmed cases, deaths, & recoveries show significant spread, with the highest variance seen in confirmed cases in December 2020, reflecting the global surge during this period.

THANK YOU

BY- PRASHANT PAL