Corona Virus Data Analysis with SQL

Adeoluwa Dotun Ogundele

Data Analysis Intern

MENTORNESS

Table of Contents

- Overview
- Description of Dataset
- Data Cleaning and Analysis (SQL Queries and Results)
- Conclusion
- Recommendation

Overview

The **coronavirus** pandemic has profoundly impacted global health, economies, and daily life. Understanding the spread, impact, and trends of the virus is crucial for formulating effective responses and policies. This analysis focuses on examining the coronavirus data, aiming to uncover patterns and meaningful insights.

This analysis utilizes a dataset provided by Mentorness, covering various aspects of the pandemic, including the number of confirmed cases, deaths, and recoveries across different regions and time periods. Through a systematic data cleaning and analysis process, this study aims to provide valuable insights and actionable recommendations.

Description of Dataset

Source of the dataset: The dataset was provided by **Mentorness.**

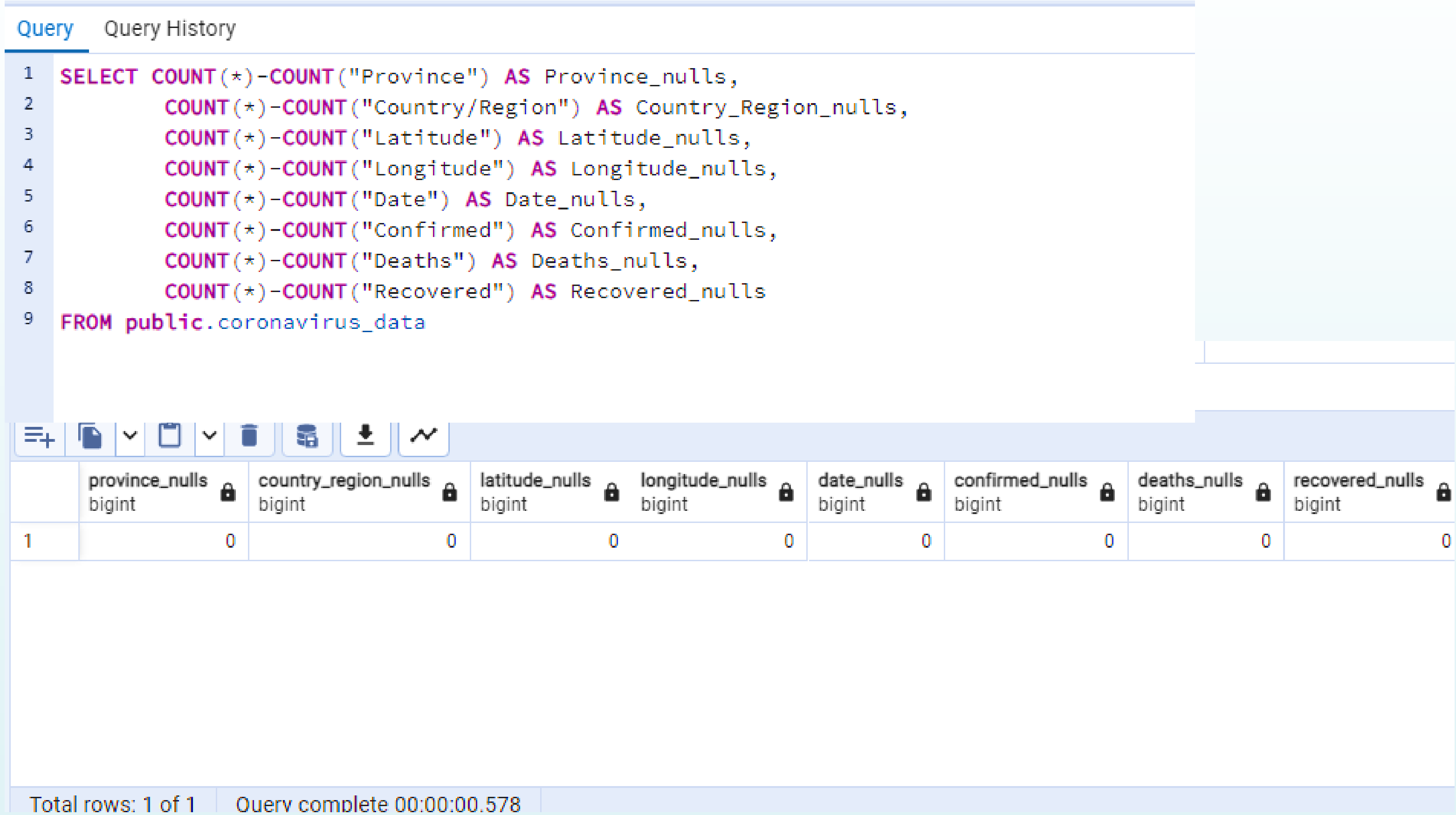
Key variables and their descriptions:

- **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.

Time period covered by the data: 22-01-2020 to 13-06-2021

Data Cleaning and Analysis (SQL Queries And Results)

1. Check for missing values



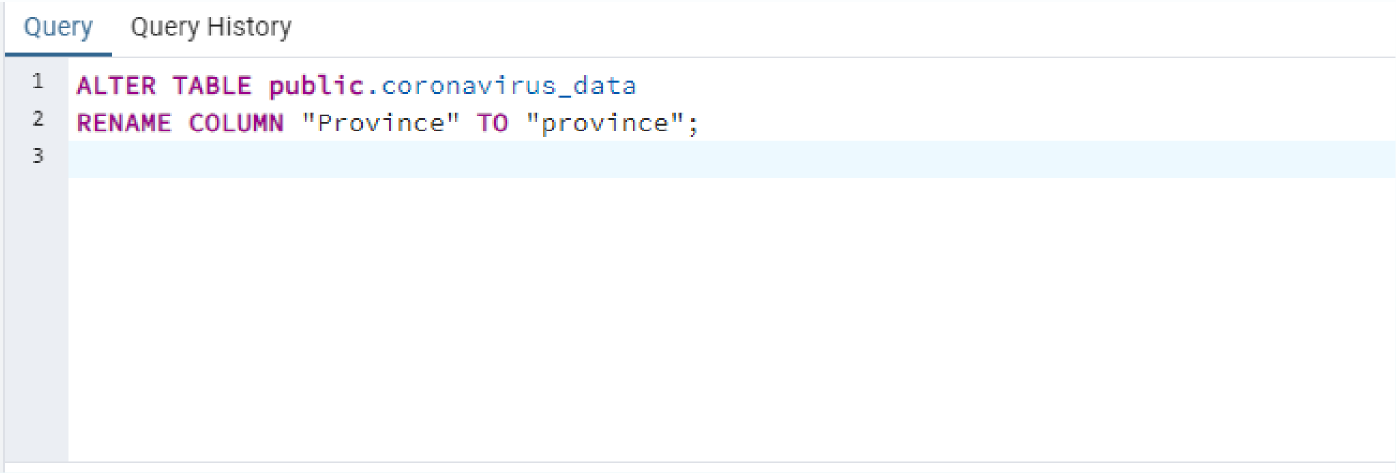
```sql
SELECT COUNT(*)-COUNT("Province") AS Province_nulls,
        COUNT(*)-COUNT("Country/Region") AS Country_Region_nulls,
        COUNT(*)-COUNT("Latitude") AS Latitude_nulls,
        COUNT(*)-COUNT("Longitude") AS Longitude_nulls,
        COUNT(*)-COUNT("Date") AS Date_nulls,
        COUNT(*)-COUNT("Confirmed") AS Confirmed_nulls,
        COUNT(*)-COUNT("Deaths") AS Deaths_nulls,
        COUNT(*)-COUNT("Recovered") AS Recovered_nulls
FROM public.coronavirus_data
```

	province_nulls bigint	country_region_nulls bigint	latitude_nulls bigint	longitude_nulls bigint	date_nulls bigint	confirmed_nulls bigint	deaths_nulls bigint	recovered_nulls bigint
1	0	0	0	0	0	0	0	0

Total rows: 1 of 1 Query complete 00:00:00.578

No missing value found.

2. Rename the Columns



```sql
ALTER TABLE public.coronavirus_data
RENAME COLUMN "Province" TO "province";
```

I did this for each of the columns.

3. Convert ‘date’ to date data type

Step 1: Add a new DATE column

```
ALTER TABLE public.coronavirus_data
ADD COLUMN new_date Date;
```

Step 2: Convert and update the new column with the correct date format

```
UPDATE public.coronavirus_data
SET new_date = TO_DATE("Date", 'DD-MM-YYYY');
```

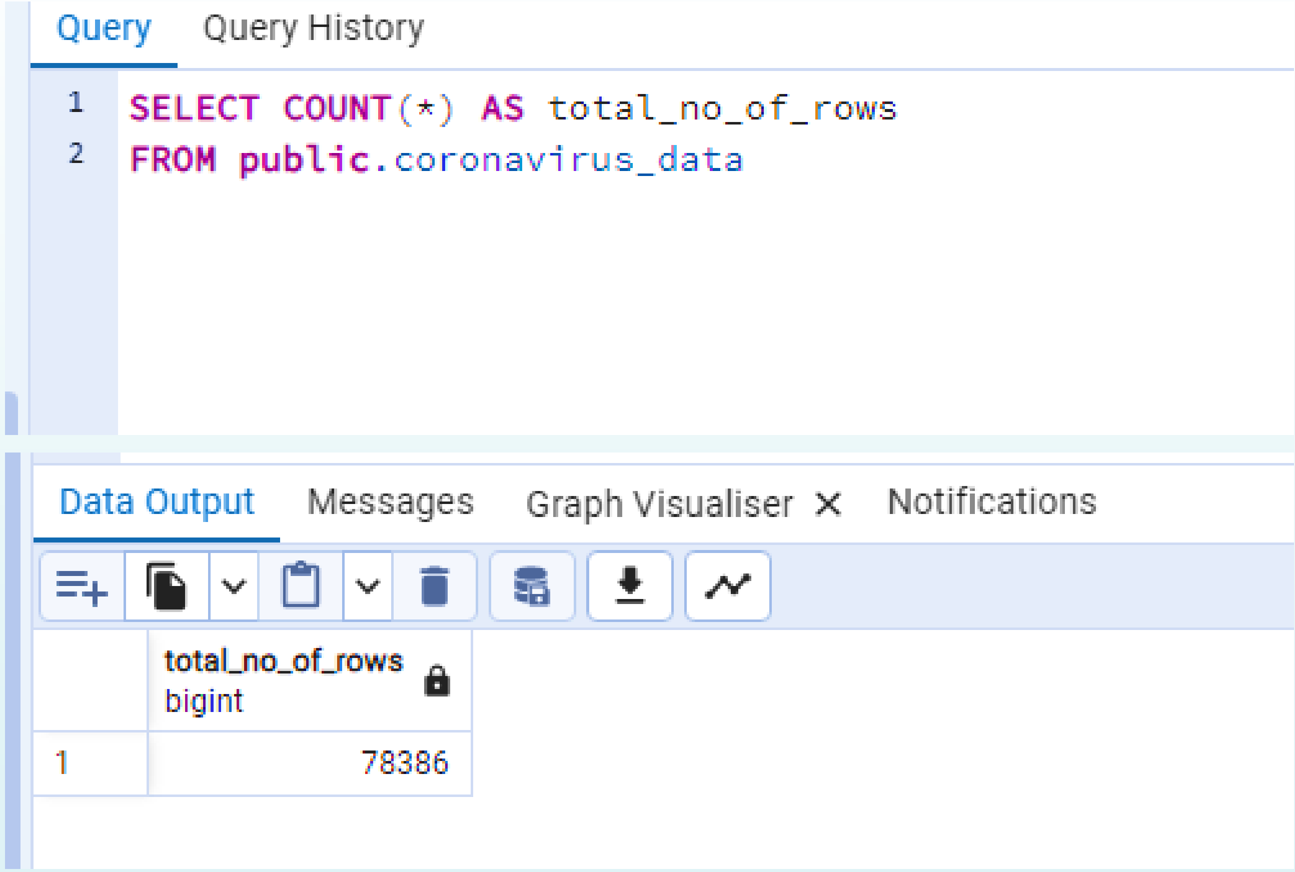
Step 3: Drop the old string-formatted date column

```
ALTER TABLE public.coronavirus_data
DROP COLUMN "Date";
```

Step 4: Rename the new DATE column to the original column name

```
ALTER TABLE public.coronavirus_data
RENAME COLUMN new_date TO date;
```

4. Total Number of Rows

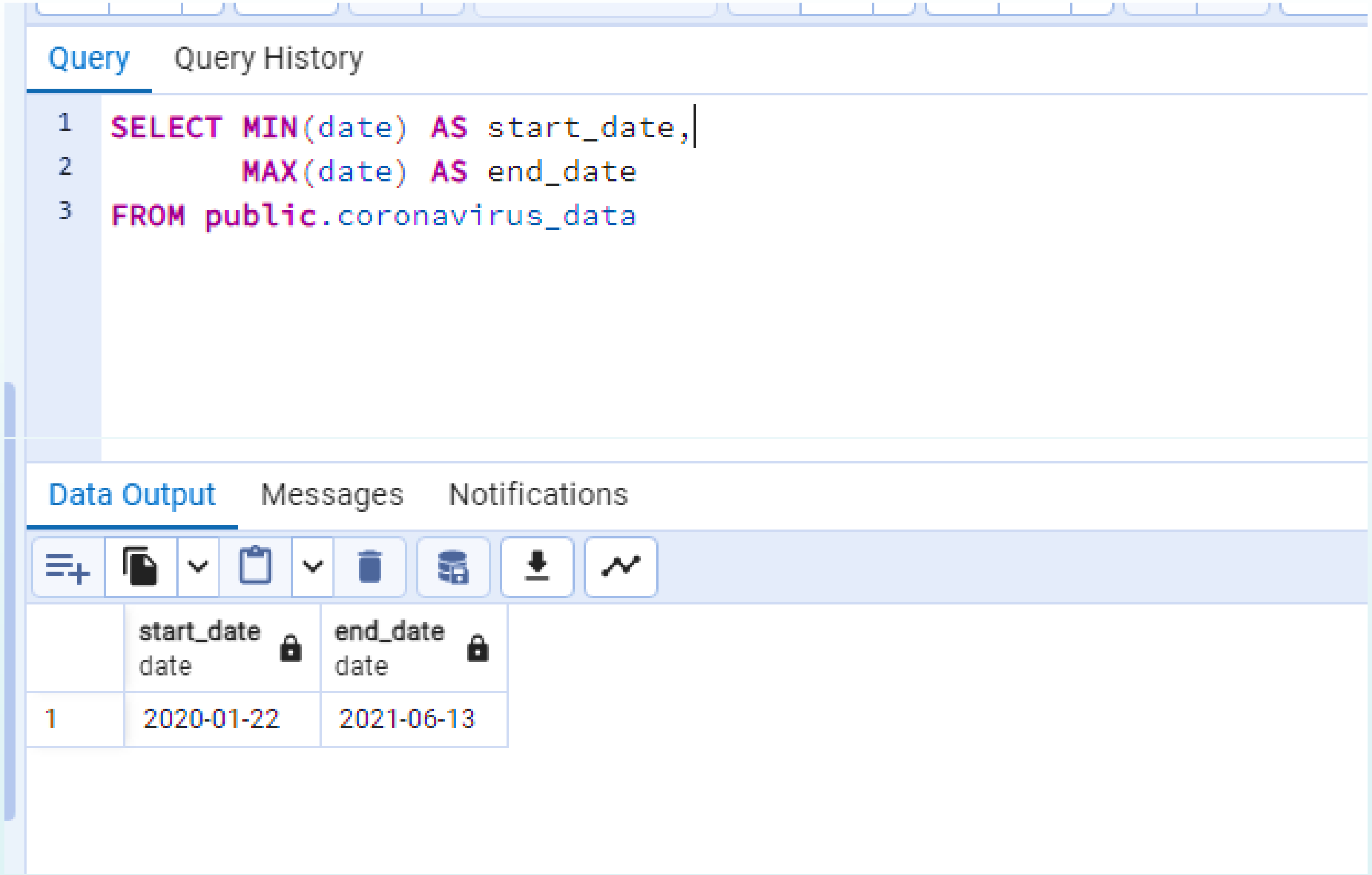


```sql
SELECT COUNT(*) AS total_no_of_rows
FROM public.coronavirus_data
```

	total_no_of_rows bigint
1	78386

There are 78386 rows.

5. Start Date and End Date



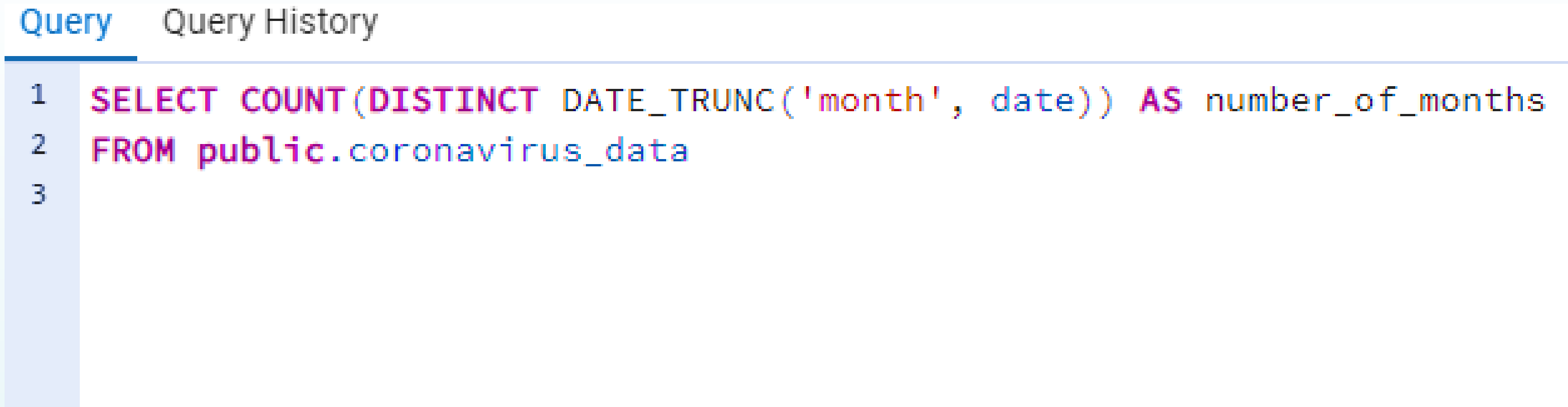
```sql
SELECT MIN(date) AS start_date,
       MAX(date) AS end_date
FROM public.coronavirus_data
```

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

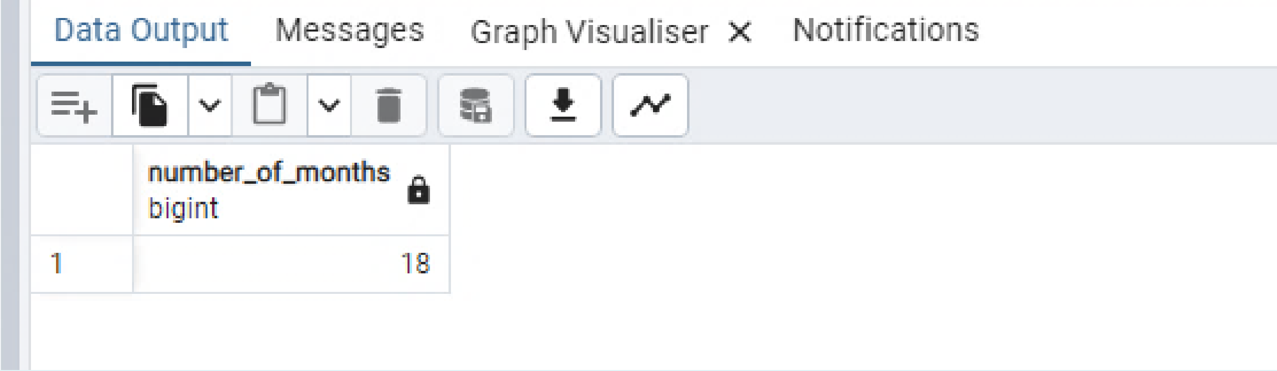
Start Date: 22 January, 2020.

End Date: 13 June, 2021.

6. Number of Months



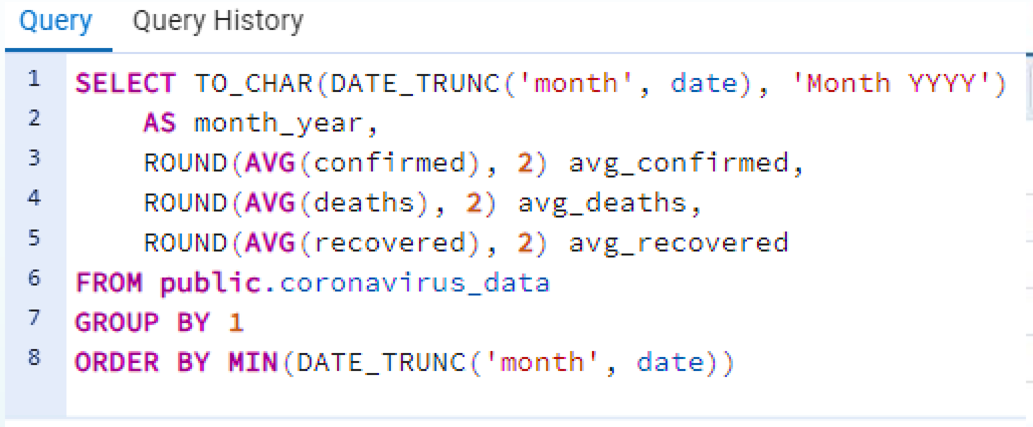
```sql
SELECT COUNT(DISTINCT DATE_TRUNC('month', date)) AS number_of_months
FROM public.coronavirus_data

```



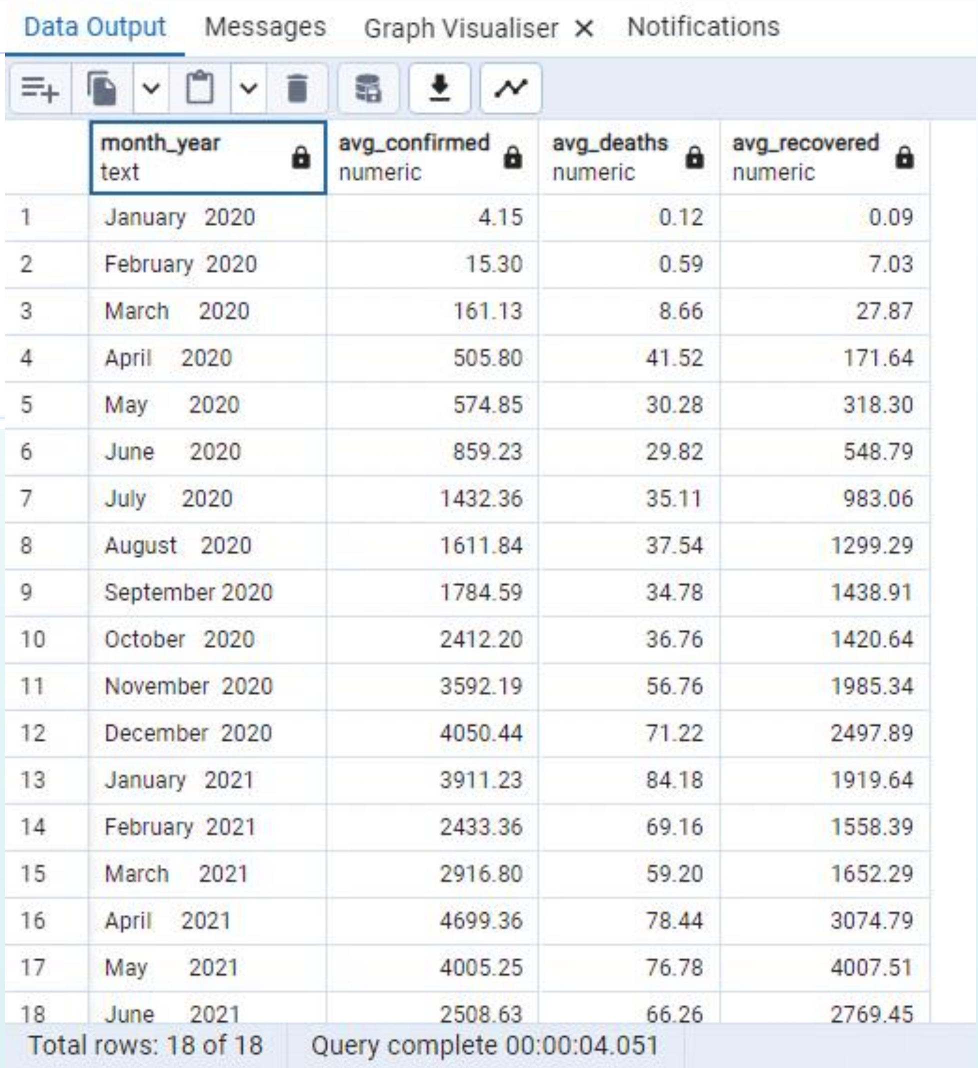
	number_of_months bigint
1	18

The dataset covered a period of 18 months.

7. Monthly Average: Confirmed Cases, Deaths and Recovered Cases



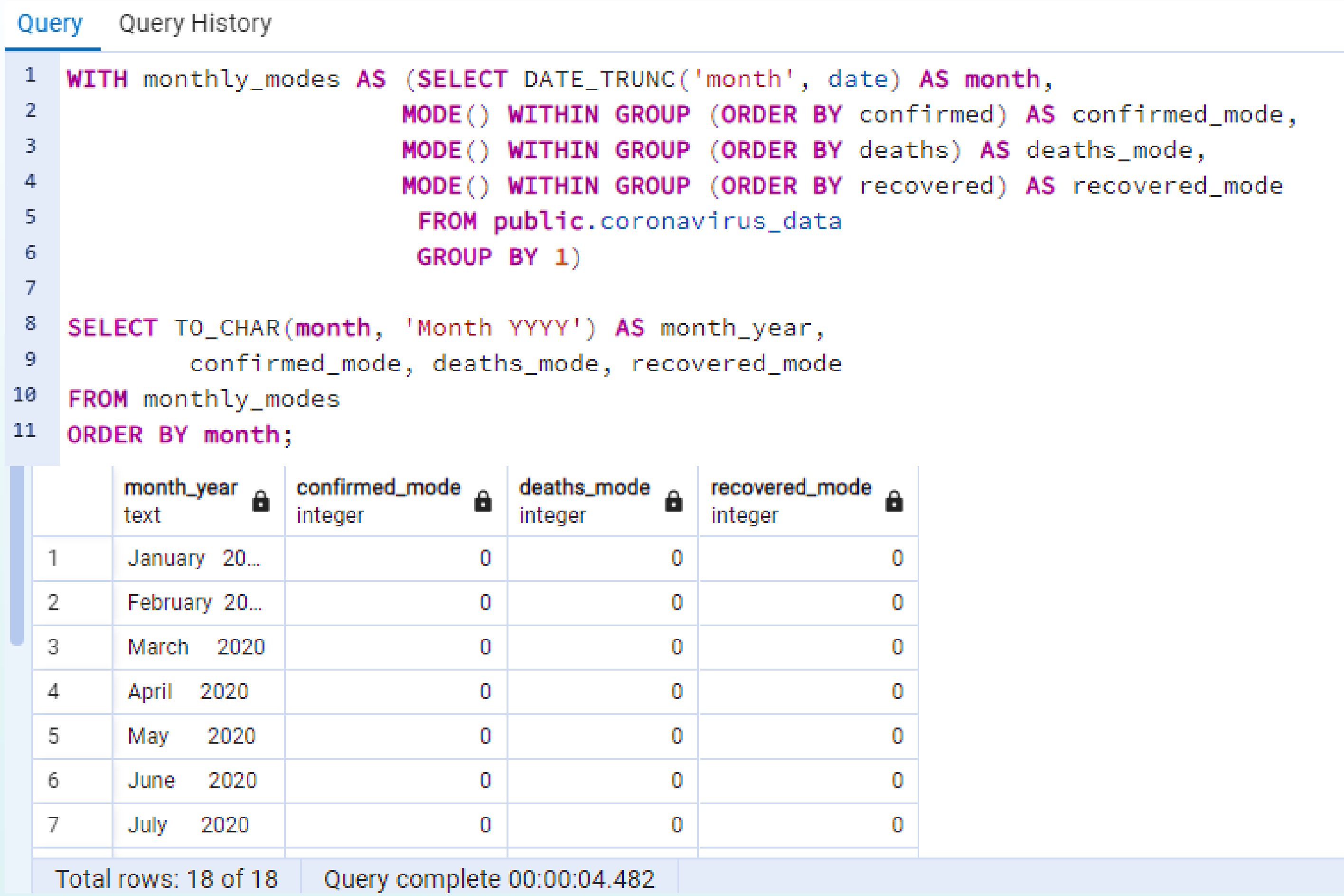
```sql
SELECT TO_CHAR(DATE_TRUNC('month', date), 'Month YYYY')
    AS month_year,
    ROUND(AVG(confirmed), 2) avg_confirmed,
    ROUND(AVG(deaths), 2) avg_deaths,
    ROUND(AVG(recovered), 2) avg_recovered
FROM public.coronavirus_data
GROUP BY 1
ORDER BY MIN(DATE_TRUNC('month', date))
```



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

Total rows: 18 of 18 Query complete 00:00:04.051

8. Most Frequent Value: Confirmed, Deaths and Recovered Cases



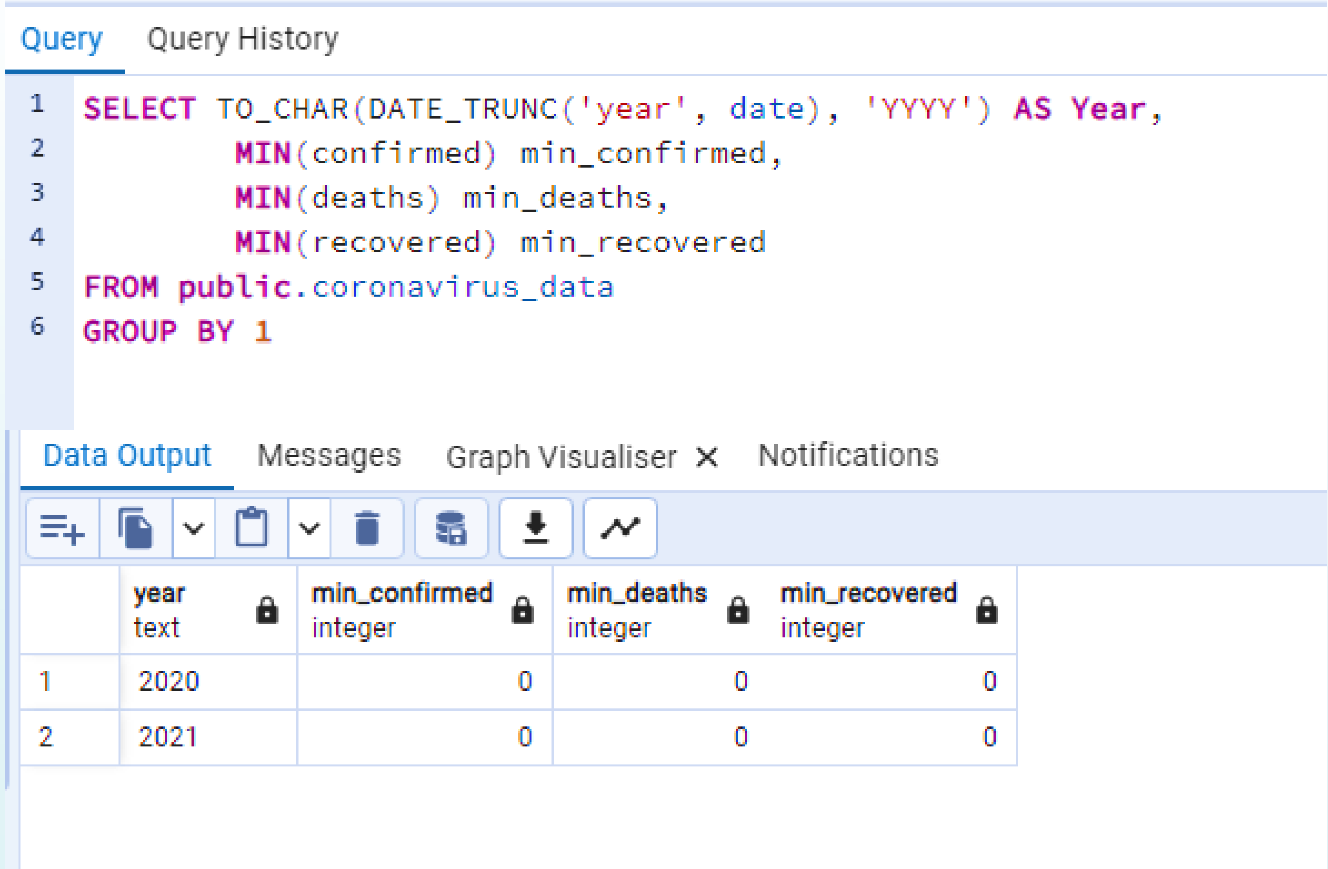
```
WITH monthly_modes AS (SELECT DATE_TRUNC('month', date) AS month,
                        MODE() WITHIN GROUP (ORDER BY confirmed) AS confirmed_mode,
                        MODE() WITHIN GROUP (ORDER BY deaths) AS deaths_mode,
                        MODE() WITHIN GROUP (ORDER BY recovered) AS recovered_mode
                         FROM public.coronavirus_data
                         GROUP BY 1)

SELECT TO_CHAR(month, 'Month YYYY') AS month_year,
        confirmed_mode, deaths_mode, recovered_mode
FROM monthly_modes
ORDER BY month;
```

	month_year text	confirmed_mode integer	deaths_mode integer	recovered_mode integer
1	January 20...	0	0	0
2	February 20...	0	0	0
3	March 2020	0	0	0
4	April 2020	0	0	0
5	May 2020	0	0	0
6	June 2020	0	0	0
7	July 2020	0	0	0

Total rows: 18 of 18 Query complete 00:00:04.482

There mode is 0 (zero) for all the categories.

9. Minimum Value Per Year

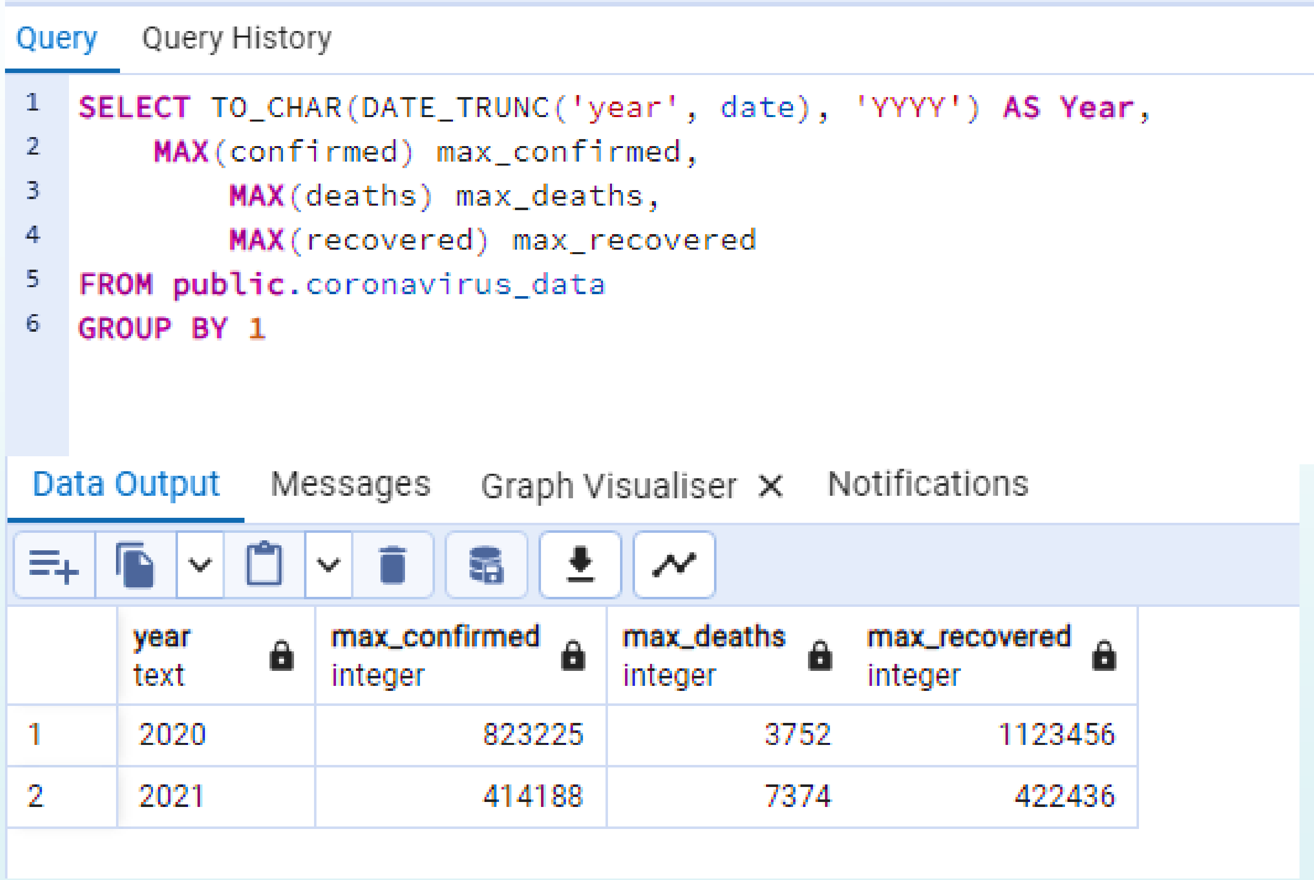


```
SELECT TO_CHAR(DATE_TRUNC('year', date), 'YYYY') AS Year,
        MIN(confirmed) min_confirmed,
        MIN(deaths) min_deaths,
        MIN(recovered) min_recovered
FROM public.coronavirus_data
GROUP BY 1
```

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

There minimum value recorded per year is 0 (zero) for all the categories.

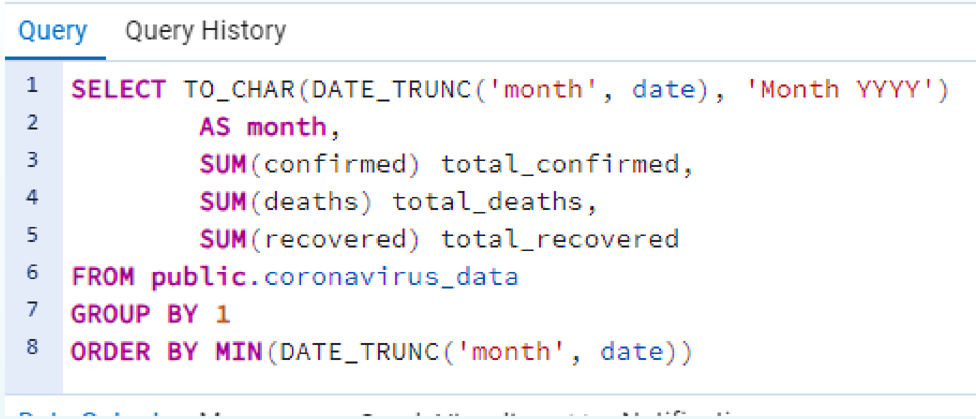
10. Maximum Value Per Year



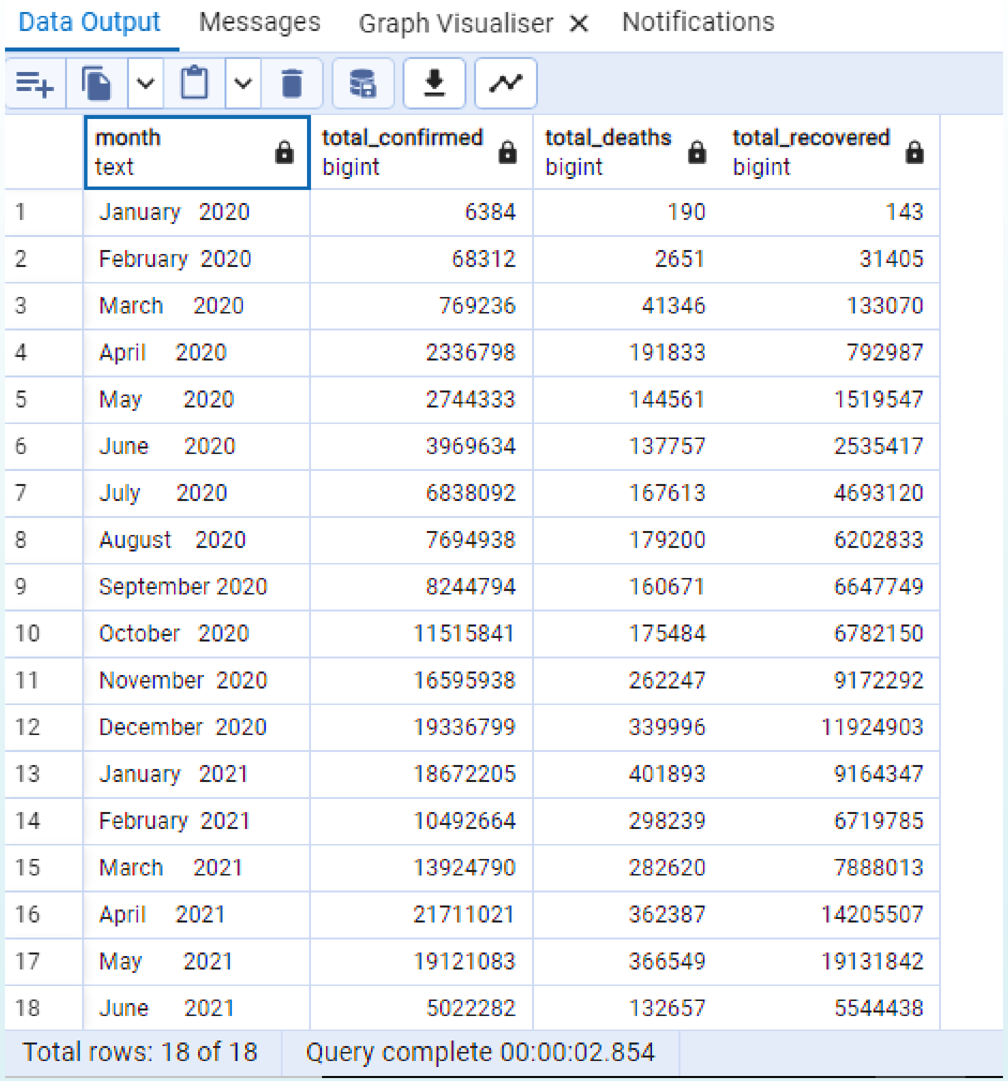
```sql
SELECT TO_CHAR(DATE_TRUNC('year', date), 'YYYY') AS Year,
    MAX(confirmed) max_confirmed,
        MAX(deaths) max_deaths,
        MAX(recovered) max_recovered
FROM public.coronavirus_data
GROUP BY 1
```

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

11. Total Number of Cases by Month



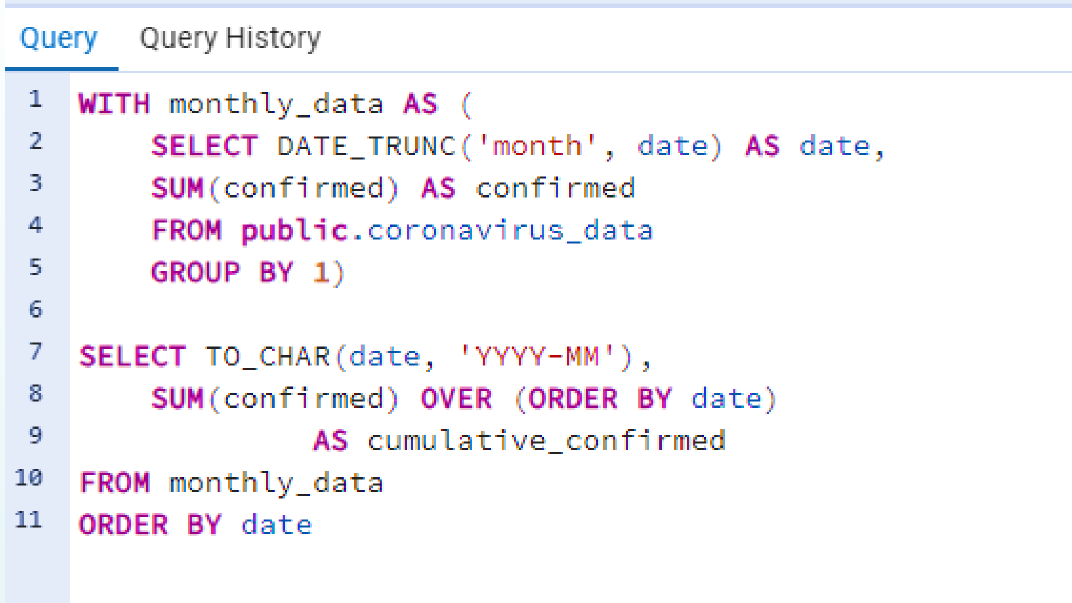
```sql
SELECT TO_CHAR(DATE_TRUNC('month', date), 'Month YYYY')
        AS month,
        SUM(confirmed) total_confirmed,
        SUM(deaths) total_deaths,
        SUM(recovered) total_recovered
FROM public.coronavirus_data
GROUP BY 1
ORDER BY MIN(DATE_TRUNC('month', date))
```



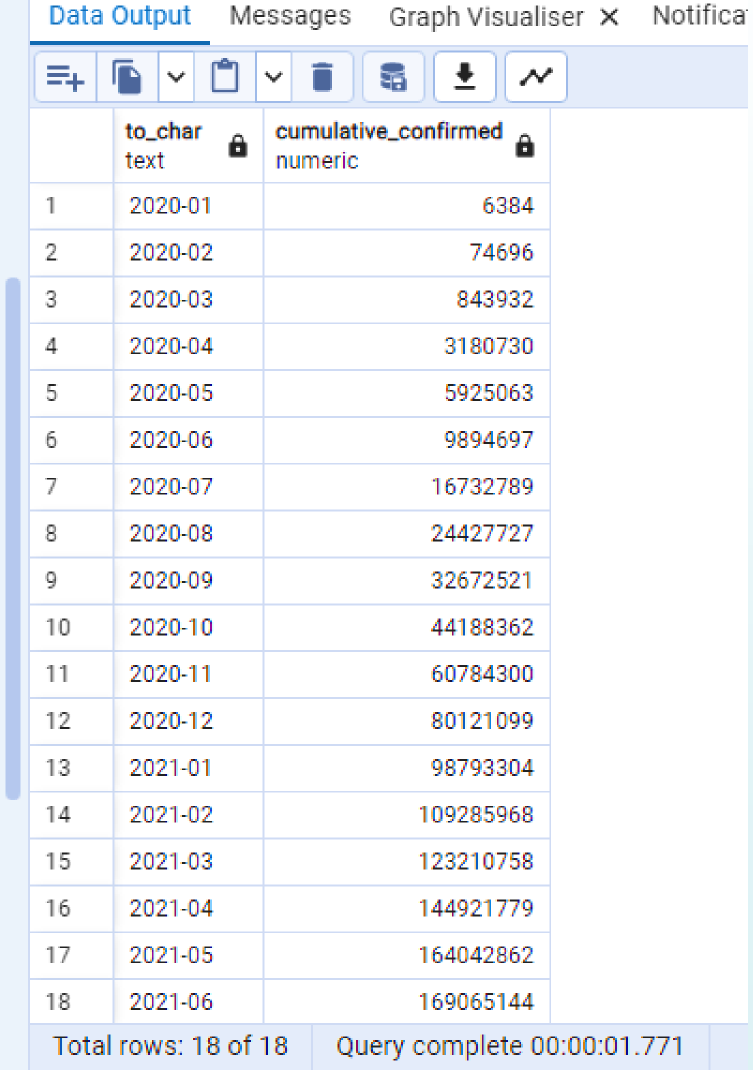
	month text	total_confirmed bigint	total_deaths bigint	total_recovered bigint
1	January 2020	6384	190	143
2	February 2020	68312	2651	31405
3	March 2020	769236	41346	133070
4	April 2020	2336798	191833	792987
5	May 2020	2744333	144561	1519547
6	June 2020	3969634	137757	2535417
7	July 2020	6838092	167613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

Total rows: 18 of 18   Query complete 00:00:02.854

12a. COVID-19 Spread Analysis: Confirmed Cases



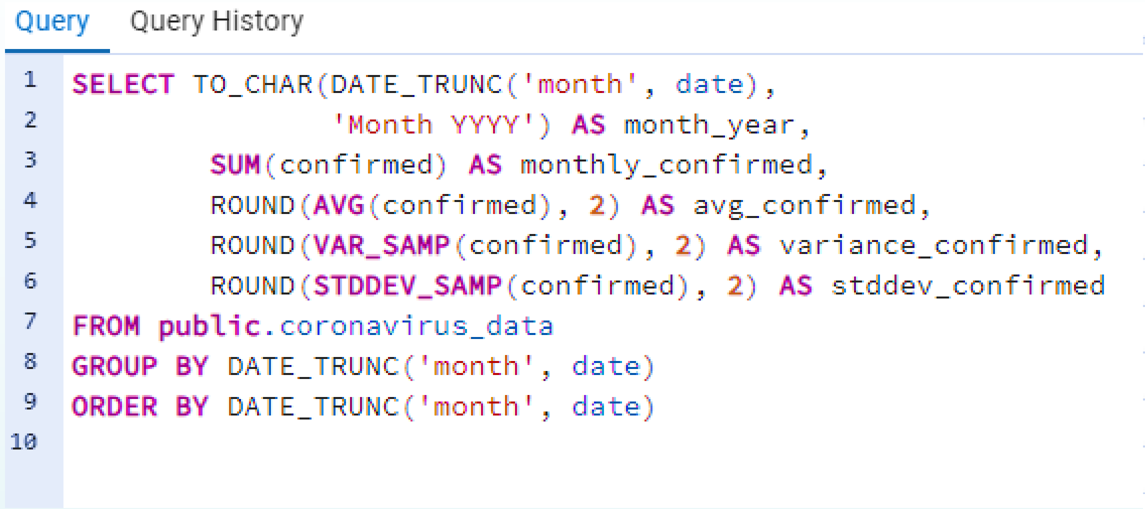
```sql
WITH monthly_data AS (
    SELECT DATE_TRUNC('month', date) AS date,
    SUM(confirmed) AS confirmed
    FROM public.coronavirus_data
    GROUP BY 1)

SELECT TO_CHAR(date, 'YYYY-MM'),
    SUM(confirmed) OVER (ORDER BY date)
              AS cumulative_confirmed
FROM monthly_data
ORDER BY date
```



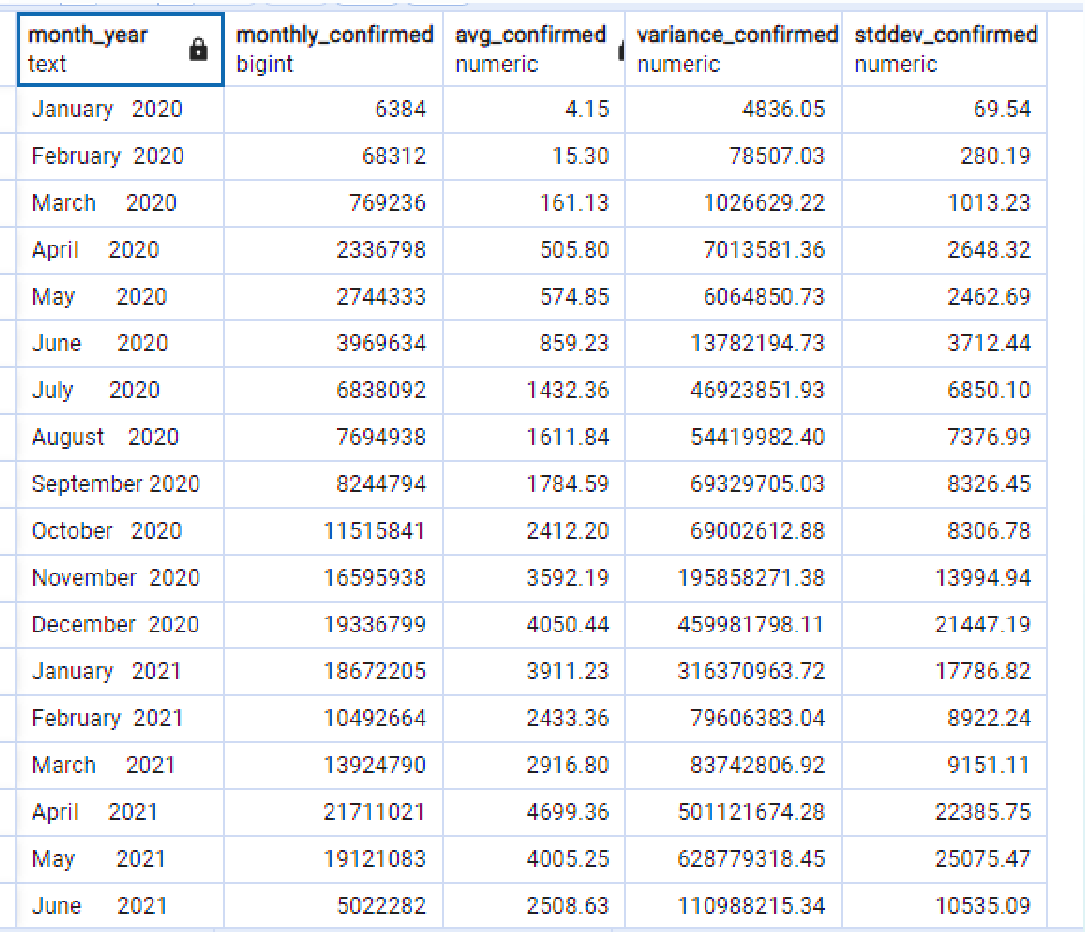
	to_char text	cumulative_confirmed numeric
1	2020-01	6384
2	2020-02	74696
3	2020-03	843932
4	2020-04	3180730
5	2020-05	5925063
6	2020-06	9894697
7	2020-07	16732789
8	2020-08	24427727
9	2020-09	32672521
10	2020-10	44188362
11	2020-11	60784300
12	2020-12	80121099
13	2021-01	98793304
14	2021-02	109285968
15	2021-03	123210758
16	2021-04	144921779
17	2021-05	164042862
18	2021-06	169065144

Total rows: 18 of 18 Query complete 00:00:01.771

12b. COVID-19 Spread Analysis: Confirmed Cases

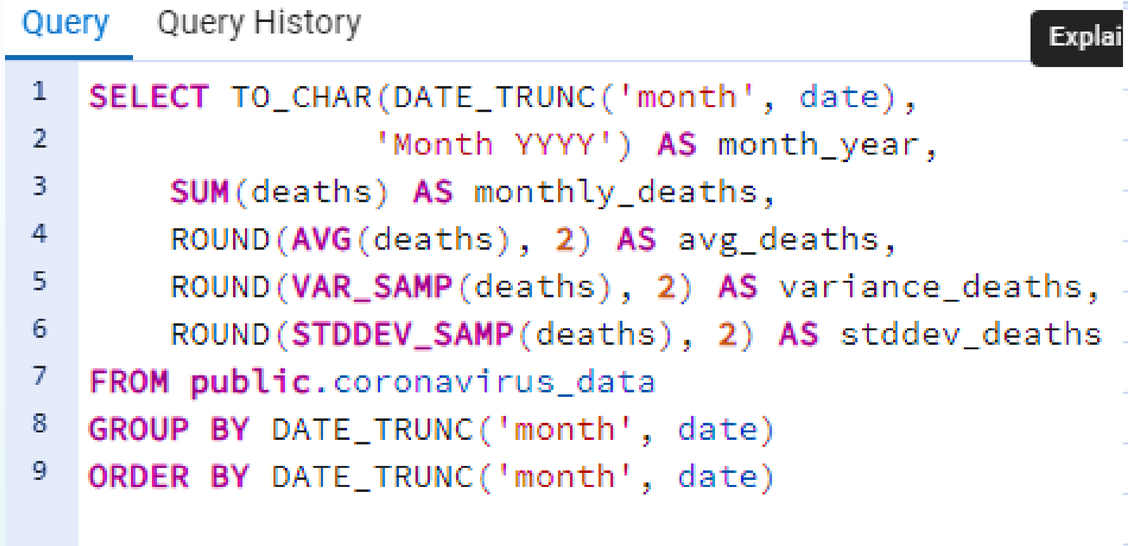


```sql
SELECT TO_CHAR(DATE_TRUNC('month', date),
               'Month YYYY') AS month_year,
        SUM(confirmed) AS monthly_confirmed,
        ROUND(AVG(confirmed), 2) AS avg_confirmed,
        ROUND(VAR_SAMP(confirmed), 2) AS variance_confirmed,
        ROUND(STDDEV_SAMP(confirmed), 2) AS stddev_confirmed
FROM public.coronavirus_data
GROUP BY DATE_TRUNC('month', date)
ORDER BY DATE_TRUNC('month', date)
```

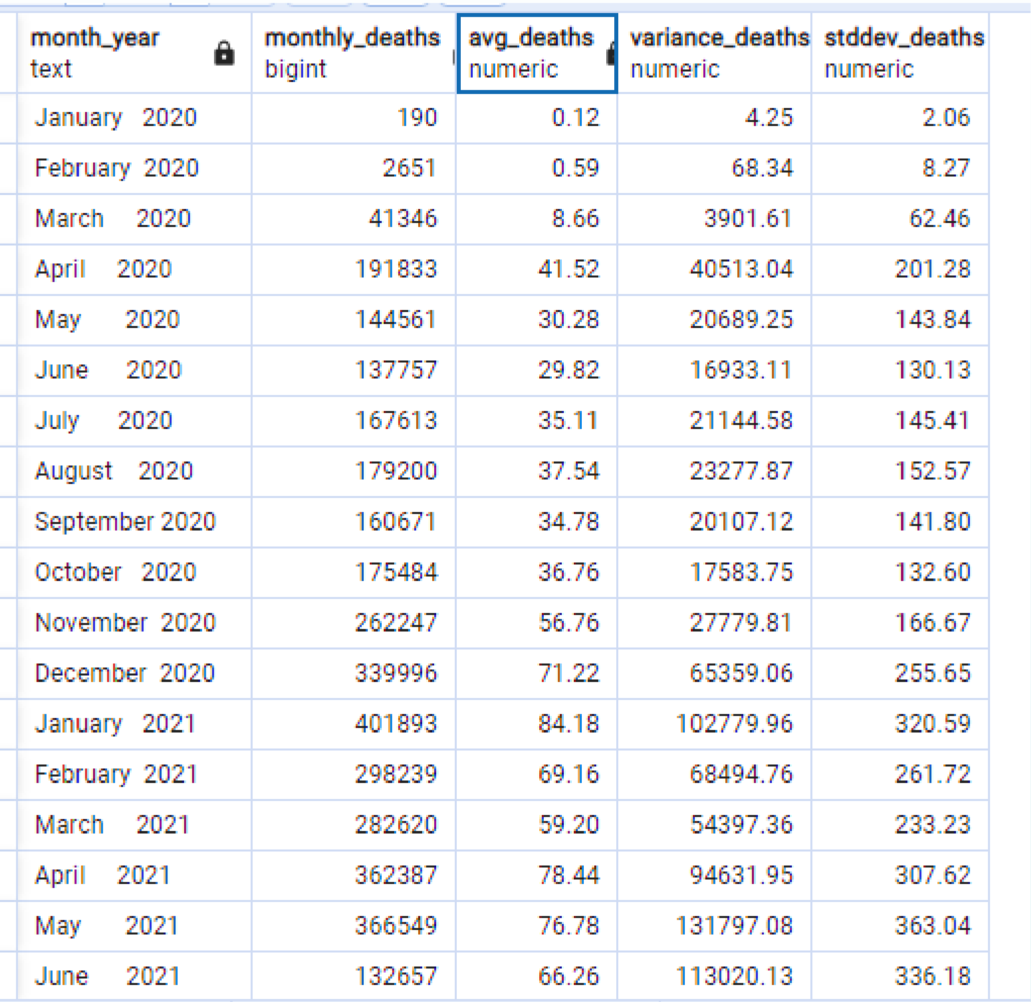


month_year text	monthly_confirmed bigint	avg_confirmed numeric	variance_confirmed numeric	stddev_confirmed numeric
January 2020	6384	4.15	4836.05	69.54
February 2020	68312	15.30	78507.03	280.19
March 2020	769236	161.13	1026629.22	1013.23
April 2020	2336798	505.80	7013581.36	2648.32
May 2020	2744333	574.85	6064850.73	2462.69
June 2020	3969634	859.23	13782194.73	3712.44
July 2020	6838092	1432.36	46923851.93	6850.10
August 2020	7694938	1611.84	54419982.40	7376.99
September 2020	8244794	1784.59	69329705.03	8326.45
October 2020	11515841	2412.20	69002612.88	8306.78
November 2020	16595938	3592.19	195858271.38	13994.94
December 2020	19336799	4050.44	459981798.11	21447.19
January 2021	18672205	3911.23	316370963.72	17786.82
February 2021	10492664	2433.36	79606383.04	8922.24
March 2021	13924790	2916.80	83742806.92	9151.11
April 2021	21711021	4699.36	501121674.28	22385.75
May 2021	19121083	4005.25	628779318.45	25075.47
June 2021	5022282	2508.63	110988215.34	10535.09

13. COVID-19 Spread Analysis: Deaths

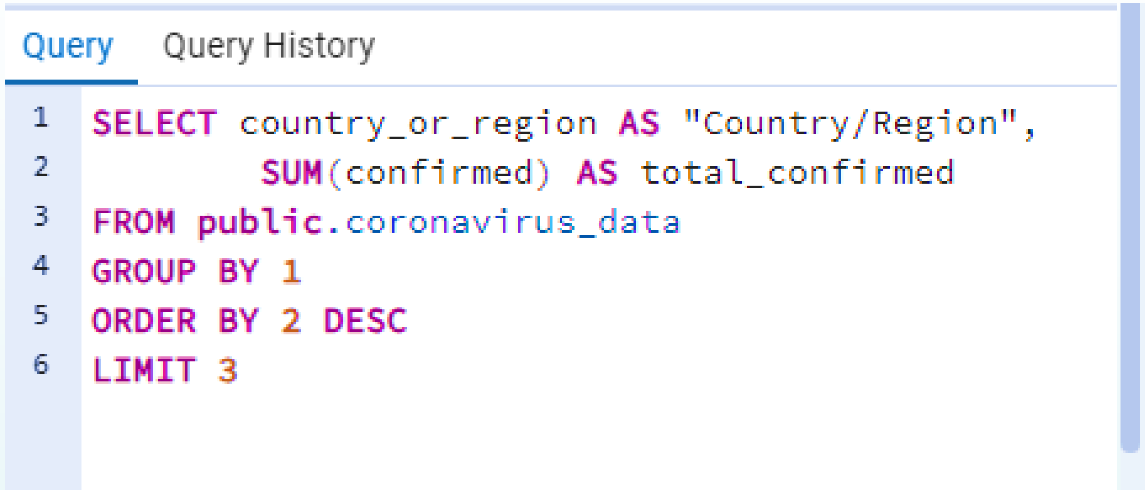


```sql
SELECT TO_CHAR(DATE_TRUNC('month', date),
               'Month YYYY') AS month_year,
    SUM(deaths) AS monthly_deaths,
    ROUND(AVG(deaths), 2) AS avg_deaths,
    ROUND(VAR_SAMP(deaths), 2) AS variance_deaths,
    ROUND(STDDEV_SAMP(deaths), 2) AS stddev_deaths
FROM public.coronavirus_data
GROUP BY DATE_TRUNC('month', date)
ORDER BY DATE_TRUNC('month', date)
```

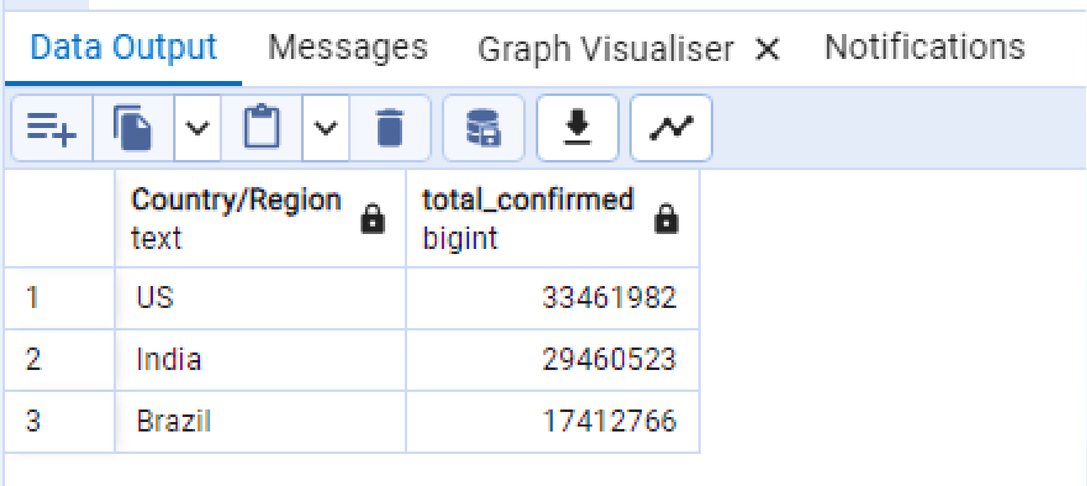


month_year text	monthly_deaths bigint	avg_deaths numeric	variance_deaths numeric	stddev_deaths numeric
January 2020	190	0.12	4.25	2.06
February 2020	2651	0.59	68.34	8.27
March 2020	41346	8.66	3901.61	62.46
April 2020	191833	41.52	40513.04	201.28
May 2020	144561	30.28	20689.25	143.84
June 2020	137757	29.82	16933.11	130.13
July 2020	167613	35.11	21144.58	145.41
August 2020	179200	37.54	23277.87	152.57
September 2020	160671	34.78	20107.12	141.80
October 2020	175484	36.76	17583.75	132.60
November 2020	262247	56.76	27779.81	166.67
December 2020	339996	71.22	65359.06	255.65
January 2021	401893	84.18	102779.96	320.59
February 2021	298239	69.16	68494.76	261.72
March 2021	282620	59.20	54397.36	233.23
April 2021	362387	78.44	94631.95	307.62
May 2021	366549	76.78	131797.08	363.04
June 2021	132657	66.26	113020.13	336.18

14. Country with the Highest Number of Confirmed Cases



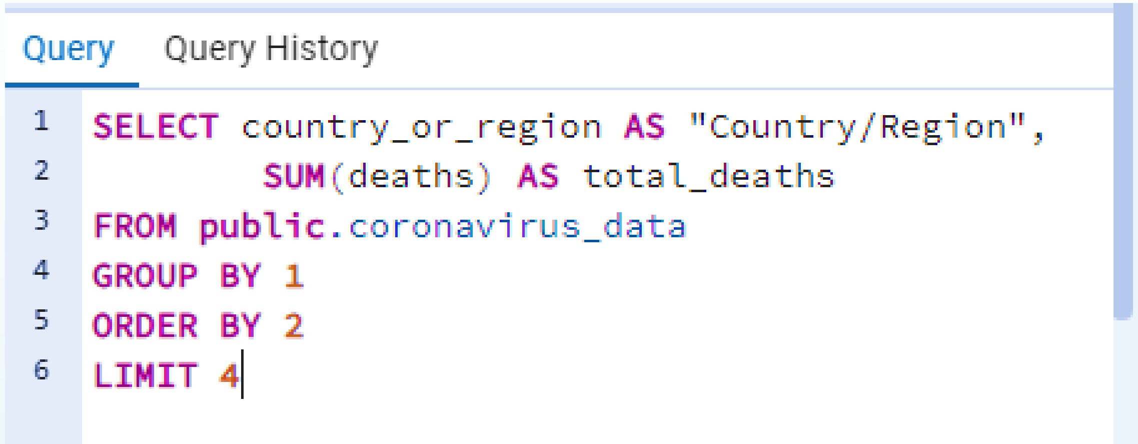
```sql
SELECT country_or_region AS "Country/Region",
        SUM(confirmed) AS total_confirmed
FROM public.coronavirus_data
GROUP BY 1
ORDER BY 2 DESC
LIMIT 3
```



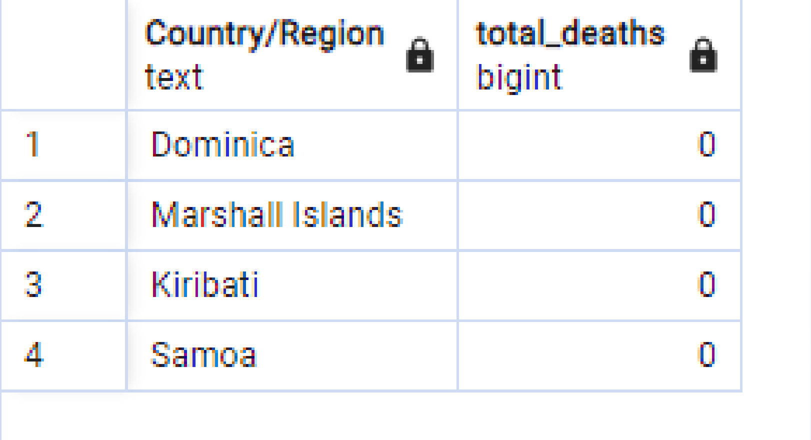
	Country/Region text	total_confirmed bigint
1	US	33461982
2	India	29460523
3	Brazil	17412766

*The **United States** had the highest number of confirmed cases.*

15. Countries with the Lowest Number of Death Cases



```sql
SELECT country_or_region AS "Country/Region",
          SUM(deaths) AS total_deaths
FROM public.coronavirus_data
GROUP BY 1
ORDER BY 2
LIMIT 4
```

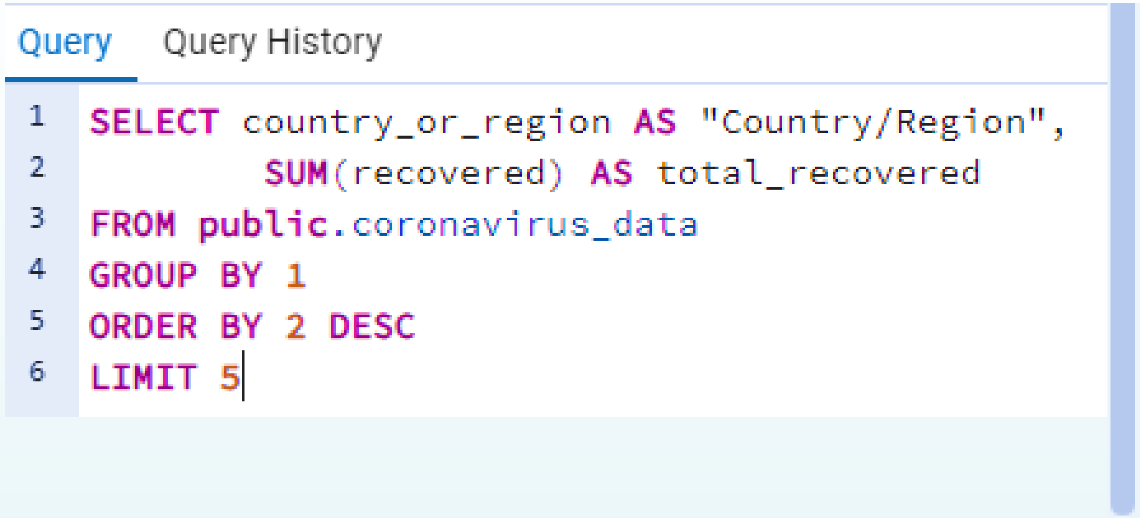


	Country/Region text	total_deaths bigint
1	Dominica	0
2	Marshall Islands	0
3	Kiribati	0
4	Samoa	0

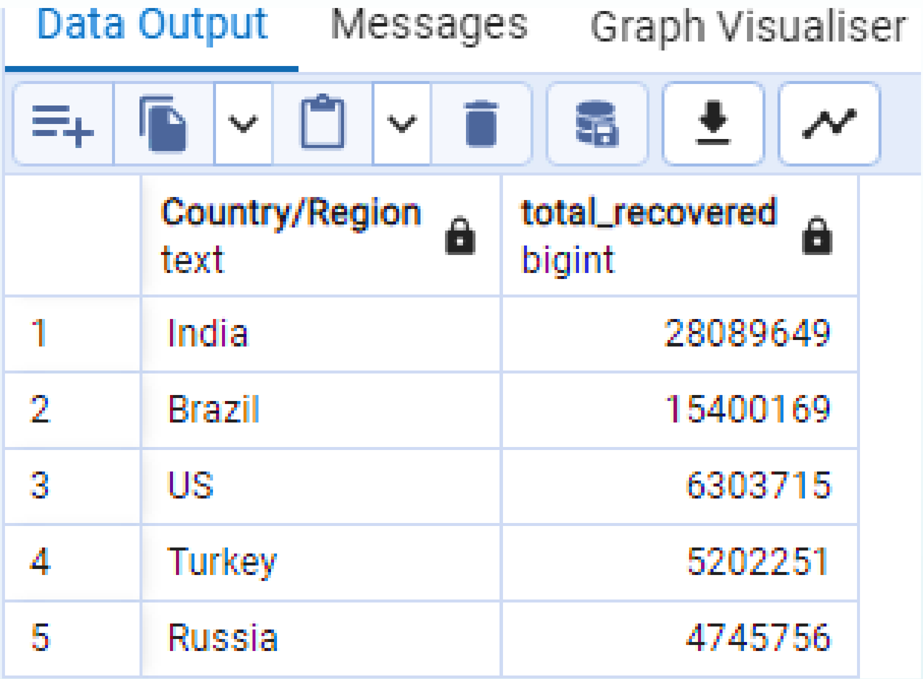
There were four (4) countries with no record of deaths.

They are all island countries.

16. Countries with the Highest Number of Recovered Cases



```sql
SELECT country_or_region AS "Country/Region",
       SUM(recovered) AS total_recovered
FROM public.coronavirus_data
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5
```



	Country/Region text	total_recovered bigint
1	India	28089649
2	Brazil	15400169
3	US	6303715
4	Turkey	5202251
5	Russia	4745756

Conclusion

- The analysis of the coronavirus dataset revealed significant trends and insights.
- Globally, the daily and monthly aggregation of confirmed cases, deaths, and recoveries showed distinct peaks corresponding to various waves of the pandemic.
- The top 5 countries with the highest recovery rates demonstrated effective management and healthcare responses.
- Mortality and recovery rates varied widely, indicating differences in healthcare infrastructure and public health policies.
- The growth rate analysis highlighted rapid spikes during initial outbreaks and subsequent waves.

Recommendations

1. **Strengthen Healthcare Systems:** Invest in healthcare infrastructure to better manage future pandemics.
2. **Improve Data Reporting:** Ensure consistent and accurate data collection for real-time analysis and decision-making.
3. **Enhance Public Health Policies:** Implement evidence-based public health measures tailored to each country's context.
4. **Promote Vaccination:** Encourage widespread vaccination to mitigate the impact of future waves.
5. **Increase Global Collaboration:** Foster international cooperation for sharing resources, knowledge, and strategies in pandemic management.

THANK YOU

Adeoluwa Ogundele

Data Analysis Intern

@Mentorness