COVID19 Data Analysis from Google Open Data using Hive and Tableau

Authors: Suleima Gonzalez, Bryan Chica, Mohammed Shodimu, Phillip Navarro
Instructor: Jongwook Woo

Objectives

- Upload data and analyze via wget
- Create a table and test data via Hive
- Clean and test data via Pig
- Store and Export data into Hadoop then Local Machine
- Visualizations and Analysis using PowerBI

Platform Specifications

- **Hostname:** bigdaiun0.sub03291929060.trainingvcn.oraclevcn.com
- **OS:** Oracle Linux Server 7.9
- **Storage:** 120 GB
- **Memory Size:** 32 GB
- **CPU:** AMC EPYX 7J13 64-Core Processor @ 2.45 Ghz
- **Cluster Version:** Hadoop 3.1.2
- **Cluster Number of Nodes:** 3

Prerequisites

- **Operating Systems**
  - Windows 10 or 11
  - MacOS
  - Linux
- **Shell terminal**
  - For windows, download and install Git Bash: *https://git-scm.com/downloads*
  - On Linux or Mac computer, use **terminal**
- **Microsoft Excel**
- **Microsoft PowerBI**
- **OpenRefine**

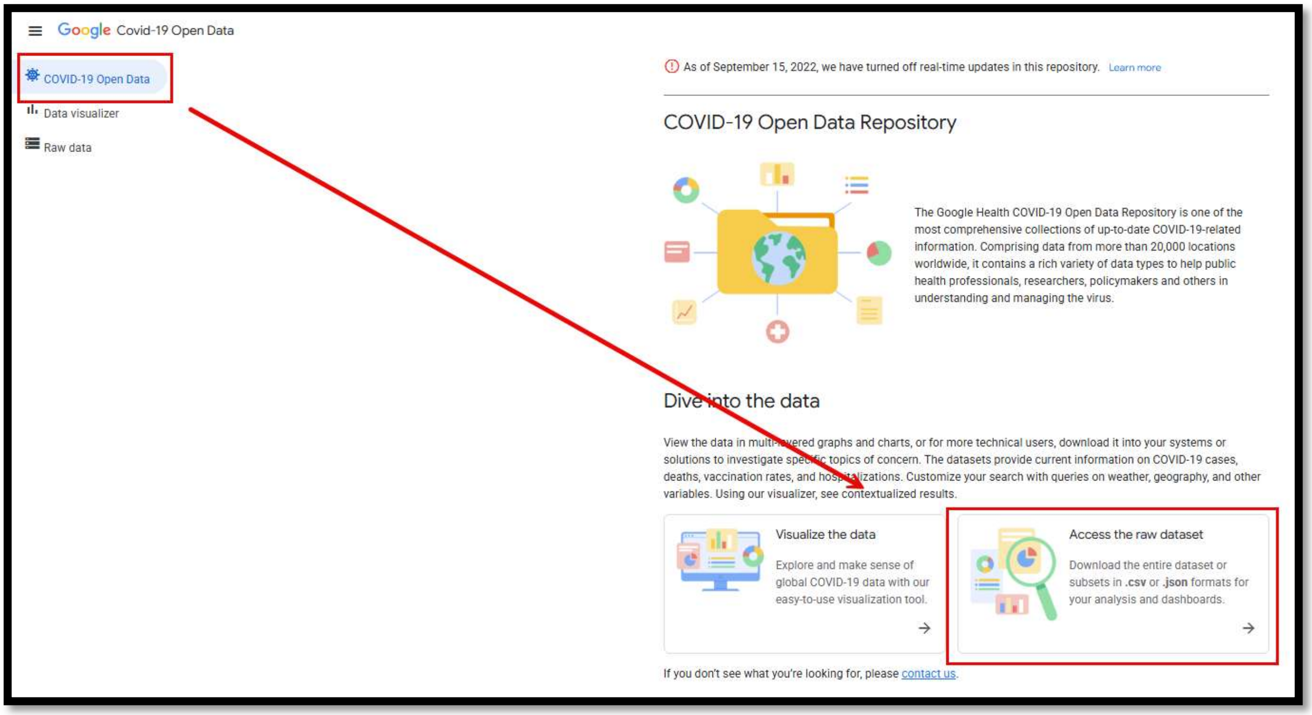
Introduction

This project demonstrates a complete big data analytic workflow using Google COVID-19 Open Data. The process includes data acquisition, data cleaning, distributed storage, table creation using Hive, aggregation of large datasets, and visualization of key results. By following this step-by-step tutorial, we apply core concepts from CIS 4560 related to big data processing, data warehousing, and analytical reporting.

Step 1: Download datasets from Google Covid-19 Open Data

The purpose of this step is to acquire raw COVID-19 data from a reliable and publicly available source. Google COVID-19 Open Data provides standardized datasets that include epidemiological, demographic, geographic, and vaccination information across multiple regions. Downloading these datasets serves as the foundation for all subsequent data processing, analysis, and visualization tasks in this project.

Go to https://health.google.com/covid-19/open-data/



As of September 15, 2022, we have turned off real-time updates in this repository. Learn more

COVID-19 Open Data Repository

The Google Health COVID-19 Open Data Repository is one of the most comprehensive collections of up-to-date COVID-19-related information. Comprising data from more than 20,000 locations worldwide, it contains a rich variety of data types to help public health professionals, researchers, policymakers and others in understanding and managing the virus.

Dive into the data

View the data in multi-layered graphs and charts, or for more technical users, download it into your systems or solutions to investigate specific topics of concern. The datasets provide current information on COVID-19 cases, deaths, vaccination rates, and hospitalizations. Customize your search with queries on weather, geography, and other variables. Using our visualizer, see contextualized results.

Visualize the data

Explore and make sense of global COVID-19 data with our easy-to-use visualization tool.

Access the raw dataset

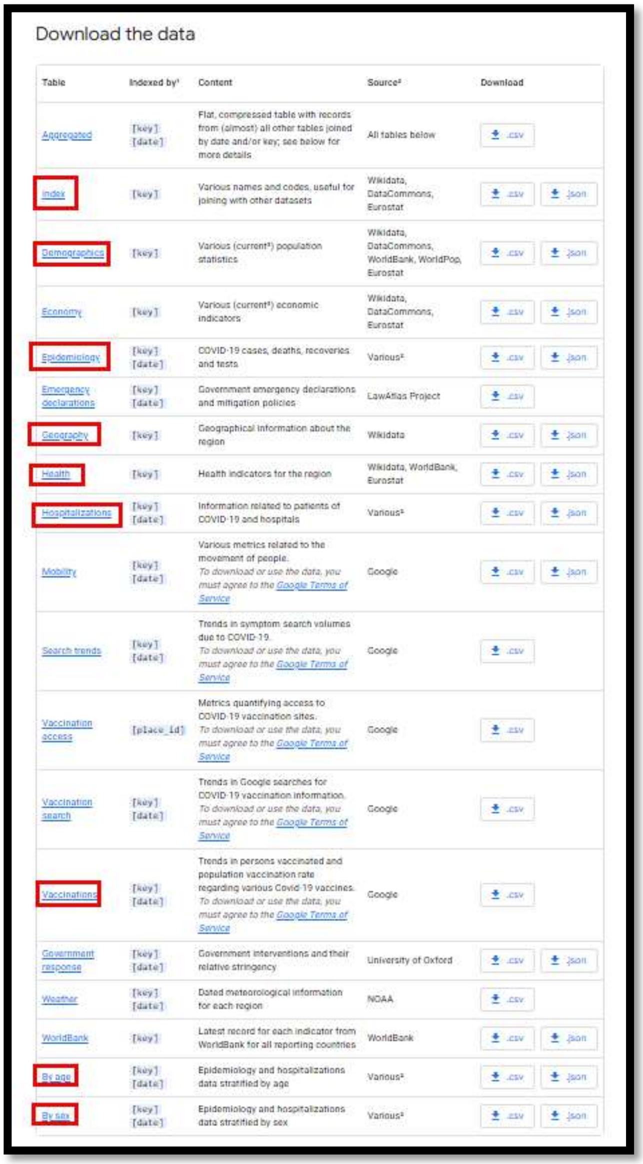
Download the entire dataset or subsets in **.csv** or **.json** formats for your analysis and dashboards.

If you don't see what you're looking for, please contact us.

Download the csv files of the following tables:

- Index
- Demographics
- Epidemiology
- Geography
- Health

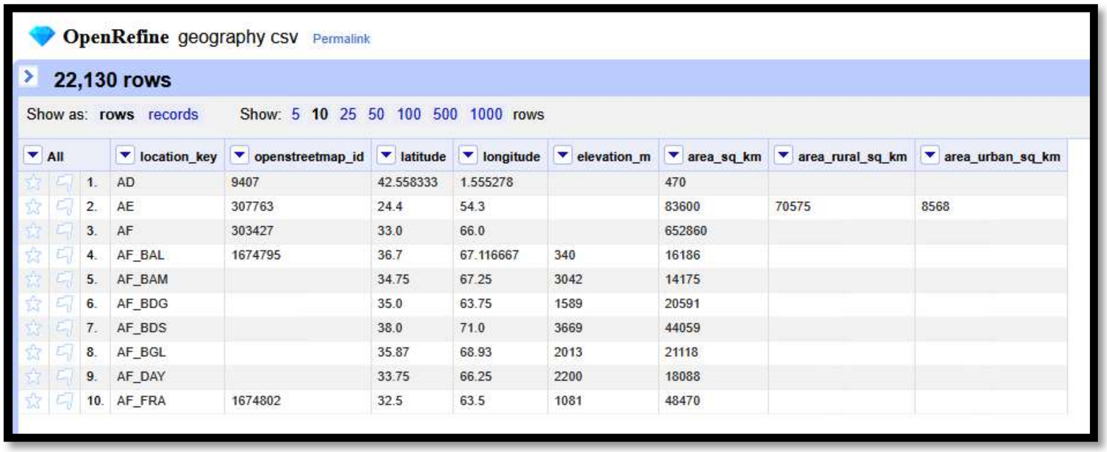
- Hospitalizations
- Vaccinations
- By age
- By sex



Download the data

Table	Indexed by[1]	Content	Source[2]	Download
Aggregated	[key] [date]	Flat, compressed table with records from (almost) all other tables joined by date and/or key; see below for more details	All tables below	.csv
Index	[key]	Various names and codes, useful for joining with other datasets	Wikidata, DataCommons, Eurostat	.csv .json
Demographics	[key]	Various (current[3]) population statistics	Wikidata, DataCommons, WorldBank, WorldPop, Eurostat	.csv .json
Economy	[key]	Various (current[3]) economic indicators	Wikidata, DataCommons, Eurostat	.csv .json
Epidemiology	[key] [date]	COVID-19 cases, deaths, recoveries and tests	Various[2]	.csv .json
Emergency declarations	[key] [date]	Government emergency declarations and mitigation policies	LawAtlas Project	.csv
Geography	[key]	Geographical information about the region	Wikidata	.csv .json
Health	[key]	Health indicators for the region	Wikidata, WorldBank, Eurostat	.csv .json
Hospitalizations	[key] [date]	Information related to patients of COVID-19 and hospitals	Various[2]	.csv .json
Mobility	[key] [date]	Various metrics related to the movement of people. *To download or use the data, you must agree to the Google Terms of Service*	Google	.csv .json
Search trends	[key] [date]	Trends in symptom search volumes due to COVID-19. *To download or use the data, you must agree to the Google Terms of Service*	Google	.csv
Vaccination access	[place_id]	Metrics quantifying access to COVID-19 vaccination sites. *To download or use the data, you must agree to the Google Terms of Service*	Google	.csv
Vaccination search	[key] [date]	Trends in Google searches for COVID-19 vaccination information. *To download or use the data, you must agree to the Google Terms of Service*	Google	.csv
Vaccinations	[key] [date]	Trends in persons vaccinated and population vaccination rate regarding various Covid-19 vaccines. *To download or use the data, you must agree to the Google Terms of Service*	Google	.csv
Government response	[key] [date]	Government interventions and their relative stringency	University of Oxford	.csv .json
Weather	[key] [date]	Dated meteorological information for each region	NOAA	.csv
WorldBank	[key]	Latest record for each indicator from WorldBank for all reporting countries	WorldBank	.csv .json
By age	[key] [date]	Epidemiology and hospitalizations data stratified by age	Various[2]	.csv .json
By sex	[key] [date]	Epidemiology and hospitalizations data stratified by sex	Various[2]	.csv .json

Clean tables using OpenRefine



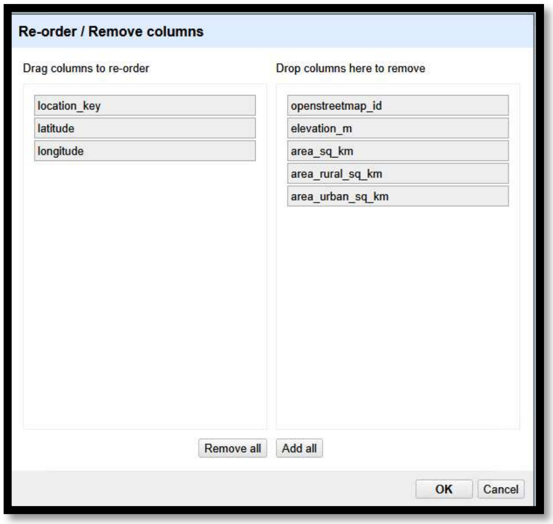
OpenRefine geography csv Permalink

22,130 rows

Show as: rows records Show: 5 10 25 50 100 500 1000 rows

All	location_key	openstreetmap_id	latitude	longitude	elevation_m	area_sq_km	area_rural_sq_km	area_urban_sq_km
1.	AD	9407	42.558333	1.555278		470		
2.	AE	307763	24.4	54.3		83600	70575	8568
3.	AF	303427	33.0	66.0		652860		
4.	AF_BAL	1674795	36.7	67.116667	340	16186		
5.	AF_BAM		34.75	67.25	3042	14175		
6.	AF_BDG		35.0	63.75	1589	20591		
7.	AF_BDS		38.0	71.0	3669	44059		
8.	AF_BGL		35.87	68.93	2013	21118		
9.	AF_DAY		33.75	66.25	2200	18088		
10.	AF_FRA	1674802	32.5	63.5	1081	48470		

Selected the fields I want to drop



Re-order / Remove columns

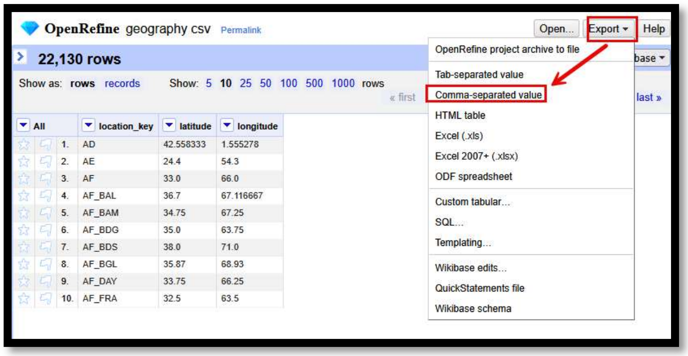
Drag columns to re-order

- location_key
- latitude
- longitude

Drop columns here to remove

- openstreetmap_id
- elevation_m
- area_sq_km
- area_rural_sq_km
- area_urban_sq_km

Export the file into a csv



OpenRefine geography csv Permalink

22,130 rows

Show as: rows records Show: 5 10 25 50 100 500 1000 rows

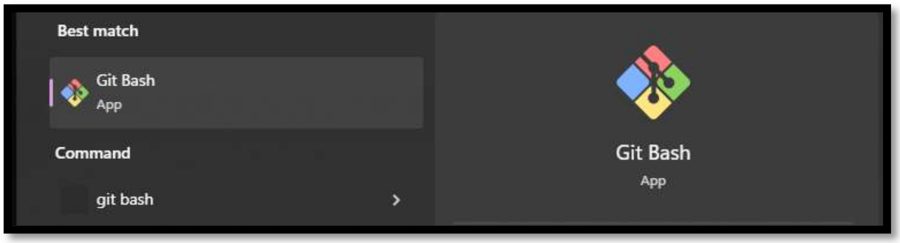
All	location_key	latitude	longitude
1.	AD	42.558333	1.555278
2.	AE	24.4	54.3
3.	AF	33.0	66.0
4.	AF_BAL	36.7	67.116667
5.	AF_BAM	34.75	67.25
6.	AF_BDG	35.0	63.75
7.	AF_BDS	38.0	71.0
8.	AF_BGL	35.87	68.93
9.	AF_DAY	33.75	66.25
10.	AF_FRA	32.5	63.5

Do this with all the csv files

Step 2: Load Data in Oracle Linux Server

This step transfers the cleaned datasets from the local machine to the Oracle Linux Server environment. Uploading the files to the cluster allows the data to be processed using distributed computing tools such as Hadoop and Hive. Secure Shell (SSH) is used to establish a remote connection to the server and manage files directly within the big data environment.

Open **Git Bash** terminal on your local computer

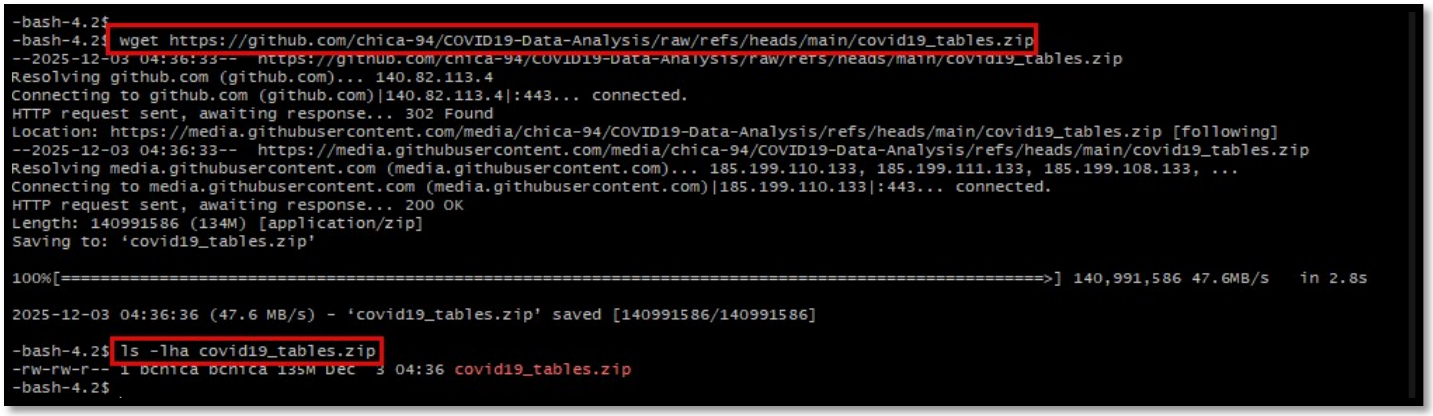


In the shell terminal, use ssh to connect to the Oracle cluster

```
ssh username@ip
```

Download the following file “covid19_tables.zip” into oracle big data server

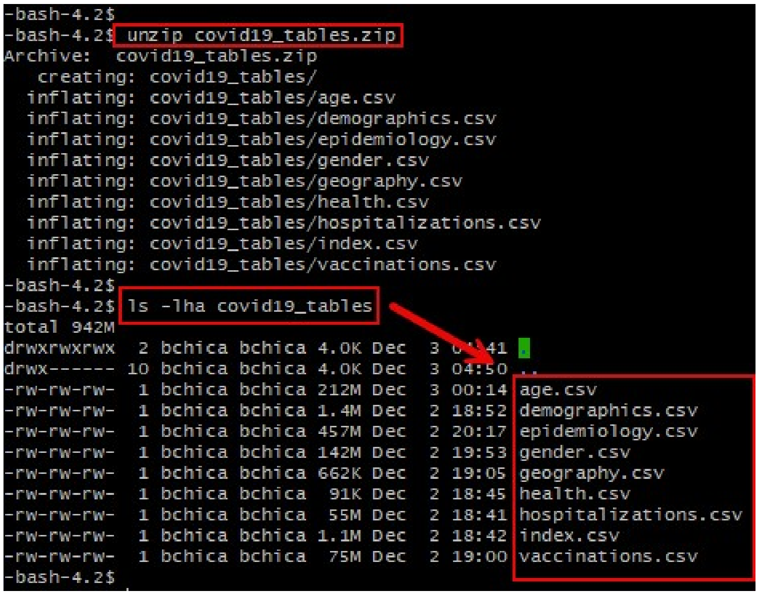
```
wget https://github.com/chica-94/COVID19-Data-Analysis/raw/refs/heads/main/covid19_tables.zip

ls -lha covid19_tables.zip
```



```
-bash-4.2$
-bash-4.2$ wget https://github.com/chica-94/COVID19-Data-Analysis/raw/refs/heads/main/covid19_tables.zip
--2025-12-03 04:36:33--  https://github.com/chica-94/COVID19-Data-Analysis/raw/refs/heads/main/covid19_tables.zip
Resolving github.com (github.com)... 140.82.113.4
Connecting to github.com (github.com)|140.82.113.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://media.githubusercontent.com/media/chica-94/COVID19-Data-Analysis/refs/heads/main/covid19_tables.zip [following]
--2025-12-03 04:36:33--  https://media.githubusercontent.com/media/chica-94/COVID19-Data-Analysis/refs/heads/main/covid19_tables.zip
Resolving media.githubusercontent.com (media.githubusercontent.com)... 185.199.110.133, 185.199.111.133, 185.199.108.133, ...
Connecting to media.githubusercontent.com (media.githubusercontent.com)|185.199.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 140991586 (134M) [application/zip]
Saving to: ‘covid19_tables.zip’

100%[===================================================================================================>] 140,991,586 47.6MB/s   in 2.8s

2025-12-03 04:36:36 (47.6 MB/s) - ‘covid19_tables.zip’ saved [140991586/140991586]

-bash-4.2$ ls -lha covid19_tables.zip
-rw-rw-r-- 1 bchica bchica 135M Dec  3 04:36 covid19_tables.zip
-bash-4.2$
```

Unzip tables folder

```
unzip covid19_tables.zip

ls -lha covid19_tables
```



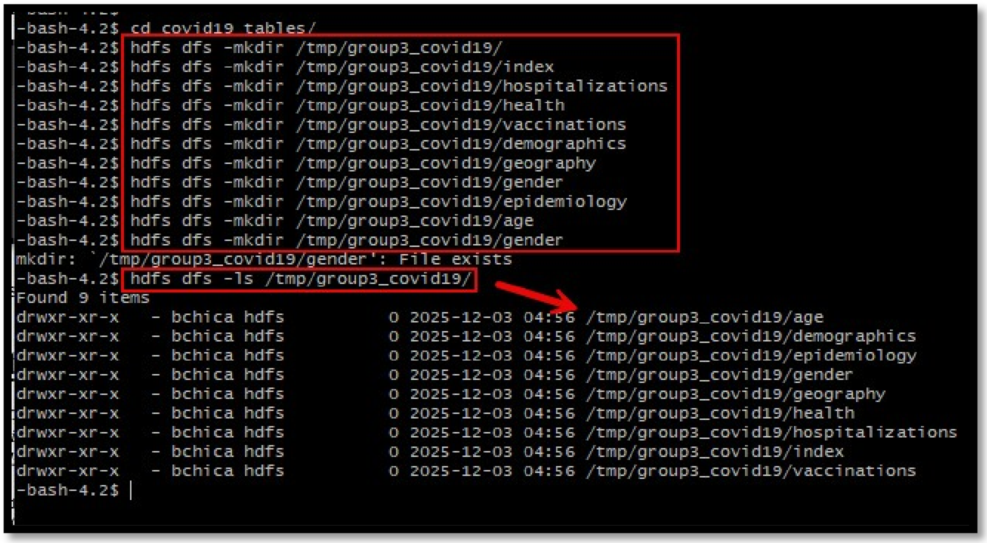
```
-bash-4.2$
-bash-4.2$ unzip covid19_tables.zip
Archive:  covid19_tables.zip
   creating: covid19_tables/
  inflating: covid19_tables/age.csv
  inflating: covid19_tables/demographics.csv
  inflating: covid19_tables/epidemiology.csv
  inflating: covid19_tables/gender.csv
  inflating: covid19_tables/geography.csv
  inflating: covid19_tables/health.csv
  inflating: covid19_tables/hospitalizations.csv
  inflating: covid19_tables/index.csv
  inflating: covid19_tables/vaccinations.csv
-bash-4.2$
-bash-4.2$ ls -lha covid19_tables
total 942M
drwxrwxrwx  2 bchica bchica 4.0K Dec  3 0[illegible]:41 .
drwx------ 10 bchica bchica 4.0K Dec  3 04:50 ..
-rw-rw-rw-  1 bchica bchica 212M Dec  3 00:14 age.csv
-rw-rw-rw-  1 bchica bchica 1.4M Dec  2 18:52 demographics.csv
-rw-rw-rw-  1 bchica bchica 457M Dec  2 20:17 epidemiology.csv
-rw-rw-rw-  1 bchica bchica 142M Dec  2 19:53 gender.csv
-rw-rw-rw-  1 bchica bchica 662K Dec  2 19:05 geography.csv
-rw-rw-rw-  1 bchica bchica  91K Dec  2 18:45 health.csv
-rw-rw-rw-  1 bchica bchica  55M Dec  2 18:41 hospitalizations.csv
-rw-rw-rw-  1 bchica bchica 1.1M Dec  2 18:42 index.csv
-rw-rw-rw-  1 bchica bchica  75M Dec  2 19:00 vaccinations.csv
-bash-4.2$
```

Step 3: Upload datasets to HDFS

Hadoop Distributed File System (HDFS) is used to store large datasets across multiple nodes for efficient access and fault tolerance. In this step, directories are created for each dataset, and the CSV files are uploaded into HDFS. Organizing the data into separate directories ensures structured storage and prepares the datasets for table creation in Hive.

Create a directory rating to put the files to HDFS. Then upload the covid data to the new directory in HDFS. Once the file is added, verify *all* files are uploaded to **covid19** directory (table):

```
cd covid19_tables

hdfs dfs -mkdir tmp/group3_covid19/index
hdfs dfs -mkdir tmp/group3_covid19/hospitalizations
hdfs dfs -mkdir tmp/group3_covid19/health
hdfs dfs -mkdir tmp/group3_covid19/vaccinations
hdfs dfs -mkdir tmp/group3_covid19/demographics
hdfs dfs -mkdir tmp/group3_covid19/geography
hdfs dfs -mkdir tmp/group3_covid19/gender
hdfs dfs -mkdir tmp/group3_covid19/epidemiology
hdfs dfs -mkdir tmp/group3_covid19/age
hdfs dfs -mkdir tmp/group3_covid19/gender
hdfs dfs -mkdir /tmp/group3_covid19/agggregated
hdfs dfs -mkdir /tmp/group3_covid19/top10_countries/

hdfs dfs -ls tmp/group3_covid19/
```



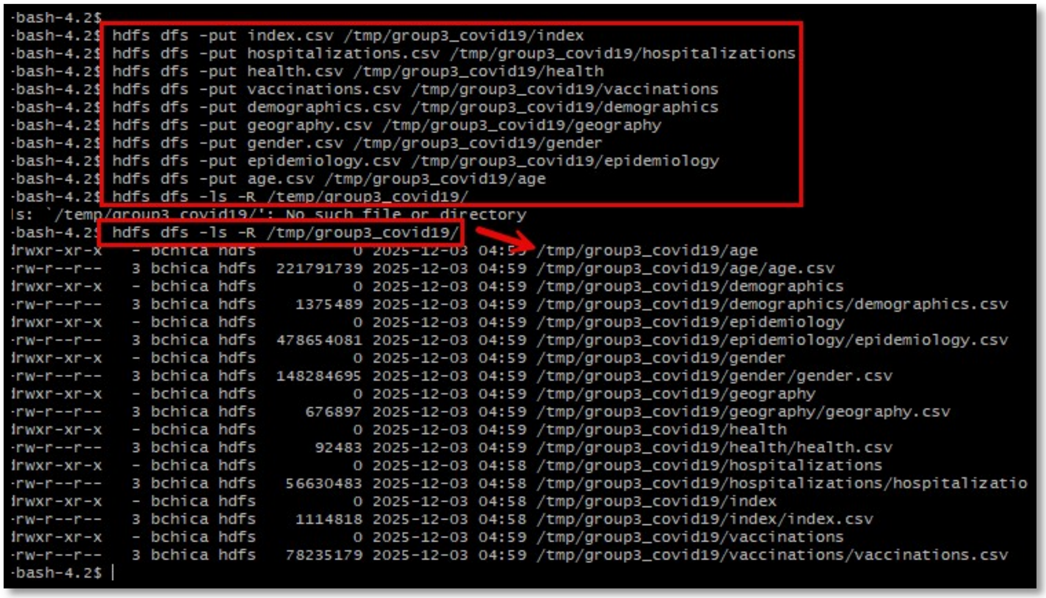
```
-bash-4.2$ cd covid19_tables/
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/index
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/hospitalizations
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/health
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/vaccinations
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/demographics
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/geography
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/gender
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/epidemiology
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/age
-bash-4.2$ hdfs dfs -mkdir /tmp/group3_covid19/gender
mkdir: `/tmp/group3_covid19/gender': File exists
-bash-4.2$ hdfs dfs -ls /tmp/group3_covid19/
Found 9 items
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/age
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/demographics
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/epidemiology
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/gender
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/geography
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/health
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/hospitalizations
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/index
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:56 /tmp/group3_covid19/vaccinations
-bash-4.2$ |
```

Upload the covid19 datasets into their respective directories in HDFS.

```
hdfs dfs -put index.csv tmp/group3_covid19/index
hdfs dfs -put hospitalizations.csv tmp/group3_covid19/hospitalizations
hdfs dfs -put health.csv tmp/group3_covid19/health
hdfs dfs -put vaccinations.csv tmp/group3_covid19/vaccinations
hdfs dfs -put demographics.csv tmp/group3_covid19/demographics
hdfs dfs -put geography.csv tmp/group3_covid19/geography
hdfs dfs -put gender.csv tmp/group3_covid19/gender
hdfs dfs -put epidemiology.csv tmp/group3_covid19/epidemiology
hdfs dfs -put age.csv tmp/group3_covid19/age
```

Ensure that all files are loaded to hdfs and are in their designated directories

```
hdfs dfs -ls -R tmp/group3_covid19/
```



```
-bash-4.2$
-bash-4.2$ hdfs dfs -put index.csv /tmp/group3_covid19/index
-bash-4.2$ hdfs dfs -put hospitalizations.csv /tmp/group3_covid19/hospitalizations
-bash-4.2$ hdfs dfs -put health.csv /tmp/group3_covid19/health
-bash-4.2$ hdfs dfs -put vaccinations.csv /tmp/group3_covid19/vaccinations
-bash-4.2$ hdfs dfs -put demographics.csv /tmp/group3_covid19/demographics
-bash-4.2$ hdfs dfs -put geography.csv /tmp/group3_covid19/geography
-bash-4.2$ hdfs dfs -put gender.csv /tmp/group3_covid19/gender
-bash-4.2$ hdfs dfs -put epidemiology.csv /tmp/group3_covid19/epidemiology
-bash-4.2$ hdfs dfs -put age.csv /tmp/group3_covid19/age
-bash-4.2$ hdfs dfs -ls -R /temp/group3_covid19/
ls: `/temp/group3_covid19/': No such file or directory
-bash-4.2$ hdfs dfs -ls -R /tmp/group3_covid19/
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/age
-rw-r--r--   3 bchica hdfs  221791739 2025-12-03 04:59 /tmp/group3_covid19/age/age.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/demographics
-rw-r--r--   3 bchica hdfs    1375489 2025-12-03 04:59 /tmp/group3_covid19/demographics/demographics.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/epidemiology
-rw-r--r--   3 bchica hdfs  478654081 2025-12-03 04:59 /tmp/group3_covid19/epidemiology/epidemiology.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/gender
-rw-r--r--   3 bchica hdfs  148284695 2025-12-03 04:59 /tmp/group3_covid19/gender/gender.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/geography
-rw-r--r--   3 bchica hdfs     676897 2025-12-03 04:59 /tmp/group3_covid19/geography/geography.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/health
-rw-r--r--   3 bchica hdfs      92483 2025-12-03 04:59 /tmp/group3_covid19/health/health.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:58 /tmp/group3_covid19/hospitalizations
-rw-r--r--   3 bchica hdfs   56630483 2025-12-03 04:58 /tmp/group3_covid19/hospitalizations/hospitalizatio
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:58 /tmp/group3_covid19/index
-rw-r--r--   3 bchica hdfs    1114818 2025-12-03 04:58 /tmp/group3_covid19/index/index.csv
drwxr-xr-x   - bchica hdfs          0 2025-12-03 04:59 /tmp/group3_covid19/vaccinations
-rw-r--r--   3 bchica hdfs   78235179 2025-12-03 04:59 /tmp/group3_covid19/vaccinations/vaccinations.csv
-bash-4.2$ |
```

Step 4: Create Tables Using Beeline

The purpose of this step is to create external Hive tables that map directly to the datasets stored in HDFS. Using Beeline and HiveQL commands, each dataset is structured into a table with defined schemas. External tables are used so that the underlying data remains in HDFS while still allowing SQL-based querying and analysis.

Now that the data is loaded in HDFS, we can now use Hive to create a table and query the data using hiveql commands

Open the Hive shell environment using **beeline**.

```
beeline
```

Use your database to create tables

```
use username;
```

Paste the following HiveQL code to create an external table called which will contain all the data from each dataset uploaded to HDFS. We start by applying a "DROP" command to remove any existing table that may come into conflict

Create **covid19_index** table

```
DROP TABLE IF EXISTS covid19_index;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_index(
        location_key STRING,
        country_code STRING,
        country_name STRING,
        subregion1_name STRING,
        subregion2_name STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/index/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_hospitalizations** table

```
DROP TABLE IF EXISTS covid19_hospitalizations;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_hospitalizations(
        event_date DATE,
        location_key STRING,
        new_hospitalized_patients INT,
        cumulative_hospitalized_patients INT,
        new_intensive_care_patients INT,
        cumulative_intensive_care_patients INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/hospitalizations/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_health** table

```
DROP TABLE IF EXISTS covid19_health;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_health(
       location_key STRING,
       adult_male_mortality_rate DOUBLE,
       adult_female_mortality_rate DOUBLE,
       life_expectancy DOUBLE,
       diabetes_prevalence DOUBLE,
       health_expenditure_usd DOUBLE,
       out_of_pocket_health_expenditure_usd DOUBLE)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/health/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_vaccinations** table

```
DROP TABLE IF EXISTS covid19_vaccinations;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_vaccinations(
       event_date DATE,
       location_key STRING,
       new_persons_fully_vaccinated INT,
cumulative_persons_fully_vaccinated INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/vaccinations/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_geography** table

```
DROP TABLE IF EXISTS covid19_geography;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_geography(
       location_key STRING,
       latitude DOUBLE,
       longitude DOUBLE)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/geography/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_demographics** table

```
DROP TABLE IF EXISTS covid19_demographics;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_demographics(
       location_key STRING,
       population INT,
       population_male INT,
       population_female INT,
       population_age_00_09 INT,
       population_age_10_19 INT,
       population_age_20_29 INT,
       population_age_30_39 INT,
       population_age_40_49 INT,
       population_age_50_59 INT,
       population_age_60_69 INT,
       population_age_70_79 INT,
       population_age_80_and_older INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/demographics/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_gender** table

```
DROP TABLE IF EXISTS covid19_gender;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_gender(
       event_date DATE,
       location_key STRING,
       new_deceased_male INT,
       cumulative_deceased_male INT,
       new_deceased_female INT,
       cumulative_deceased_female INT,
       new_recovered_male INT,
       cumulative_recovered_male INT,
       new_recovered_female INT,
       cumulative_recovered_female INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/gender/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

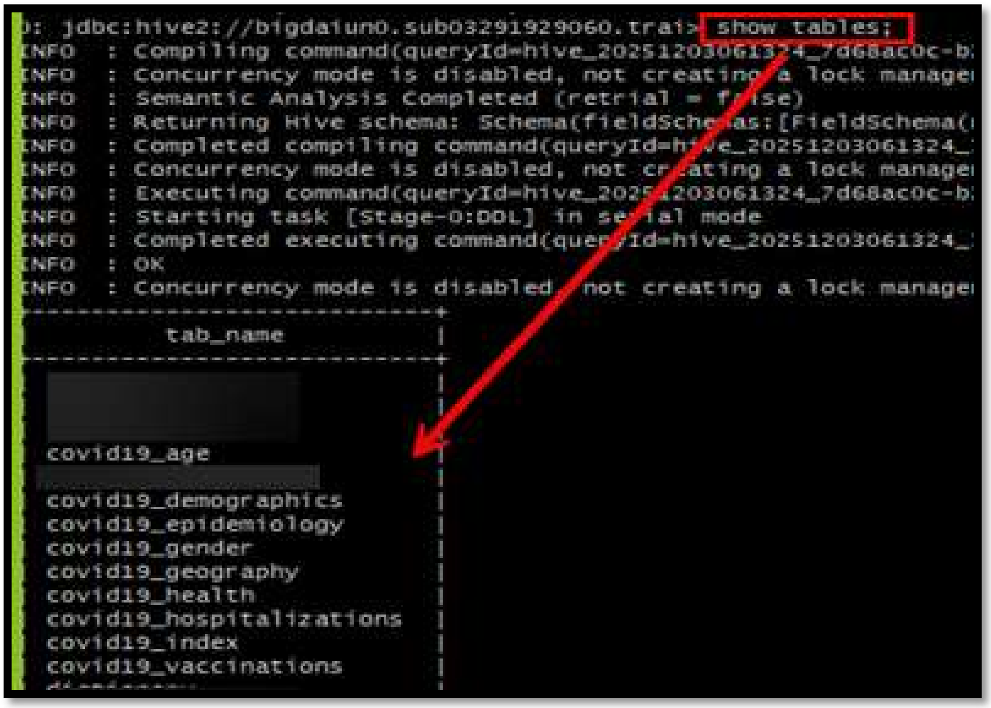
Create **covid19_epidemiology** table

```
DROP TABLE IF EXISTS covid19_epidemiology;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_epidemiology(
       event_date DATE,
       location_key STRING,
       new_confirmed INT,
       cumulative_confirmed INT,
       new_deceased INT,
       cumulative_deceased INT,
       new_recovered INT,
       cumulative_recovered INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/epidemiology/'
TBLPROPERTIES ("skip.header.line.count"="1");
```

Create **covid19_age** table

```
DROP TABLE IF EXISTS covid19_age;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_age(
        event_date DATE,
        location_key STRING,
        new_recovered_age_0-9 INT,
        new_recovered_age_10-19 INT,
        new_recovered_age_20-29 INT,
        new_recovered_age_30-39 INT,
        new_recovered_age_40-49 INT,
        new_recovered_age_50-59 INT,
        new_recovered_age_60-69 INT,
        new_recovered_age_70-79 INT,
        new_recovered_age_80-89 INT,
        new_recovered_age_90-99 INT,
        cumulative_recovered_age_0-9 INT,
        cumulative_recovered_age_10-19 INT,
        cumulative_recovered_age_20-29 INT,
        cumulative_recovered_age_30-39 INT,
        cumulative_recovered_age_40-49 INT,
        cumulative_recovered_age_50-59 INT,
        cumulative_recovered_age_60-69 INT,
        cumulative_recovered_age_70-79 INT,
        cumulative_recovered_age_80-89 INT,
        cumulative_recovered_age_90-99 INT )
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3 covid19/age/'
```

Then, in the Hive shell, make sure that the tables **covid19** are created and is shown:



```
0: jdbc:hive2://bigdaiun0.sub03291929060.trai> show tables;
INFO  : Compiling command(queryId=hive_20251203061324_7d68ac0c-b
INFO  : Concurrency mode is disabled, not creating a lock manage
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(
INFO  : Completed compiling command(queryId=hive_20251203061324_
INFO  : Concurrency mode is disabled, not creating a lock manage
INFO  : Executing command(queryId=hive_20251203061324_7d68ac0c-b
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20251203061324_
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manage
+----------------------------+
|          tab_name          |
+----------------------------+
| [redacted]                 |
| covid19_age                |
| [redacted]                 |
| covid19_demographics       |
| covid19_epidemiology       |
| covid19_gender             |
| covid19_geography          |
| covid19_health             |
| covid19_hospitalizations   |
| covid19_index              |
| covid19_vaccinations       |
```

Step 4: Create external aggregate table

An aggregated table is created to combine multiple COVID-19 datasets into a single unified structure. This table enables comprehensive analysis by integrating epidemiology, hospitalizations, vaccinations, health indicators, demographics, and geographic data. Aggregating the data simplifies querying and improves analytical efficiency.

Paste the following HiveQL code to create a table called “**covid19_aggregated**,” to store data from the **covid19** tables.

```
DROP TABLE IF EXISTS covid19_aggregated;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_aggregated(
       event_date DATE, location_key STRING,country_code STRING,country_name
       STRING, subregion1_name STRING,subregion2_name STRING, new_confirmed INT,
       cumulative_confirmed INT,new_deceased INT,cumulative_deceased
       INT,new_recovered INT,cumulative_recovered INT,new_hospitalized_patients
       INT,cumulative_hospitalized_patients INT,new_intensive_care_patients
       INT,cumulative_intensive_care_patients INT,new_persons_fully_vaccinated
       INT,cumulative_persons_fully_vaccinated INT,adult_male_mortality_rate
       DOUBLE,adult_female_mortality_rate DOUBLE,life_expectancy
       DOUBLE,diabetes_prevalence DOUBLE,health_expenditure_usd
       DOUBLE,out_of_pocket_health_expenditure_usd DOUBLE,population
       INT,population_male INT,population_female INT,population_age_00_09
       INT,population_age_10_19 INT,population_age_20_29 INT,population_age_30_39
       INT,population_age_40_49 INT,population_age_50_59 INT,population_age_60_69
       INT,population_age_70_79 INT,population_age_80_and_older INTlatitude
       DOUBLE,longitude DOUBLE,new_deceased_male INT,cumulative_deceased_male
       INT,new_deceased_female INT,cumulative_deceased_female
       INT,new_recovered_male INT,cumulative_recovered_male
       INT,new_recovered_female INT,cumulative_recovered_female
       INTnew_recovered_age_0-9 INT,new_recovered_age_10-19
       INT,new_recovered_age_20-29 INT,new_recovered_age_30-39
       INT,new_recovered_age_40-49 INT,new_recovered_age_50-59
       INT,new_recovered_age_60-69 INT,new_recovered_age_70-79
       INT,new_recovered_age_80-89 INT,new_recovered_age_90-99
       INT,cumulative_recovered_age_0-9 INT,cumulative_recovered_age_10-19
       INT,cumulative_recovered_age_20-29 INT,cumulative_recovered_age_30-39
       INT,cumulative_recovered_age_40-49 INT,cumulative_recovered_age_50-59
       INT,cumulative_recovered_age_60-69 INT,cumulative_recovered_age_70-79
       INT,cumulative_recovered_age_80-89 INT,cumulative_recovered_age_90-99 INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/tmp/group3_covid19/aggregated';
```

Now you need to populate data from the **covid19 tables** to **covid19_aggregated** using SELECT statement as follows:

```
INSERT OVERWRITE TABLE covid19_aggregated
SELECT
       epi.event_date,epi.location_key,idx.country_code,idx.country_name,idx.subregion1_nam
       e,idx.subregion2_name,epi.new_confirmed,epi.cumulative_confirmed,epi.new_deceased,
       epi.cumulative_deceased,epi.new_recovered,epi.cumulative_recovered,hosp.new_hospital
       ized_patients,hosp.cumulative_hospitalized_patients,hosp.new_intensive_care_patients
       ,hosp.cumulative_intensive_care_patients,vac.new_persons_fully_vaccinated,vac.cumula
       tive_persons_fully_vaccinated,health.adult_male_mortality_rate,health.adult_female_m
       ortality_rate,health.life_expectancy,health.diabetes_prevalence,health.health_expend
       iture_usd,health.out_of_pocket_health_expenditure_usd,demo.population,demo.populatio
       n_male,demo.population_female,demo.population_age_00_09,demo.population_age_10_19,de
       mo.population_age_20_29,demo.population_age_30_39,demo.population_age_40_49,demo.pop
       ulation_age_50_59,demo.population_age_60_69,demo.population_age_70_79,demo.populatio
       n_age_80_and_older,geo.latitude,geo.longitude,gender.new_deceased_male,gender.cumula
       tive_deceased_male,gender.new_deceased_female,gender.cumulative_deceased_female,gend
       er.new_recovered_male,gender.cumulative_recovered_male,gender.new_recovered_female,g
       ender.cumulative_recovered_female,age.new_recovered_age_0_9,age.new_recovered_age_10
       _19,age.new_recovered_age_20_29,age.new_recovered_age_30_39,age.new_recovered_age_40
       _49,age.new_recovered_age_50_59,age.new_recovered_age_60_69,age.new_recovered_age_70
       _79,age.new_recovered_age_80_89,age.new_recovered_age_90_99,age.cumulative_recovered
       _age_0_9,age.cumulative_recovered_age_10_19,age.cumulative_recovered_age_20_29,age.c
       umulative_recovered_age_30_39,age.cumulative_recovered_age_40_49,age.cumulative_reco
       vered_age_50_59,age.cumulative_recovered_age_60_69,age.cumulative_recovered_age_70_7
       9,age.cumulative_recovered_age_80_89,age.cumulative_recovered_age_90_99
       FROM covid19_epidemiology epi
LEFT JOIN covid19_index idx ON epi.location_key = idx.location_key
LEFT JOIN covid19_hospitalizations hosp ON epi.location_key = hosp.location_key AND
epi.event_date = hosp.event_date
LEFT JOIN covid19_vaccinations vac ON epi.location_key = vac.location_key AND
epi.event_date = vac.event_date
LEFT JOIN covid19_health health ON epi.location_key = health.location_key
LEFT JOIN covid19_demographics demo ON epi.location_key = demo.location_key
LEFT JOIN covid19_geography geo ON epi.location_key = geo.location_key
LEFT JOIN covid19_gender gender ON epi.location_key = gender.location_key AND
epi.event_date = gender.event_date
LEFT JOIN covid19_age age ON epi.location_key = age.location_key AND epi.event_date =
age.event_date;
```

Check if the table **covid19_aggregated** tables are shown

```
show tables;
```

Query the content of the **covid19_aggregated** table to ensure that the table has been created successfully.

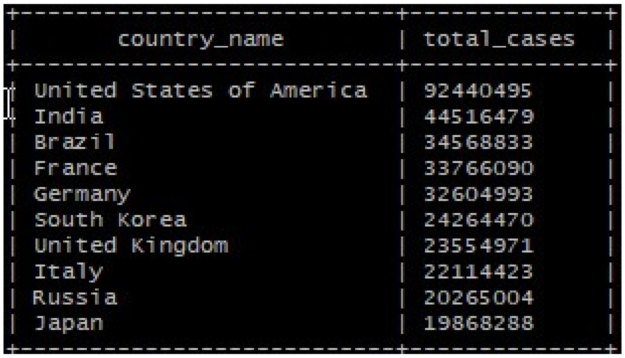
```
SELECT * FROM covid19_aggregated LIMIT 3;
```

Step 5: Data Cleaning and Preparation

This step focuses on refining the dataset to support meaningful analysis. By querying the aggregated table, the top ten countries with the highest cumulative confirmed COVID-19 cases are identified. Reducing the dataset to the most relevant records improves performance and allows the analysis to focus on the most impacted regions.

Query the aggregated tables to find the top 10 countries with the most confirmed COVID-19 cases

```
SELECT country_name, MAX(cumulative_confirmed) AS total_cases
FROM covid19_aggregated
WHERE cumulative_confirmed IS NOT NULL
GROUP BY country_name
ORDER BY total_cases DESC
LIMIT 10;
```



```
+---------------------------+---------------+
|       country_name        | total_cases   |
+---------------------------+---------------+
| United States of America  | 92440495      |
| India                     | 44516479      |
| Brazil                    | 34568833      |
| France                    | 33766090      |
| Germany                   | 32604993      |
| South Korea               | 24264470      |
| United Kingdom            | 23554971      |
| Italy                     | 22114423      |
| Russia                    | 20265004      |
| Japan                     | 19868288      |
+---------------------------+---------------+
```

In the Hive shell, use the following code to create an empty table called **covid19_top10_countries**

```
DROP TABLE IF EXISTS covid19_top10_countries;

CREATE EXTERNAL TABLE IF NOT EXISTS covid19_top10_countries (
      location_key STRING, country_code STRING, country_name STRING, subregion1_name
      STRING, subregion2_name STRING, cumulative_confirmed INT, cumulative_deceased INT,
      cumulative_recovered INT, cumulative_hospitalized_patients INT,
      cumulative_intensive_care_patients INT, cumulative_persons_fully_vaccinated INT,
      adult_male_mortality_rate DOUBLE, adult_female_mortality_rate DOUBLE,
      life_expectancy DOUBLE, diabetes_prevalence DOUBLE, health_expenditure_usd DOUBLE,
      out_of_pocket_health_expenditure_usd DOUBLE, population INT, population_male INT,
      population_female INT, population_age_00_09 INT, population_age_10_19 INT,
      population_age_20_29 INT, population_age_30_39 INT,population_age_40_49 INT,
      population_age_50_59 INT, population_age_60_69 INT, population_age_70_79 INT,
      population_age_80_and_older INT, latitude DOUBLE, longitude DOUBLE,
      cumulative_deceased_male INT, cumulative_deceased_female INT,
      cumulative_recovered_male INT, cumulative_recovered_female INT,
      cumulative_recovered_age_0_9 INT, cumulative_recovered_age_10_19 INT,
      cumulative_recovered_age_20_29 INT, cumulative_recovered_age_30_39 INT,
      cumulative_recovered_age_40_49 INT, cumulative_recovered_age_50_59 INT,
      cumulative_recovered_age_60_69 INT, cumulative_recovered_age_70_79 INT,
      cumulative_recovered_age_80_89 INT, cumulative_recovered_age_90_99 INT)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
LOCATION '/tmp/group3_covid19/top10_countries/';
```

Copy and paste the following Hive code to Hive shell in order to create a table **covid19_top10_countries.** It will be data from the top 10 countries with the most confirmed cases from the **covid19_aggregated** tables

```
WITH top10 AS (
    SELECT country_name, MAX(cumulative_confirmed) AS total_cases
    FROM covid19_aggregated
    WHERE cumulative_confirmed IS NOT NULL
    GROUP BY country_name
    ORDER BY total_cases DESC
    LIMIT 10)
INSERT OVERWRITE TABLE covid19_top10_countries
SELECT
       a.location_key, a.country_code, a.country_name, a.subregion1_name, a.subregion2_name,
       MAX(a.cumulative_confirmed) AS cumulative_confirmed, MAX(a.cumulative_deceased) AS
       cumulative_deceased, MAX(a.cumulative_recovered) AS cumulative_recovered,
       MAX(a.cumulative_hospitalized_patients) AS cumulative_hospitalized_patients,
       MAX(a.cumulative_intensive_care_patients) AS cumulative_intensive_care_patients,
       MAX(a.cumulative_persons_fully_vaccinated) AS cumulative_persons_fully_vaccinated,
       MAX(a.adult_male_mortality_rate), MAX(a.adult_female_mortality_rate),
       MAX(a.life_expectancy), MAX(a.diabetes_prevalence), MAX(a.health_expenditure_usd),
       MAX(a.out_of_pocket_health_expenditure_usd), MAX(a.population),
       MAX(a.population_male), MAX(a.population_female), MAX(a.population_age_00_09),
       MAX(a.population_age_10_19), MAX(a.population_age_20_29),
       MAX(a.population_age_30_39), MAX(a.population_age_40_49),
       MAX(a.population_age_50_59), MAX(a.population_age_60_69),
       MAX(a.population_age_70_79), MAX(a.population_age_80_and_older), a.latitude,
       a.longitude, MAX(a.cumulative_deceased_male), MAX(a.cumulative_deceased_female),
       MAX(a.cumulative_recovered_male), MAX(a.cumulative_recovered_female),
       MAX(a.cumulative_recovered_age_0_9), MAX(a.cumulative_recovered_age_10_19),
       MAX(a.cumulative_recovered_age_20_29), MAX(a.cumulative_recovered_age_30_39),
       MAX(a.cumulative_recovered_age_40_49), MAX(a.cumulative_recovered_age_50_59),
       MAX(a.cumulative_recovered_age_60_69), MAX(a.cumulative_recovered_age_70_79),
       MAX(a.cumulative_recovered_age_80_89), MAX(a.cumulative_recovered_age_90_99)
FROM covid19_aggregated a
JOIN top10 t ON a.country_name = t.country_name
WHERE a.subregion1_name IS NOT NULL
GROUP BY a.location_key,a.country_code,a.country_name,a.subregion1_name,a.subregion2_name;
```

Query from the table and to see if it has the correct data and values:

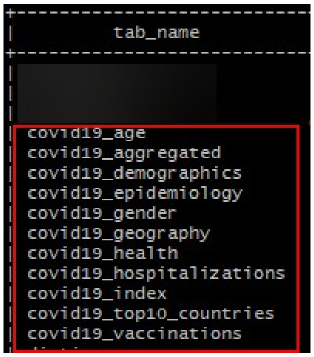
```
SELECT * FROM covid19_top10_countries LIMIT 10;
```

In the Beeline shell, you need to check if the table **covid19_top10_countries** shows your database and hdfs directory

```
describe formatted covid19_top10_countries;
```

Then, in the hive shell, you need to check if the tables **covid19_aggregated** and **covid19_top10_countries** are shown:

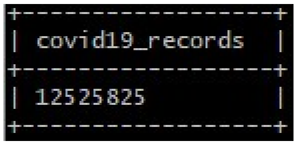
```
show tables;
```



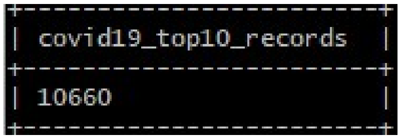
```
+---------------------------
|         tab_name
+---------------------------
[redacted]
| covid19_age
| covid19_aggregated
| covid19_demographics
| covid19_epidemiology
| covid19_gender
| covid19_geography
| covid19_health
| covid19_hospitalizations
| covid19_index
| covid19_top10_countries
| covid19_vaccinations
```

Compare the number of records between the tables **covid19_aggregated** and **covid19_ top10_countries.** You will see the table was reduced from **12,525,825** to **10,660.**

```
SELECT COUNT(*) FROM covid19_aggregated;
SELECT COUNT(*) FROM covid19_top10_countries;
```



```
+-------------------+
| covid19_records   |
+-------------------+
| 12525825          |
+-------------------+
```



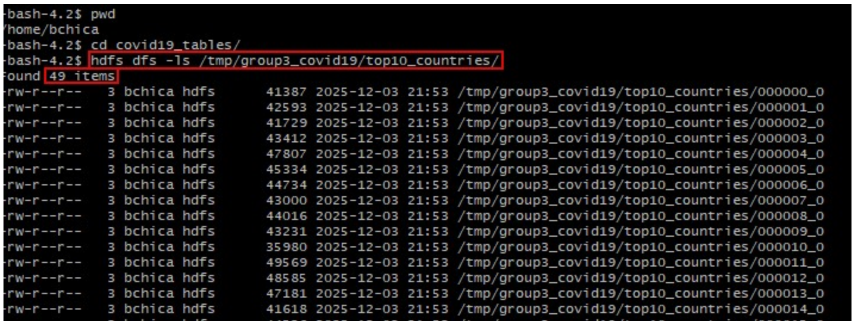
```
+-------------------------+
| covid19_top10_records   |
+-------------------------+
| 10660                   |
+-------------------------+
```

Step 6: Export Cleaned Table

After identifying the top ten countries, the cleaned dataset is exported from HDFS to a single CSV file. The (-getmerge) command is used to combine multiple output files into one consolidated file, which is then downloaded to the local machine. This exported dataset is used for visualization and further analysis.

After the **covid19_top10_countries** table is created, listing the contents inside the ***/tmp/group3_covid19/top10_countries/*** will show 49 files

```
hdfs dfs -ls /tmp/group3_covid19/top10_countries/
```



```
-bash-4.2$ pwd
/home/bchica
-bash-4.2$ cd covid19_tables/
-bash-4.2$ hdfs dfs -ls /tmp/group3_covid19/top10_countries/
Found 49 items
-rw-r--r--   3 bchica hdfs      41387 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000000_0
-rw-r--r--   3 bchica hdfs      42593 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000001_0
-rw-r--r--   3 bchica hdfs      41729 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000002_0
-rw-r--r--   3 bchica hdfs      43412 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000003_0
-rw-r--r--   3 bchica hdfs      47807 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000004_0
-rw-r--r--   3 bchica hdfs      45334 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000005_0
-rw-r--r--   3 bchica hdfs      44734 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000006_0
-rw-r--r--   3 bchica hdfs      43000 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000007_0
-rw-r--r--   3 bchica hdfs      44016 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000008_0
-rw-r--r--   3 bchica hdfs      43231 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000009_0
-rw-r--r--   3 bchica hdfs      35980 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000010_0
-rw-r--r--   3 bchica hdfs      49569 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000011_0
-rw-r--r--   3 bchica hdfs      48585 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000012_0
-rw-r--r--   3 bchica hdfs      47181 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000013_0
-rw-r--r--   3 bchica hdfs      41618 2025-12-03 21:53 /tmp/group3_covid19/top10_countries/000014_0
```

Merge the 49 output files to a file named **covid19_top10_countries.csv** using the following:

```
        hdfs dfs -getmerge /tmp/group3_covid19/top10_countries/0000*
        covid19_top10_countries.csv
```

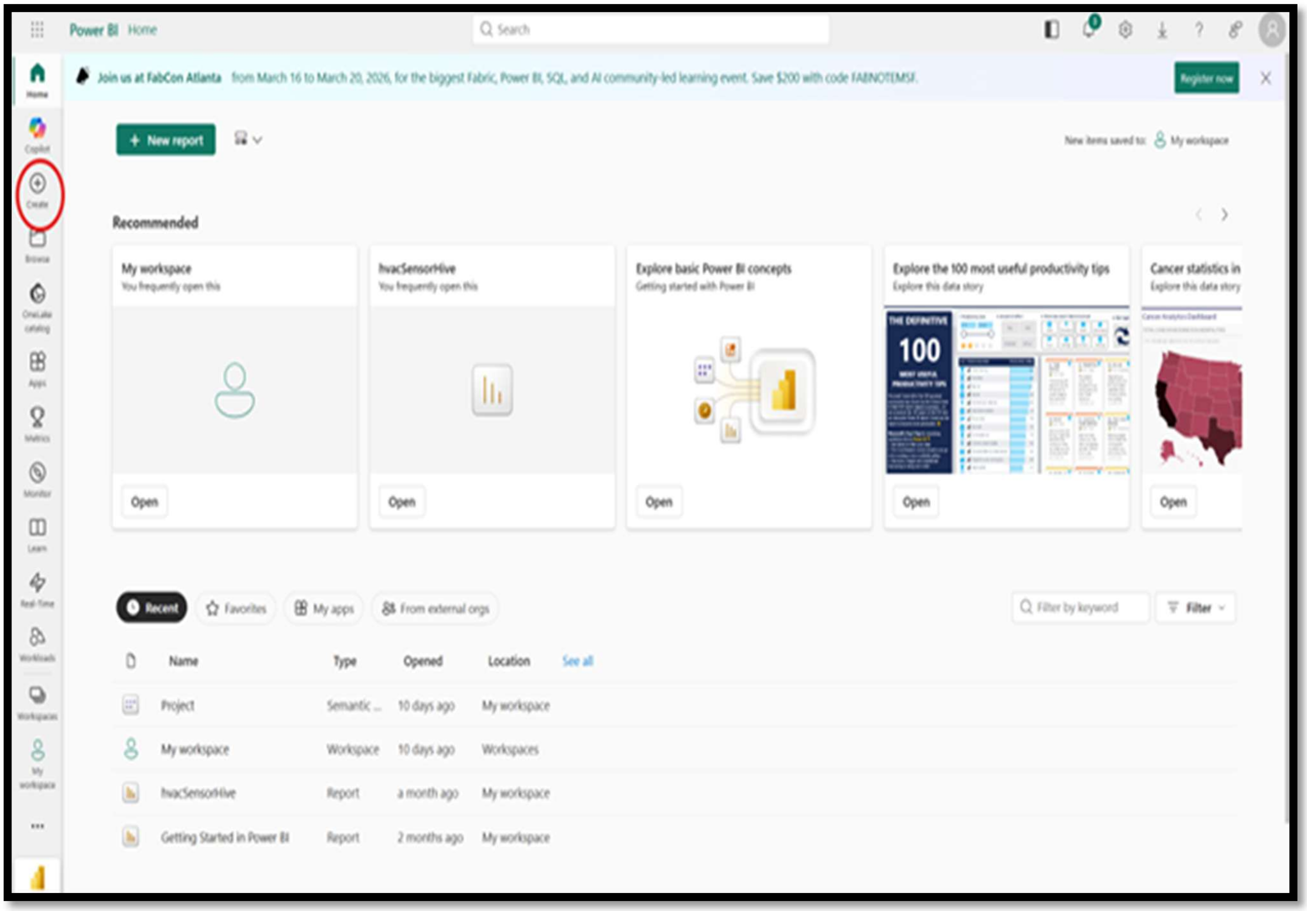
Logout of the cluster and download the csv file into your local machine

```
        scp username@ip:/home/bchica/covid19_tables/covid19_top10_countries.csv
        ~/Downloads/
```

Step 7: Visualization using PowerBI

Power BI is used to visualize the results of the aggregated COVID-19 analysis. A bubble map is created to display the top ten countries by cumulative confirmed COVID-19 cases, where bubble size represents the relative magnitude of total confirmed cases. This visualization provides a clear geographic comparison of COVID-19 impact among the most affected countries and supports the findings derived from the Hive-based analysis.

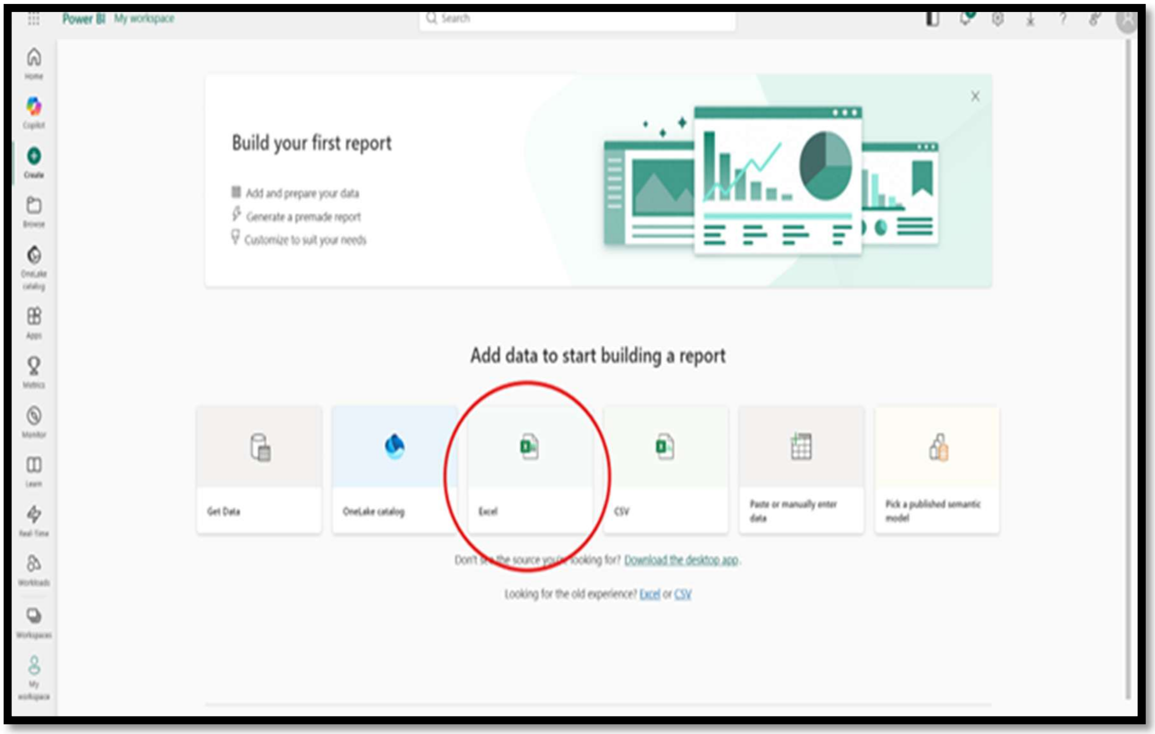
Open your MS Power BI Desktop at your local computer. Select **Create**



Join us at FabCon Atlanta from March 16 to March 20, 2026, for the biggest Fabric, Power BI, SQL, and AI community-led learning event. Save $200 with code FABNOTEMSF.

Name	Type	Opened	Location
Project	Semantic ...	10 days ago	My workspace
My workspace	Workspace	10 days ago	Workspaces
hvacSensorHive	Report	a month ago	My workspace
Getting Started in Power BI	Report	2 months ago	My workspace

Once you see the following page, select **csv**



Build your first report

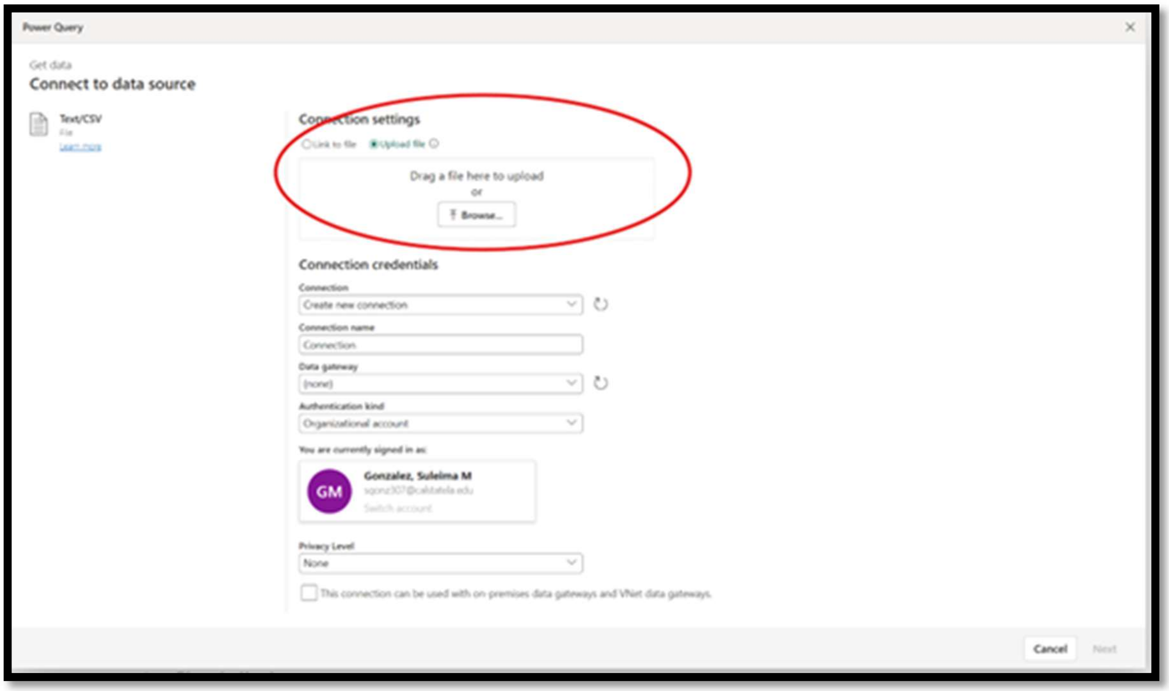
- Add and prepare your data
- Generate a premade report
- Customize to suit your needs

Add data to start building a report

Don't see the source you're looking for? Download the desktop app.

Looking for the old experience? Excel or CSV

Select the **Browse** button. Choose the file you want to use on the File Explorer, then hit **Next**



Get data

Connect to data source

Connection settings

○ Link to file ● Upload file

Drag a file here to upload
or

Connection credentials

Connection: Create new connection

Connection name: Connection

Data gateway: (none)

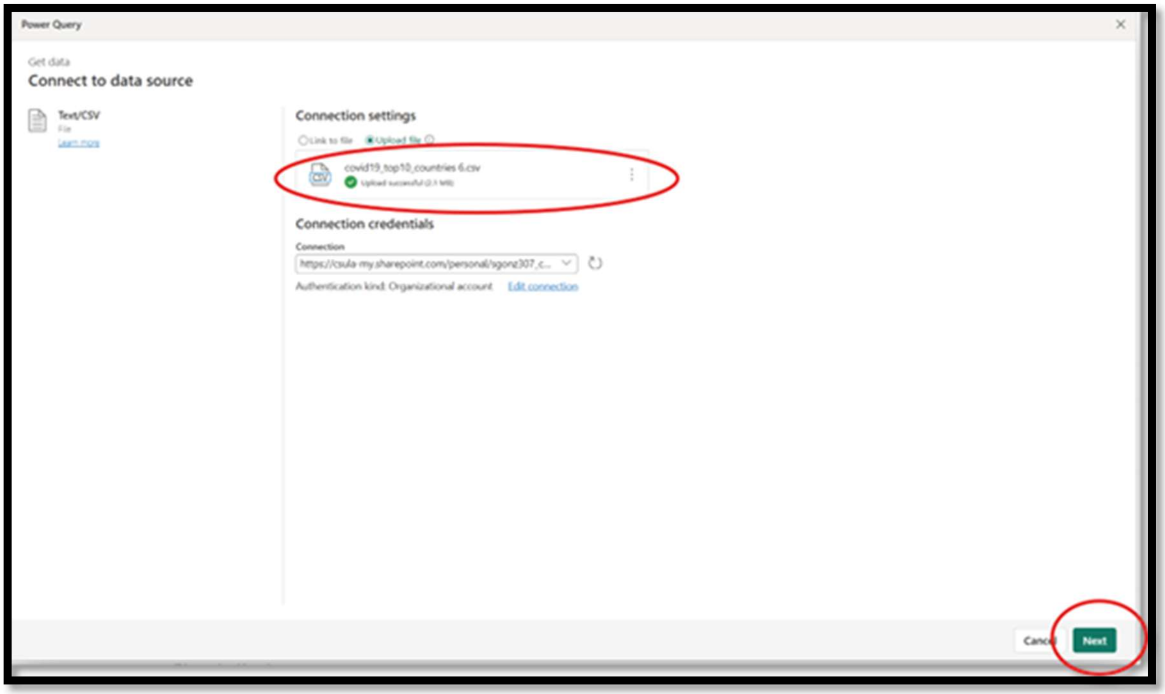
Authentication kind: Organizational account

You are currently signed in as:

Gonzalez, Suleima M
sgonz307@calstatela.edu

Privacy Level: None

□ This connection can be used with on-premises data gateways and VNet data gateways.



Power Query

Get data

Connect to data source

Text/CSV
File
Learn more

Connection settings

○ Link to file ● Upload file

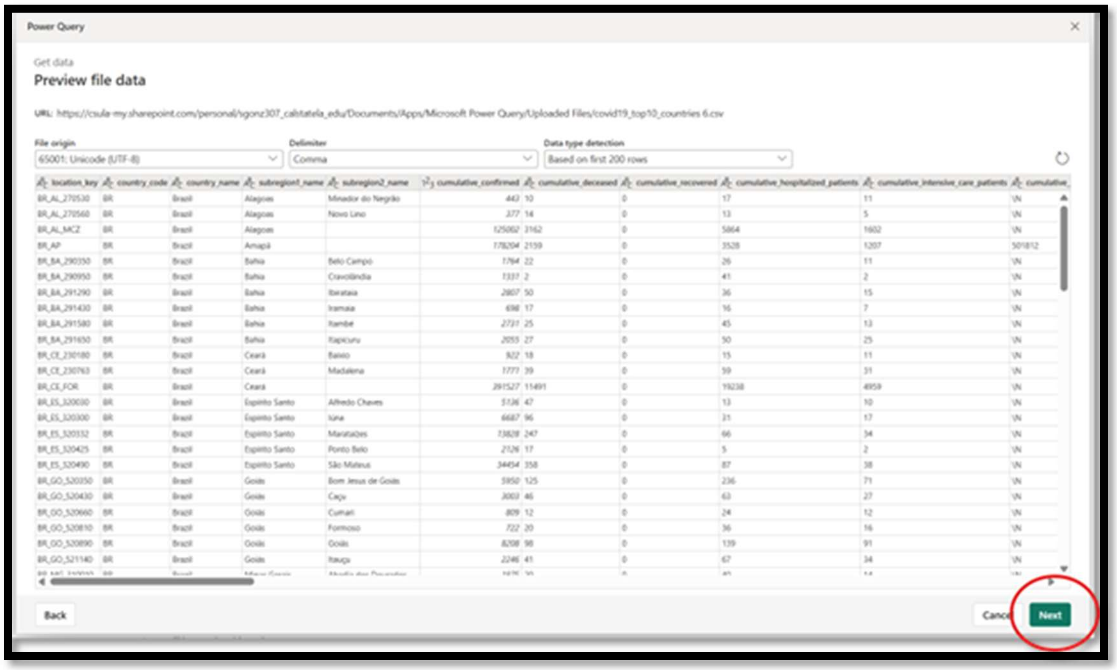
covid19_top10_countries 6.csv
Upload successful (2.1 MB)

Connection credentials

Connection: https://csula-my.sharepoint.com/personal/sgonz307_c...

Authentication kind: Organizational account

You will see the following page in PowerBI to **preview** your dataset



Power Query

Get data

Preview file data

URL: https://csula-my.sharepoint.com/personal/sgonz307_calstatela_edu/Documents/Apps/Microsoft Power Query/Uploaded Files/covid19_top10_countries 6.csv

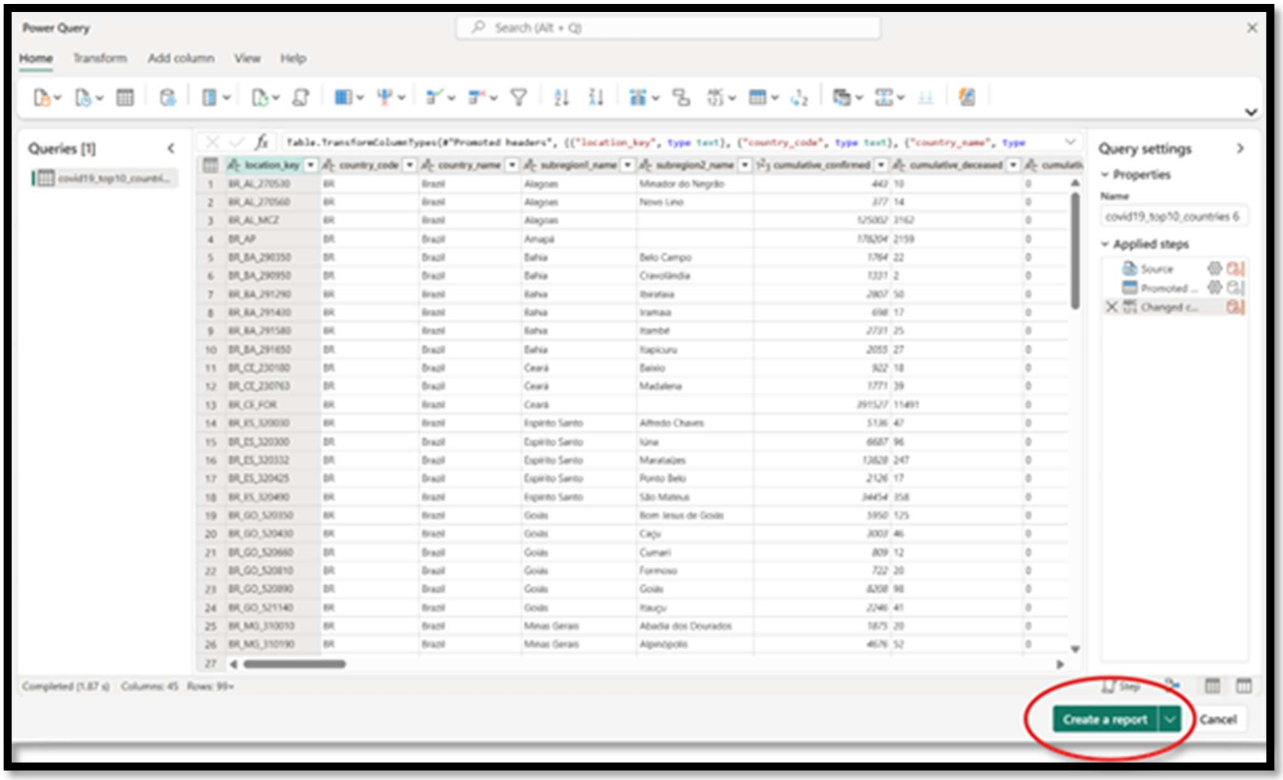
File origin: 65001: Unicode (UTF-8)

Delimiter: Comma

Data type detection: Based on first 200 rows

location_key	country_code	country_name	subregion1_name	subregion2_name	cumulative_confirmed	cumulative_deceased	cumulative_recovered	cumulative_hospitalized_patients	cumulative_intensive_care_patients	cumulative_
BR_AL_270530	BR	Brazil	Alagoas	Minador do Negrão	443	10	0	17	11	\N
BR_AL_270560	BR	Brazil	Alagoas	Novo Lino	377	14	0	13	5	\N
BR_AL_MCZ	BR	Brazil	Alagoas		125002	3162	0	5864	1602	\N
BR_AP	BR	Brazil	Amapá		178204	2159	0	3528	1207	501812
BR_BA_290350	BR	Brazil	Bahia	Belo Campo	1764	22	0	26	11	\N
BR_BA_290950	BR	Brazil	Bahia	Cravolândia	1337	2	0	41	2	\N
BR_BA_291290	BR	Brazil	Bahia	Ibirataia	2807	50	0	36	15	\N
BR_BA_291430	BR	Brazil	Bahia	Iramaia	698	17	0	16	7	\N
BR_BA_291580	BR	Brazil	Bahia	Itambé	2731	25	0	45	13	\N
BR_BA_291650	BR	Brazil	Bahia	Itapicuru	2055	27	0	50	25	\N
BR_CE_230180	BR	Brazil	Ceará	Baixio	922	18	0	15	11	\N
BR_CE_230763	BR	Brazil	Ceará	Madalena	1777	39	0	59	31	\N
BR_CE_FOR	BR	Brazil	Ceará		291527	11491	0	19238	4959	\N
BR_ES_320030	BR	Brazil	Espírito Santo	Alfredo Chaves	5136	47	0	13	10	\N
BR_ES_320300	BR	Brazil	Espírito Santo	Iúna	6687	96	0	31	17	\N
BR_ES_320332	BR	Brazil	Espírito Santo	Marataízes	13828	247	0	66	34	\N
BR_ES_320425	BR	Brazil	Espírito Santo	Ponto Belo	2126	17	0	5	2	\N
BR_ES_320490	BR	Brazil	Espírito Santo	São Mateus	34454	358	0	87	38	\N
BR_GO_520350	BR	Brazil	Goiás	Bom Jesus de Goiás	5950	125	0	236	71	\N
BR_GO_520430	BR	Brazil	Goiás	Caçu	3003	46	0	63	27	\N
BR_GO_520660	BR	Brazil	Goiás	Cumari	809	12	0	24	12	\N
BR_GO_520810	BR	Brazil	Goiás	Formoso	722	20	0	36	16	\N
BR_GO_520890	BR	Brazil	Goiás	Goiás	8208	98	0	139	91	\N
BR_GO_521140	BR	Brazil	Goiás	Itauçu	2246	41	0	67	34	\N
[illegible]	BR	Brazil	Minas Gerais	Abadia dos Dourados	[illegible]	30	0	40	14	\N

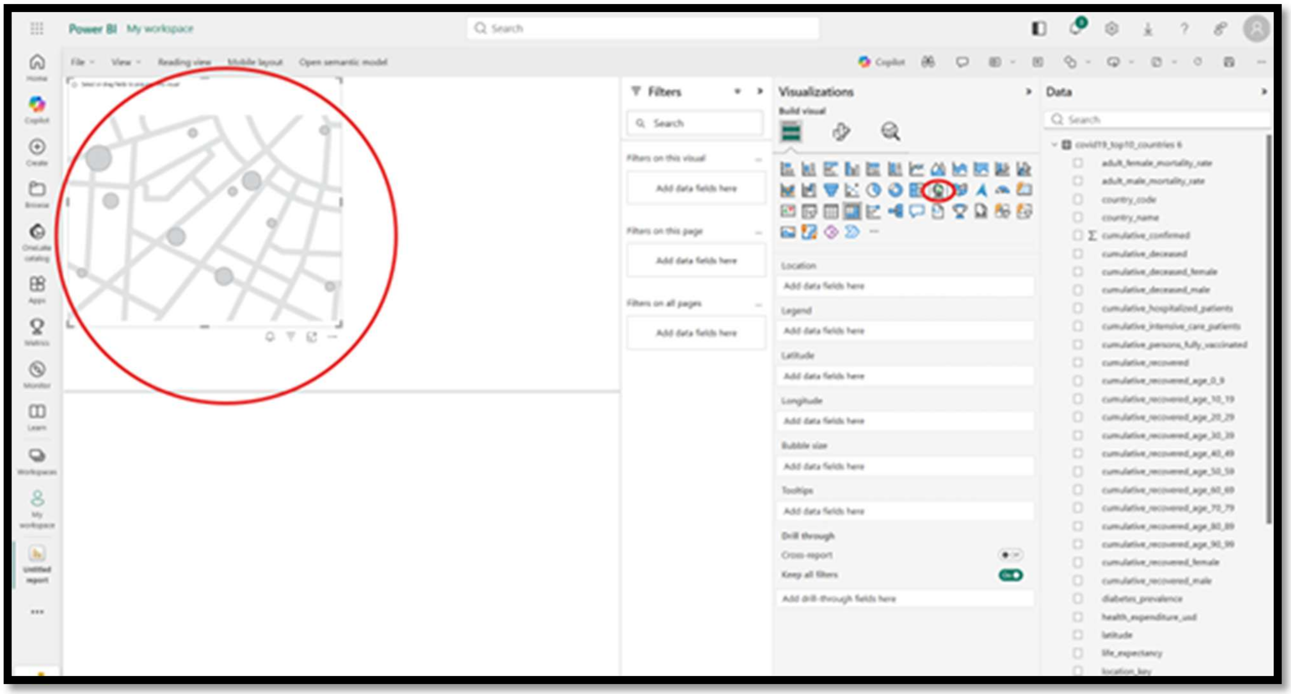
On this page, select **Create a report -> Create**



```
Table.TransformColumnTypes(#"Promoted headers", {{"location_key", type text}, {"country_code", type text}, {"country_name", type
```

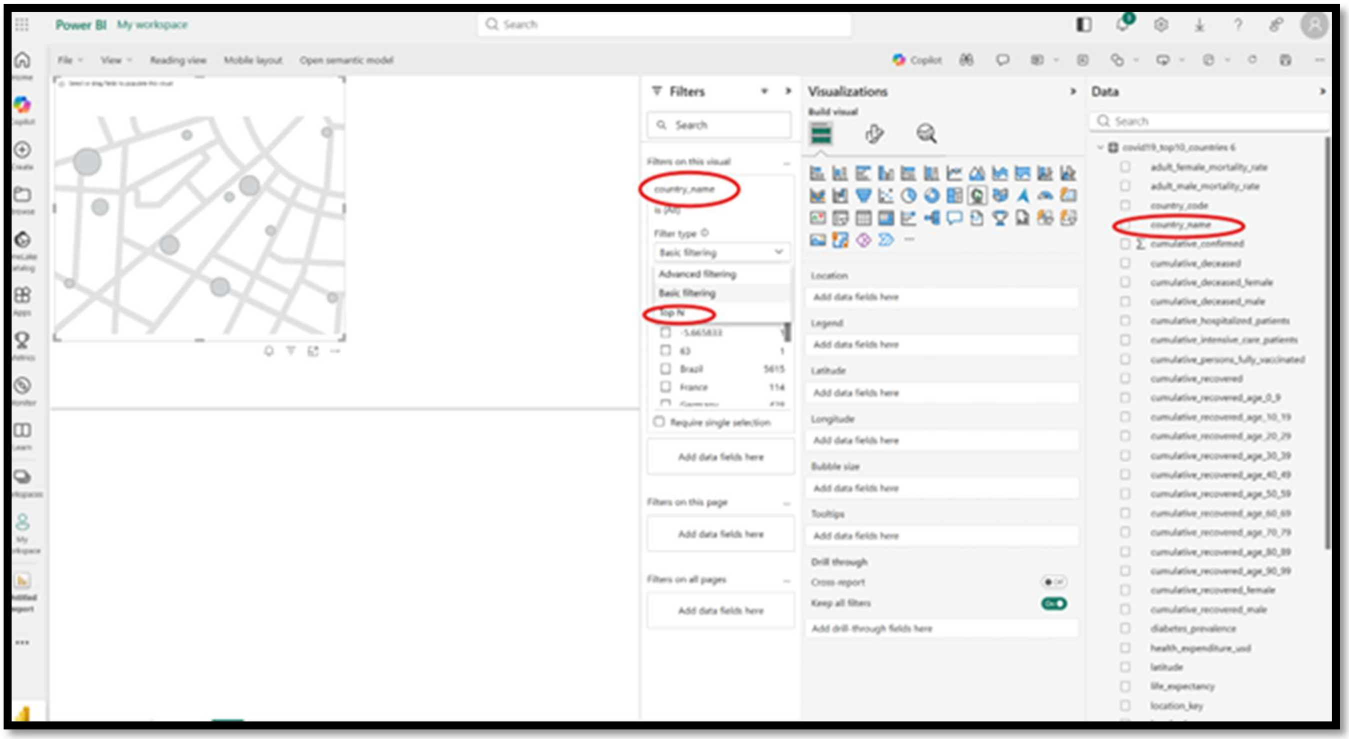
	location_key	country_code	country_name	subregion1_name	subregion2_name	cumulative_confirmed	cumulative_deceased	cumulati
1	BR_AL_270530	BR	Brazil	Alagoas	Minador do Negrão	443	10	0
2	BR_AL_270560	BR	Brazil	Alagoas	Novo Lino	377	14	0
3	BR_AL_MCZ	BR	Brazil	Alagoas		125002	3162	0
4	BR_AP	BR	Brazil	Amapá		178204	2159	0
5	BR_BA_290350	BR	Brazil	Bahia	Belo Campo	1764	22	0
6	BR_BA_290950	BR	Brazil	Bahia	Cravolândia	1331	2	0
7	BR_BA_291290	BR	Brazil	Bahia	Ibiratáia	2807	50	0
8	BR_BA_291430	BR	Brazil	Bahia	Iramaia	698	17	0
9	BR_BA_291580	BR	Brazil	Bahia	Itambé	2731	25	0
10	BR_BA_291650	BR	Brazil	Bahia	Itapicuru	2055	27	0
11	BR_CE_230180	BR	Brazil	Ceará	Baixio	922	18	0
12	BR_CE_230763	BR	Brazil	Ceará	Madalena	1771	39	0
13	BR_CE_FOR	BR	Brazil	Ceará		391527	11491	0
14	BR_ES_320030	BR	Brazil	Espírito Santo	Alfredo Chaves	5136	47	0
15	BR_ES_320300	BR	Brazil	Espírito Santo	Iúna	6687	96	0
16	BR_ES_320332	BR	Brazil	Espírito Santo	Marataízes	13828	247	0
17	BR_ES_320425	BR	Brazil	Espírito Santo	Ponto Belo	2126	17	0
18	BR_ES_320490	BR	Brazil	Espírito Santo	São Mateus	34454	358	0
19	BR_GO_520350	BR	Brazil	Goiás	Bom Jesus de Goiás	5950	125	0
20	BR_GO_520430	BR	Brazil	Goiás	Caçu	3003	46	0
21	BR_GO_520660	BR	Brazil	Goiás	Cumari	809	12	0
22	BR_GO_520810	BR	Brazil	Goiás	Formoso	722	20	0
23	BR_GO_520890	BR	Brazil	Goiás	Goiás	8208	98	0
24	BR_GO_521140	BR	Brazil	Goiás	Itauçu	2246	41	0
25	BR_MG_310010	BR	Brazil	Minas Gerais	Abadia dos Dourados	1875	20	0
26	BR_MG_310190	BR	Brazil	Minas Gerais	Alpinópolis	4676	52	0
27								

Click on the **map** icon to begin setting up the visualization



covid19_top10_countries 6
- □ adult_female_mortality_rate
- □ adult_male_mortality_rate
- □ country_code
- □ country_name
- □ Σ cumulative_confirmed
- □ cumulative_deceased
- □ cumulative_deceased_female
- □ cumulative_deceased_male
- □ cumulative_hospitalized_patients
- □ cumulative_intensive_care_patients
- □ cumulative_persons_fully_vaccinated
- □ cumulative_recovered
- □ cumulative_recovered_age_0_9
- □ cumulative_recovered_age_10_19
- □ cumulative_recovered_age_20_29
- □ cumulative_recovered_age_30_39
- □ cumulative_recovered_age_40_49
- □ cumulative_recovered_age_50_59
- □ cumulative_recovered_age_60_69
- □ cumulative_recovered_age_70_79
- □ cumulative_recovered_age_80_89
- □ cumulative_recovered_age_90_99
- □ cumulative_recovered_female
- □ cumulative_recovered_male
- □ diabetes_prevalence
- □ health_expenditure_usd
- □ latitude
- □ life_expectancy
- □ location_key

Drag the **country_name** to the filters on this visual


Power BI
My workspace
Search
File
View
Reading view
Mobile layout
Open semantic model
Copilot
Filters
Search
Filters on this visual
Visualizations
Build visual
Location
Add data fields here
Legend
Add data fields here
Latitude
Add data fields here
Longitude
Add data fields here
Bubble size
Add data fields here
Tooltips
Add data fields here
Drill through
Cross-report
Keep all filters
Add drill-through fields here
Data
Search
covid19_top10_countries 6
adult_female_mortality_rate
adult_male_mortality_rate
country_code
country_name
cumulative_confirmed
cumulative_deceased
cumulative_deceased_female
cumulative_deceased_male
cumulative_hospitalized_patients
cumulative_intensive_care_patients
cumulative_persons_fully_vaccinated
cumulative_recovered
cumulative_recovered_age_0_9
cumulative_recovered_age_10_19
cumulative_recovered_age_20_29
cumulative_recovered_age_30_39
cumulative_recovered_age_40_49
cumulative_recovered_age_50_59
cumulative_recovered_age_60_69
cumulative_recovered_age_70_79
cumulative_recovered_age_80_89
cumulative_recovered_age_90_99
cumulative_recovered_female
cumulative_recovered_male
diabetes_prevalence
health_expenditure_usd
latitude
life_expectancy
location_key

country_name
is (All)

Filter type
Basic filtering
- Advanced filtering
- Basic filtering
- Top N

□ -5.665833	1
□ 63	1
□ Brazil	5615
□ France	114
□ Germany	419

□ Require single selection

Add data fields here

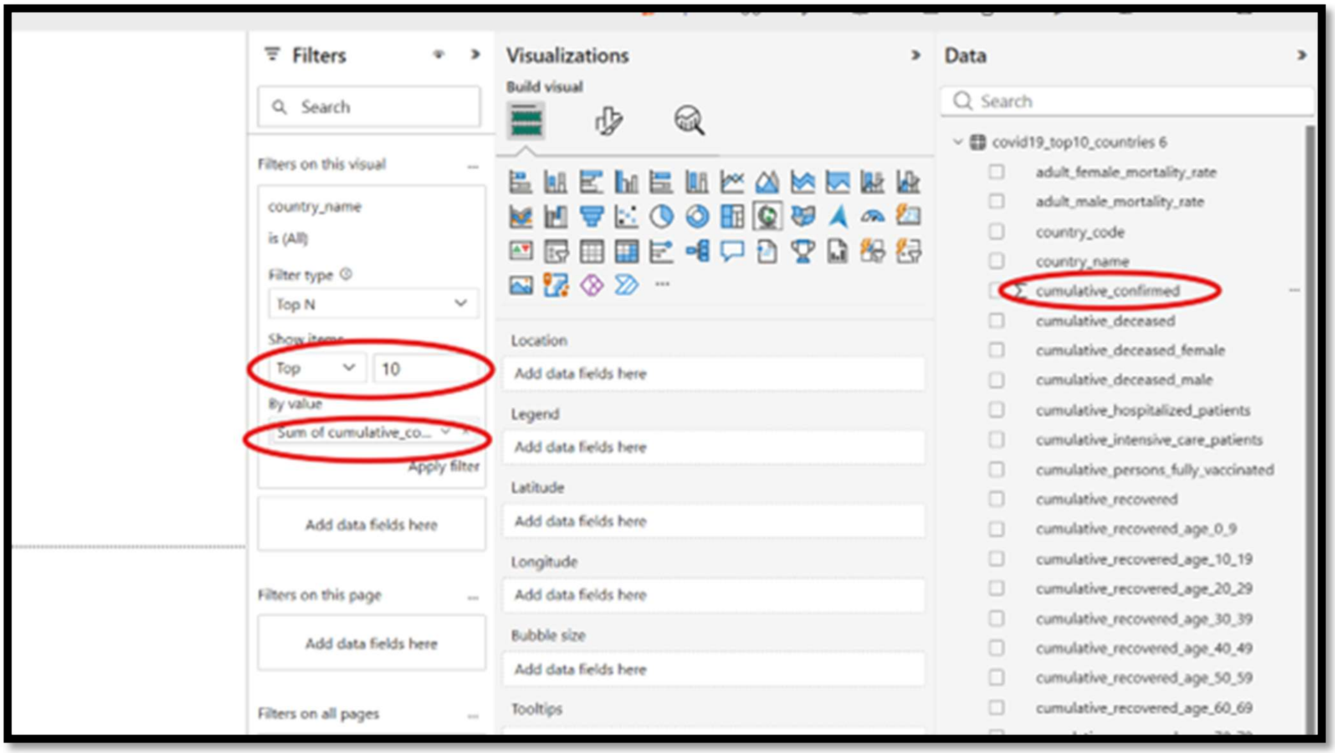
Filters on this page

Add data fields here

Filters on all pages

Add data fields here

On Filter type click Top N, then on show items click Top then next to it change it to **10** then drag the **cumulative_confirmed** to By value, then click **Apply filer**


Visualizations
Build visual
Location
Add data fields here
Legend
Add data fields here
Latitude
Add data fields here
Longitude
Add data fields here
Bubble size
Add data fields here
Tooltips
Data
Search
covid19_top10_countries 6
adult_female_mortality_rate
adult_male_mortality_rate
country_code
country_name
cumulative_confirmed
cumulative_deceased
cumulative_deceased_female
cumulative_deceased_male
cumulative_hospitalized_patients
cumulative_intensive_care_patients
cumulative_persons_fully_vaccinated
cumulative_recovered
cumulative_recovered_age_0_9
cumulative_recovered_age_10_19
cumulative_recovered_age_20_29
cumulative_recovered_age_30_39
cumulative_recovered_age_40_49
cumulative_recovered_age_50_59
cumulative_recovered_age_60_69

Filters

Search

Filters on this visual

country_name
is (All)

Filter type
Top N

Show items
Top 10

By value
Sum of cumulative_co...

Apply filter

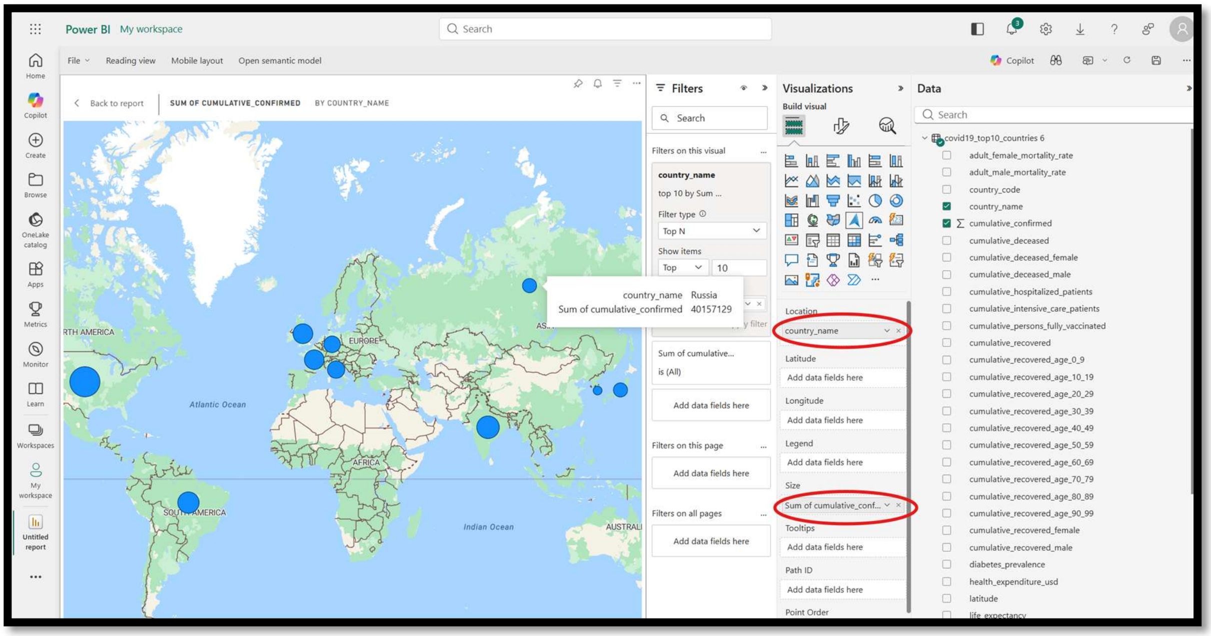
Add data fields here

Filters on this page

Add data fields here

Filters on all pages

Drag the **country_name** field to the **Location** area and the **cumulative_confirmed** field to the **Size** area



country_name Russia
Sum of cumulative_confirmed 40157129

Filters on this visual

country_name
top 10 by Sum ...
Filter type: Top N
Show items: Top 10

Sum of cumulative...
is (All)

Add data fields here

Filters on this page

Add data fields here

Filters on all pages

Add data fields here

Location: country_name

Latitude: Add data fields here

Longitude: Add data fields here

Legend: Add data fields here

Size: Sum of cumulative_conf...

Tooltips: Add data fields here

Path ID: Add data fields here

Point Order

covid19_top10_countries 6

- □ adult_female_mortality_rate
- □ adult_male_mortality_rate
- □ country_code
- ☑ country_name
- ☑ Σ cumulative_confirmed
- □ cumulative_deceased
- □ cumulative_deceased_female
- □ cumulative_deceased_male
- □ cumulative_hospitalized_patients
- □ cumulative_intensive_care_patients
- □ cumulative_persons_fully_vaccinated
- □ cumulative_recovered
- □ cumulative_recovered_age_0_9
- □ cumulative_recovered_age_10_19
- □ cumulative_recovered_age_20_29
- □ cumulative_recovered_age_30_39
- □ cumulative_recovered_age_40_49
- □ cumulative_recovered_age_50_59
- □ cumulative_recovered_age_60_69
- □ cumulative_recovered_age_70_79
- □ cumulative_recovered_age_80_89
- □ cumulative_recovered_age_90_99
- □ cumulative_recovered_female
- □ cumulative_recovered_male
- □ diabetes_prevalence
- □ health_expenditure_usd
- □ latitude
- □ life_expectancy

References

1. **COVID19-Data-Analysis: Final Project for CIS 4560. COVID-19 data analysis using HDFS Hive -** https://github.com/chica-94/COVID19-Data-Analysis