Mentorness

Task 2: Corona Virus Analysis with SQL

Aayush Jitendrabhai Vataliya

Data Analyst Intern

MIP-DA-03

Project Overview

The effects of COVID-19 on public health highlight the necessity of **data-driven insights** to understand the virus's progress.

As a **data analyst**, your job is to search for important insights by analyzing a COVID-19 dataset.

We want to identify **patterns and trends** through in-depth study to improve our understanding of virus transmission.

Insights derived from data will help fight the pandemic and safeguard public health.

Dataset Attributes Description

Information on each attribute in the dataset :

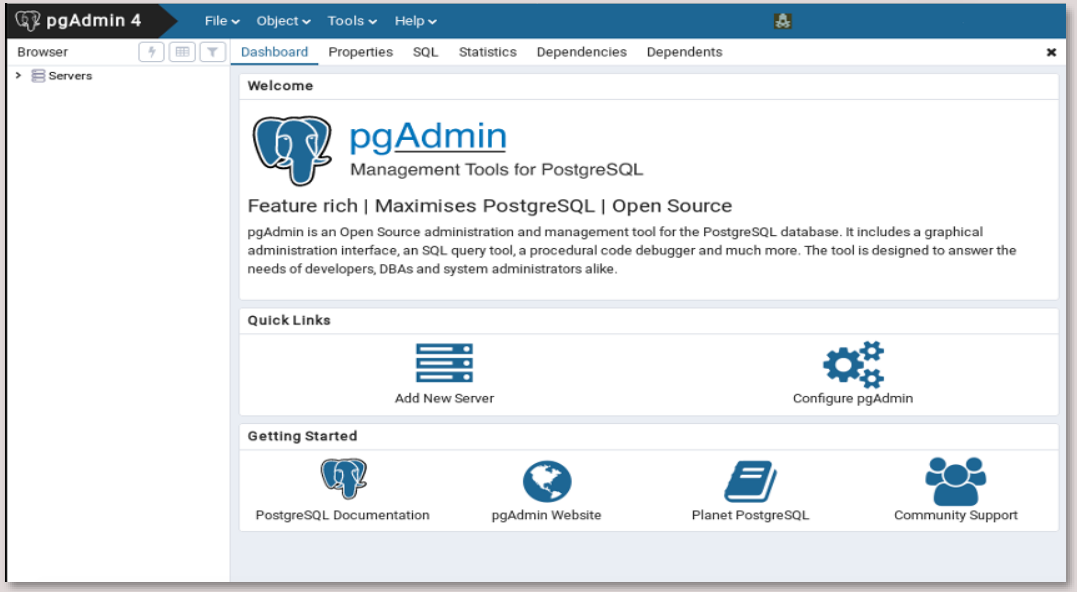
- **Province:** Geographic subdivision within a country/region.
- **Country/Region:** Geographic entity where data is recorded.
- **Latitude:** North-south position on Earth's surface.
- **Longitude:** East-west position on Earth's surface.
- **Date:** Recorded date of CORONA VIRUS data.
- **Confirmed:**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

SQL Data Analysis

Database used for the Analysis:

PostgreSQL

DBMS Tool used for the Analysis:

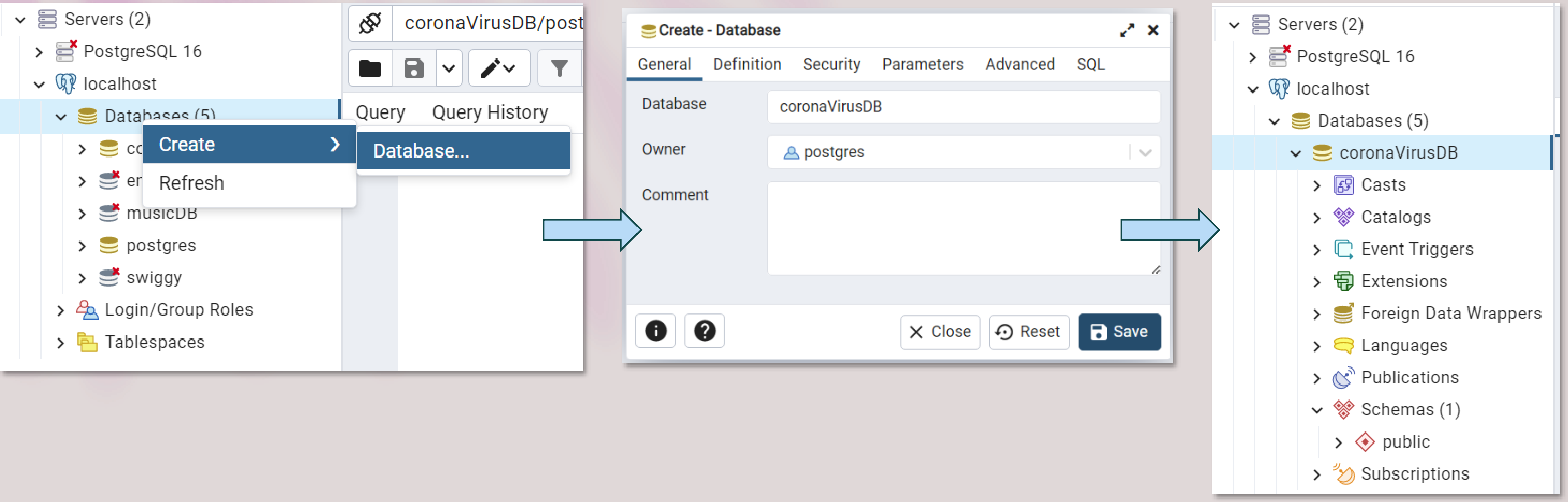


Feature rich | Maximises PostgreSQL | Open Source

pgAdmin is an Open Source administration and management tool for the PostgreSQL database. It includes a graphical administration interface, an SQL query tool, a procedural code debugger and much more. The tool is designed to answer the needs of developers, DBAs and system administrators alike.

Data Gathering Phase

Creating “CoronaVirusDB” Database:



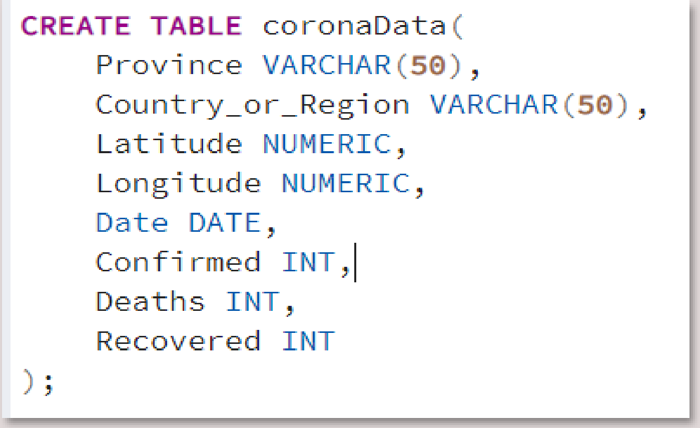
Database: coronaVirusDB

Owner: postgres

Comment:

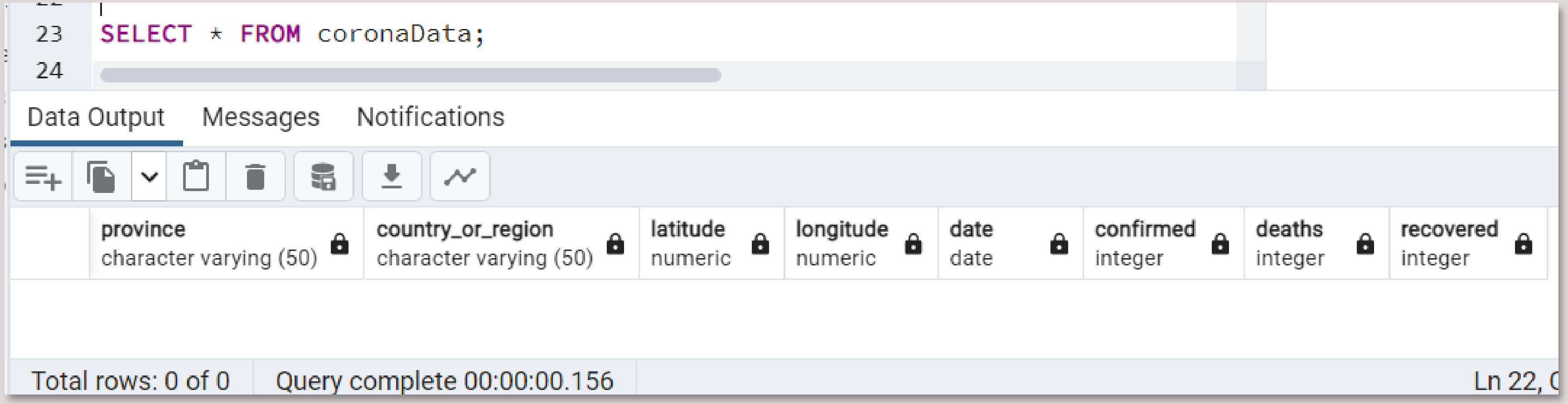
Data Gathering Phase

Creating “ coronaData ” Table:



```sql
CREATE TABLE coronaData(
    Province VARCHAR(50),
    Country_or_Region VARCHAR(50),
    Latitude NUMERIC,
    Longitude NUMERIC,
    Date DATE,
    Confirmed INT,
    Deaths INT,
    Recovered INT
);
```

Getting Table
with no records



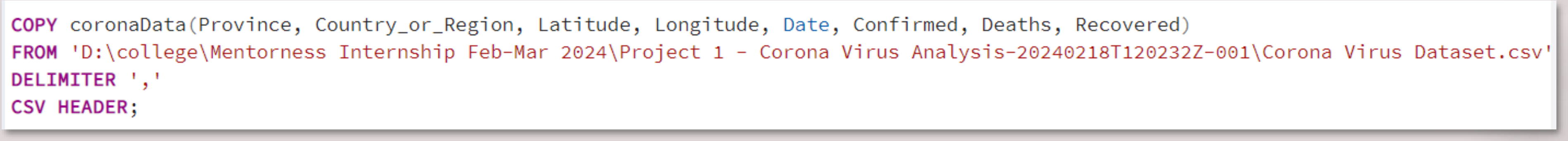
```sql
SELECT * FROM coronaData;
```

province character varying (50)	country_or_region character varying (50)	latitude numeric	longitude numeric	date date	confirmed integer	deaths integer	recovered integer

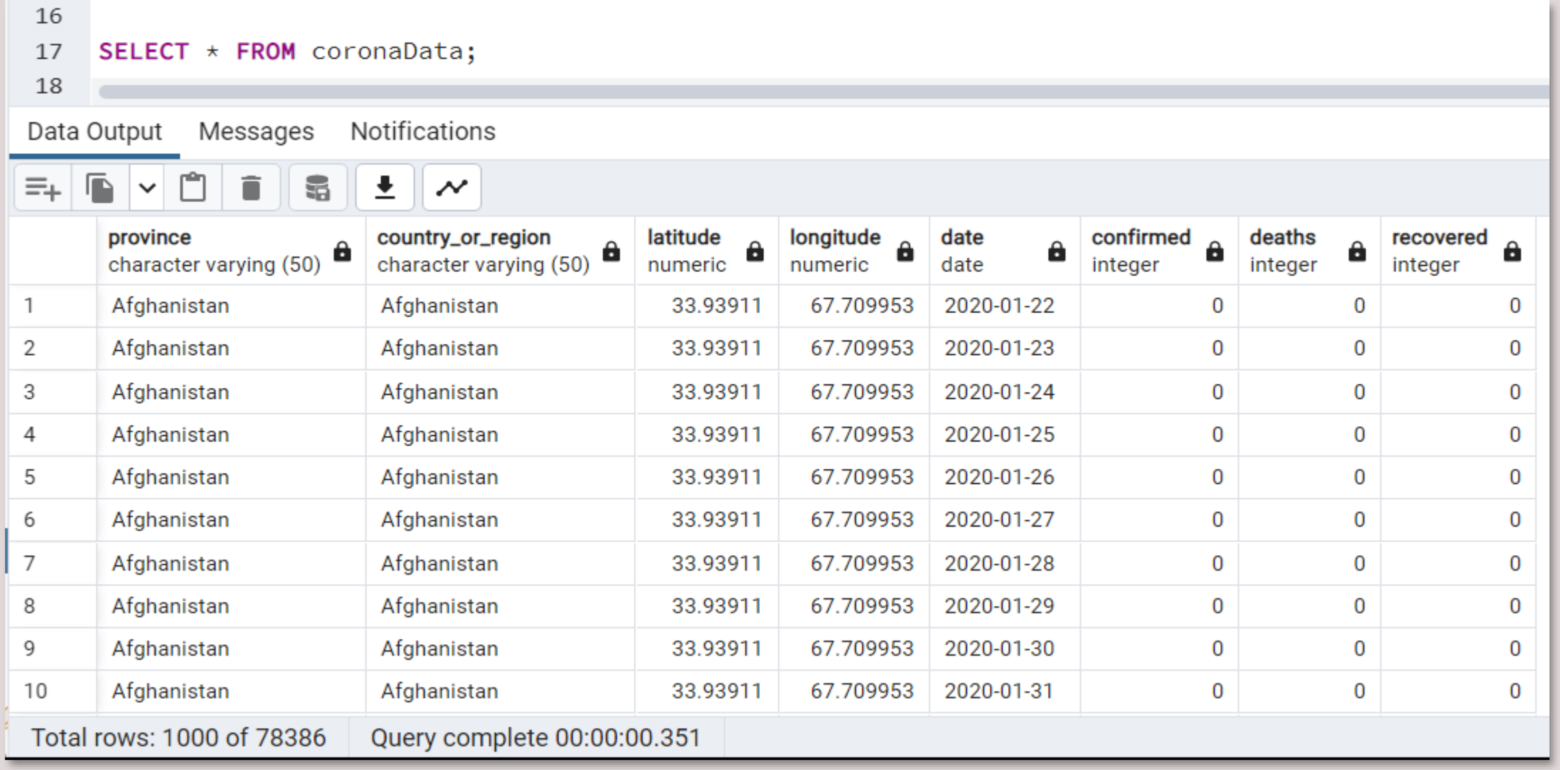
Total rows: 0 of 0 Query complete 00:00:00.156 Ln 22, C

Data Gathering Phase

Import Data from “ Corona Virus Dataset.csv ” file:



```
COPY coronaData(Province, Country_or_Region, Latitude, Longitude, Date, Confirmed, Deaths, Recovered)
FROM 'D:\college\Mentorness Internship Feb-Mar 2024\Project 1 – Corona Virus Analysis-20240218T120232Z-001\Corona Virus Dataset.csv'
DELIMITER ','
CSV HEADER;
```



```
SELECT * FROM coronaData;
```

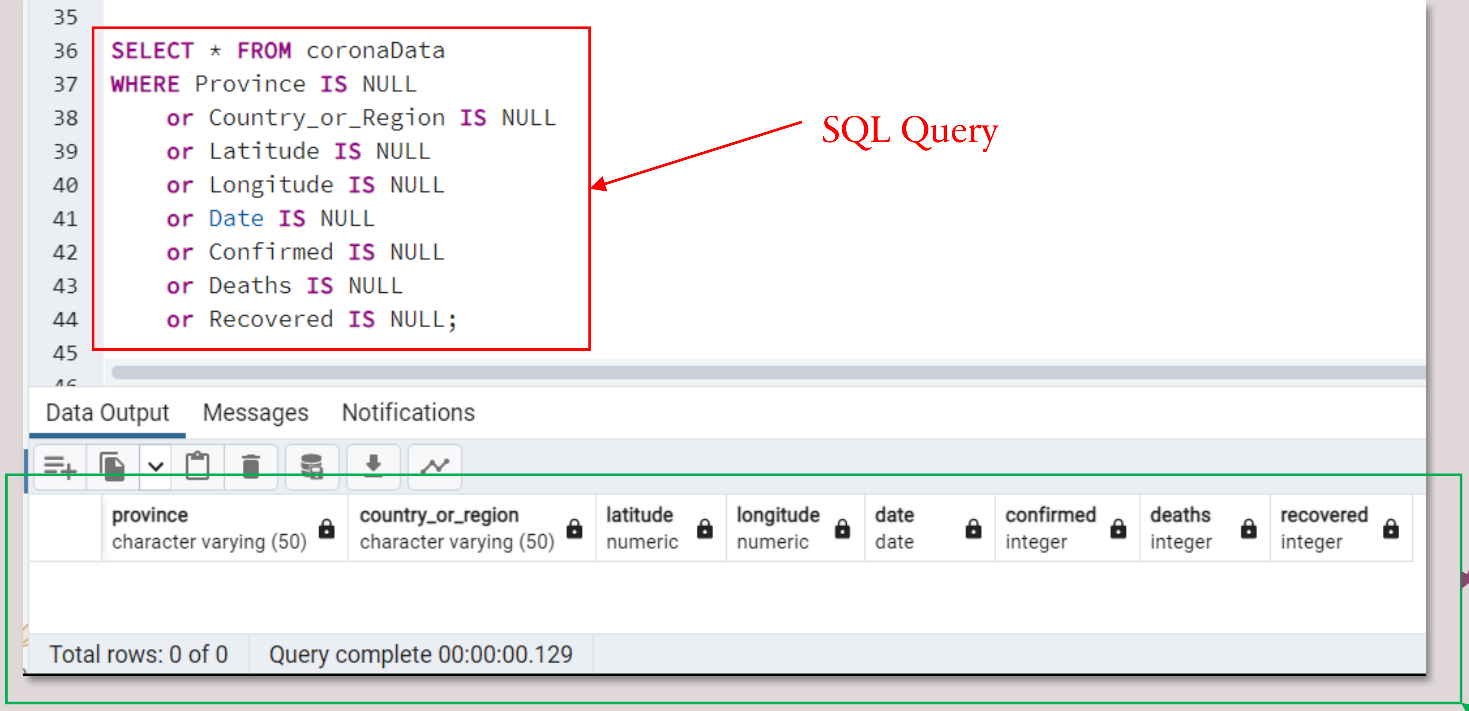
	province character varying (50)	country_or_region character varying (50)	latitude numeric	longitude numeric	date date	confirmed integer	deaths integer	recovered integer
1	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-22	0	0	0
2	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-23	0	0	0
3	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-24	0	0	0
4	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-25	0	0	0
5	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-26	0	0	0
6	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-27	0	0	0
7	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-28	0	0	0
8	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-29	0	0	0
9	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-30	0	0	0
10	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-31	0	0	0

Total rows: 1000 of 78386 Query complete 00:00:00.351

Data Cleaning Phase

To avoid any errors, check missing value / null value

1. Write a code to check NULL values:



```sql
SELECT * FROM coronaData
WHERE Province IS NULL
    or Country_or_Region 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;
```

```
province                 country_or_region        latitude   longitude   date   confirmed   deaths    recovered
character varying (50)   character varying (50)   numeric    numeric     date   integer     integer   integer

Total rows: 0 of 0     Query complete 00:00:00.129
```

SQL Query

Output

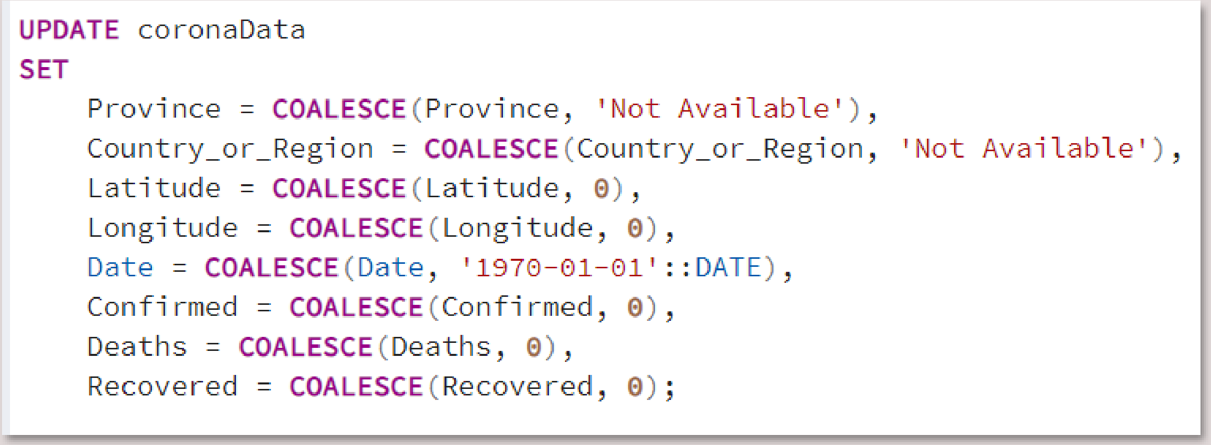
By this query, it is evident that there is **not a single NULL / Missing value** present in the entire dataset.

Data Cleaning Phase

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

- By the previous query, it is evident that there is **not a single NULL / Missing value** present in the entire dataset.

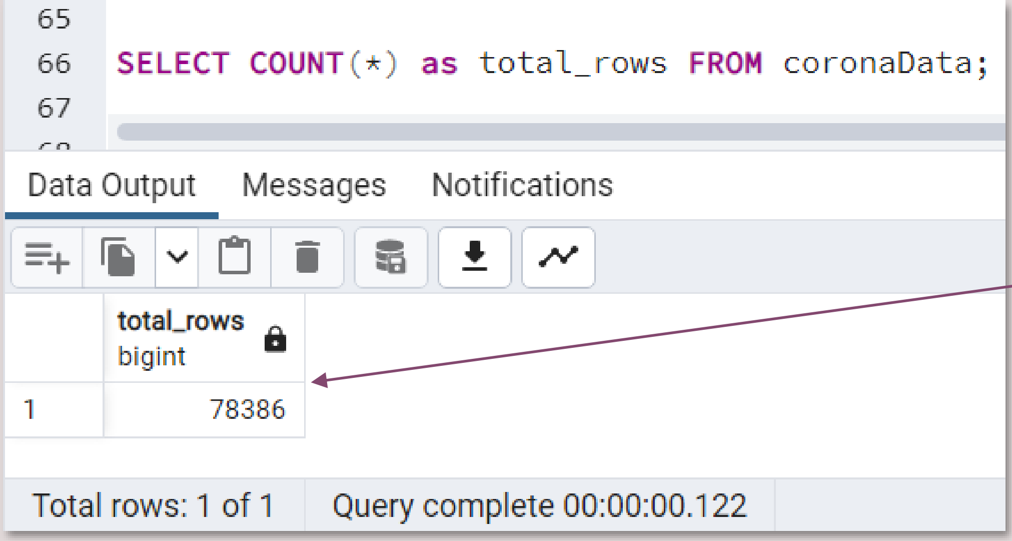
- If missing values are present, then below mentioned query can be used to replace missing values with default values.



```
UPDATE coronaData
SET
    Province = COALESCE(Province, 'Not Available'),
    Country_or_Region = COALESCE(Country_or_Region, 'Not Available'),
    Latitude = COALESCE(Latitude, 0),
    Longitude = COALESCE(Longitude, 0),
    Date = COALESCE(Date, '1970-01-01'::DATE),
    Confirmed = COALESCE(Confirmed, 0),
    Deaths = COALESCE(Deaths, 0),
    Recovered = COALESCE(Recovered, 0);
```

Data Cleaning Phase

3. Check Total number of rows:



```
SELECT COUNT(*) as total_rows FROM coronaData;
```

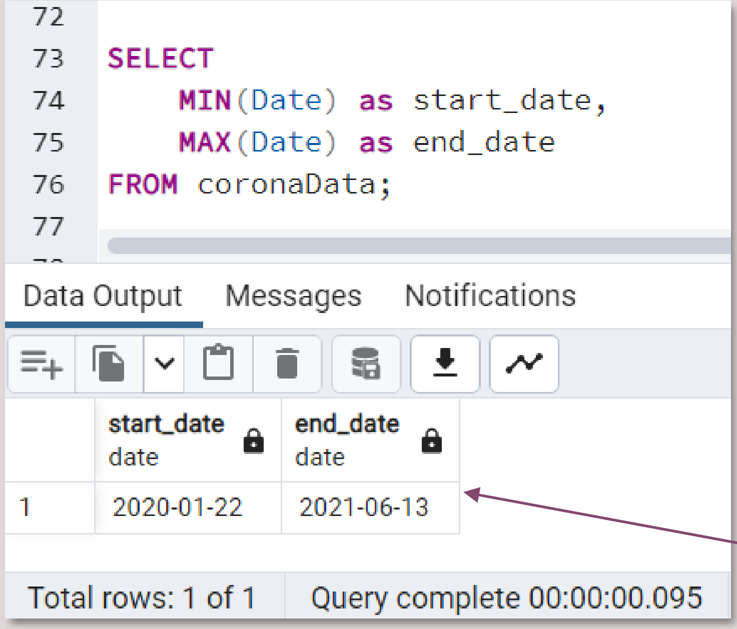
```
   total_rows
   bigint
1       78386
```

Total rows: 1 of 1 Query complete 00:00:00.122

Total Number of Records stored in the table is **78386**

Insightful Queries

4. Check what is start_date and end_date:



```
SELECT
    MIN(Date) as start_date,
    MAX(Date) as end_date
FROM coronaData;
```

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

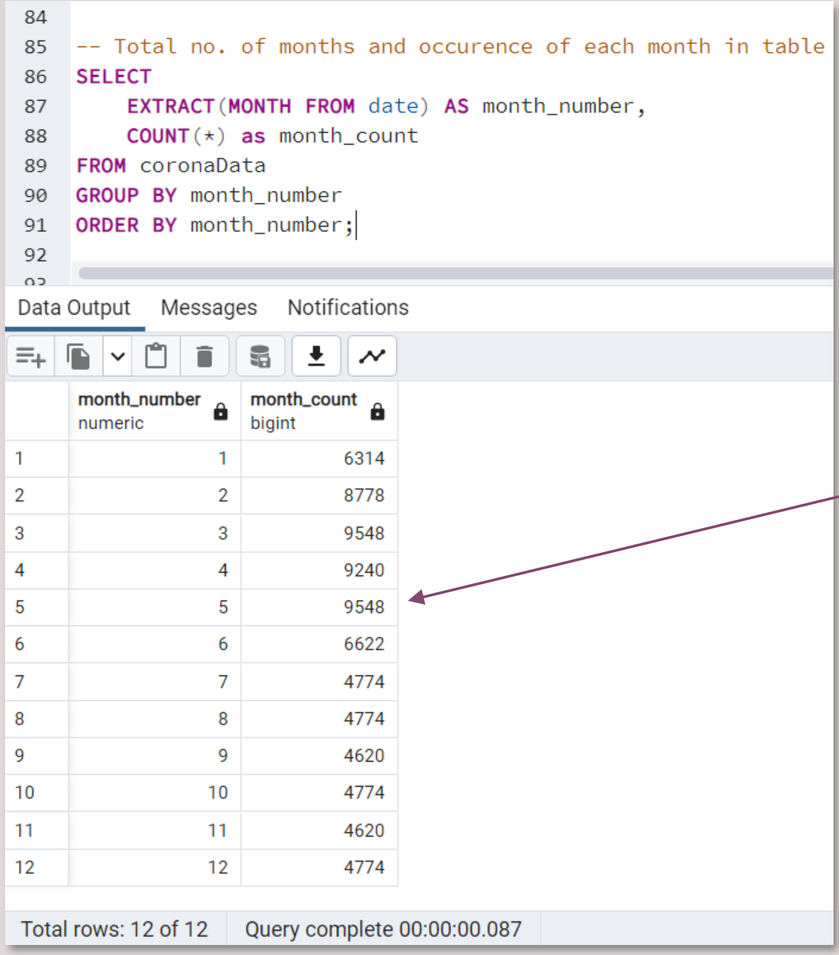
Total rows: 1 of 1   Query complete 00:00:00.095

- Thus it can be observed that the data is ranging from **22nd January 2020** to **13th June 2021.**

- Hence, according to the dataset, the **first case** of Covid-19 was recorded on **22-01-2020** and the **last case** was recorded on **13-06-2021.**

Insightful Queries

5. Number of month present in dataset:



```sql

-- Total no. of months and occurence of each month in table
SELECT
    EXTRACT(MONTH FROM date) AS month_number,
    COUNT(*) as month_count
FROM coronaData
GROUP BY month_number
ORDER BY month_number;

```

	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

Total rows: 12 of 12   Query complete 00:00:00.087

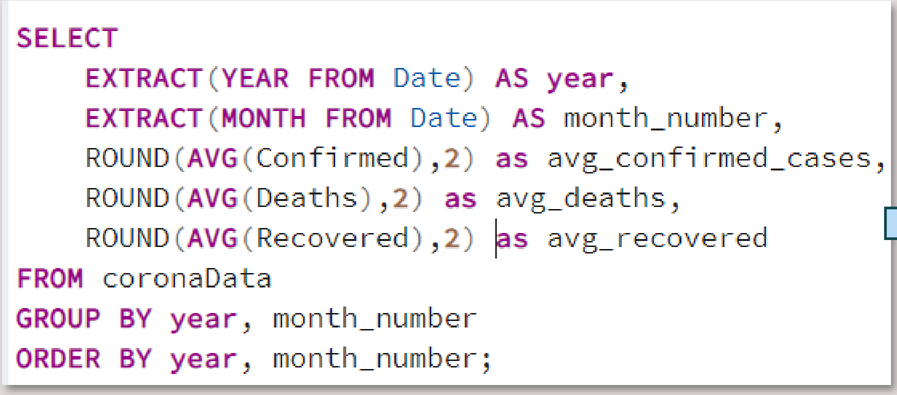
- Here, **month_number** is the number of corresponding months and **month_count** is the number of times a particular month is associated with the Covid case.

- Let's say, **January** month (month_number = 1) has month_count of **6314**, i.e. all over the world, there are **6314 instances** of covid-19 that happened in the month of January in 2020 and 2021

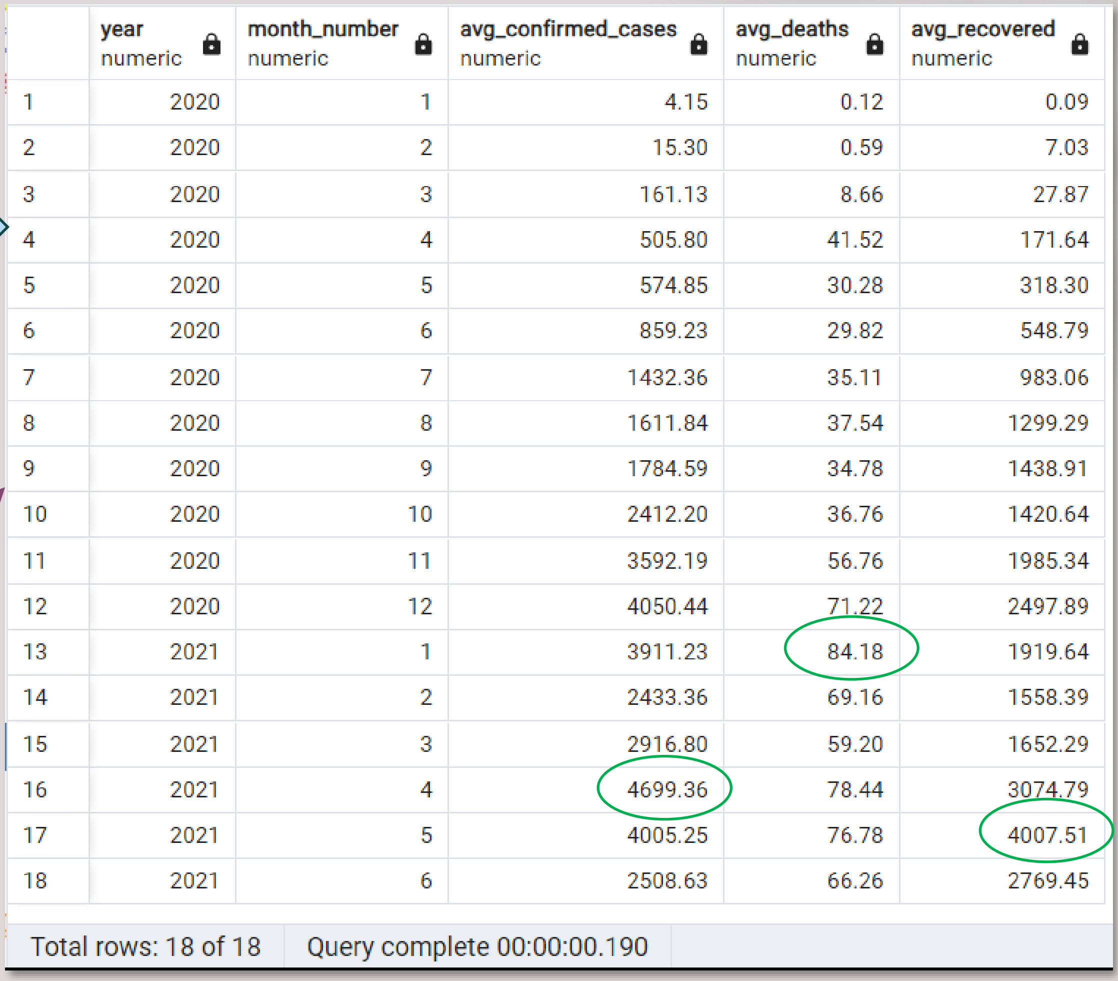
The dataset contains a total of **12 unique months**

Insightful Queries

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



```
SELECT
    EXTRACT(YEAR FROM Date) AS year,
    EXTRACT(MONTH FROM Date) AS month_number,
    ROUND(AVG(Confirmed),2) as avg_confirmed_cases,
    ROUND(AVG(Deaths),2) as avg_deaths,
    ROUND(AVG(Recovered),2) as avg_recovered
FROM coronaData
GROUP BY year, month_number
ORDER BY year, month_number;
```



	year numeric	month_number numeric	avg_confirmed_cases 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

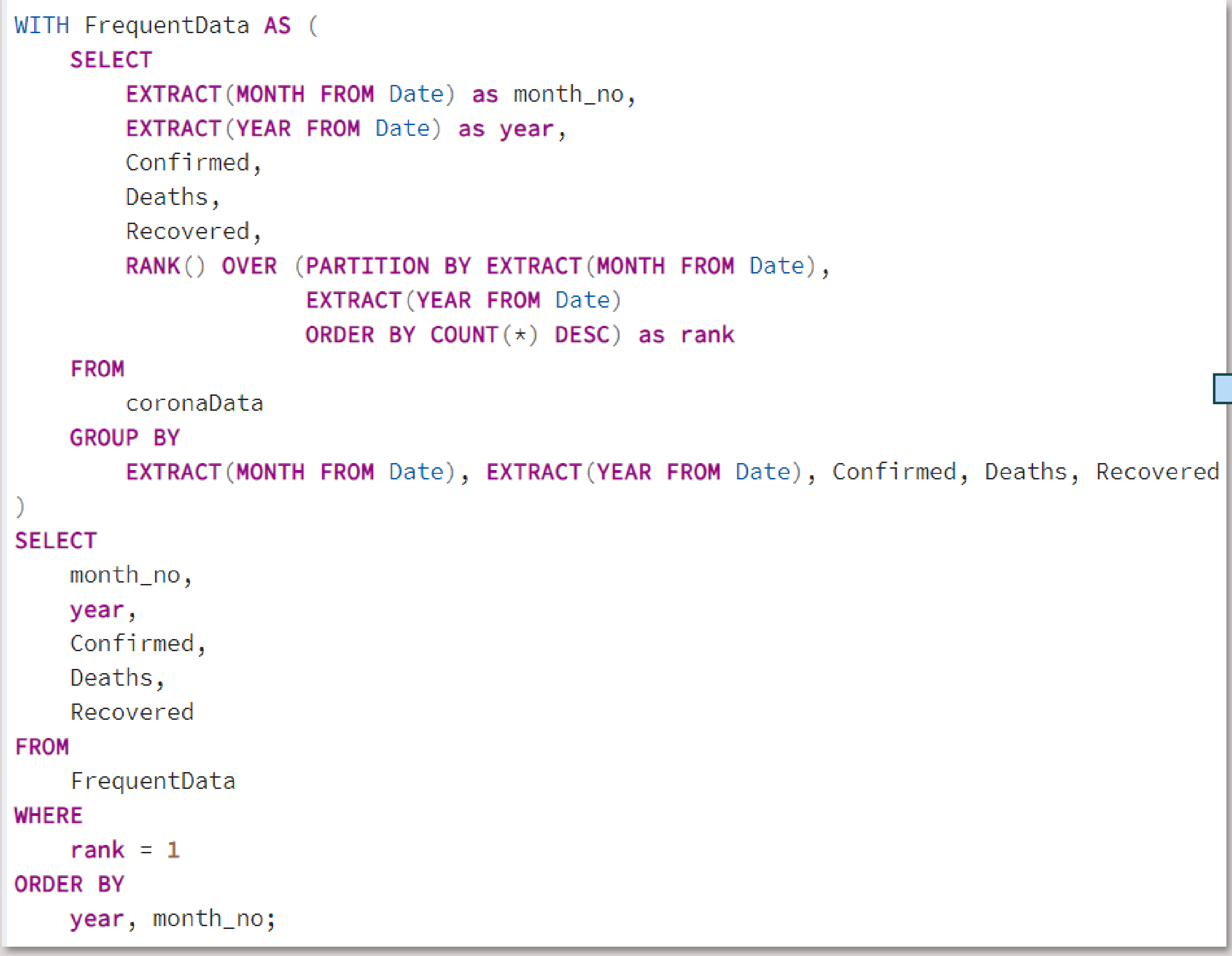
Total rows: 18 of 18   Query complete 00:00:00.190

From the output, it is evident that the **highest average values** of confirmed cases, deaths, & recovered cases are:

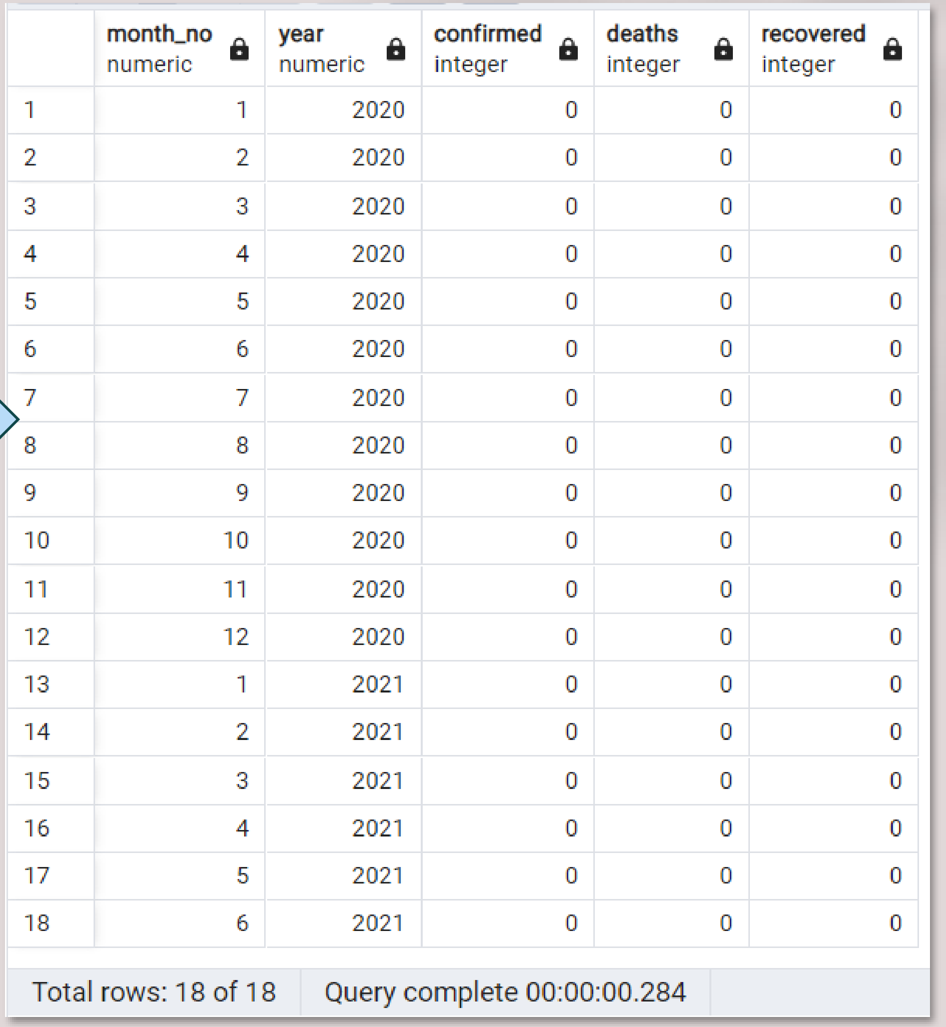
- Confirmed - 4699.36 (July-21)
- Deaths - 84.18 (Jan-21)
- Recovered - 4007.51 (May-21)

Insightful Queries

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



```
WITH FrequentData AS (
    SELECT
        EXTRACT(MONTH FROM Date) as month_no,
        EXTRACT(YEAR FROM Date) as year,
        Confirmed,
        Deaths,
        Recovered,
        RANK() OVER (PARTITION BY EXTRACT(MONTH FROM Date),
                     EXTRACT(YEAR FROM Date)
                     ORDER BY COUNT(*) DESC) as rank
    FROM
        coronaData
    GROUP BY
        EXTRACT(MONTH FROM Date), EXTRACT(YEAR FROM Date), Confirmed, Deaths, Recovered
)
SELECT
    month_no,
    year,
    Confirmed,
    Deaths,
    Recovered
FROM
    FrequentData
WHERE
    rank = 1
ORDER BY
    year, month_no;
```

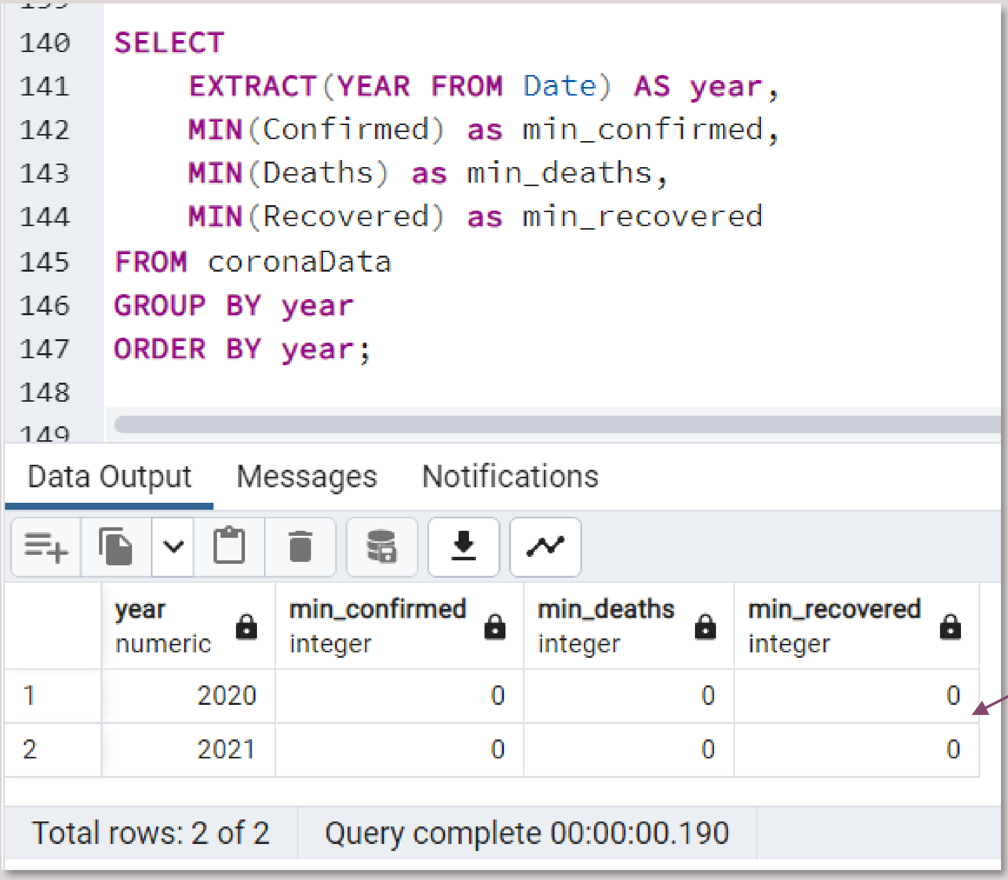


	month_no numeric	year numeric	confirmed integer	deaths integer	recovered integer
1	1	2020	0	0	0
2	2	2020	0	0	0
3	3	2020	0	0	0
4	4	2020	0	0	0
5	5	2020	0	0	0
6	6	2020	0	0	0
7	7	2020	0	0	0
8	8	2020	0	0	0
9	9	2020	0	0	0
10	10	2020	0	0	0
11	11	2020	0	0	0
12	12	2020	0	0	0
13	1	2021	0	0	0
14	2	2021	0	0	0
15	3	2021	0	0	0
16	4	2021	0	0	0
17	5	2021	0	0	0
18	6	2021	0	0	0

Total rows: 18 of 18   Query complete 00:00:00.284

Insightful Queries

8. Find minimum values for confirmed, deaths, recovered per year:



```sql
SELECT
    EXTRACT(YEAR FROM Date) AS year,
    MIN(Confirmed) as min_confirmed,
    MIN(Deaths) as min_deaths,
    MIN(Recovered) as min_recovered
FROM coronaData
GROUP BY year
ORDER BY year;
```

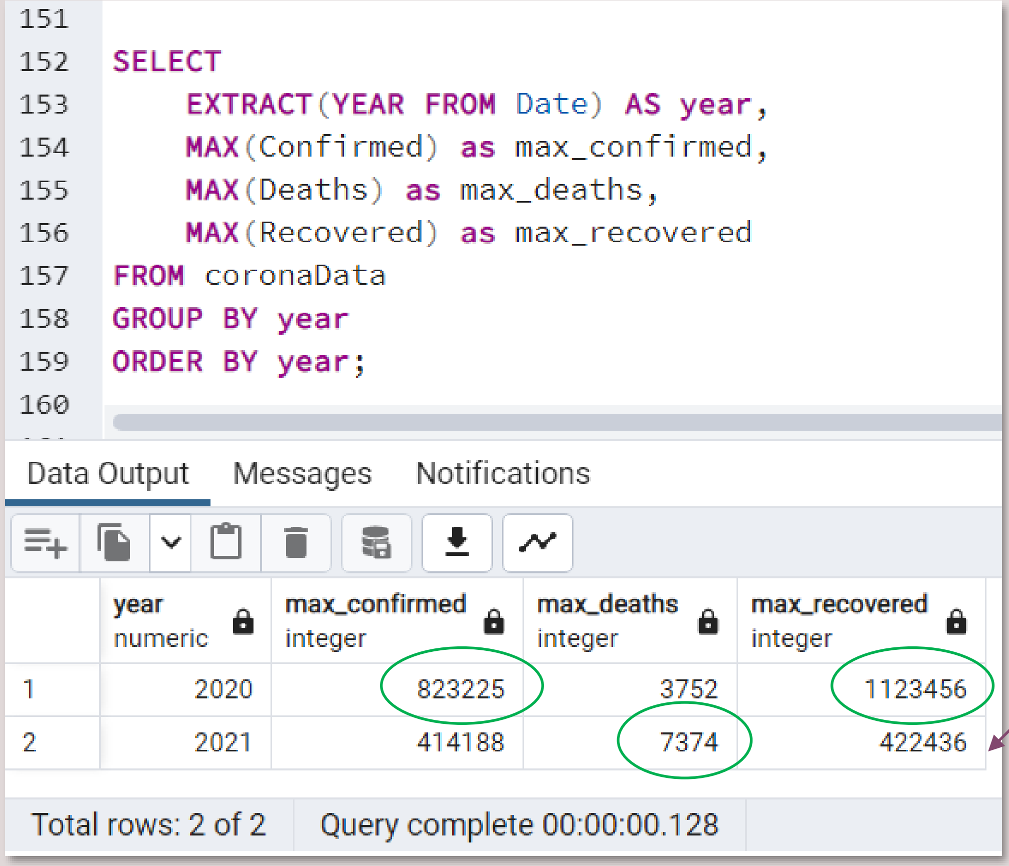
	year numeric	min_confirmed integer	min_deaths integer	min_recovered integer
1	2020	0	0	0
2	2021	0	0	0

Total rows: 2 of 2   Query complete 00:00:00.190

It can be seen that the minimum reported value for each category in 2020 and 2021 is **0**.

Insightful Queries

9. Find maximum values for confirmed, deaths, recovered per year:



```sql

SELECT
    EXTRACT(YEAR FROM Date) AS year,
    MAX(Confirmed) as max_confirmed,
    MAX(Deaths) as max_deaths,
    MAX(Recovered) as max_recovered
FROM coronaData
GROUP BY year
ORDER BY year;

```

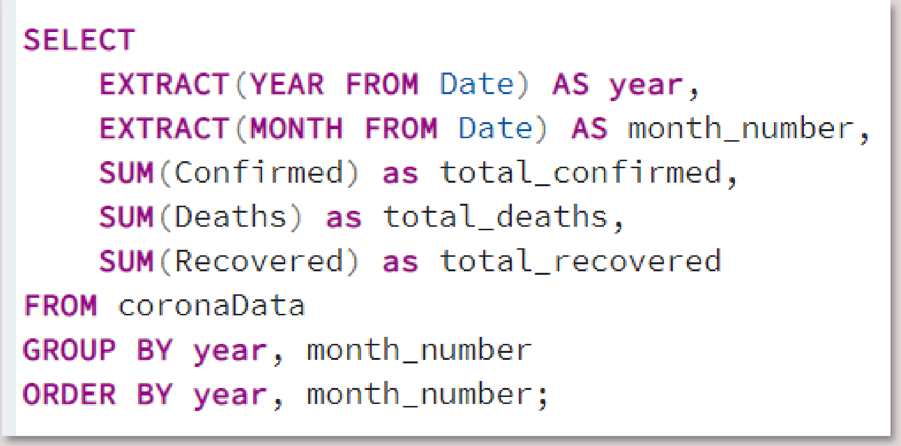
	year	max_confirmed	max_deaths	max_recovered
	numeric	integer	integer	integer
1	2020	823225	3752	1123456
2	2021	414188	7374	422436

Total rows: 2 of 2 Query complete 00:00:00.128

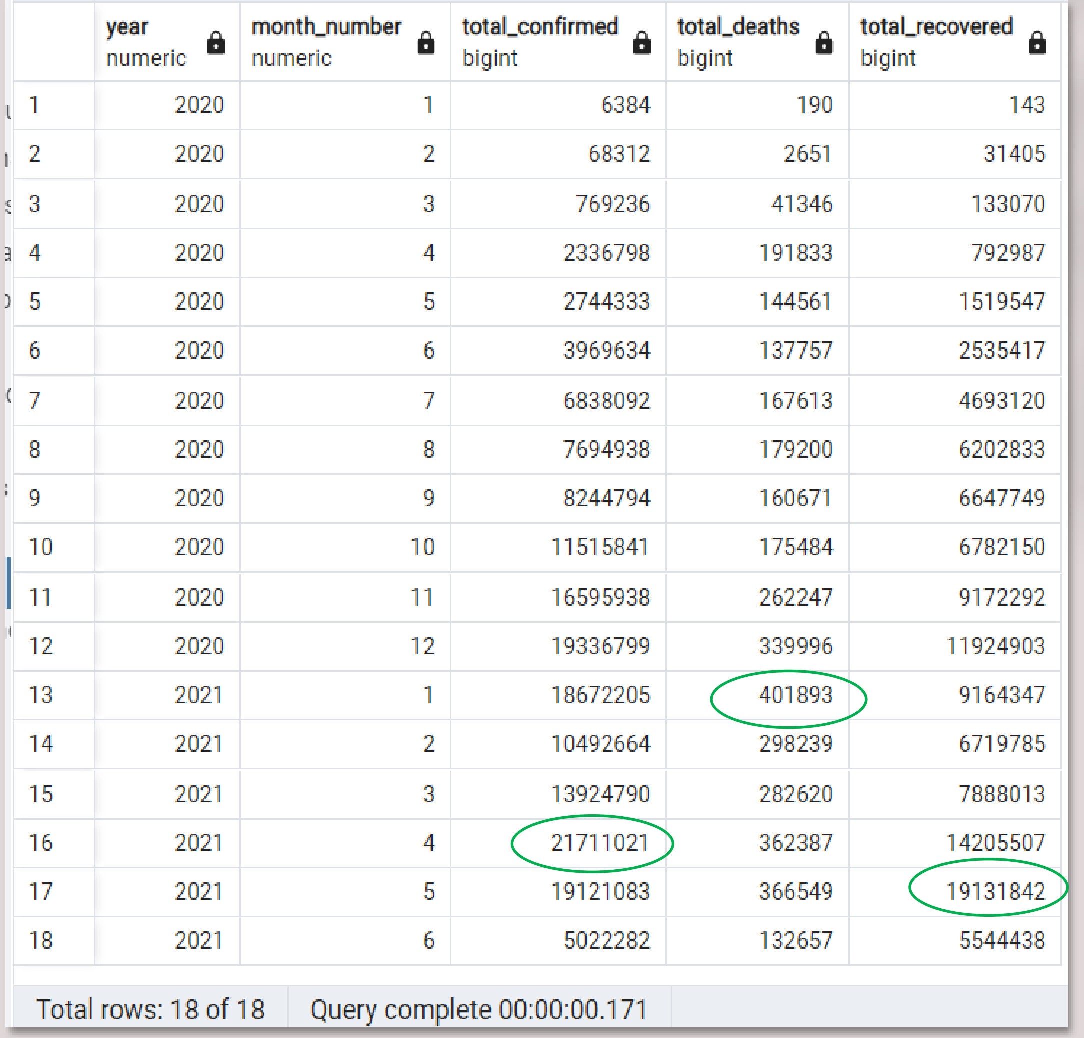
- **max_confirmed** cases topped the table in the year **2020** with **8,23,225** diagnosed cases.
- On the flip side, **max_deaths** were at its apex in the year **2021** with a count of **7,374** deaths.
- Nevertheless, the **recovery rate** of COVID-19 cases was at its zenith in the year **2020** with a recovery rate of **11,23,456** cases.

Insightful Queries

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



```
SELECT
    EXTRACT(YEAR FROM Date) AS year,
    EXTRACT(MONTH FROM Date) AS month_number,
    SUM(Confirmed) as total_confirmed,
    SUM(Deaths) as total_deaths,
    SUM(Recovered) as total_recovered
FROM coronaData
GROUP BY year, month_number
ORDER BY year, month_number;
```



	year numeric	month_number 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

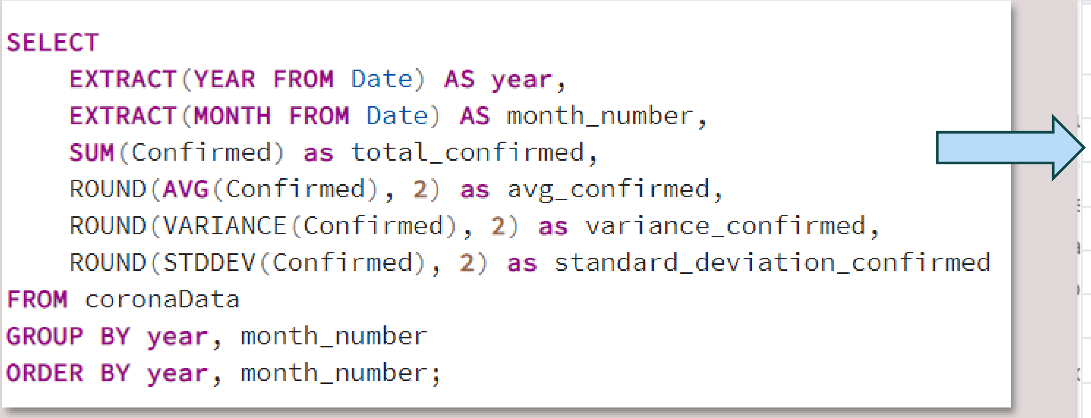
Total rows: 18 of 18 Query complete 00:00:00.171

- The total number of **Confirmed Cases** was at its zenith in **April 2021** with a count of **2,17,11,021**.
- Conversely, the maximum count of **total deaths** was reported in **January 2021** with **4,01,893** deaths.
- However, the total **recovery rate** skyrocketed in the **second Quarter of 2021** with **1,91,31,842** recovered cases in **May 2021** topped the table.

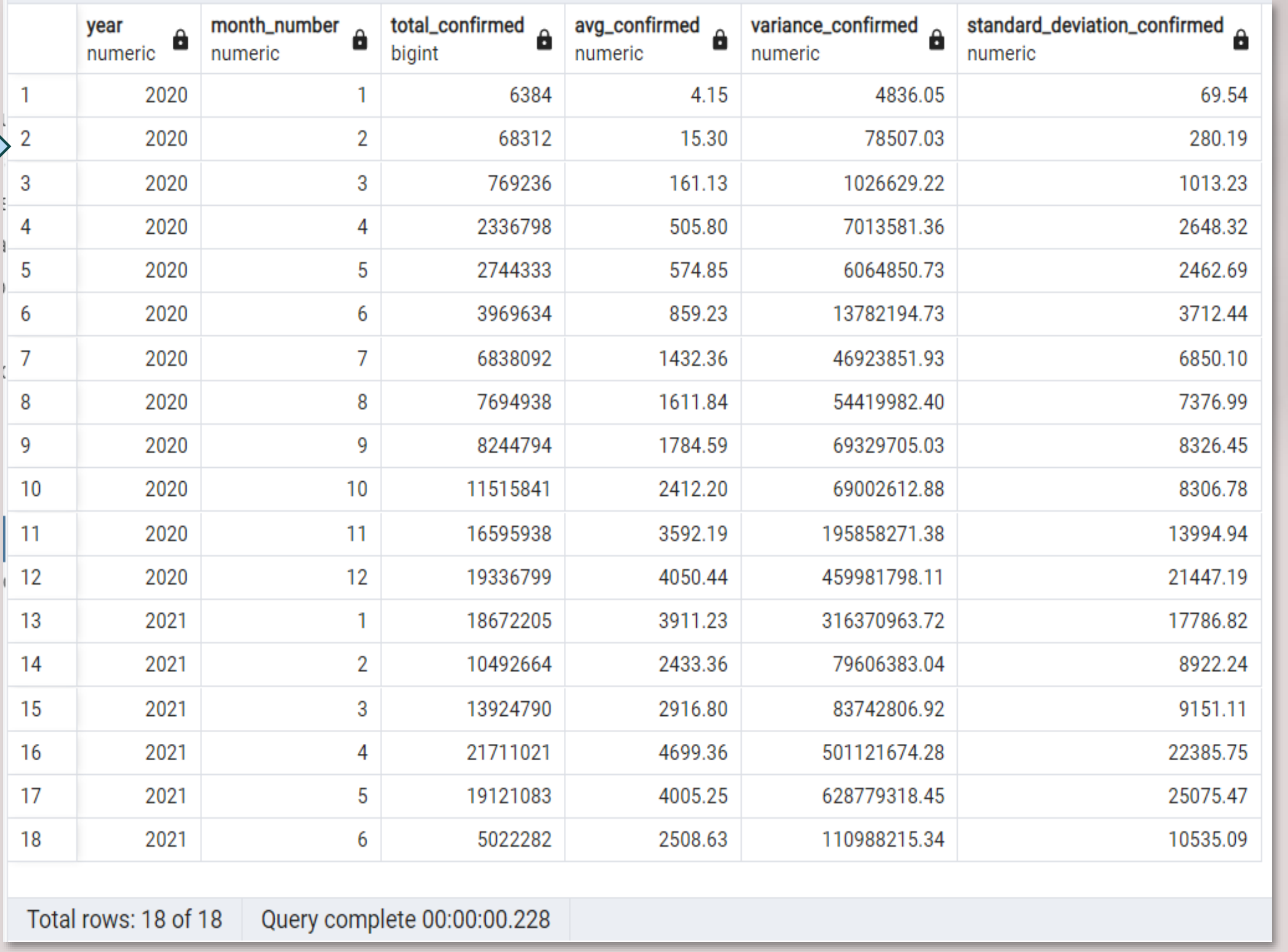
Insightful Queries

11. Check how coronavirus spread out with respect to confirmed cases per month:

    (Eg: Total confirmed cases, their average, variance & STDEV)



```
SELECT
    EXTRACT(YEAR FROM Date) AS year,
    EXTRACT(MONTH FROM Date) AS month_number,
    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_deviation_confirmed
FROM coronaData
GROUP BY year, month_number
ORDER BY year, month_number;
```



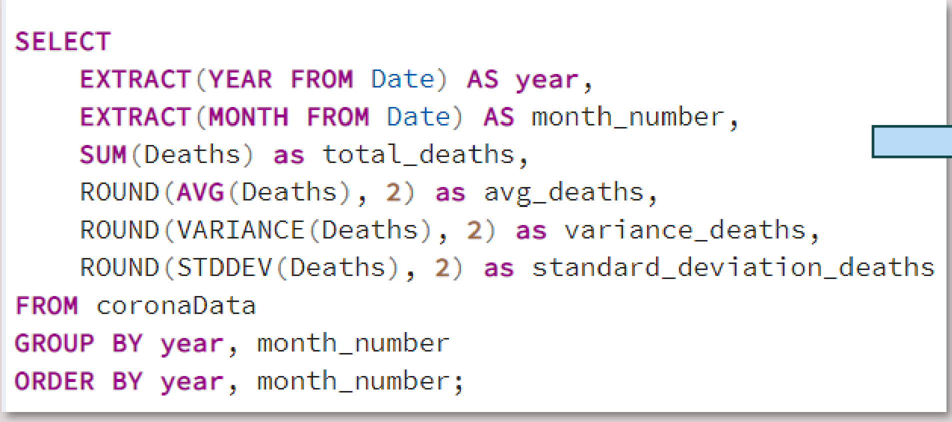
	year	month_number	total_confirmed	avg_confirmed	variance_confirmed	standard_deviation_confirmed
	numeric	numeric	bigint	numeric	numeric	numeric
1	2020	1	6384	4.15	4836.05	69.54
2	2020	2	68312	15.30	78507.03	280.19
3	2020	3	769236	161.13	1026629.22	1013.23
4	2020	4	2336798	505.80	7013581.36	2648.32
5	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

Total rows: 18 of 18   Query complete 00:00:00.228

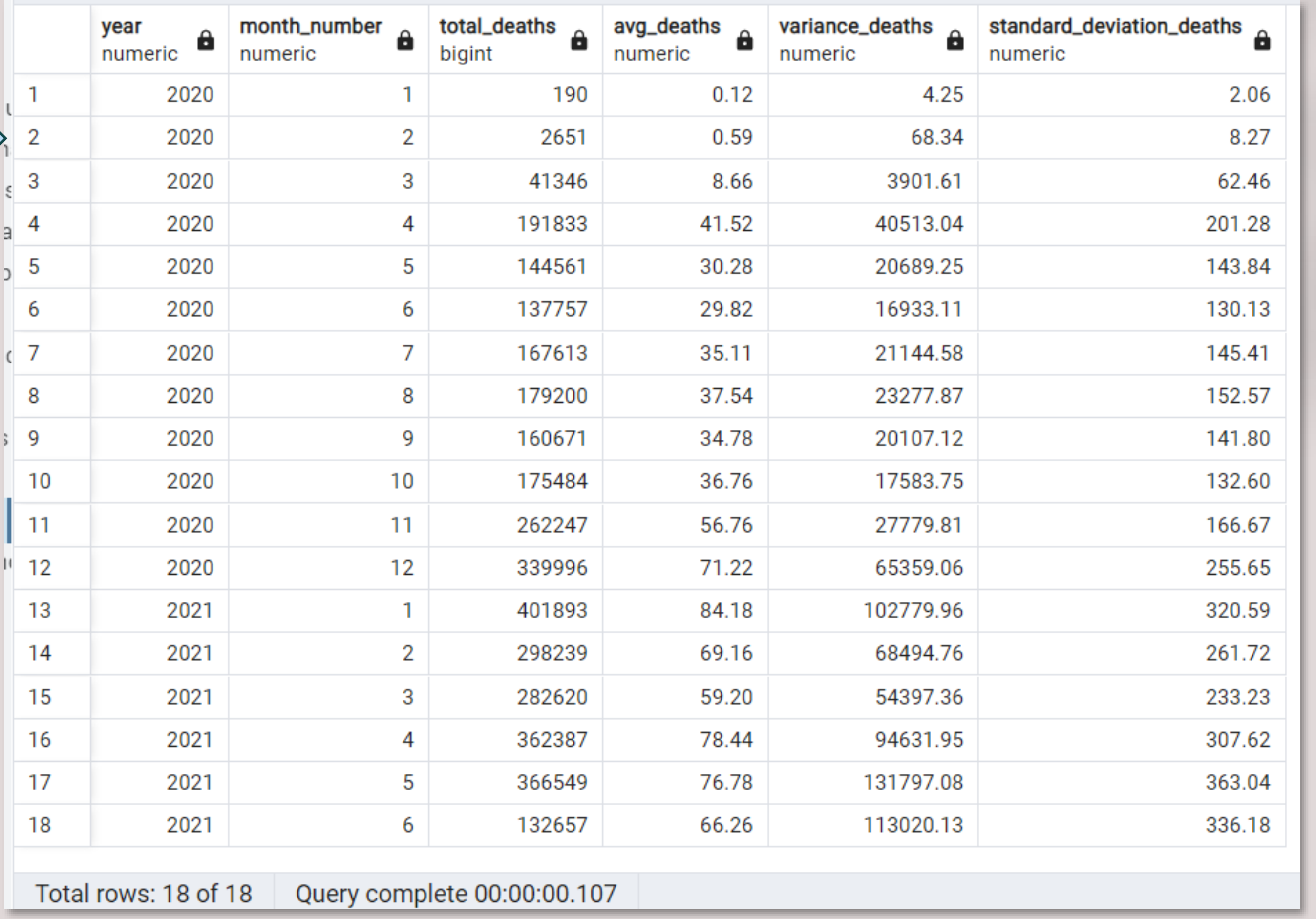
Insightful Queries

12. Check how coronavirus spread out with respect to death cases per month:

(Eg: total death cases, their average, variance & STDEV)



```sql
SELECT
    EXTRACT(YEAR FROM Date) AS year,
    EXTRACT(MONTH FROM Date) AS month_number,
    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_deviation_deaths
FROM coronaData
GROUP BY year, month_number
ORDER BY year, month_number;
```



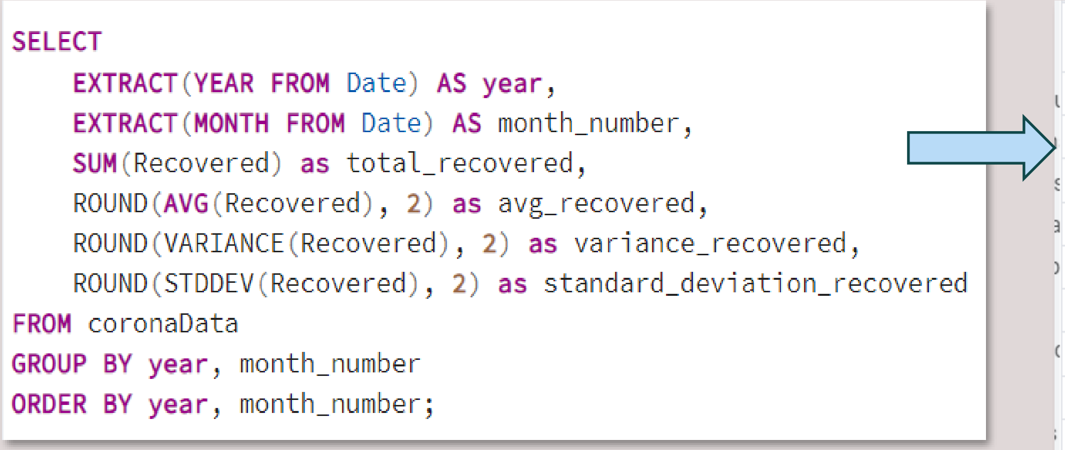
	year numeric	month_number numeric	total_deaths bigint	avg_deaths numeric	variance_deaths numeric	standard_deviation_deaths numeric
1	2020	1	190	0.12	4.25	2.06
2	2020	2	2651	0.59	68.34	8.27
3	2020	3	41346	8.66	3901.61	62.46
4	2020	4	191833	41.52	40513.04	201.28
5	2020	5	144561	30.28	20689.25	143.84
6	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

Total rows: 18 of 18 Query complete 00:00:00.107

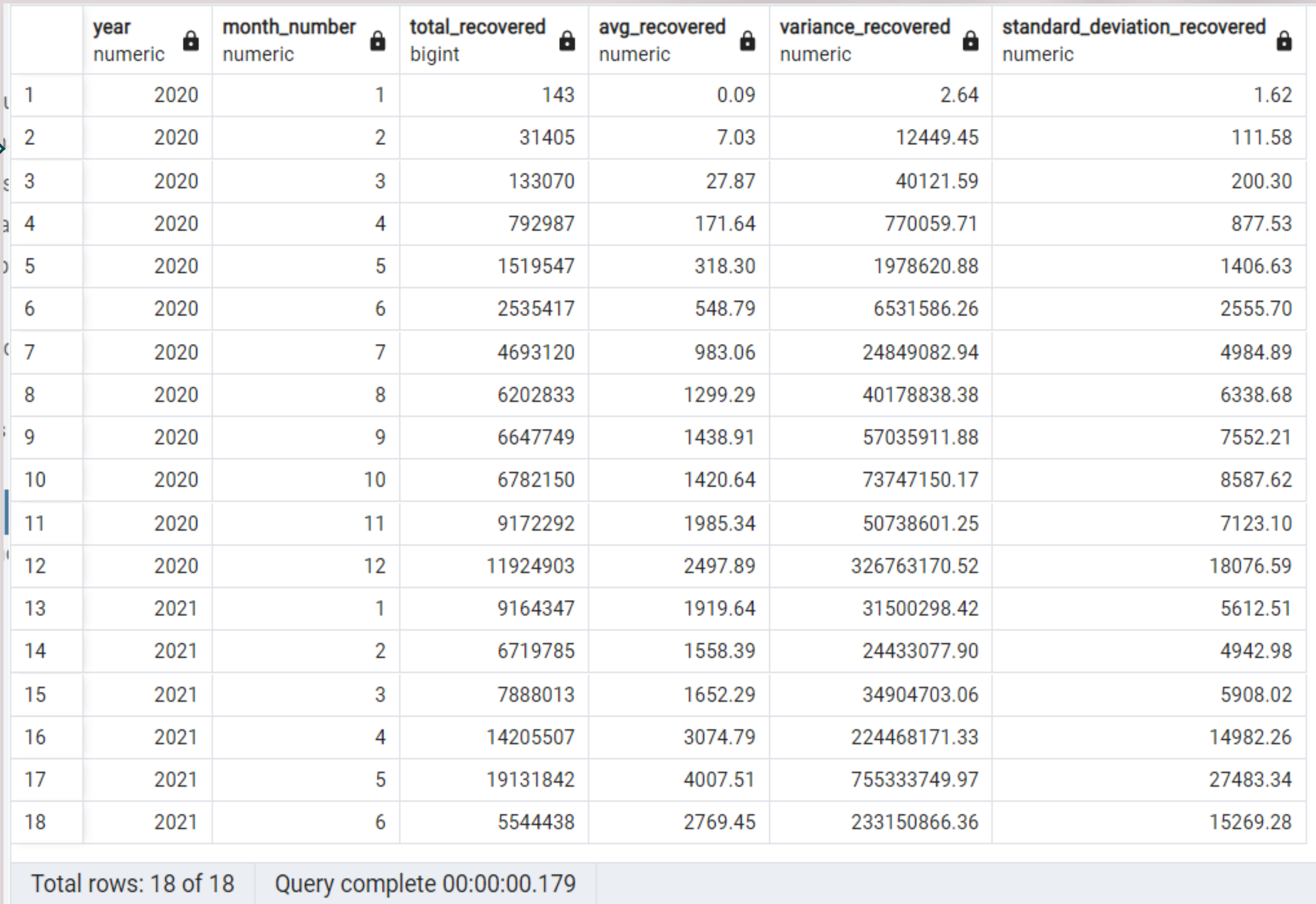
Insightful Queries

13. Check how coronavirus spread out with respect to recovered cases per month:

(Eg: total recovered cases, their average, variance & STDEV)



```
SELECT
    EXTRACT(YEAR FROM Date) AS year,
    EXTRACT(MONTH FROM Date) AS month_number,
    SUM(Recovered) as total_recovered,
    ROUND(AVG(Recovered), 2) as avg_recovered,
    ROUND(VARIANCE(Recovered), 2) as variance_recovered,
    ROUND(STDDEV(Recovered), 2) as standard_deviation_recovered
FROM coronaData
GROUP BY year, month_number
ORDER BY year, month_number;
```

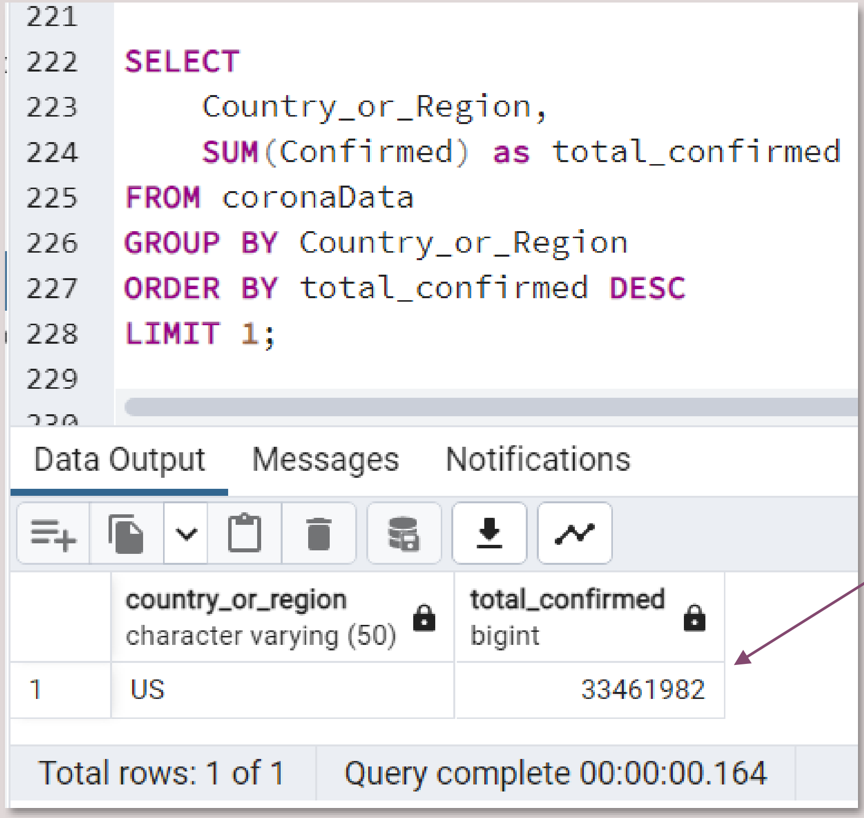


	year	month_number	total_recovered	avg_recovered	variance_recovered	standard_deviation_recovered
	numeric	numeric	bigint	numeric	numeric	numeric
1	2020	1	143	0.09	2.64	1.62
2	2020	2	31405	7.03	12449.45	111.58
3	2020	3	133070	27.87	40121.59	200.30
4	2020	4	792987	171.64	770059.71	877.53
5	2020	5	1519547	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

Total rows: 18 of 18   Query complete 00:00:00.179

Insightful Queries

14. Find the Country having the highest number of Confirmed cases:



```sql

SELECT
    Country_or_Region,
    SUM(Confirmed) as total_confirmed
FROM coronaData
GROUP BY Country_or_Region
ORDER BY total_confirmed DESC
LIMIT 1;

```

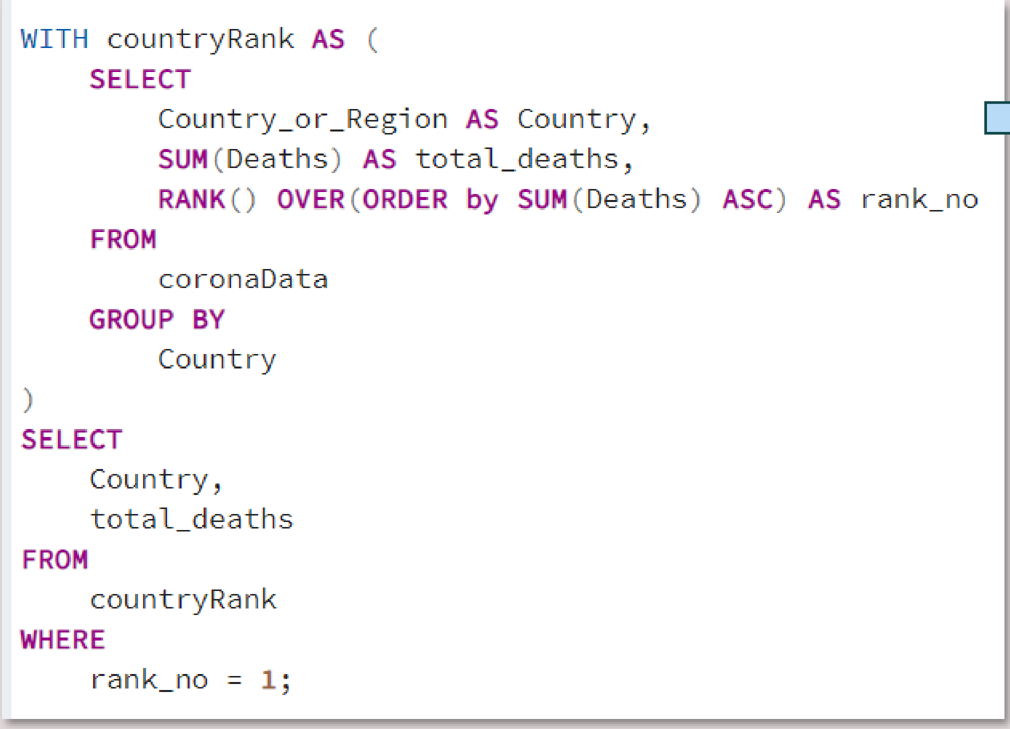
	country_or_region character varying (50)	total_confirmed bigint
1	US	33461982

Total rows: 1 of 1 Query complete 00:00:00.164

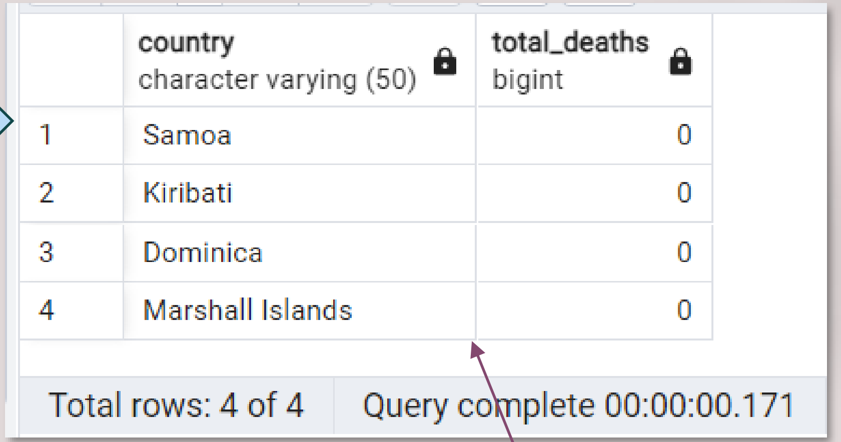
US is the country with the **highest** number of COVID-19 confirmed cases with an aggregate count of **3,34,61,982**

Insightful Queries

15. **Find the Country having the lowest number of death cases:**



```
WITH countryRank AS (
    SELECT
        Country_or_Region AS Country,
        SUM(Deaths) AS total_deaths,
        RANK() OVER(ORDER by SUM(Deaths) ASC) AS rank_no
    FROM
        coronaData
    GROUP BY
        Country
)
SELECT
    Country,
    total_deaths
FROM
    countryRank
WHERE
    rank_no = 1;
```



	country character varying (50)	total_deaths bigint
1	Samoa	0
2	Kiribati	0
3	Dominica	0
4	Marshall Islands	0

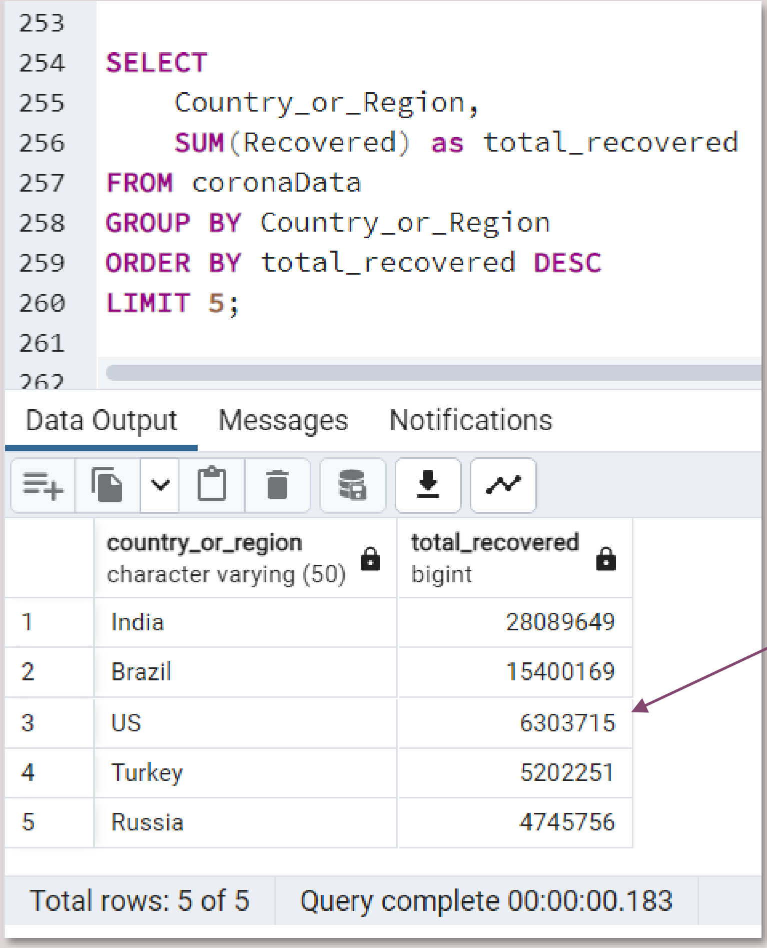
Total rows: 4 of 4   Query complete 00:00:00.171

The 4 countries with the **lowest death** count i.e. **0** deaths are:

- Samoa
- Kiribati
- Dominica
- Marshall Islands

Insightful Queries

16. Find the top 5 countries having the highest recovered cases:



```sql
SELECT
    Country_or_Region,
    SUM(Recovered) as total_recovered
FROM coronaData
GROUP BY Country_or_Region
ORDER BY total_recovered DESC
LIMIT 5;
```

	country_or_region character varying (50)	total_recovered bigint
1	India	28089649
2	Brazil	15400169
3	US	6303715
4	Turkey	5202251
5	Russia	4745756

Total rows: 5 of 5   Query complete 00:00:00.183

Top 5 countries with the **highest Recovered** COVID-19 cases are:

- India (topped the table)
- Brazil
- US
- Turkey
- Russia

Insights

After the detailed analysis of the COVID-19 dataset in SQL, we can draw several conclusions from it.

COVID-19 Pandemic Duration:
22nd January 2020 to 13 June 2021.

Highest Confirmed COVID-19 Cases in:
USA

Highest Recovered COVID-19 Cases in:
India

Peak Confirmed Cases in:
April 2021

Peak Death Rate in:
January 2021

Lowest Death Rates in:
Samoa, Kiribati, Dominica, Marshall Islands

Thank You