

[Back](#)

Insights from Data with BigQuery: Challenge Lab

Task 1. Total confirmed cases

- Build a query that will answer "What was the total count of confirmed cases on May 10, 2020 ?" The query needs to return a single row containing the sum of confirmed cases across all countries. The name of the column should be **total_cases_worldwide**.

Columns to reference:

- cumulative_confirmed
- date

SELECT

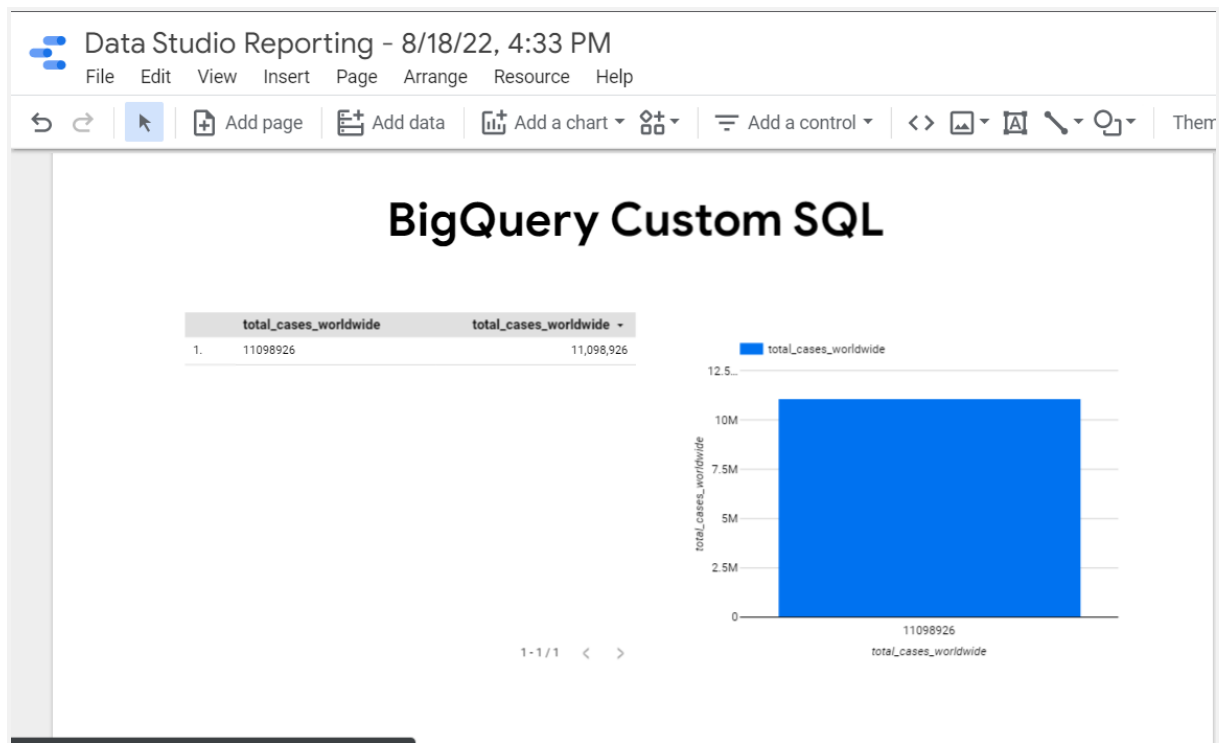
SUM(cumulative_confirmed) AS total_cases_worldwide

FROM

`bigquery-public-data.covid19_open_data.covid19_open_data`

WHERE

date = "2020-05-10"



Task 2. Worst affected areas

- Build a query for answering "How many states in the US had more than 200 deaths on May 10, 2020 ?" The query needs to list the output in the field **count_of_states**.

Note: Don't include NULL values.

Columns to reference:

- `country_name`
- `subregion1_name` (for state information)
- `cumulative_deceased`

```

with deaths_by_states as (

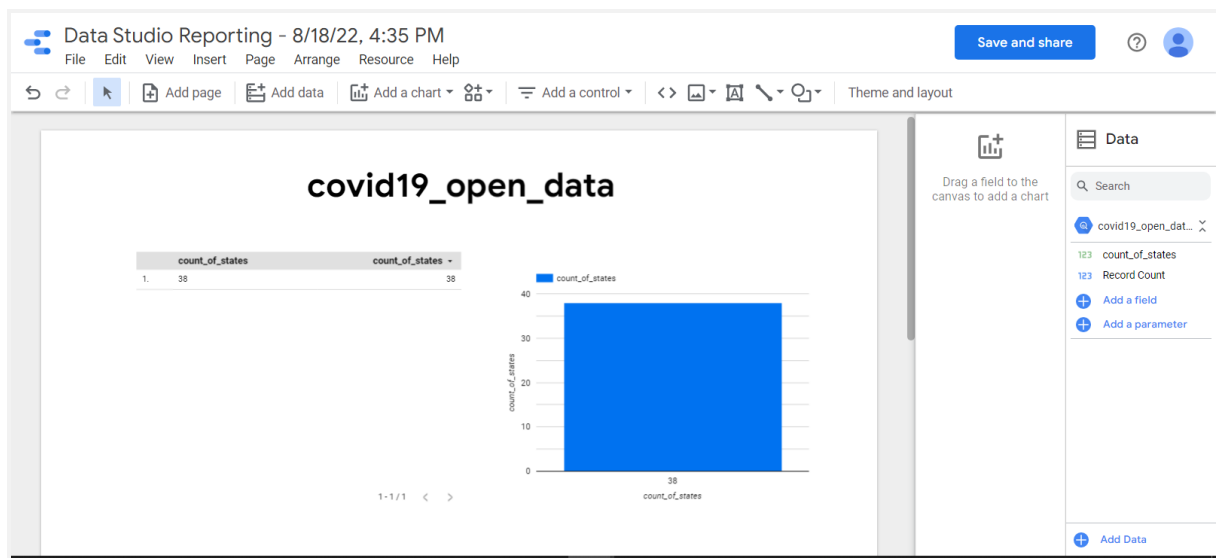
SELECT subregion1_name as state, sum(cumulative_deceased) as death_count

FROM `bigquery-public-data.covid19_open_data.covid19_open_data` where
country_name="United States of America" and date='2020-05-10' and subregion1_name is
NOT NULL

group by subregion1_name )

select count(*) as count_of_states from deaths_by_states where death_count > 200

```



Task 3. Identifying hotspots

- Build a query that will answer "List all the states in the United States of America that had more than 3000 confirmed cases on May 10, 2020 ?" The query needs to return the State Name and the corresponding confirmed cases arranged in descending order. Name of the fields to return **state** and **total_confirmed_cases**.

Columns to reference:

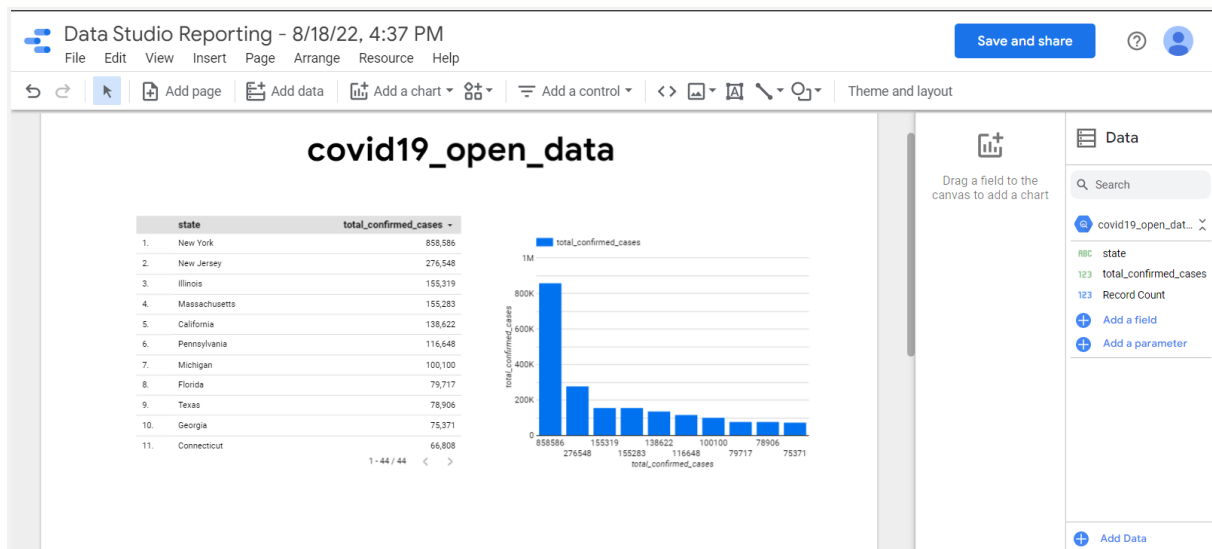
- `country_code`
- `subregion1_name` (for state information)
- `cumulative_confirmed`

```
SELECT * FROM (
```

```
SELECT subregion1_name as state, sum(cumulative_confirmed) as total_confirmed_cases
```

```
FROM `bigquery-public-data.covid19_open_data.covid19_open_data` WHERE  
country_code="US" AND date='2020-05-10' AND subregion1_name is NOT NULL
```

```
GROUP BY subregion1_name ORDER BY total_confirmed_cases DESC ) WHERE  
total_confirmed_cases > 3000
```



Task 4. Fatality ratio

1. Build a query that will answer "What was the case-fatality ratio in Italy for the month of June 2020?" Case-fatality ratio here is defined as (total deaths / total confirmed cases) * 100.
2. Write a query to return the ratio for the month of June 2020 and contain the following fields in the output: **total_confirmed_cases**, **total_deaths**, **case_fatality_ratio**.

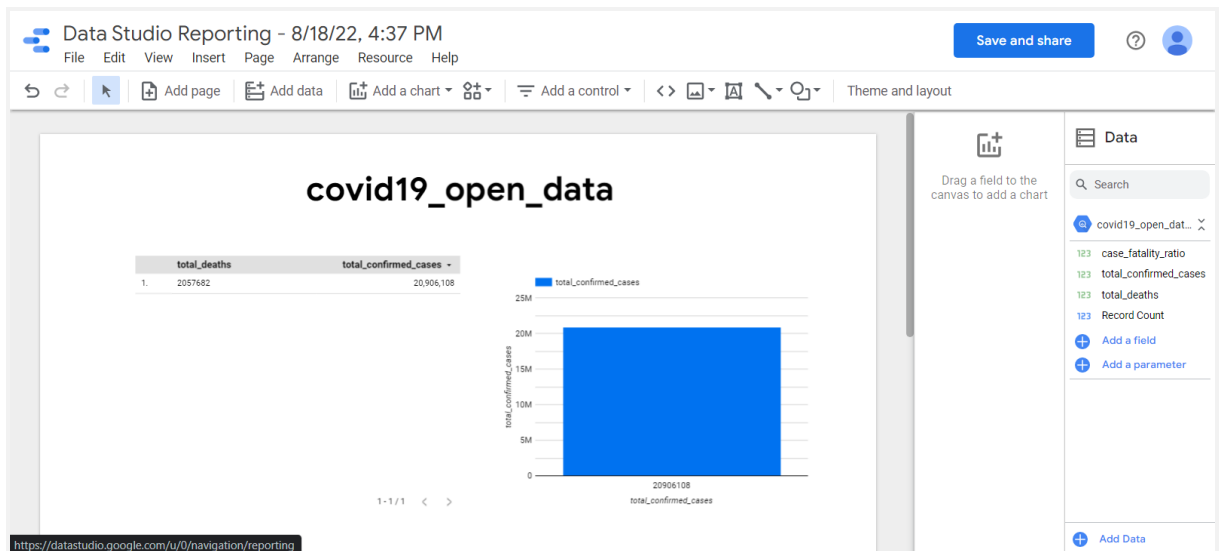
Columns to reference:

- `country_name`
- `cumulative_confirmed`
- `cumulative_deceased`

```
SELECT SUM(cumulative_confirmed) AS total_confirmed_cases,  
SUM(cumulative_deceased) AS total_deaths,  
(SUM(cumulative_deceased)/SUM(cumulative_confirmed))*100 AS  
case_fatality_ratio
```

```
FROM `bigquery-public-data.covid19_open_data.covid19_open_data`
```

```
WHERE country_name="Italy" AND date BETWEEN "2020-06-01" AND "2020-06-30"
```



Task 5. Identifying specific day

- Build a query that will answer: "On what day did the total number of deaths cross 14000 in Italy?" The query should return the date in the format **yyyy-mm-dd**.

Columns to reference:

- `country_name`
- `cumulative_deceased`

`SELECT`

`date`

`FROM`

```
`bigquery-public-data.covid19_open_data.covid19_open_data`
```

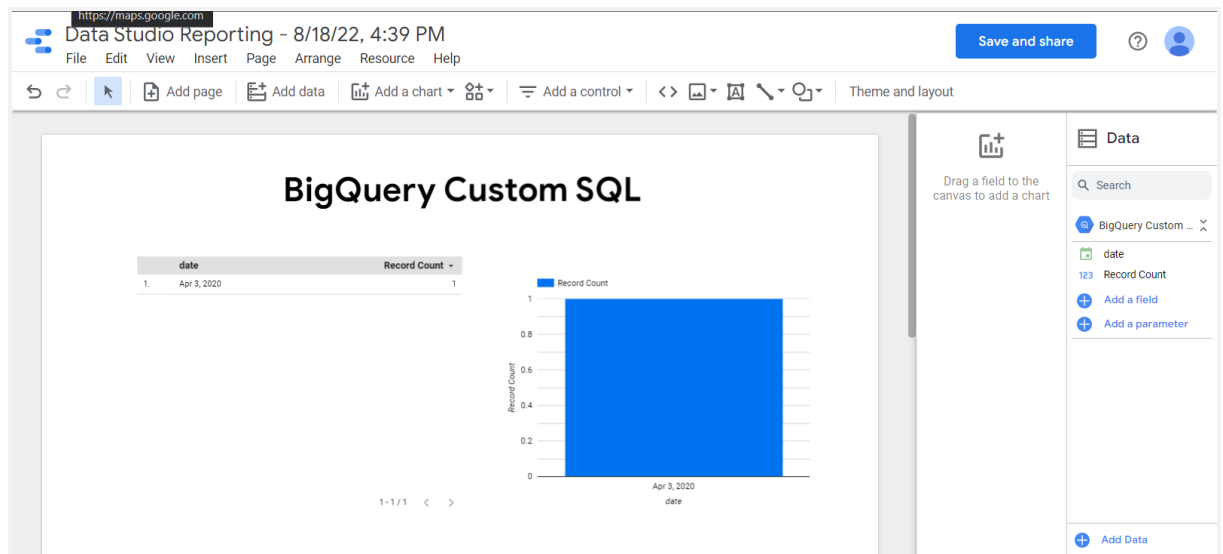
WHERE

```
country_name = 'Italy'
```

```
AND cumulative_deceased > 14000
```

ORDER BY date

LIMIT 1



Task 6. Finding days with zero net new cases

The following query is written to identify the number of days in India between 24, Feb 2020 and 14, March 2020 when there were zero increases in the number of confirmed cases. However it is not executing properly.

- You need to update the query to complete it and obtain the result:

```
WITH india_cases_by_date AS (
```

```
SELECT
```

```
date,
```

```
SUM(cumulative_confirmed) AS cases
```

```
FROM
```

```
`bigquery-public-data.covid19_open_data.covid19_open_data`
```

```
WHERE
```

```
country_name="India"
```

```
AND date between '2020-02-21' and '2020-03-15'
```

```
GROUP BY
```


date

ORDER BY

date ASC

)

, *india_previous_day_comparison AS*

(*SELECT*

date,

cases,

LAG(cases) OVER(ORDER BY date) AS previous_day,

cases - LAG(cases) OVER(ORDER BY date) AS net_new_cases

FROM india_cases_by_date

)

Solution:

```
WITH india_cases_by_date AS (
```

```
SELECT
```

```
date,
```

```
SUM(cumulative_confirmed) AS cases
```

```
FROM
```

```
`bigquery-public-data.covid19_open_data.covid19_open_data`
```

```
WHERE
```

```
country_name="India"
```

```
AND date between '2020-02-24' and '2020-03-14'
```

```
GROUP BY
```

```
date
```

```
ORDER BY
```

```
date ASC
```

```
)
```

```
, india_previous_day_comparison AS
```

```
(SELECT
```

```
date,
```

```
cases,
```

```
LAG(cases) OVER(ORDER BY date) AS previous_day,
```

```
cases - LAG(cases) OVER(ORDER BY date) AS net_new_cases
```

```
FROM india_cases_by_date
```

```
)
```

```
SELECT
```

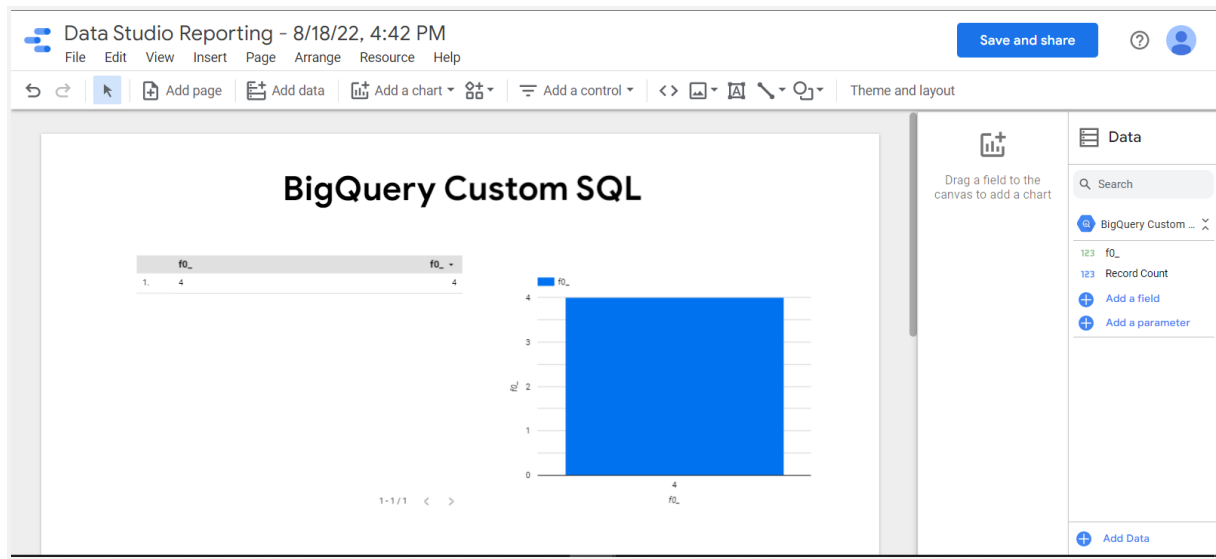
```
COUNT(date)
```

```
FROM
```

```
india_previous_day_comparison
```

```
WHERE
```

```
net_new_cases = 0
```



Task 7. Doubling rate

- Using the previous query as a template, write a query to find out the dates on which the confirmed cases increased by more than 15 % compared to the previous day (indicating doubling rate of ~ 7 days) in the US between the dates March 22, 2020 and April 20, 2020. The query needs to return the list of dates, the confirmed cases on that day, the confirmed cases the previous day, and the percentage increase in cases between the days.
 - Use the following names for the returned fields: **Date**, **Confirmed_Cases_On_Day**, **Confirmed_Cases_Previous_Day** and **Percentage_Increase_In_Cases**.

```
WITH us_cases_by_date AS (
```

```
SELECT
```

```
date,
```

```
SUM( cumulative_confirmed ) AS cases
```

```
FROM
```

```
`bigquery-public-data.covid19_open_data.covid19_open_data`
```

```
WHERE
```

```
country_name="United States of America"
```

```
AND date between '2020-03-22' and '2020-04-20'
```

```
GROUP BY
```

```
date
```

```
ORDER BY
```

```
date ASC
```

```
)
```

```
, us_previous_day_comparison AS
```

```
(SELECT
```

```
date,
```

```
cases,
```

```
LAG(cases) OVER(ORDER BY date) AS previous_day,
```

```
cases - LAG(cases) OVER(ORDER BY date) AS net_new_cases,
```

```
(cases - LAG(cases) OVER(ORDER BY date))*100/LAG(cases) OVER(ORDER  
BY date) AS percentage_increase
```

```
FROM us_cases_by_date
```

```
)
```

```
SELECT
```

```
Date,
```

```
cases AS Confirmed_Cases_On_Day,
```

```
previous_day AS Confirmed_Cases_Previous_Day,
```

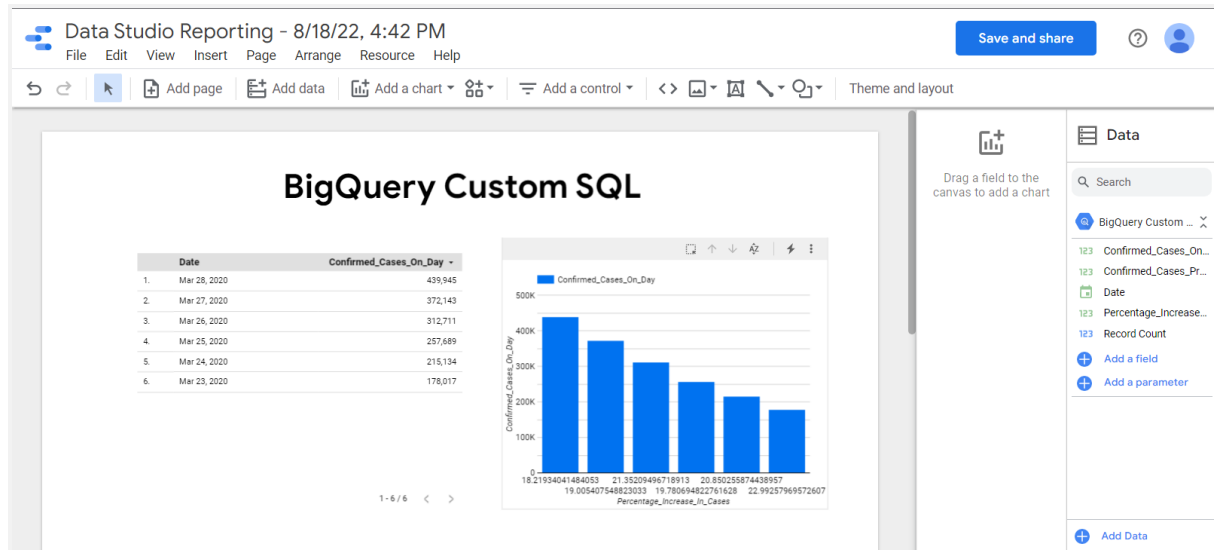
```
percentage_increase AS Percentage_Increase_In_Cases
```

```
FROM
```

```
us_previous_day_comparison
```

```
WHERE
```


percentage_increase > 15



Task 8. Recovery rate

1. Build a query to list the recovery rates of countries arranged in descending order (limit to 15) upto the date May 10, 2020.

2. Restrict the query to only those countries having more than 50K confirmed cases.

- The query needs to return the following fields: **country**, **recovered_cases**, **confirmed_cases**, **recovery_rate**.

Columns to reference:

* country_name

* cumulative_confirmed

* cumulative_recovered

WITH

```
france_cases AS (
```

```
SELECT
```

```
date,
```

```
SUM(cumulative_confirmed) AS total_cases
```

```
FROM
```

```
`bigquery-public-data.covid19_open_data.covid19_open_data`
```

```
WHERE
```

```
country_name="France"
```

```
AND date BETWEEN '2020-01-24' AND
```

```
'2020-05-10'
```

```
GROUP BY
```

```
date
```

```
ORDER BY
```

```
date)
```

```
, summary as (
```

```
SELECT
```

```
total_cases AS first_day_cases,
```

```
LEAD(total_cases) OVER(ORDER BY date) AS last_day_cases,
```

```
DATE_DIFF(LEAD(date) OVER(ORDER BY date),date, day) AS days_diff
```

```
FROM
```

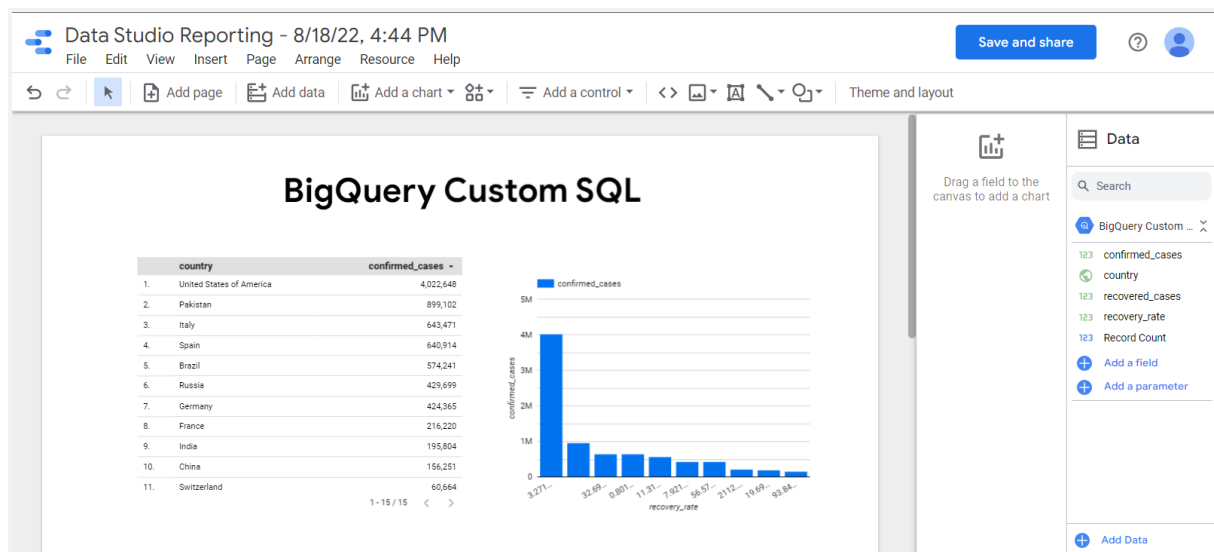
```
france_cases
```

```
LIMIT 1
```

```
)
```

```
select first_day_cases, last_day_cases, days_diff,  
POWER(last_day_cases/first_day_cases,1/days_diff)-1 as cdgr
```

from summary



Task 9. CDGR - Cumulative daily growth rate

- The following query is trying to calculate the CDGR on May 10, 2020 (Cumulative Daily Growth Rate) for France since the day the first case was reported. The first case was reported on Jan 24, 2020.
- The CDGR is calculated as:

$$((\text{last_day_cases} / \text{first_day_cases})^{1/\text{days_diff}}) - 1$$

Where :

- last_day_cases is the number of confirmed cases on May 10, 2020
- first_day_cases is the number of confirmed cases on Jan 24, 2020
- days_diff is the number of days between Jan 24 - May 10, 2020
- The query isn't executing properly. Can you fix the error to make the query execute successfully?

WITH

```
france_cases AS (  
  SELECT  
    date,  
    SUM(cumulative_confirmed) AS total_cases  
  FROM  
    `bigquery-public-data.covid19_open_data.covid19_open_data`  
  WHERE  
    country_name="France"  
    AND date IN ('2020-01-24',  
                 '2020-05-10')  
  GROUP BY  
    date  
  ORDER BY  
    date)  
, summary as (  
  SELECT  
    total_cases AS first_day_cases,  
    LEAD(total_cases) AS last_day_cases,  
    DATE_DIFF(LEAD(date) OVER(ORDER BY date),date, day) AS days_diff  
  FROM  
    france_cases  
  LIMIT 1  
)  
select first_day_cases, last_day_cases, days_diff,  
SQRT((last_day_cases/first_day_cases),(1/days_diff))-1 as cdgr  
from summary
```

SELECT

date, SUM(cumulative_confirmed) AS country_cases,

SUM(cumulative_deceased) AS country_deaths

FROM

`bigquery-public-data.covid19_open_data.covid19_open_data`

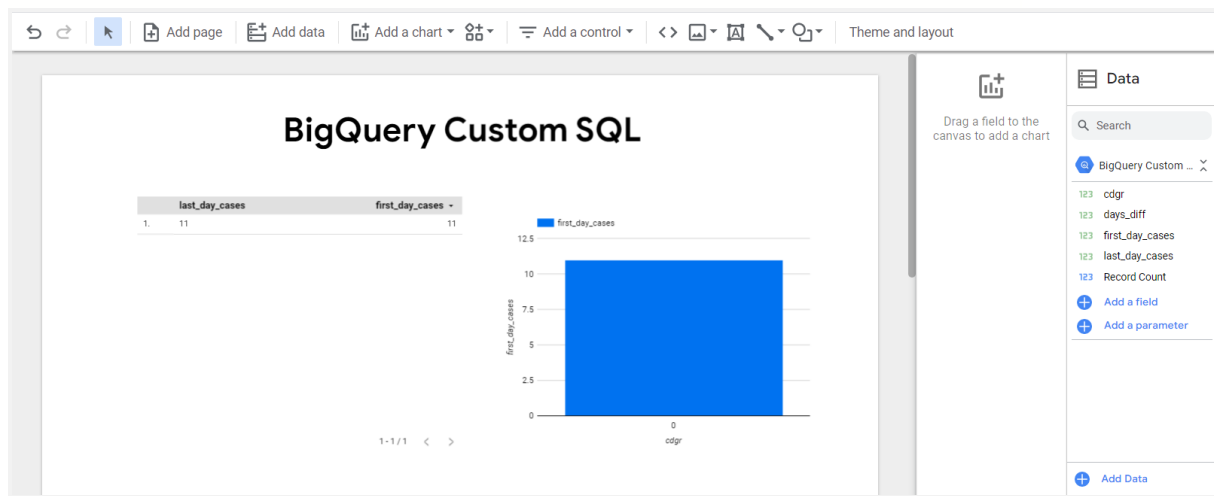
WHERE

date BETWEEN '2020-03-15'

AND '2020-04-25'


```
AND country_name='United States of America'
```

```
GROUP BY date
```



Task 10. Create a Data Studio report

- Create a [Google Data Studio](#) report that plots the following for the United States:
 - Number of Confirmed Cases
 - Number of Deaths
 - Date range : 2020-03-15 to 2020-04-25

```
SELECT
```

```
date, SUM(cumulative_confirmed) AS country_cases,
```

```
SUM(cumulative_deceased) AS country_deaths
```

```
FROM
```

```
`bigquery-public-data.covid19_open_data.covid19_open_data`
```

```
WHERE
```

```
date BETWEEN '2020-03-15'
```

AND '2020-04-25'

AND country_name='United States of America'

GROUP BY date

