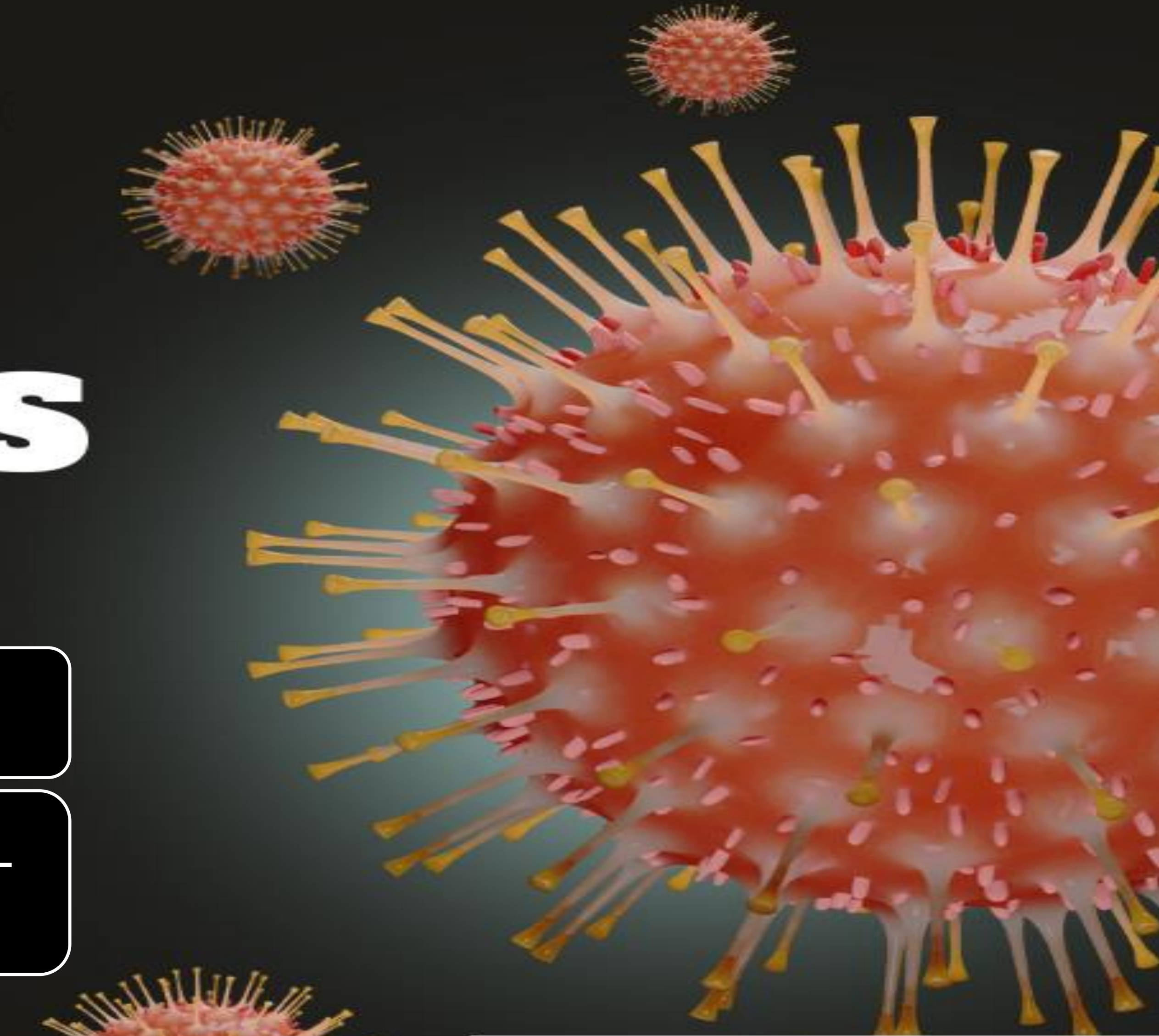
CORONA VIRUS ANALYSIS



Program: Internship with Mentorness:

Purpose: This project is designed to test my SQL and data analysis skills in a real-world context.

Data Analyst Intern: Omotosho Taiwo

Date: April to May , 2024

MENTOR NESS

COVID-19 Data Analysis Project

Project Overview

This project involves analyzing a COVID-19 dataset using SQL and data analysis skills. The task is to derive insights from the dataset containing information such as geographic location, dates, confirmed cases, deaths, and recoveries. Detailed tasks are outlined in a reference file with 16 questions for analysis. The dataset contains **78,386 rows** and **8 columns**, representing COVID-19 data collected between **2020 and 2021**. It includes information from **121 unique countries/regions** and **153 provinces**, along with their geographic coordinates (**latitude and longitude**).

Objective

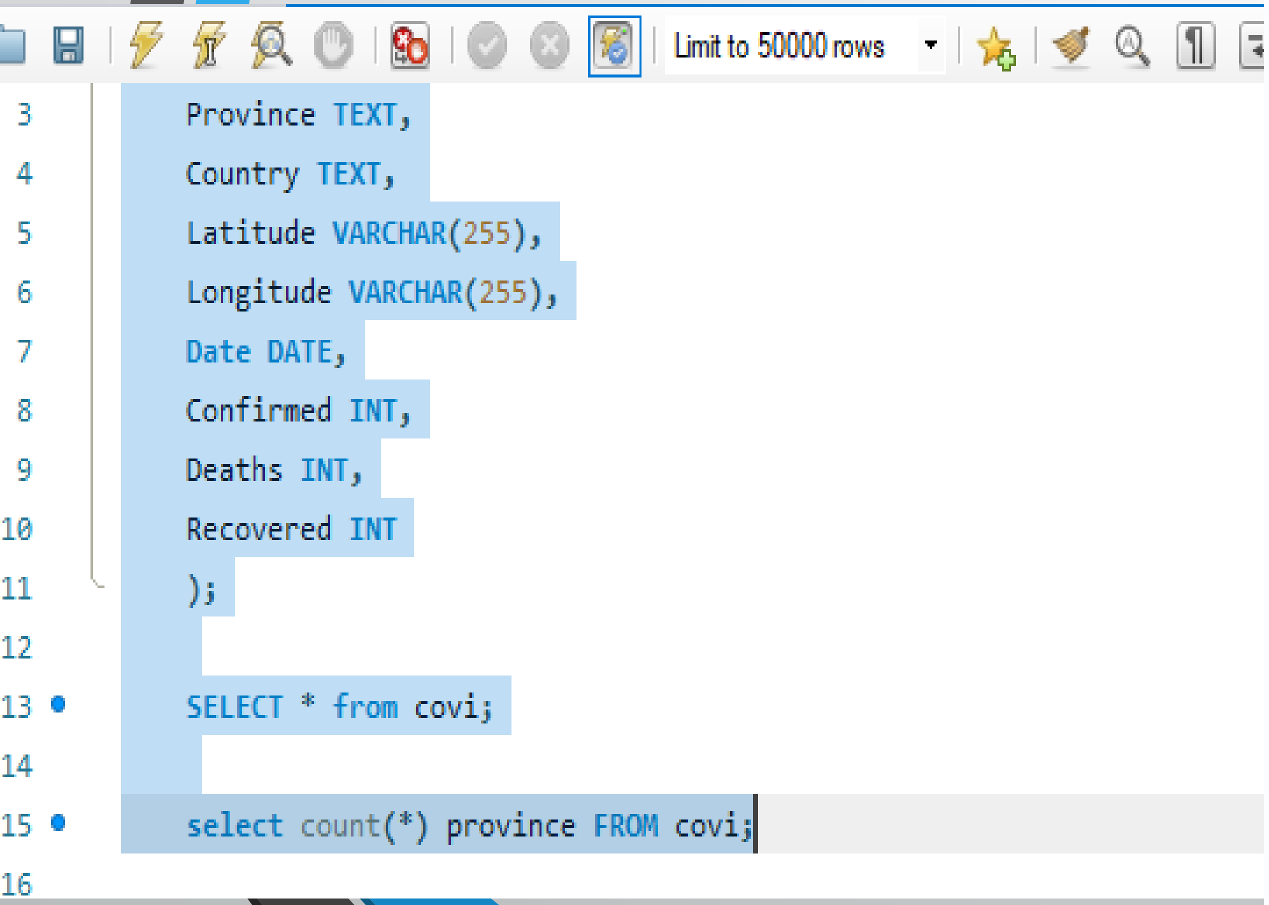
This project aims to derive meaningful insights from a COVID-19 dataset using SQL. By querying and analyzing the data with PostgreSQL, we seek to uncover patterns related to case numbers, recoveries, fatalities, and regional impacts, providing valuable insights for public health strategies.

QUESTION ONE: CHECK FOR ANY NULL VALUE

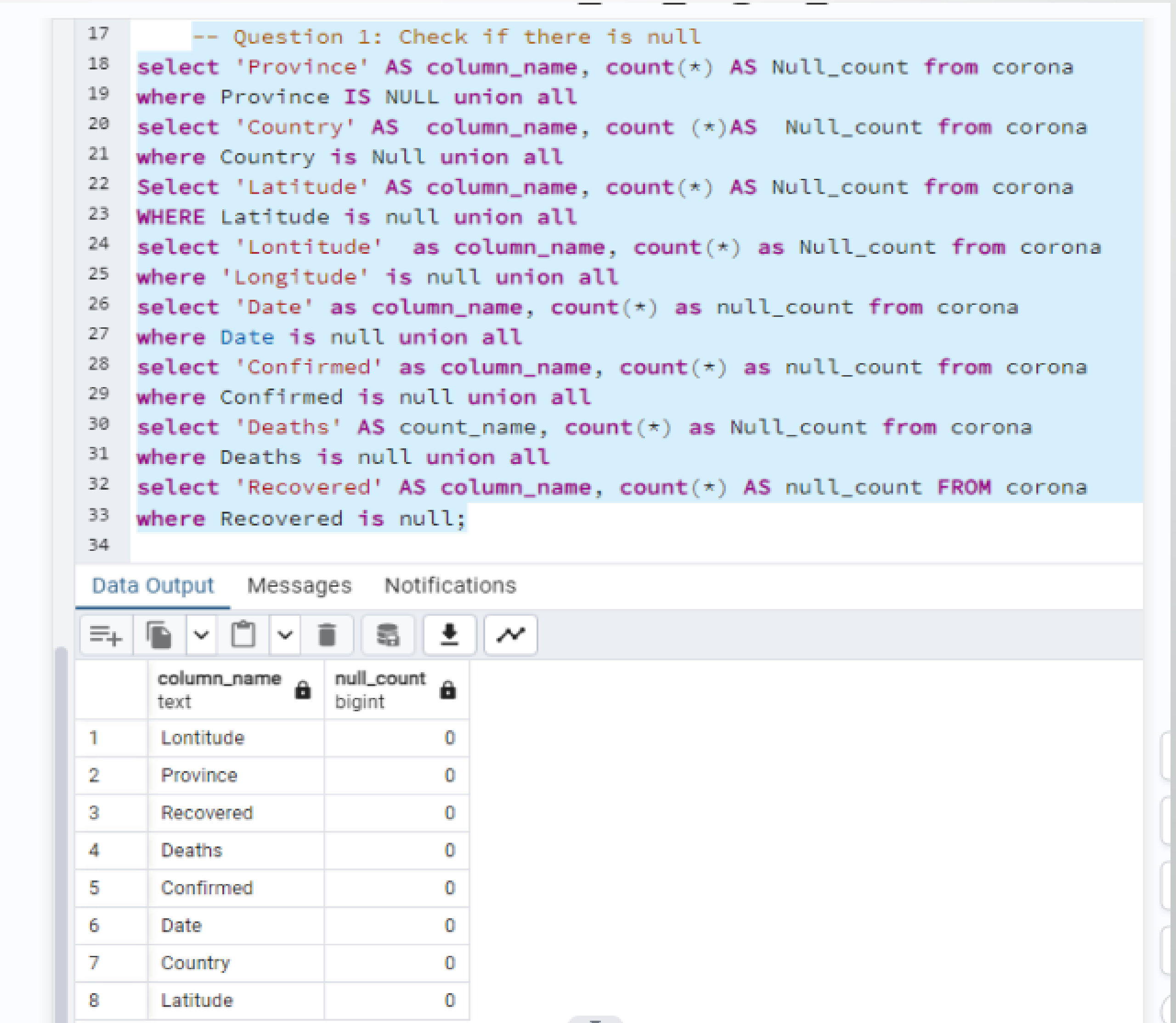
QUESTION TWO: IF NULL VALUES ARE PRESENT, UPDATE THEM WITH ZEROS FOR ALL COLUMNS

Answer:

The start date is January 1st, 2020, end date is June 13th, 2021.



```
    Province TEXT,
    Country TEXT,
    Latitude VARCHAR(255),
    Longitude VARCHAR(255),
    Date DATE,
    Confirmed INT,
    Deaths INT,
    Recovered INT
    );

    SELECT * from covi;

    select count(*) province FROM covi;
```

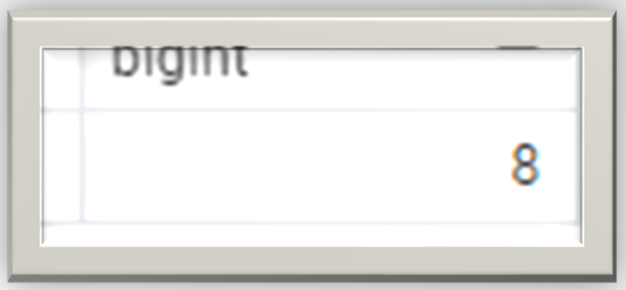


```
        -- Question 1: Check if there is null
select 'Province' AS column_name, count(*) AS Null_count from corona
where Province IS NULL union all
select 'Country' AS  column_name, count (*)AS  Null_count from corona
where Country is Null union all
Select 'Latitude' AS column_name, count(*) AS Null_count from corona
WHERE Latitude is null union all
select 'Lontitude'  as column_name, count(*) as Null_count from corona
where 'Longitude' is null union all
select 'Date' as column_name, count(*) as null_count from corona
where Date is null union all
select 'Confirmed' as column_name, count(*) as null_count from corona
where Confirmed is null union all
select 'Deaths' AS count_name, count(*) as Null_count from corona
where Deaths is null union all
select 'Recovered' AS column_name, count(*) AS null_count FROM corona
where Recovered is null;
```

	column_name text	null_count bigint
1	Lontitude	0
2	Province	0
3	Recovered	0
4	Deaths	0
5	Confirmed	0
6	Date	0
7	Country	0
8	Latitude	0

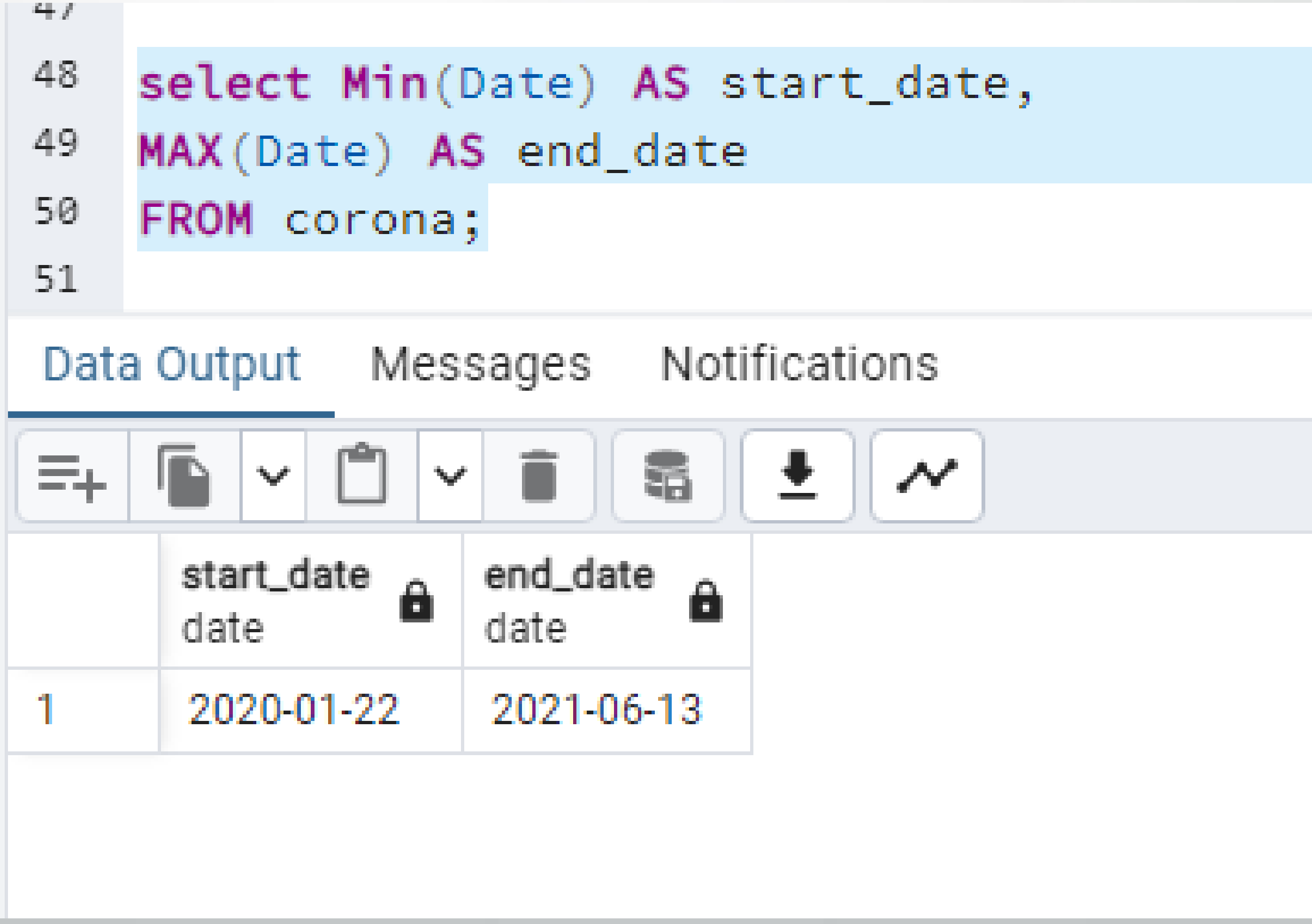
QUESTION 3: check total number of rows

- **ANSWER: Total number is 8**



bigint
8

Question 4: Check what is start_date and end_date



```
select Min(Date) AS start_date,
MAX(Date) AS end_date
FROM corona;
```

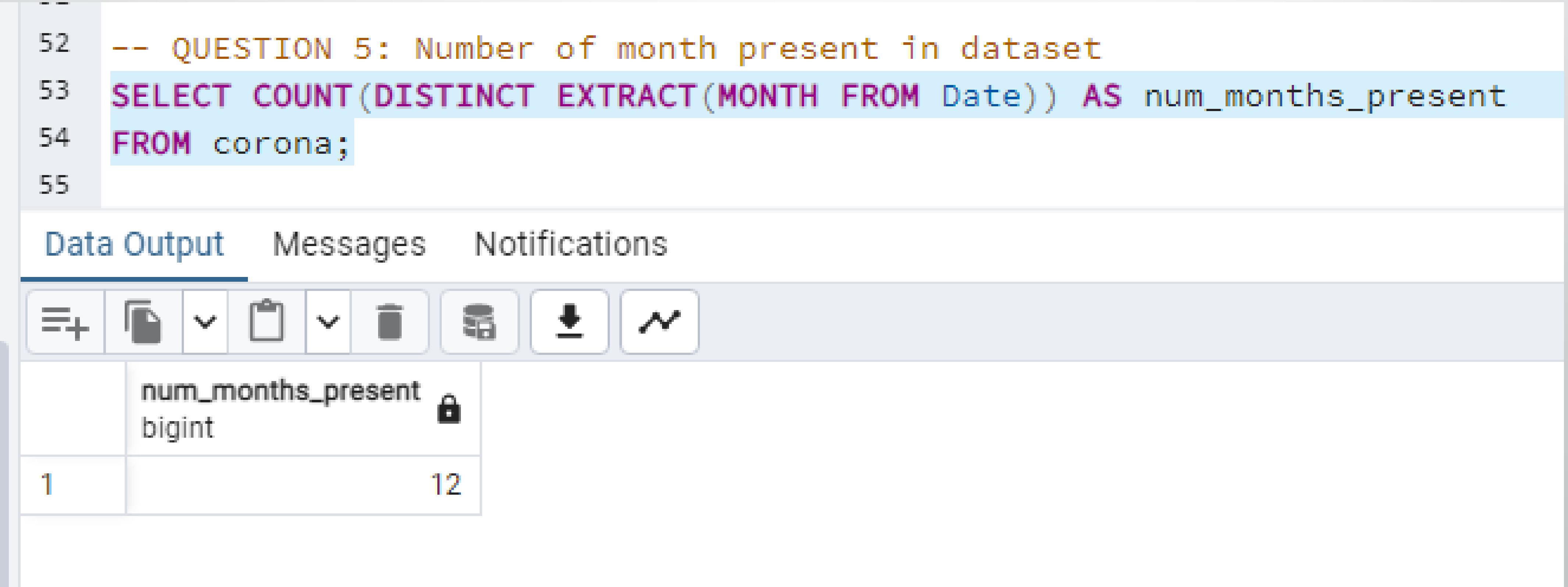
	start_date date	end_date date
1	2020-01-22	2021-06-13

Answer:
The start date 22nd January 2020, the end date is June 13th, 2021.

Question 5:
What is the Number of month present in dataset

Answer:

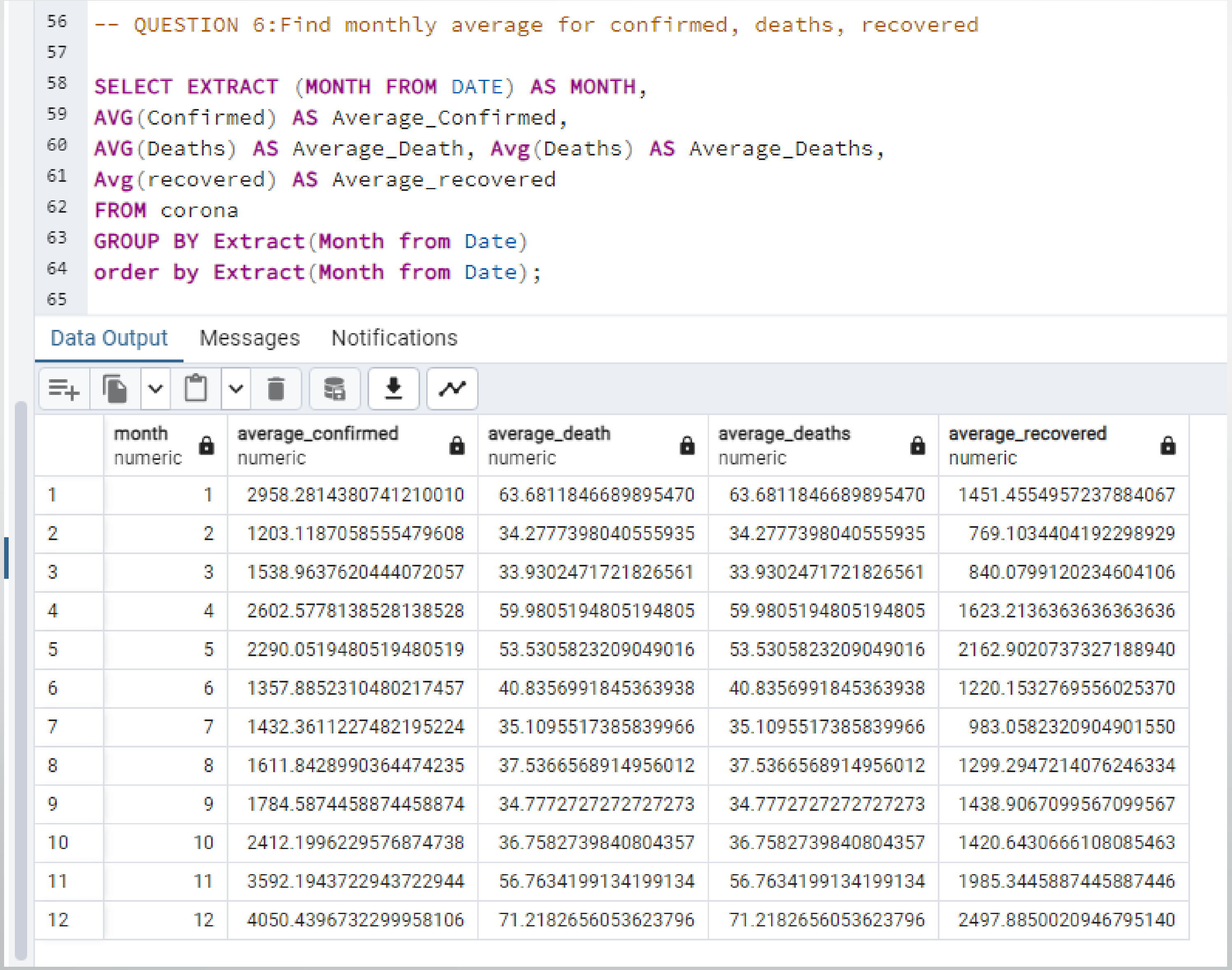
12 months from January to December.



```
-- QUESTION 5: Number of month present in dataset
SELECT COUNT(DISTINCT EXTRACT(MONTH FROM Date)) AS num_months_present
FROM corona;
```

	num_months_present bigint
1	12

Question 6: Find a monthly average for confirmed, deaths, recovered

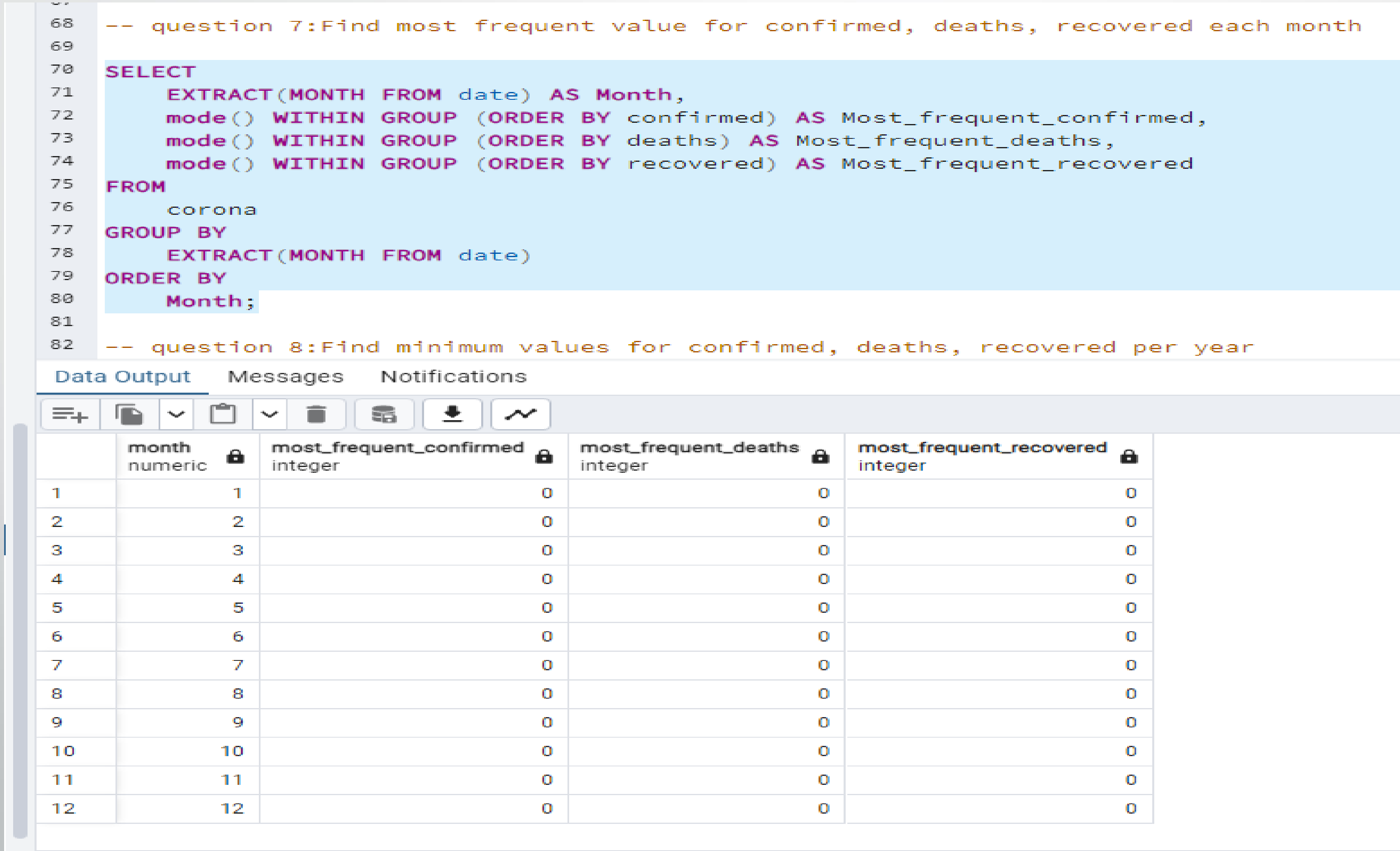


```
-- QUESTION 6:Find monthly average for confirmed, deaths, recovered

SELECT EXTRACT (MONTH FROM DATE) AS MONTH,
AVG(Confirmed) AS Average_Confirmed,
AVG(Deaths) AS Average_Death, Avg(Deaths) AS Average_Deaths,
Avg(recovered) AS Average_recovered
FROM corona
GROUP BY Extract(Month from Date)
order by Extract(Month from Date);
```

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

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

Answer: The most frequent value is Zero

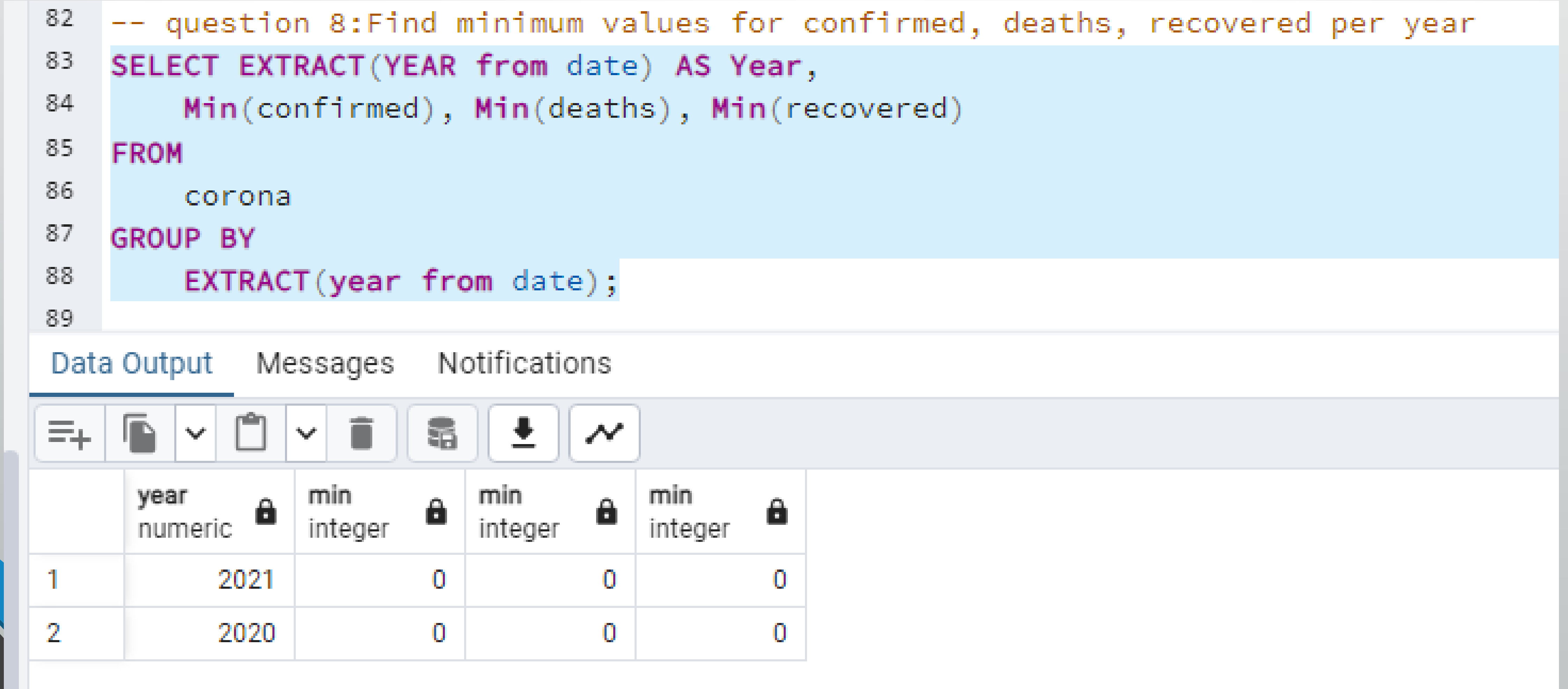


```sql
-- question 7:Find most frequent value for confirmed, deaths, recovered each month

SELECT
    EXTRACT(MONTH FROM date) AS Month,
    mode() WITHIN GROUP (ORDER BY confirmed) AS Most_frequent_confirmed,
    mode() WITHIN GROUP (ORDER BY deaths) AS Most_frequent_deaths,
    mode() WITHIN GROUP (ORDER BY recovered) AS Most_frequent_recovered
FROM
    corona
GROUP BY
    EXTRACT(MONTH FROM date)
ORDER BY
    Month;

-- question 8:Find minimum values for confirmed, deaths, recovered per year
```

	month numeric	most_frequent_confirmed integer	most_frequent_deaths integer	most_frequent_recovered integer
1	1	0	0	0
2	2	0	0	0
3	3	0	0	0
4	4	0	0	0
5	5	0	0	0
6	6	0	0	0
7	7	0	0	0
8	8	0	0	0
9	9	0	0	0
10	10	0	0	0
11	11	0	0	0
12	12	0	0	0

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

Answer: The minimum for confirmed, deaths and recovered cases is zero frequent value is Zero

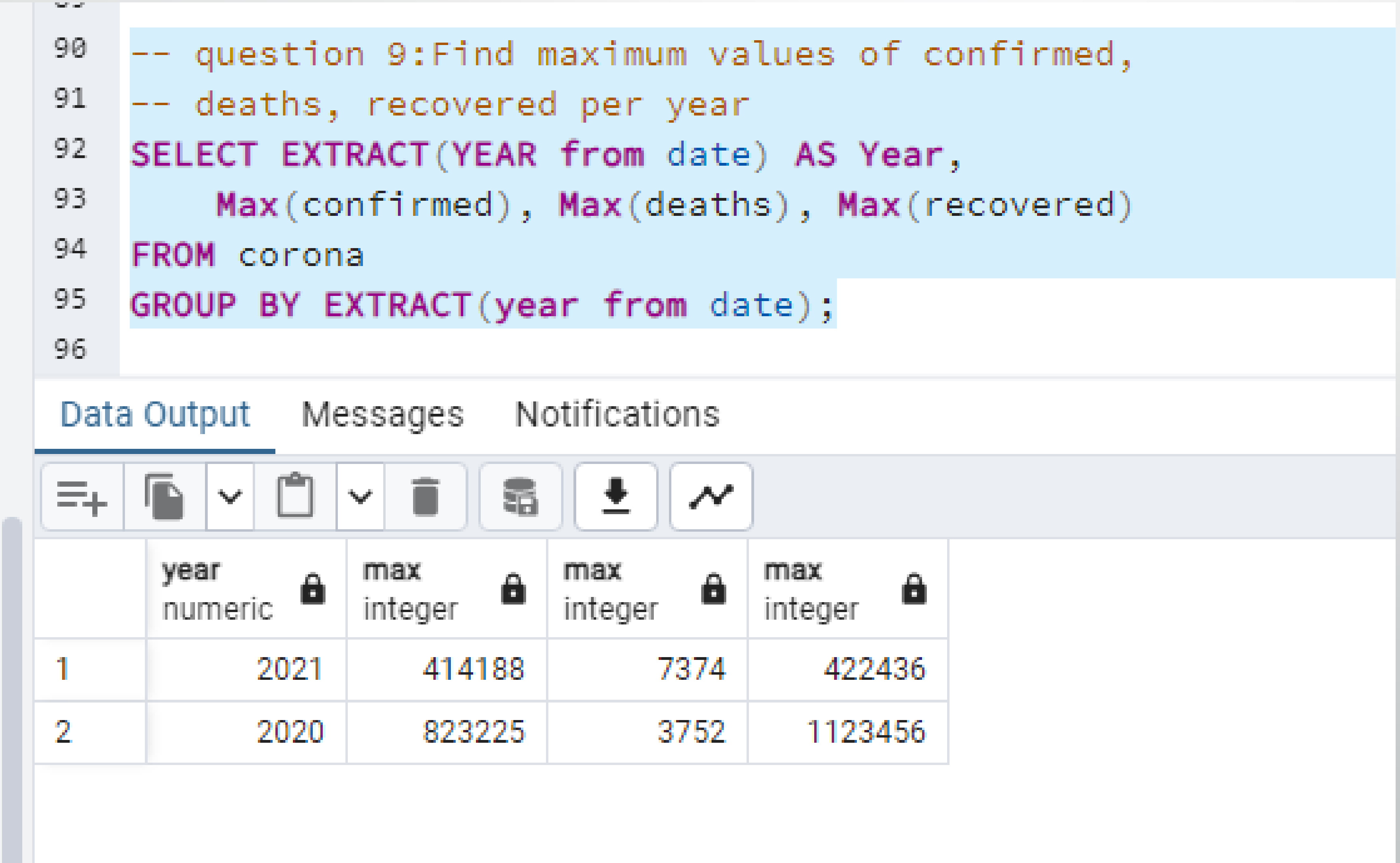


```
-- question 8:Find minimum values for confirmed, deaths, recovered per year
SELECT EXTRACT(YEAR from date) AS Year,
    Min(confirmed), Min(deaths), Min(recovered)
FROM
    corona
GROUP BY
    EXTRACT(year from date);
```

	year numeric	min integer	min integer	min integer
1	2021	0	0	0
2	2020	0	0	0

Question 9: Find maximum values of confirmed, deaths, recovered per year

Answer: 2020 had the highest confirmed and recovered case, while 2021 had the highest death case.

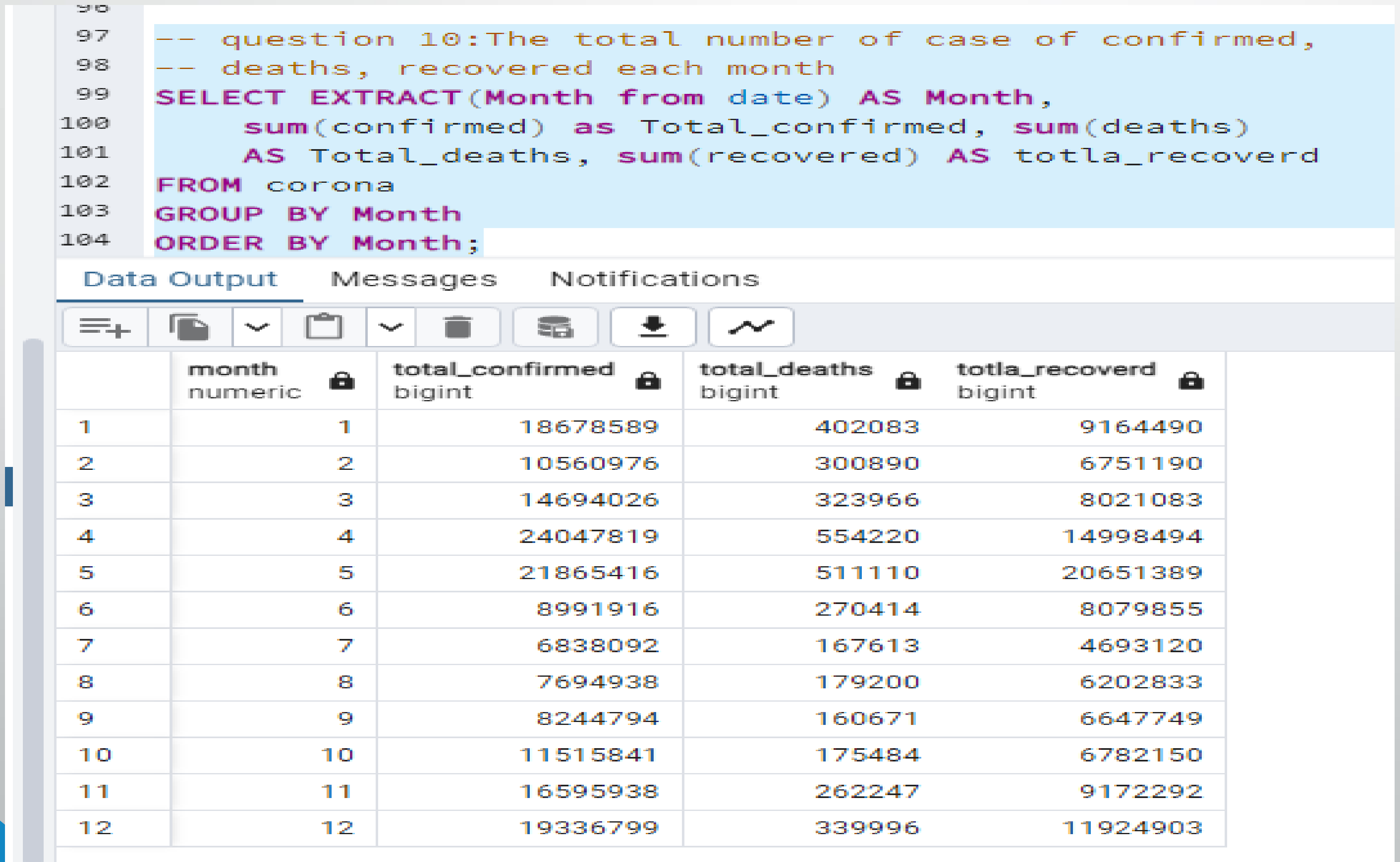


```sql
-- question 9:Find maximum values of confirmed,
-- deaths, recovered per year
SELECT EXTRACT(YEAR from date) AS Year,
    Max(confirmed), Max(deaths), Max(recovered)
FROM corona
GROUP BY EXTRACT(year from date);
```

	year numeric	max integer	max integer	max integer
1	2021	414188	7374	422436
2	2020	823225	3752	1123456

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

Answer: The month of April had the highest number of confirmed case and death case, and it still had the highest recovered case.



```
-- question 10:The total number of case of confirmed,
-- deaths, recovered each month
SELECT EXTRACT(Month from date) AS Month,
    sum(confirmed) as Total_confirmed, sum(deaths)
    AS Total_deaths, sum(recovered) AS totla_recoverd
FROM corona
GROUP BY Month
ORDER BY Month;
```

	month numeric	total_confirmed bigint	total_deaths bigint	totla_recoverd bigint
1	1	18678589	402083	9164490
2	2	10560976	300890	6751190
3	3	14694026	323966	8021083
4	4	24047819	554220	14998494
5	5	21865416	511110	20651389
6	6	8991916	270414	8079855
7	7	6838092	167613	4693120
8	8	7694938	179200	6202833
9	9	8244794	160671	6647749
10	10	11515841	175484	6782150
11	11	16595938	262247	9172292
12	12	19336799	339996	11924903

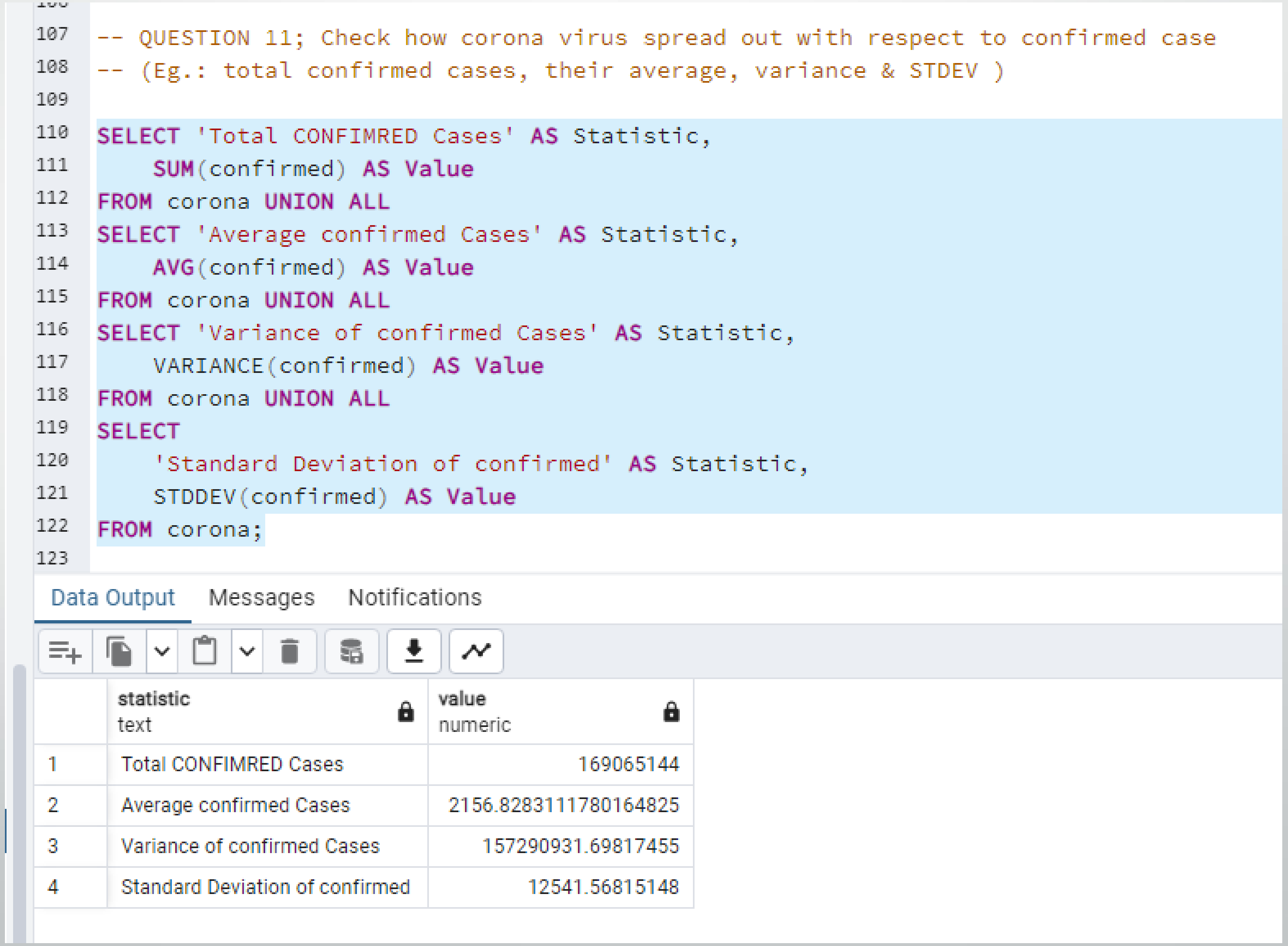
Question 11: Check how corona virus spread out with respect to confirmed case (Eg.: total confirmed cases, their average, variance & STDEV)

Total confirmed cases: 169,065,144

Average confirmed cases: 2,156.828

Variance of confirmed cases: 157,290,931.698

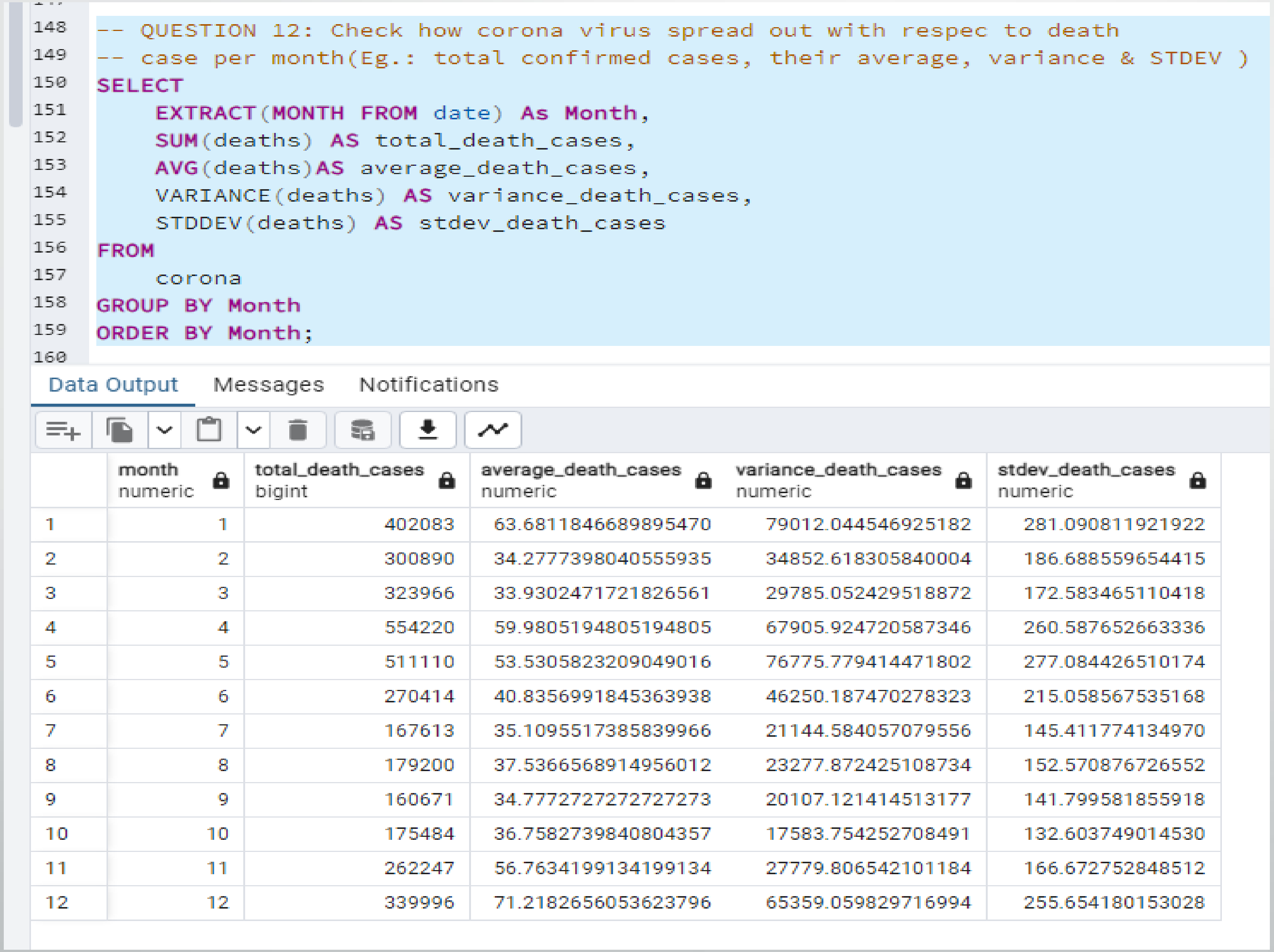
Stdev of confirmed cases: 12,541.568



```sql
-- QUESTION 11; Check how corona virus spread out with respect to confirmed case
-- (Eg.: total confirmed cases, their average, variance & STDEV )

SELECT 'Total CONFIMRED Cases' AS Statistic,
    SUM(confirmed) AS Value
FROM corona UNION ALL
SELECT 'Average confirmed Cases' AS Statistic,
    AVG(confirmed) AS Value
FROM corona UNION ALL
SELECT 'Variance of confirmed Cases' AS Statistic,
    VARIANCE(confirmed) AS Value
FROM corona UNION ALL
SELECT
    'Standard Deviation of confirmed' AS Statistic,
    STDDEV(confirmed) AS Value
FROM corona;
```

	statistic text	value numeric
1	Total CONFIMRED Cases	169065144
2	Average confirmed Cases	2156.8283111780164825
3	Variance of confirmed Cases	157290931.69817455
4	Standard Deviation of confirmed	12541.56815148

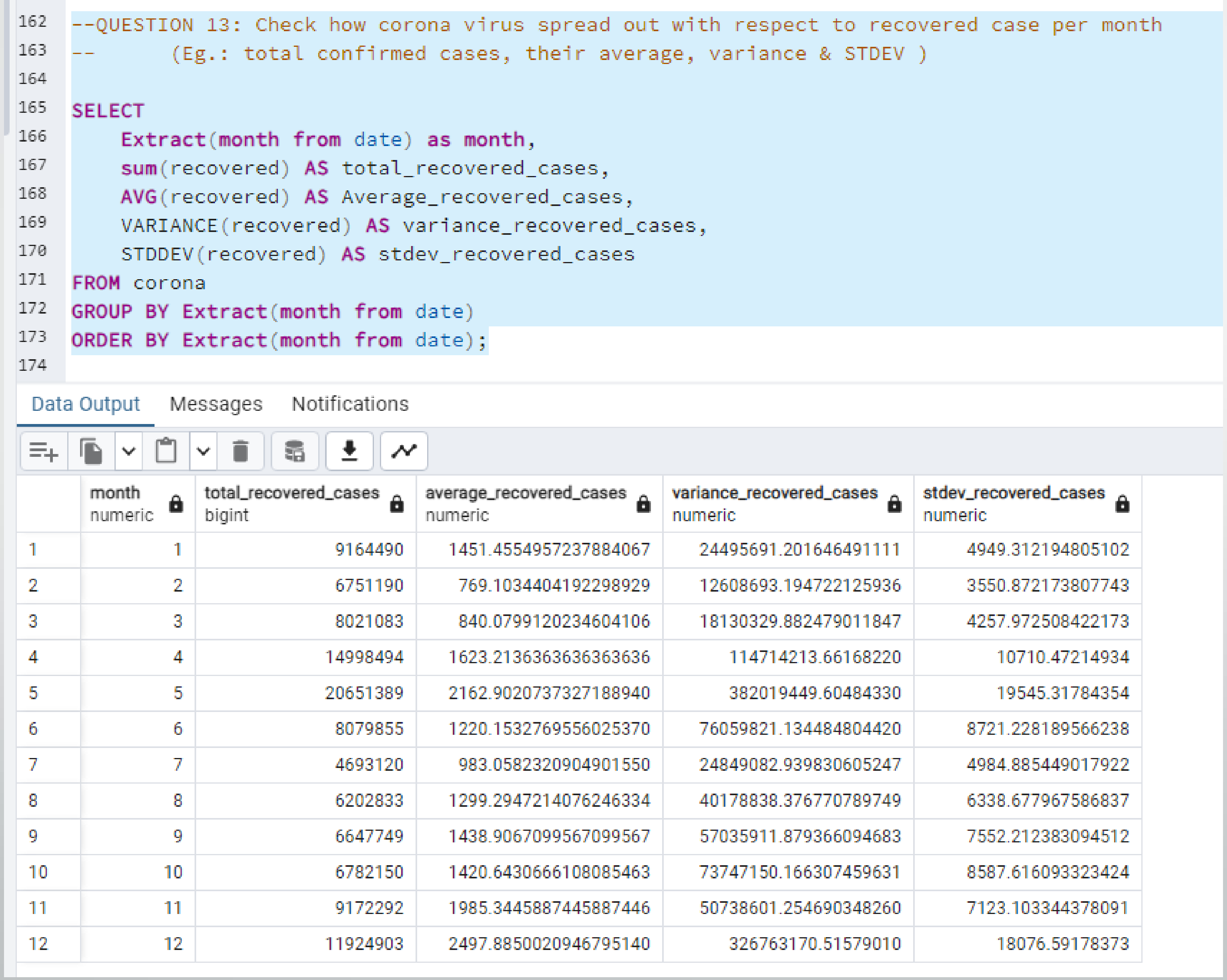
Question 12: Check how coronavirus spread out for death case per month (Eg: total confirmed cases, their average, variance & STDEV)



```sql
-- QUESTION 12: Check how corona virus spread out with respec to death
-- case per month(Eg.: total confirmed cases, their average, variance & STDEV )
SELECT
    EXTRACT(MONTH FROM date) As Month,
    SUM(deaths) AS total_death_cases,
    AVG(deaths)AS average_death_cases,
    VARIANCE(deaths) AS variance_death_cases,
    STDDEV(deaths) AS stdev_death_cases
FROM
    corona
GROUP BY Month
ORDER BY Month;
```

	month numeric	total_death_cases bigint	average_death_cases numeric	variance_death_cases numeric	stdev_death_cases numeric
1	1	402083	63.6811846689895470	79012.044546925182	281.090811921922
2	2	300890	34.2777398040555935	34852.618305840004	186.688559654415
3	3	323966	33.9302471721826561	29785.052429518872	172.583465110418
4	4	554220	59.9805194805194805	67905.924720587346	260.587652663336
5	5	511110	53.5305823209049016	76775.779414471802	277.084426510174
6	6	270414	40.8356991845363938	46250.187470278323	215.058567535168
7	7	167613	35.1095517385839966	21144.584057079556	145.411774134970
8	8	179200	37.5366568914956012	23277.872425108734	152.570876726552
9	9	160671	34.7772727272727273	20107.121414513177	141.799581855918
10	10	175484	36.7582739840804357	17583.754252708491	132.603749014530
11	11	262247	56.7634199134199134	27779.806542101184	166.672752848512
12	12	339996	71.2182656053623796	65359.059829716994	255.654180153028

Question 13: Check how corona virus spread out for recovered cases (Eg.: total confirmed cases, their average, variance & STDEV)

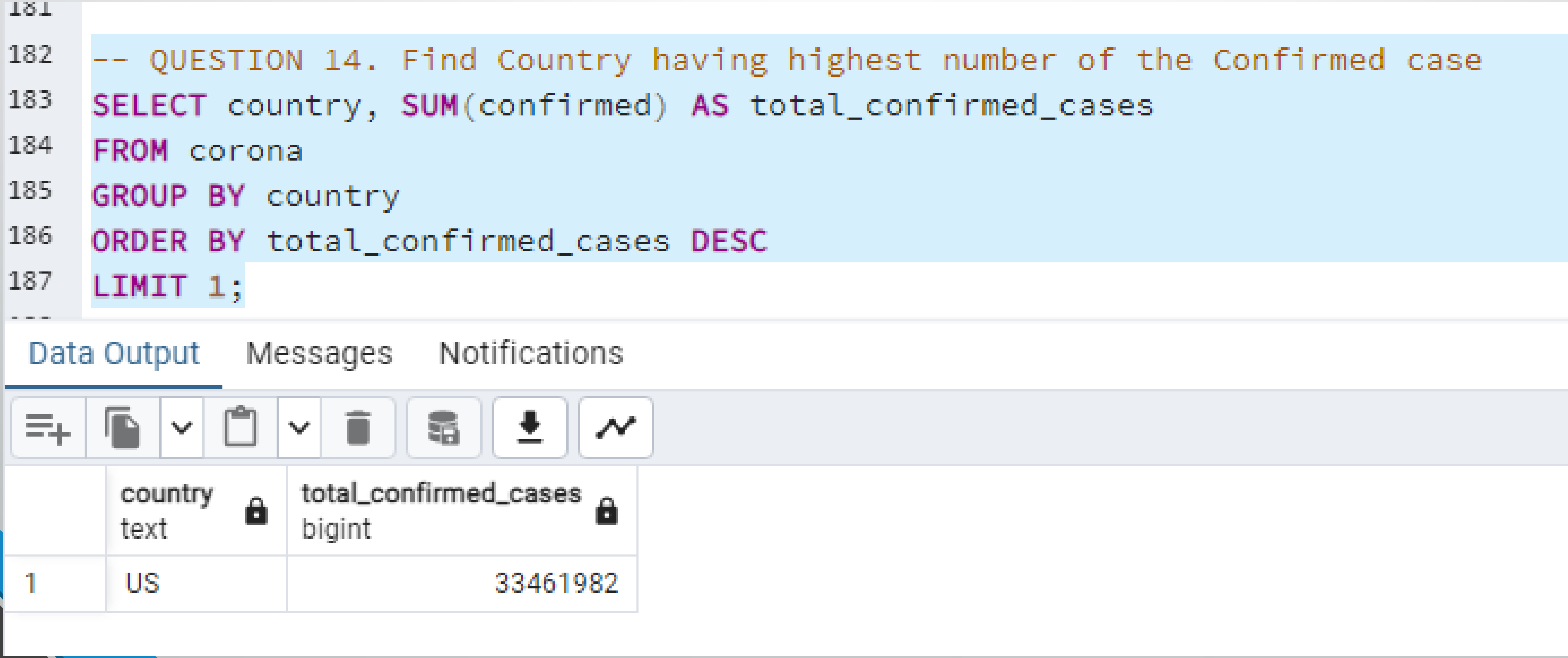


```sql
--QUESTION 13: Check how corona virus spread out with respect to recovered case per month
--      (Eg.: total confirmed cases, their average, variance & STDEV )

SELECT
    Extract(month from date) as month,
    sum(recovered) AS total_recovered_cases,
    AVG(recovered) AS Average_recovered_cases,
    VARIANCE(recovered) AS variance_recovered_cases,
    STDDEV(recovered) AS stdev_recovered_cases
FROM corona
GROUP BY Extract(month from date)
ORDER BY Extract(month from date);

```

	month numeric	total_recovered_cases bigint	average_recovered_cases numeric	variance_recovered_cases numeric	stdev_recovered_cases numeric
1	1	9164490	1451.4554957237884067	24495691.201646491111	4949.312194805102
2	2	6751190	769.1034404192298929	12608693.194722125936	3550.872173807743
3	3	8021083	840.0799120234604106	18130329.882479011847	4257.972508422173
4	4	14998494	1623.2136363636363636	114714213.66168220	10710.47214934
5	5	20651389	2162.9020737327188940	382019449.60484330	19545.31784354
6	6	8079855	1220.1532769556025370	76059821.134484804420	8721.228189566238
7	7	4693120	983.0582320904901550	24849082.939830605247	4984.885449017922
8	8	6202833	1299.2947214076246334	40178838.376770789749	6338.677967586837
9	9	6647749	1438.9067099567099567	57035911.879366094683	7552.212383094512
10	10	6782150	1420.6430666108085463	73747150.166307459631	8587.616093323424
11	11	9172292	1985.3445887445887446	50738601.254690348260	7123.103344378091
12	12	11924903	2497.8850020946795140	326763170.51579010	18076.59178373

Question 14: Find the Country having the highest number of Confirmed case

Answer: The United States had the highest number of 33461982 confirmed cases.

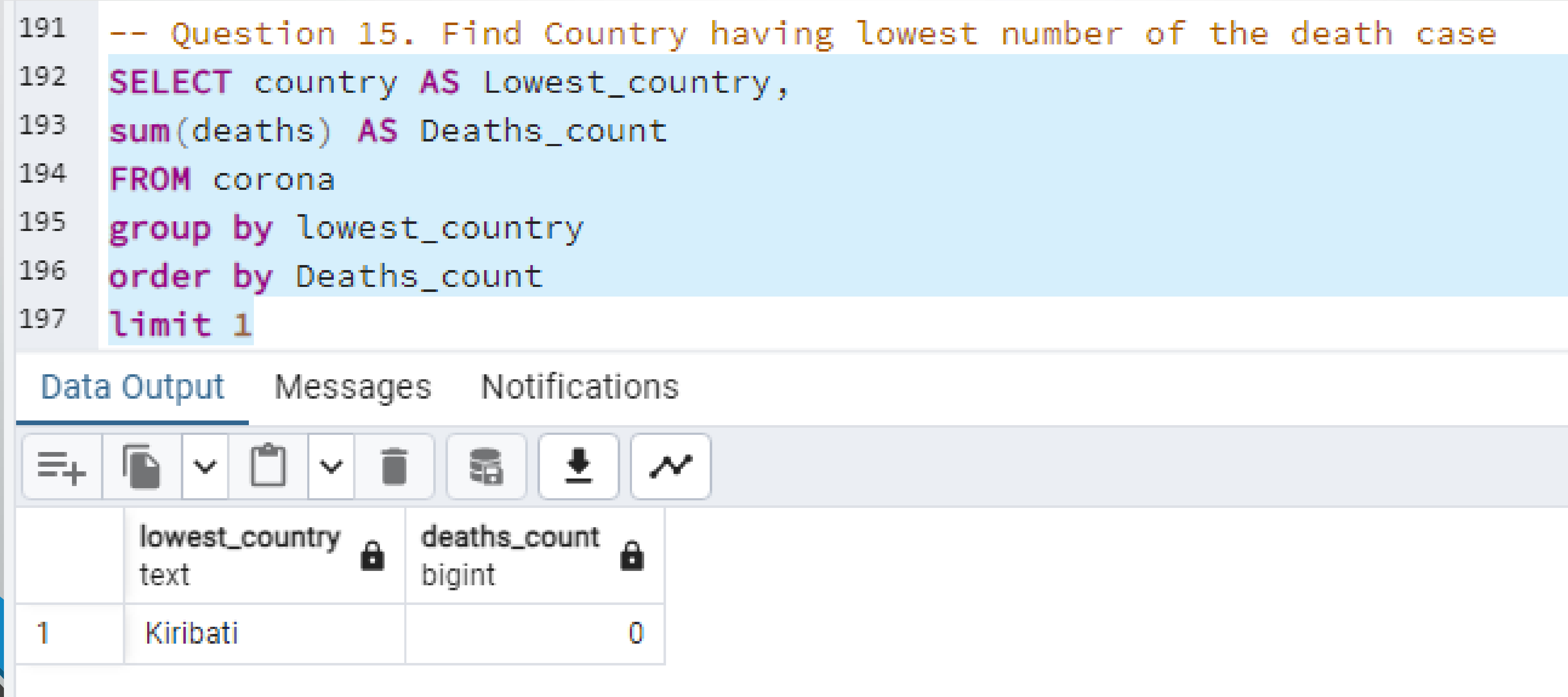


```sql
-- QUESTION 14. Find Country having highest number of the Confirmed case
SELECT country, SUM(confirmed) AS total_confirmed_cases
FROM corona
GROUP BY country
ORDER BY total_confirmed_cases DESC
LIMIT 1;
```

	country text	total_confirmed_cases bigint
1	US	33461982

Question 15: Find the Country having the lowest number of death case

Answer: Kiribati had the lowest number of death cases with zero cases.



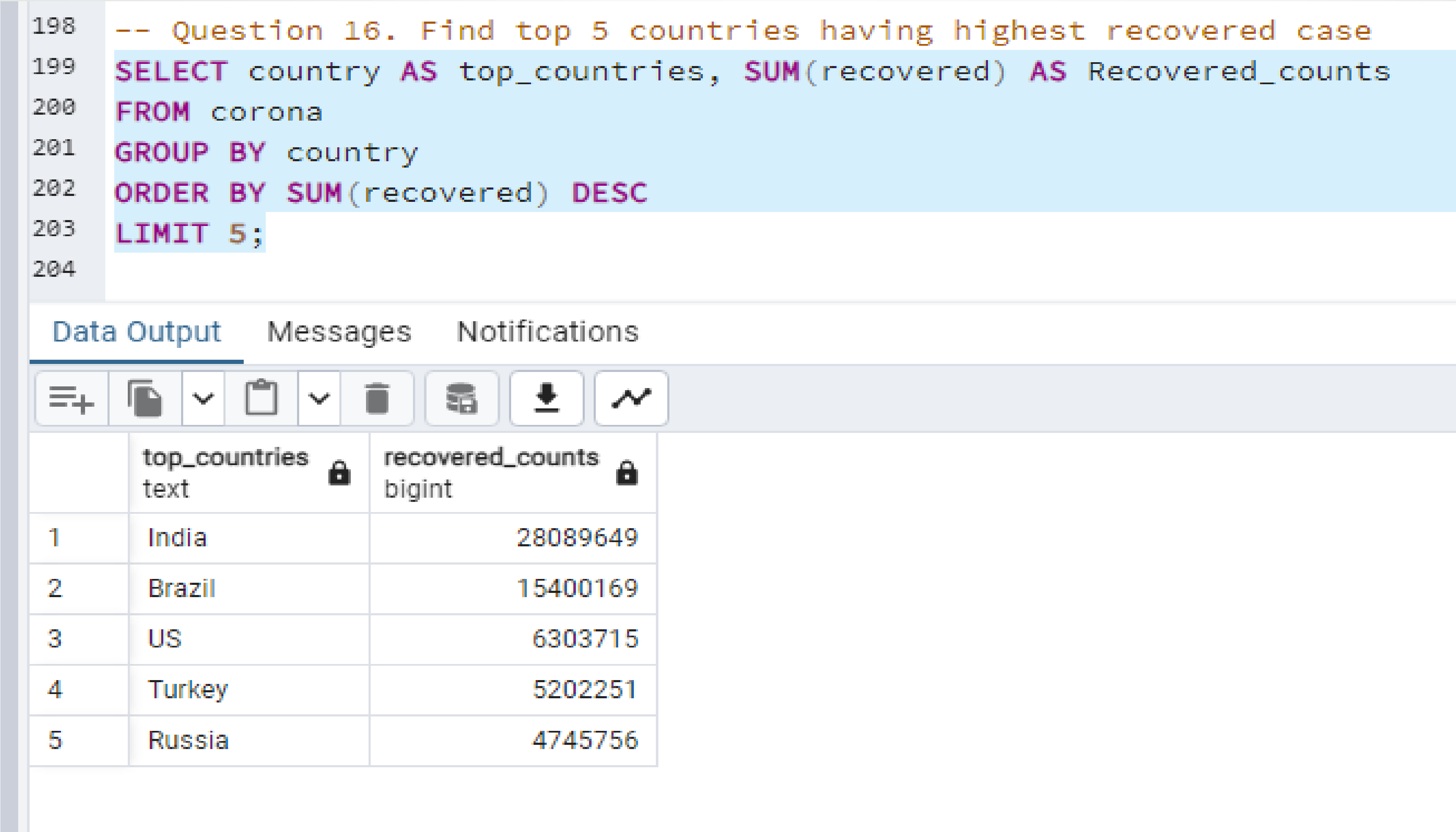
```sql
-- Question 15. Find Country having lowest number of the death case
SELECT country AS Lowest_country,
sum(deaths) AS Deaths_count
FROM corona
group by lowest_country
order by Deaths_count
limit 1
```

	lowest_country text	deaths_count bigint
1	Kiribati	0

Question 16: Find top 5 countries having highest recovered case

Answer:

India had the highest recovered cases of 28089649, followed by Brazil with 15400169, followed by the US with 6303715, then turkey with 5202251, and lastly Russia with 4745756.



```sql
-- Question 16. Find top 5 countries having highest recovered case
SELECT country AS top_countries, SUM(recovered) AS Recovered_counts
FROM corona
GROUP BY country
ORDER BY SUM(recovered) DESC
LIMIT 5;
```

	top_countries text	recovered_counts bigint
1	India	28089649
2	Brazil	15400169
3	US	6303715
4	Turkey	5202251
5	Russia	4745756

Conclusion & Recommendations

Conclusion:

•SQL analysis of COVID-19 data reveals trends in infection rates, mortality, and vaccination progress.

•Early intervention, high testing, and vaccination led to lower fatality rates.

•Delayed response and limited healthcare resources worsened outbreaks.

Recommendations:

✓ **Early Detection & Response** – Strengthen real-time data monitoring.

✓ **Boost Vaccination Campaigns** – Increase awareness & accessibility.

✓ **Improve Healthcare Infrastructure** – Expand testing & hospital capacity.

✓ **Use Data-Driven Decisions** – SQL dashboards for forecasting & resource allocation.

✓ **Encourage Global Collaboration** – Share real-time data for better outbreak control.