pymongo_queries

December 30, 2022

1 BDS Assignment 1

This submission is made by Group 136 for BDS Assignment 1.

Group 136

Members as follows

Name	BITS ID	Contribution
Vinayak Nayak	2021fc04135	100%
Shreysi Kalra	2021fc04586	100%

We have used `MongoDB Atlas` with `pymongo` and structured our solution as follows - An Exploratory Analysis of data - Populating MongoDB Atlas with our data - Executing queries with pymongo and verifying ther results with pandas

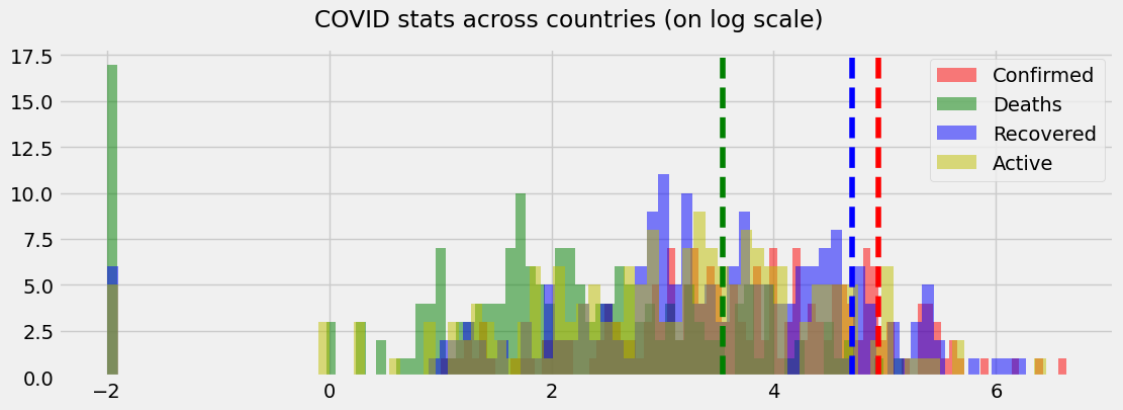
Setup Details - System: Ubuntu 20.04 LTS - pymongo - version 4.3.3 (For interfacing with the database) - MongoDB Atlas (As a cloud hosted database to store our data)

2 Exploratory Data Analysis

[ ]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

plt.style.use("fivethirtyeight")
import seaborn as sns

%matplotlib inline
```

[ ]:
```
# Define the data sources
country_data = "data/country_wise_latest.csv"
covid_data = "data/covid_19_clean_complete.csv"
day_data = "data/day_wise.csv"
full_data = "data/full_grouped.csv"
usa_data = "data/usa_county_wise.csv"
world_data = "data/worldometer_data.csv"
```

[ ]:
```
country = pd.read_csv(country_data)
fig, axis = plt.subplots(1, 1, figsize=(12, 4))

axis.hist(np.log10(country.Confirmed + 1e-2), bins=80, color="r", alpha=0.5)
axis.hist(np.log10(country.Deaths + 1e-2), bins=80, color="g", alpha=0.5)
axis.hist(np.log10(country.Recovered + 1e-2), bins=80, color="b", alpha=0.5)
axis.hist(np.log10(country.Active + 1e-2), bins=80, color="y", alpha=0.5)
axis.legend(["Confirmed", "Deaths", "Recovered", "Active"])

axis.axvline(np.log10(country.Confirmed.mean()), color="r", linestyle="--")
axis.axvline(np.log10(country.Deaths.mean()), color="g", linestyle="--")
axis.axvline(np.log10(country.Recovered.mean()), color="b", linestyle="--")
fig.suptitle("COVID stats across countries (on log scale)");
```



[ ]:
```
country.describe()
```

[ ]:

	Confirmed	Deaths	Recovered	Active	New cases	\
count	1.870000e+02	187.000000	1.870000e+02	1.870000e+02	187.000000	
mean	8.813094e+04	3497.518717	5.063148e+04	3.400194e+04	1222.957219	
std	3.833187e+05	14100.002482	1.901882e+05	2.133262e+05	5710.374790	
min	1.000000e+01	0.000000	0.000000e+00	0.000000e+00	0.000000	
25%	1.114000e+03	18.500000	6.265000e+02	1.415000e+02	4.000000	
50%	5.059000e+03	108.000000	2.815000e+03	1.600000e+03	49.000000	
75%	4.046050e+04	734.000000	2.260600e+04	9.149000e+03	419.500000	
max	4.290259e+06	148011.000000	1.846641e+06	2.816444e+06	56336.000000	

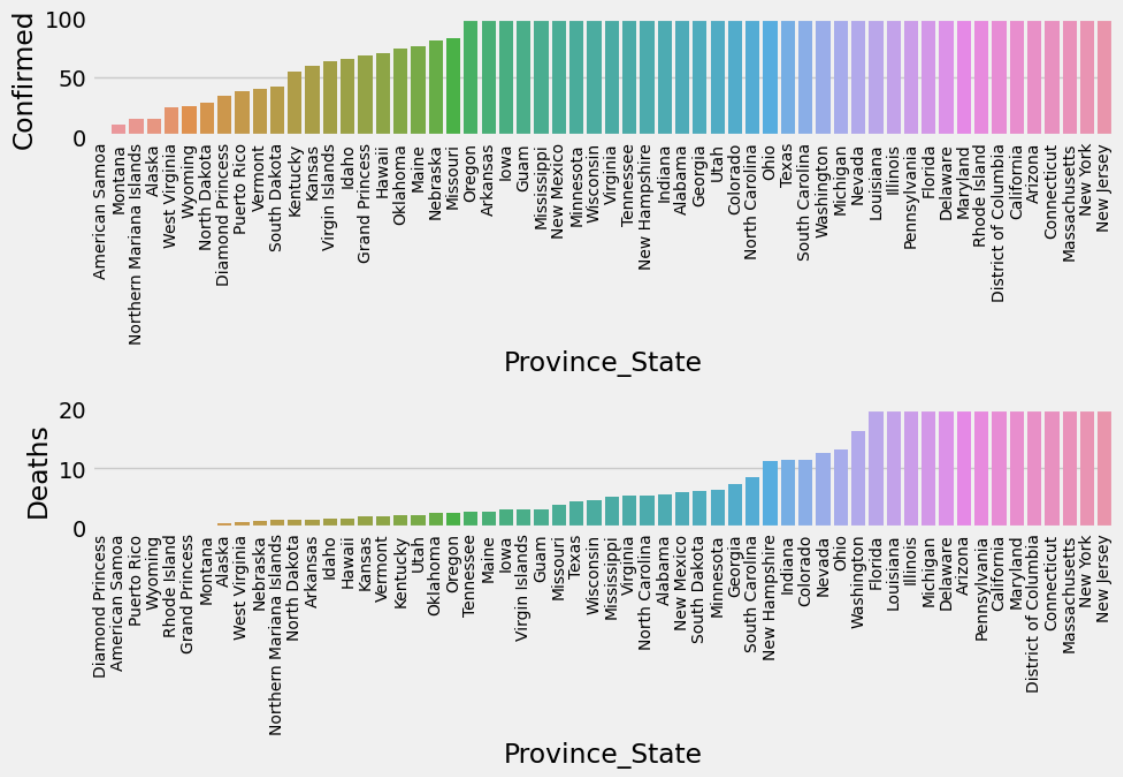
	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	\
count	187.000000	187.000000	187.000000	187.000000	
mean	28.957219	933.812834	3.019519	64.820535	
std	120.037173	4197.719635	3.454302	26.287694	
min	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.945000	48.770000	
50%	1.000000	22.000000	2.150000	71.320000	

```
75%       6.000000     221.000000            3.875000              86.885000
max    1076.000000   33728.000000           28.560000             100.000000

       Deaths / 100 Recovered  Confirmed last week  1 week change  \
count                  187.00         1.870000e+02     187.000000
mean                      inf         7.868248e+04    9448.459893
std                       NaN         3.382737e+05   47491.127684
min                      0.00         1.000000e+01     -47.000000
25%                      1.45         1.051500e+03      49.000000
50%                      3.62         5.020000e+03     432.000000
75%                      6.44         3.708050e+04    3172.000000
max                       inf         3.834677e+06  455582.000000

       1 week % increase
count         187.000000
mean           13.606203
std            24.509838
min            -3.840000
25%             2.775000
50%             6.890000
75%            16.855000
max           226.320000
```

Observations

1. There are observations with zero values (maybe missing observations or genuine data as well) in the dataset hence the spike at -2 which we have manually induced to restrict the ranges.
2. Confirmed distribution is peaks to the right of Recovered distribution which means that the recovery rate is slower than the contagion rate for the given data.
3. Deaths are peaking to the left of recovered with an order of magnitude's difference which is a very good sign. More people are getting recovered than dying.

[ ]:
```
usa = pd.read_csv(usa_data)
usa = usa.groupby(by=["Province_State"])[["Confirmed", "Deaths"]].mean().
 ↪reset_index()
confirmed_cases = usa.sort_values(by="Confirmed")
death_cases = usa.sort_values(by="Deaths")

fig, ax = plt.subplots(2, 1, figsize=(10, 7))

sns.barplot(data=confirmed_cases, x="Province_State", y="Confirmed", ax=ax[0])
sns.barplot(data=death_cases, x="Province_State", y="Deaths", ax=ax[1])
ax[0].set_xticklabels(confirmed_cases.Province_State, rotation=90, fontsize=10)
ax[0].set_ylim([0, 100])

ax[1].set_xticklabels(death_cases.Province_State, rotation=90, fontsize=10)
ax[1].set_ylim([0, 20])
fig.tight_layout();
```



[ ]:
```
usa.describe()
```

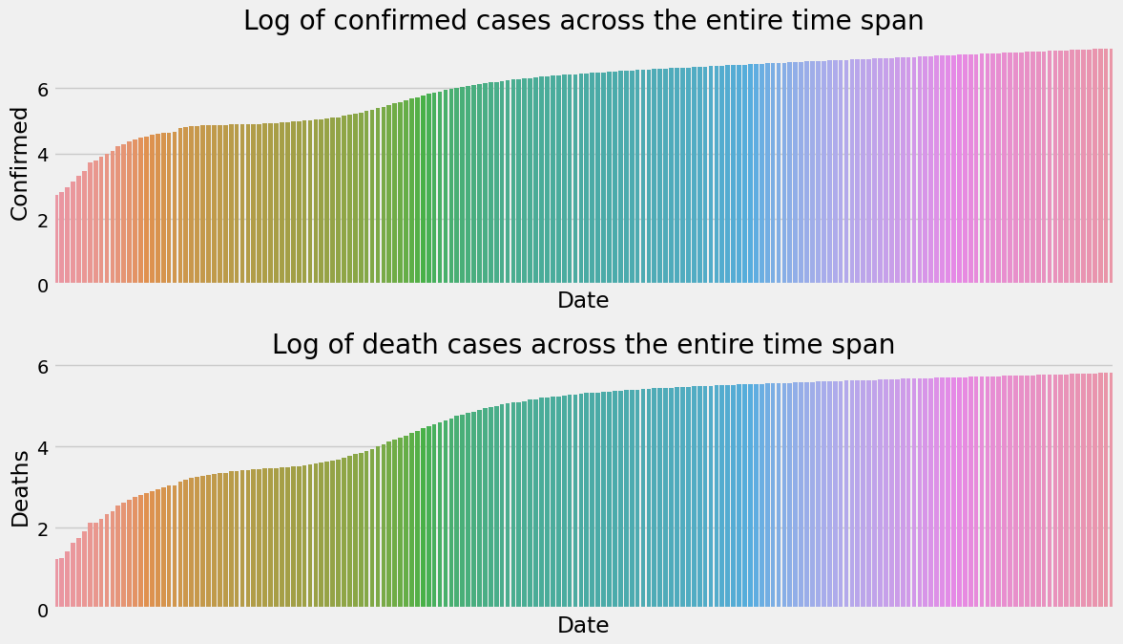
[ ]:

	Confirmed	Deaths
count	58.000000	58.000000
mean	501.599235	26.559350
std	836.286029	60.990893
min	0.000000	0.000000
25%	66.175416	1.618250
50%	175.022023	4.852940
75%	418.920583	15.446665
max	3817.463922	282.455828

Observations

1. In the USA, `NY` and `NJ` seem to have the highest number of deaths and confirmed cases respectively.
2. `Puerto Rico`, `Diamond Princess` and `American Samoa` have the lowest deaths and confirmed cases.
3. From the five point summary it seems there are several provinces where there's no deaths.
4. Thankfully for most counties number of deaths are low in comparison to compared cases ($<$ 16 deaths for $>$75% of the counties).

[ ]:
```
day = pd.read_csv(day_data)
day = day.groupby(by="Date").sum().reset_index()
fig, axis = plt.subplots(2, 1, figsize=(12, 7))

sns.barplot(x=day.Date, y=np.log10(day.Confirmed), ax=axis[0])
axis[0].set_xticks([])
axis[0].set_title("Log of confirmed cases across the entire time span")

sns.barplot(x=day.Date, y=np.log10(day.Deaths), ax=axis[1])
axis[1].set_xticks([])
axis[1].set_title("Log of death cases across the entire time span")

fig.tight_layout();
```



[ ]:
```
day[["Deaths", "Confirmed"]].describe()
```

[ ]:

	Deaths	Confirmed
count	188.000000	1.880000e+02
mean	230770.760638	4.406960e+06
std	217929.094183	4.757988e+06
min	17.000000	5.550000e+02
25%	3935.000000	1.121910e+05
50%	204190.000000	2.848733e+06
75%	418634.500000	7.422046e+06
max	654036.000000	1.648048e+07

Observations

1. Across the world, over time, we can see that the number of confirmed cases have reached over 10 million owing to this horrible disease.
2. Also, the number of deaths have almost reached the scale of million by the last timestamp available in the data.
3. The growth in death cases as well as confirmed cases is very nearly exponential in nature... (Since their log plot seems to be linearly growing).

Now we have a coarse understanding of our underlying data distribution quite well. We can start using mongodb to further dig deep and get insights about our data distribution.

3 Pushing Data to MongoDB Atlas

[ ]:
```
# Import all the necessary libraries
import os

# To suppress unnecessary warnings
import warnings

# For printing and for handling paths
from pathlib import Path
from pprint import pprint

# For cross-checking mongodb query results
import pandas as pd

# For interacting with mongodb database
import pymongo
from pymongo import MongoClient

# For watching the progress
from tqdm import tqdm

warnings.filterwarnings("ignore")
```

[ ]:
```
# Get the connection string from mongodb atlas
# Substitute the password from environment variable
CONNECTION_STRING = f"mongodb+srv://2021fc04135:{os.
 ↪environ['PASSWORD']}@bits-assignment-cluster.xwgiyxv.mongodb.net/?
 ↪retryWrites=true&w=majority"
```

[ ]:
```
# Create a mongo client
client = MongoClient(CONNECTION_STRING)

# Create a temporary database for testing out our queries
db = client["temp"]
```

[ ]:
```
# Define the data sources
country_data = "data/country_wise_latest.csv"
covid_data = "data/covid_19_clean_complete.csv"
day_data = "data/day_wise.csv"
full_data = "data/full_grouped.csv"
usa_data = "data/usa_county_wise.csv"
world_data = "data/worldometer_data.csv"
```

[ ]:
```
# Function to create a collection and add documents to that collection
def create_db_and_add_records(f, batch_size=128):
    # Give a suitable name to the collection
    collection_name = Path(f).stem
    coll = db[collection_name]

    # Read and upload entries from our csvs to the mongo collection
    df = pd.read_csv(f)
    records = df.to_dict(orient="record")

    # Insert every single record in our collection in batches of a fixed size
    n_batches = len(records) // batch_size
    if batch_size * n_batches <= len(records):
        n_batches += 1

    for idx in tqdm(
        range(n_batches), total=n_batches, desc=f"Adding to {collection_name}␣
 ↪coll."
    ):
        # Create a batch of data
        start_index = idx * batch_size
        end_index = (idx + 1) * batch_size
        recs = records[start_index:end_index]
        for num, rec in enumerate(recs):
            rec["_id"] = start_index + num

        # Bulk insert data in batches of 64 items
        try:
            coll.insert_many(recs)
        # If some of the records already exists, then move over to the next
        # record without creating a fuss
        except pymongo.errors.BulkWriteError as e:
            pass
```

[ ]:
```
# # Create all the databases
# create_db_and_add_records(country_data)
# create_db_and_add_records(day_data)
# create_db_and_add_records(usa_data)
# create_db_and_add_records(world_data)
```

```
# create_db_and_add_records(covid_data)
# create_db_and_add_records(full_data)
```

[ ]:
```
# Have a look at the created collections from above
list(db.list_collection_names())
```

[ ]:
```
['country_wise_latest',
 'worldometer_data',
 'full_grouped',
 'usa_county_wise',
 'day_wise',
 'covid_19_clean_complete']
```

[ ]:
```
# Create references to all the collections for answering subsequent questions.
country_coll = db["country_wise_latest"]
world_coll = db["worldometer_data"]
full_coll = db["full_grouped"]
usa_coll = db["usa_county_wise"]
day_coll = db["day_wise"]
covid_coll = db["covid_19_clean_complete"]
```

4 Querying the database for respective questions

4.1 Q1

The number of new cases, new deaths and new recovered

[ ]:
```
qry = country_coll.aggregate(
    [
        {
            "$group": {
                "_id": "Aggregate Stats",
                "Total Cases": {"$sum": "$New cases"},
                "Total Recovered": {"$sum": "$New recovered"},
                "Total Deaths": {"$sum": "$New deaths"},
            }
        }
    ]
)
pprint(list(qry))
```

```
[{'Total Cases': 228693,
  'Total Deaths': 5415,
  'Total Recovered': 174623,
  '_id': 'Aggregate Stats'}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(country_data)
df[["New cases", "New deaths", "New recovered"]].sum()
```

[ ]:
```
New cases        228693
New deaths         5415
New recovered    174623
dtype: int64
```

4.2 Q2

The number of death cases in each country of continent Asia and also the corresponding WHO regions

[ ]:
```
qry = world_coll.find(
    {"Continent": "Asia"},
    projection={"TotalDeaths": 1, "WHO Region": 1, "Country/Region": 1, "_id":␣
 ↪0},
)

pprint(list(qry))
```

```
[{'Country/Region': 'India',
  'TotalDeaths': 41638.0,
  'WHO Region': 'South-EastAsia'},
 {'Country/Region': 'Iran',
  'TotalDeaths': 17976.0,
  'WHO Region': 'EasternMediterranean'},
 {'Country/Region': 'Saudi Arabia',
  'TotalDeaths': 3055.0,
  'WHO Region': 'EasternMediterranean'},
 {'Country/Region': 'Pakistan',
  'TotalDeaths': 6035.0,
  'WHO Region': 'EasternMediterranean'},
 {'Country/Region': 'Bangladesh',
  'TotalDeaths': 3306.0,
  'WHO Region': 'South-EastAsia'},
 {'Country/Region': 'Turkey', 'TotalDeaths': 5798.0, 'WHO Region': 'Europe'},
 {'Country/Region': 'Iraq',
  'TotalDeaths': 5161.0,
  'WHO Region': 'EasternMediterranean'},
 {'Country/Region': 'Philippines',
  'TotalDeaths': 2150.0,
  'WHO Region': 'WesternPacific'},
 {'Country/Region': 'Indonesia',
  'TotalDeaths': 5521.0,
  'WHO Region': 'South-EastAsia'},
 {'Country/Region': 'Qatar',
```

```
 'TotalDeaths': 178.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Kazakhstan',
 'TotalDeaths': 1058.0,
 'WHO Region': 'Europe'},
{'Country/Region': 'Oman',
 'TotalDeaths': 492.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Israel', 'TotalDeaths': 576.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Kuwait',
 'TotalDeaths': 469.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'UAE',
 'TotalDeaths': 354.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Singapore',
 'TotalDeaths': 27.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Bahrain',
 'TotalDeaths': 156.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Japan',
 'TotalDeaths': 1026.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Armenia', 'TotalDeaths': 772.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Kyrgyzstan',
 'TotalDeaths': 1447.0,
 'WHO Region': 'Europe'},
{'Country/Region': 'Afghanistan',
 'TotalDeaths': 1298.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Azerbaijan', 'TotalDeaths': 479.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Uzbekistan', 'TotalDeaths': 175.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Nepal',
 'TotalDeaths': 65.0,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'S. Korea',
 'TotalDeaths': 303.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Palestine',
 'TotalDeaths': 92.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Malaysia',
 'TotalDeaths': 125.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Tajikistan', 'TotalDeaths': 62.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Lebanon',
 'TotalDeaths': 70.0,
```

```
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Maldives',
 'TotalDeaths': 19.0,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'Hong Kong',
 'TotalDeaths': 46.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Thailand',
 'TotalDeaths': 58.0,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'Sri Lanka',
 'TotalDeaths': 11.0,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'Yemen',
 'TotalDeaths': 508.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Jordan',
 'TotalDeaths': 11.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Cyprus', 'TotalDeaths': 19.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Georgia', 'TotalDeaths': 17.0, 'WHO Region': 'Europe'},
{'Country/Region': 'Syria',
 'TotalDeaths': 48.0,
 'WHO Region': 'EasternMediterranean'},
{'Country/Region': 'Vietnam',
 'TotalDeaths': 10.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Taiwan',
 'TotalDeaths': 7.0,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Myanmar',
 'TotalDeaths': 6.0,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'Mongolia',
 'TotalDeaths': nan,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Cambodia',
 'TotalDeaths': nan,
 'WHO Region': 'WesternPacific'},
{'Country/Region': 'Brunei ', 'TotalDeaths': 3.0, 'WHO Region': nan},
{'Country/Region': 'Bhutan',
 'TotalDeaths': nan,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'Macao', 'TotalDeaths': nan, 'WHO Region': nan},
{'Country/Region': 'Timor-Leste',
 'TotalDeaths': nan,
 'WHO Region': 'South-EastAsia'},
{'Country/Region': 'Laos', 'TotalDeaths': nan, 'WHO Region': 'WesternPacific'}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv("data/worldometer_data.csv")
df[df.Continent == "Asia"][["Country/Region", "TotalDeaths", "WHO Region"]]
```

[ ]:

	Country/Region	TotalDeaths	WHO Region
2	India	41638.0	South-EastAsia
10	Iran	17976.0	EasternMediterranean
12	Saudi Arabia	3055.0	EasternMediterranean
13	Pakistan	6035.0	EasternMediterranean
14	Bangladesh	3306.0	South-EastAsia
16	Turkey	5798.0	Europe
20	Iraq	5161.0	EasternMediterranean
21	Philippines	2150.0	WesternPacific
22	Indonesia	5521.0	South-EastAsia
24	Qatar	178.0	EasternMediterranean
25	Kazakhstan	1058.0	Europe
30	Oman	492.0	EasternMediterranean
31	Israel	576.0	Europe
36	Kuwait	469.0	EasternMediterranean
38	UAE	354.0	EasternMediterranean
41	Singapore	27.0	WesternPacific
47	Bahrain	156.0	EasternMediterranean
48	Japan	1026.0	WesternPacific
49	Armenia	772.0	Europe
51	Kyrgyzstan	1447.0	Europe
52	Afghanistan	1298.0	EasternMediterranean
55	Azerbaijan	479.0	Europe
57	Uzbekistan	175.0	Europe
63	Nepal	65.0	South-EastAsia
72	S. Korea	303.0	WesternPacific
74	Palestine	92.0	EasternMediterranean
83	Malaysia	125.0	WesternPacific
86	Tajikistan	62.0	Europe
95	Lebanon	70.0	EasternMediterranean
101	Maldives	19.0	South-EastAsia
107	Hong Kong	46.0	WesternPacific
110	Thailand	58.0	South-EastAsia
114	Sri Lanka	11.0	South-EastAsia
131	Yemen	508.0	EasternMediterranean
137	Jordan	11.0	EasternMediterranean
140	Cyprus	19.0	Europe
141	Georgia	17.0	Europe
145	Syria	48.0	EasternMediterranean
154	Vietnam	10.0	WesternPacific
162	Taiwan	7.0	WesternPacific
165	Myanmar	6.0	South-EastAsia
168	Mongolia	NaN	WesternPacific

```
174       Cambodia          NaN        WesternPacific
181         Brunei          3.0                   NaN
186         Bhutan          NaN        South-EastAsia
193          Macao          NaN                   NaN
197    Timor-Leste          NaN        South-EastAsia
200           Laos          NaN        WesternPacific
```

4.3 Q3

The number of deaths that occurred on 12-02-2020

[ ]:
```
qry = day_coll.find({"Date": "2020-02-12"}, projection={"Deaths": 1, "_id": 0})
pprint(list(qry))
```

```
[{'Deaths': 1118}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(day_data)
df[df.Date == "2020-02-12"][["Deaths"]]
```

[ ]:
```
    Deaths
21    1118
```

4.4 Q4

The number of active new cases (new cases-(new death+new recovered)) in a reverse sorted order based on the country name

> Here we create an additional field called `active new cases` which is defined as the difference between `new cases` and the sum of `new death` and `new recovered` cases. We then sort based on the country name in a alphabetically decreasing order and only project the country and these cases to get the final result.

[ ]:
```
qry = country_coll.aggregate(
    [
        {
            "$group": {
                "_id": "$Country/Region",
                "active new cases": {
                    "$sum": {
                        "$add": [
                            "$New cases",
                            {
                                "$multiply": [
                                    {"$add": ["$New deaths", "$New recovered"]},
                                    -1,
                                ]
                            },
```

```
                        ]
                    }
                },
            }
        },
        {"$sort": {"_id": -1}},
    ]
)
pprint(list(qry))
```

```
[{'_id': 'Zimbabwe', 'active new cases': 166},
 {'_id': 'Zambia', 'active new cases': -395},
 {'_id': 'Yemen', 'active new cases': -30},
 {'_id': 'Western Sahara', 'active new cases': 0},
 {'_id': 'West Bank and Gaza', 'active new cases': 150},
 {'_id': 'Vietnam', 'active new cases': 11},
 {'_id': 'Venezuela', 'active new cases': 308},
 {'_id': 'Uzbekistan', 'active new cases': 104},
 {'_id': 'Uruguay', 'active new cases': 6},
 {'_id': 'United Kingdom', 'active new cases': 678},
 {'_id': 'United Arab Emirates', 'active new cases': -65},
 {'_id': 'Ukraine', 'active new cases': 507},
 {'_id': 'Uganda', 'active new cases': 9},
 {'_id': 'US', 'active new cases': 27319},
 {'_id': 'Turkey', 'active new cases': -80},
 {'_id': 'Tunisia', 'active new cases': -12},
 {'_id': 'Trinidad and Tobago', 'active new cases': 1},
 {'_id': 'Togo', 'active new cases': -2},
 {'_id': 'Timor-Leste', 'active new cases': 0},
 {'_id': 'Thailand', 'active new cases': 4},
 {'_id': 'Tanzania', 'active new cases': 0},
 {'_id': 'Tajikistan', 'active new cases': -16},
 {'_id': 'Taiwan*', 'active new cases': 4},
 {'_id': 'Syria', 'active new cases': 22},
 {'_id': 'Switzerland', 'active new cases': -136},
 {'_id': 'Sweden', 'active new cases': 395},
 {'_id': 'Suriname', 'active new cases': 8},
 {'_id': 'Sudan', 'active new cases': -13},
 {'_id': 'Sri Lanka', 'active new cases': 8},
 {'_id': 'Spain', 'active new cases': 0},
 {'_id': 'South Sudan', 'active new cases': 42},
 {'_id': 'South Korea', 'active new cases': -75},
 {'_id': 'South Africa', 'active new cases': -3050},
 {'_id': 'Somalia', 'active new cases': -4},
 {'_id': 'Slovenia', 'active new cases': -50},
 {'_id': 'Slovakia', 'active new cases': -37},
 {'_id': 'Singapore', 'active new cases': 298},
```

{'_id': 'Sierra Leone', 'active new cases': -4},
{'_id': 'Seychelles', 'active new cases': 0},
{'_id': 'Serbia', 'active new cases': 402},
{'_id': 'Senegal', 'active new cases': 12},
{'_id': 'Saudi Arabia', 'active new cases': -647},
{'_id': 'Sao Tome and Principe', 'active new cases': -36},
{'_id': 'San Marino', 'active new cases': 0},
{'_id': 'Saint Vincent and the Grenadines', 'active new cases': 0},
{'_id': 'Saint Lucia', 'active new cases': 0},
{'_id': 'Saint Kitts and Nevis', 'active new cases': 0},
{'_id': 'Rwanda', 'active new cases': 1},
{'_id': 'Russia', 'active new cases': 2445},
{'_id': 'Romania', 'active new cases': 934},
{'_id': 'Qatar', 'active new cases': -12},
{'_id': 'Portugal', 'active new cases': -25},
{'_id': 'Poland', 'active new cases': 229},
{'_id': 'Philippines', 'active new cases': 1243},
{'_id': 'Peru', 'active new cases': 8484},
{'_id': 'Paraguay', 'active new cases': -9},
{'_id': 'Papua New Guinea', 'active new cases': 0},
{'_id': 'Panama', 'active new cases': 163},
{'_id': 'Pakistan', 'active new cases': -2436},
{'_id': 'Oman', 'active new cases': -685},
{'_id': 'Norway', 'active new cases': 15},
{'_id': 'North Macedonia', 'active new cases': -16},
{'_id': 'Nigeria', 'active new cases': -183},
{'_id': 'Niger', 'active new cases': 0},
{'_id': 'Nicaragua', 'active new cases': 0},
{'_id': 'New Zealand', 'active new cases': 0},
{'_id': 'Netherlands', 'active new cases': 418},
{'_id': 'Nepal', 'active new cases': -490},
{'_id': 'Namibia', 'active new cases': 42},
{'_id': 'Mozambique', 'active new cases': 32},
{'_id': 'Morocco', 'active new cases': 491},
{'_id': 'Montenegro', 'active new cases': 22},
{'_id': 'Mongolia', 'active new cases': -3},
{'_id': 'Monaco', 'active new cases': 0},
{'_id': 'Moldova', 'active new cases': -138},
{'_id': 'Mexico', 'active new cases': -3957},
{'_id': 'Mauritius', 'active new cases': 0},
{'_id': 'Mauritania', 'active new cases': -186},
{'_id': 'Malta', 'active new cases': 1},
{'_id': 'Mali', 'active new cases': 0},
{'_id': 'Maldives', 'active new cases': 48},
{'_id': 'Malaysia', 'active new cases': 6},
{'_id': 'Malawi', 'active new cases': 18},
{'_id': 'Madagascar', 'active new cases': -292},
{'_id': 'Luxembourg', 'active new cases': -129},

{'_id': 'Lithuania', 'active new cases': 7},
{'_id': 'Liechtenstein', 'active new cases': 0},
{'_id': 'Libya', 'active new cases': 130},
{'_id': 'Liberia', 'active new cases': 0},
{'_id': 'Lesotho', 'active new cases': 0},
{'_id': 'Lebanon', 'active new cases': 115},
{'_id': 'Latvia', 'active new cases': 0},
{'_id': 'Laos', 'active new cases': 0},
{'_id': 'Kyrgyzstan', 'active new cases': -358},
{'_id': 'Kuwait', 'active new cases': -83},
{'_id': 'Kosovo', 'active new cases': 206},
{'_id': 'Kenya', 'active new cases': 277},
{'_id': 'Kazakhstan', 'active new cases': -307},
{'_id': 'Jordan', 'active new cases': 8},
{'_id': 'Japan', 'active new cases': 230},
{'_id': 'Jamaica', 'active new cases': 11},
{'_id': 'Italy', 'active new cases': 16},
{'_id': 'Israel', 'active new cases': 1917},
{'_id': 'Ireland', 'active new cases': 11},
{'_id': 'Iraq', 'active new cases': 530},
{'_id': 'Iran', 'active new cases': 291},
{'_id': 'Indonesia', 'active new cases': -50},
{'_id': 'India', 'active new cases': 10222},
{'_id': 'Iceland', 'active new cases': 7},
{'_id': 'Hungary', 'active new cases': 13},
{'_id': 'Honduras', 'active new cases': 298},
{'_id': 'Holy See', 'active new cases': 0},
{'_id': 'Haiti', 'active new cases': 24},
{'_id': 'Guyana', 'active new cases': 19},
{'_id': 'Guinea-Bissau', 'active new cases': 0},
{'_id': 'Guinea', 'active new cases': -60},
{'_id': 'Guatemala', 'active new cases': -614},
{'_id': 'Grenada', 'active new cases': 0},
{'_id': 'Greenland', 'active new cases': 1},
{'_id': 'Greece', 'active new cases': 34},
{'_id': 'Ghana', 'active new cases': 348},
{'_id': 'Germany', 'active new cases': 185},
{'_id': 'Georgia', 'active new cases': 4},
{'_id': 'Gambia', 'active new cases': 41},
{'_id': 'Gabon', 'active new cases': -14},
{'_id': 'France', 'active new cases': 2267},
{'_id': 'Finland', 'active new cases': 5},
{'_id': 'Fiji', 'active new cases': 0},
{'_id': 'Ethiopia', 'active new cases': 404},
{'_id': 'Eswatini', 'active new cases': 68},
{'_id': 'Estonia', 'active new cases': -1},
{'_id': 'Eritrea', 'active new cases': 0},
{'_id': 'Equatorial Guinea', 'active new cases': 0},

{'_id': 'El Salvador', 'active new cases': 267},
{'_id': 'Egypt', 'active new cases': -633},
{'_id': 'Ecuador', 'active new cases': 450},
{'_id': 'Dominican Republic', 'active new cases': -373},
{'_id': 'Dominica', 'active new cases': 0},
{'_id': 'Djibouti', 'active new cases': -2},
{'_id': 'Denmark', 'active new cases': 32},
{'_id': 'Czechia', 'active new cases': 190},
{'_id': 'Cyprus', 'active new cases': 3},
{'_id': 'Cuba', 'active new cases': 35},
{'_id': 'Croatia', 'active new cases': -49},
{'_id': "Cote d'Ivoire", 'active new cases': -124},
{'_id': 'Costa Rica', 'active new cases': 513},
{'_id': 'Congo (Kinshasa)', 'active new cases': -181},
{'_id': 'Congo (Brazzaville)', 'active new cases': 86},
{'_id': 'Comoros', 'active new cases': 0},
{'_id': 'Colombia', 'active new cases': 4304},
{'_id': 'China', 'active new cases': 202},
{'_id': 'Chile', 'active new cases': 199},
{'_id': 'Chad', 'active new cases': 7},
{'_id': 'Central African Republic', 'active new cases': 0},
{'_id': 'Canada', 'active new cases': 671},
{'_id': 'Cameroon', 'active new cases': 396},
{'_id': 'Cambodia', 'active new cases': -3},
{'_id': 'Cabo Verde', 'active new cases': -82},
{'_id': 'Burundi', 'active new cases': -5},
{'_id': 'Burma', 'active new cases': -2},
{'_id': 'Burkina Faso', 'active new cases': 8},
{'_id': 'Bulgaria', 'active new cases': -43},
{'_id': 'Brunei', 'active new cases': 0},
{'_id': 'Brazil', 'active new cases': -11058},
{'_id': 'Botswana', 'active new cases': 41},
{'_id': 'Bosnia and Herzegovina', 'active new cases': 342},
{'_id': 'Bolivia', 'active new cases': 1379},
{'_id': 'Bhutan', 'active new cases': 3},
{'_id': 'Benin', 'active new cases': 0},
{'_id': 'Belize', 'active new cases': 0},
{'_id': 'Belgium', 'active new cases': 387},
{'_id': 'Belarus', 'active new cases': 48},
{'_id': 'Barbados', 'active new cases': 0},
{'_id': 'Bangladesh', 'active new cases': 934},
{'_id': 'Bahrain', 'active new cases': -71},
{'_id': 'Bahamas', 'active new cases': 40},
{'_id': 'Azerbaijan', 'active new cases': -168},
{'_id': 'Austria', 'active new cases': 48},
{'_id': 'Australia', 'active new cases': 225},
{'_id': 'Armenia', 'active new cases': -120},
{'_id': 'Argentina', 'active new cases': 2713},

```
{'_id': 'Antigua and Barbuda', 'active new cases': -1},
{'_id': 'Angola', 'active new cases': 17},
{'_id': 'Andorra', 'active new cases': 10},
{'_id': 'Algeria', 'active new cases': -141},
{'_id': 'Albania', 'active new cases': 48},
{'_id': 'Afghanistan', 'active new cases': 78}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(country_data)
df["Active New"] = df["New cases"] - (df["New deaths"] + df["New recovered"])
df.sort_values(by=["Country/Region"], ascending=False)[["Country/Region",␣
↪"Active New"]]
```

[ ]:
```
          Country/Region  Active New
186             Zimbabwe         166
185               Zambia        -395
184                Yemen         -30
183      Western Sahara           0
182  West Bank and Gaza         150
..                   …         …
4                 Angola          17
3                Andorra          10
2                Algeria        -141
1                Albania          48
0            Afghanistan          78

[187 rows x 2 columns]
```

4.5 Q5

The names of the countries with more than 9000 active cases and more than 800 deaths

[ ]:
```
qry = country_coll.aggregate(
    [
        {"$match": {"$and": [{"Active": {"$gt": 9000}}, {"Deaths": {"$gt":␣
↪800}}]}},
        {"$project": {"Country/Region": -1, "_id": 0}},
    ]
)

pprint(list(qry))
```

```
[{'Country/Region': 'Afghanistan'},
 {'Country/Region': 'Argentina'},
 {'Country/Region': 'Bangladesh'},
 {'Country/Region': 'Belgium'},
 {'Country/Region': 'Bolivia'},
 {'Country/Region': 'Brazil'},
```

```
{'Country/Region': 'Canada'},
{'Country/Region': 'Chile'},
{'Country/Region': 'Colombia'},
{'Country/Region': 'Dominican Republic'},
{'Country/Region': 'Ecuador'},
{'Country/Region': 'Egypt'},
{'Country/Region': 'France'},
{'Country/Region': 'Guatemala'},
{'Country/Region': 'Honduras'},
{'Country/Region': 'India'},
{'Country/Region': 'Indonesia'},
{'Country/Region': 'Iran'},
{'Country/Region': 'Iraq'},
{'Country/Region': 'Italy'},
{'Country/Region': 'Kyrgyzstan'},
{'Country/Region': 'Mexico'},
{'Country/Region': 'Netherlands'},
{'Country/Region': 'Nigeria'},
{'Country/Region': 'Pakistan'},
{'Country/Region': 'Panama'},
{'Country/Region': 'Peru'},
{'Country/Region': 'Philippines'},
{'Country/Region': 'Portugal'},
{'Country/Region': 'Romania'},
{'Country/Region': 'Russia'},
{'Country/Region': 'Saudi Arabia'},
{'Country/Region': 'South Africa'},
{'Country/Region': 'Spain'},
{'Country/Region': 'Sweden'},
{'Country/Region': 'Turkey'},
{'Country/Region': 'US'},
{'Country/Region': 'Ukraine'},
{'Country/Region': 'United Kingdom'}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(country_data)
df = df[(df["Active"] > 9000) & (df["Deaths"] > 800)]
df[["Country/Region"]]
```

[ ]:
```
      Country/Region
0        Afghanistan
6          Argentina
13        Bangladesh
16           Belgium
20           Bolivia
23            Brazil
32            Canada
```

```
35                Chile
37             Colombia
50   Dominican Republic
51              Ecuador
52                Egypt
61               France
70            Guatemala
76             Honduras
79                India
80            Indonesia
81                 Iran
82                 Iraq
85                Italy
93            Kyrgyzstan
111               Mexico
120          Netherlands
124              Nigeria
128             Pakistan
129               Panama
132                 Peru
133          Philippines
135             Portugal
137              Romania
138               Russia
145         Saudi Arabia
154         South Africa
157                Spain
161               Sweden
172               Turkey
173                   US
175              Ukraine
177       United Kingdom
```

4.6 Q6

The country with the highest number of active cases and also with second highest death rate.

[ ]:
```
qry = country_coll.aggregate(
    [
        {
            "$sort": {"Active": -1},
        },
        {"$limit": 1},
        {"$project": {"Country/Region": 1, "Active": 1, "_id": 0}},
    ]
)
```

```
pprint(list(qry))
```

```
[{'Active': 2816444, 'Country/Region': 'US'}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(country_data)
df.sort_values(by=["Active"]).tail(1)
```

[ ]:
```
    Country/Region  Confirmed  Deaths  Recovered   Active  New cases  \
173             US    4290259  148011    1325804  2816444      56336

     New deaths  New recovered  Deaths / 100 Cases  Recovered / 100 Cases  \
173        1076          27941                3.45                   30.9

     Deaths / 100 Recovered  Confirmed last week  1 week change  \
173                   11.16              3834677         455582

     1 week % increase WHO Region
173              11.88   Americas
```

[ ]:
```
qry = country_coll.aggregate(
    [
        {
            "$sort": {"Deaths / 100 Cases": -1},
        },
        {"$limit": 2},
        {
            "$sort": {"Deaths / 100 Cases": 1},
        },
        {"$limit": 1},
        {
            "$project": {
                "Country/Region": 1,
                "Active": 1,
                "Deaths / 100 Cases": 1,
                "_id": 0,
            }
        },
    ]
)

pprint(list(qry)[-1])
```

```
{'Active': 254427,
 'Country/Region': 'United Kingdom',
 'Deaths / 100 Cases': 15.19}
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df.sort_values(by=["Deaths / 100 Cases"], ascending=False).head(2).tail(1)
```

[ ]:
```
     Country/Region  Confirmed  Deaths  Recovered  Active  New cases  \
177  United Kingdom     301708   45844       1437  254427        688

     New deaths  New recovered  Deaths / 100 Cases  Recovered / 100 Cases  \
177           7              3               15.19                   0.48

     Deaths / 100 Recovered  Confirmed last week  1 week change  \
177                 3190.26               296944           4764

     1 week % increase WHO Region
177                1.6     Europe
```

4.7 Q7

The total number of deaths all around the world

[ ]:
```
qry = country_coll.aggregate(
    [{"$group": {"_id": "Worldwide Deaths", "total": {"$sum": "$Deaths"}}}]
)
pprint(list(qry))
```

```
[{'_id': 'Worldwide Deaths', 'total': 654036}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(country_data)
df.Deaths.sum()
```

[ ]:
```
654036
```

4.8 Q8

The number of death cases and active cases between 28-01-2020 and 21-02-2020

[ ]:
```
start = "2020-01-28"
end = "2020-02-21"

qry = day_coll.aggregate(
    [
        {"$match": {"$and": [{"Date": {"$gte": start}}, {"Date": {"$lte":␣
 ↪end}}]}},
        {"$project": {"Deaths": 1, "Active": 1, "_id": 0}},
        {
            "$group": {
                "_id": "Case counts",
                "Total Active": {"$sum": "$Active"},
```

```
                "Total Dead": {"$sum": "$Deaths"},
            }
        },
    ]
)
pprint(list(qry))
```

```
[{'Total Active': 886425, 'Total Dead': 25885, '_id': 'Case counts'}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(day_data)
df[(df.Date >= start) & (df.Date <= end)][["Active", "Deaths"]].sum()
```

[ ]:
```
Active    886425
Deaths     25885
dtype: int64
```

4.9 Q9

The latitude and longitude of countries ending with "ia" and the number of countries

The data provided here consists of multiple countries and multiple provinces/states within a country. In order to get an exact latitude/longitude on a country level is not possible. We could tackle this in several ways

- Provide lat and long for the capital of that country
- Provide lat and long for the region with most cases in that country
- More simply, provide the average lat and average long of all the provinces/states in the country so we get an approximate coordinate for center of mass of the country.

We have chosen approach 3 to address the above query as showcased below

[ ]:
```
qry = covid_coll.aggregate(
    [
        {"$match": {"Country/Region": {"$regex": ".ia$"}}},
        {
            "$project": {
                "Lat": {"$ifNull": ["$Lat", 0]},
                "Long": {"$ifNull": ["$Long", 0]},
                "Country/Region": 1,
            }
        },
        {
            "$group": {
                "_id": "$Country/Region",
                "lat": {"$avg": "$Lat"},
                "long": {"$avg": "$Long"},
            }
        },
```

```
        {"$sort": {"_id": 1}},
    ]
)
pprint(list(qry))
```

```
[{'_id': 'Albania', 'lat': 41.1533, 'long': 20.1683},
 {'_id': 'Algeria', 'lat': 28.0339, 'long': 1.6596},
 {'_id': 'Armenia', 'lat': 40.0691, 'long': 45.038199999999996},
 {'_id': 'Australia', 'lat': -32.106275, 'long': 141.3554875},
 {'_id': 'Austria', 'lat': 47.5162, 'long': 14.550099999999999},
 {'_id': 'Bolivia', 'lat': -16.2902, 'long': -63.5887},
 {'_id': 'Bulgaria', 'lat': 42.7339, 'long': 25.485799999999998},
 {'_id': 'Cambodia', 'lat': 11.55, 'long': 104.91669999999999},
 {'_id': 'Colombia', 'lat': 4.5709, 'long': -74.2973},
 {'_id': 'Croatia', 'lat': 45.10000000000001, 'long': 15.2},
 {'_id': 'Czechia', 'lat': 49.8175, 'long': 15.473},
 {'_id': 'Estonia', 'lat': 58.59530000000001, 'long': 25.0136},
 {'_id': 'Ethiopia', 'lat': 9.145, 'long': 40.4897},
 {'_id': 'Gambia', 'lat': 13.4432, 'long': -15.3101},
 {'_id': 'Georgia', 'lat': 42.3154, 'long': 43.3569},
 {'_id': 'India', 'lat': 20.593684, 'long': 78.96288},
 {'_id': 'Indonesia', 'lat': -0.7893, 'long': 113.92129999999999},
 {'_id': 'Latvia', 'lat': 56.8796, 'long': 24.6032},
 {'_id': 'Liberia', 'lat': 6.428055000000001, 'long': -9.429499},
 {'_id': 'Lithuania', 'lat': 55.1694, 'long': 23.8813},
 {'_id': 'Malaysia', 'lat': 4.210483999999999, 'long': 101.975766},
 {'_id': 'Mauritania', 'lat': 21.0079, 'long': -10.9408},
 {'_id': 'Mongolia', 'lat': 46.8625, 'long': 103.8467},
 {'_id': 'Namibia', 'lat': -22.9576, 'long': 18.4904},
 {'_id': 'Nigeria', 'lat': 9.082, 'long': 8.6753},
 {'_id': 'North Macedonia', 'lat': 41.6086, 'long': 21.7453},
 {'_id': 'Romania', 'lat': 45.9432, 'long': 24.9668},
 {'_id': 'Russia', 'lat': 61.52400999999999, 'long': 105.318756},
 {'_id': 'Saint Lucia', 'lat': 13.9094, 'long': -60.978899999999996},
 {'_id': 'Saudi Arabia', 'lat': 23.885942000000004, 'long': 45.079162},
 {'_id': 'Serbia', 'lat': 44.0165, 'long': 21.0059},
 {'_id': 'Slovakia', 'lat': 48.669, 'long': 19.699},
 {'_id': 'Slovenia', 'lat': 46.1512, 'long': 14.9955},
 {'_id': 'Somalia', 'lat': 5.152149, 'long': 46.19961599999999},
 {'_id': 'Syria', 'lat': 34.802075, 'long': 38.99681500000001},
 {'_id': 'Tanzania', 'lat': -6.369028, 'long': 34.888822},
 {'_id': 'Tunisia', 'lat': 33.886917, 'long': 9.537499},
 {'_id': 'Zambia', 'lat': -13.133897, 'long': 27.849332}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(covid_data)
df = df[df["Country/Region"].apply(lambda x: x.endswith("ia"))]
```

```
len(df["Country/Region"].unique())
```

[ ]: 38

```
df.groupby(by=["Country/Region"])[["Lat", "Long"]].mean()
```

[ ]:

Country/Region	Lat	Long
Albania	41.153300	20.168300
Algeria	28.033900	1.659600
Armenia	40.069100	45.038200
Australia	-32.106275	141.355488
Austria	47.516200	14.550100
Bolivia	-16.290200	-63.588700
Bulgaria	42.733900	25.485800
Cambodia	11.550000	104.916700
Colombia	4.570900	-74.297300
Croatia	45.100000	15.200000
Czechia	49.817500	15.473000
Estonia	58.595300	25.013600
Ethiopia	9.145000	40.489700
Gambia	13.443200	-15.310100
Georgia	42.315400	43.356900
India	20.593684	78.962880
Indonesia	-0.789300	113.921300
Latvia	56.879600	24.603200
Liberia	6.428055	-9.429499
Lithuania	55.169400	23.881300
Malaysia	4.210484	101.975766
Mauritania	21.007900	-10.940800
Mongolia	46.862500	103.846700
Namibia	-22.957600	18.490400
Nigeria	9.082000	8.675300
North Macedonia	41.608600	21.745300
Romania	45.943200	24.966800
Russia	61.524010	105.318756
Saint Lucia	13.909400	-60.978900
Saudi Arabia	23.885942	45.079162
Serbia	44.016500	21.005900
Slovakia	48.669000	19.699000
Slovenia	46.151200	14.995500
Somalia	5.152149	46.199616
Syria	34.802075	38.996815
Tanzania	-6.369028	34.888822
Tunisia	33.886917	9.537499
Zambia	-13.133897	27.849332

4.10 Q10

The countries with active cases on 30/03/2020

[ ]:
```
qry = full_coll.find(
    {"Date": "2020-03-30", "Active": {"$gt": 0}},
    projection={"Country/Region": 1, "_id": 0},
)
pprint(list(qry))
```

```
[{'Country/Region': 'Afghanistan'},
 {'Country/Region': 'Albania'},
 {'Country/Region': 'Algeria'},
 {'Country/Region': 'Andorra'},
 {'Country/Region': 'Angola'},
 {'Country/Region': 'Antigua and Barbuda'},
 {'Country/Region': 'Argentina'},
 {'Country/Region': 'Armenia'},
 {'Country/Region': 'Australia'},
 {'Country/Region': 'Austria'},
 {'Country/Region': 'Azerbaijan'},
 {'Country/Region': 'Bahamas'},
 {'Country/Region': 'Bahrain'},
 {'Country/Region': 'Bangladesh'},
 {'Country/Region': 'Barbados'},
 {'Country/Region': 'Belarus'},
 {'Country/Region': 'Belgium'},
 {'Country/Region': 'Belize'},
 {'Country/Region': 'Benin'},
 {'Country/Region': 'Bhutan'},
 {'Country/Region': 'Bolivia'},
 {'Country/Region': 'Bosnia and Herzegovina'},
 {'Country/Region': 'Botswana'},
 {'Country/Region': 'Brazil'},
 {'Country/Region': 'Brunei'},
 {'Country/Region': 'Bulgaria'},
 {'Country/Region': 'Burkina Faso'},
 {'Country/Region': 'Burma'},
 {'Country/Region': 'Cabo Verde'},
 {'Country/Region': 'Cambodia'},
 {'Country/Region': 'Cameroon'},
 {'Country/Region': 'Canada'},
 {'Country/Region': 'Central African Republic'},
 {'Country/Region': 'Chad'},
 {'Country/Region': 'Chile'},
 {'Country/Region': 'China'},
 {'Country/Region': 'Colombia'},
 {'Country/Region': 'Congo (Brazzaville)'},
```

{'Country/Region': 'Congo (Kinshasa)'},
{'Country/Region': 'Costa Rica'},
{'Country/Region': "Cote d'Ivoire"},
{'Country/Region': 'Croatia'},
{'Country/Region': 'Cuba'},
{'Country/Region': 'Cyprus'},
{'Country/Region': 'Czechia'},
{'Country/Region': 'Denmark'},
{'Country/Region': 'Djibouti'},
{'Country/Region': 'Dominica'},
{'Country/Region': 'Dominican Republic'},
{'Country/Region': 'Ecuador'},
{'Country/Region': 'Egypt'},
{'Country/Region': 'El Salvador'},
{'Country/Region': 'Equatorial Guinea'},
{'Country/Region': 'Eritrea'},
{'Country/Region': 'Estonia'},
{'Country/Region': 'Eswatini'},
{'Country/Region': 'Ethiopia'},
{'Country/Region': 'Fiji'},
{'Country/Region': 'Finland'},
{'Country/Region': 'France'},
{'Country/Region': 'Gabon'},
{'Country/Region': 'Gambia'},
{'Country/Region': 'Georgia'},
{'Country/Region': 'Germany'},
{'Country/Region': 'Ghana'},
{'Country/Region': 'Greece'},
{'Country/Region': 'Greenland'},
{'Country/Region': 'Grenada'},
{'Country/Region': 'Guatemala'},
{'Country/Region': 'Guinea'},
{'Country/Region': 'Guinea-Bissau'},
{'Country/Region': 'Guyana'},
{'Country/Region': 'Haiti'},
{'Country/Region': 'Holy See'},
{'Country/Region': 'Honduras'},
{'Country/Region': 'Hungary'},
{'Country/Region': 'Iceland'},
{'Country/Region': 'India'},
{'Country/Region': 'Indonesia'},
{'Country/Region': 'Iran'},
{'Country/Region': 'Iraq'},
{'Country/Region': 'Ireland'},
{'Country/Region': 'Israel'},
{'Country/Region': 'Italy'},
{'Country/Region': 'Jamaica'},
{'Country/Region': 'Japan'},

{'Country/Region': 'Jordan'},
{'Country/Region': 'Kazakhstan'},
{'Country/Region': 'Kenya'},
{'Country/Region': 'Kosovo'},
{'Country/Region': 'Kuwait'},
{'Country/Region': 'Kyrgyzstan'},
{'Country/Region': 'Laos'},
{'Country/Region': 'Latvia'},
{'Country/Region': 'Lebanon'},
{'Country/Region': 'Liberia'},
{'Country/Region': 'Libya'},
{'Country/Region': 'Liechtenstein'},
{'Country/Region': 'Lithuania'},
{'Country/Region': 'Luxembourg'},
{'Country/Region': 'Madagascar'},
{'Country/Region': 'Malaysia'},
{'Country/Region': 'Maldives'},
{'Country/Region': 'Mali'},
{'Country/Region': 'Malta'},
{'Country/Region': 'Mauritania'},
{'Country/Region': 'Mauritius'},
{'Country/Region': 'Mexico'},
{'Country/Region': 'Moldova'},
{'Country/Region': 'Monaco'},
{'Country/Region': 'Mongolia'},
{'Country/Region': 'Montenegro'},
{'Country/Region': 'Morocco'},
{'Country/Region': 'Mozambique'},
{'Country/Region': 'Namibia'},
{'Country/Region': 'Nepal'},
{'Country/Region': 'Netherlands'},
{'Country/Region': 'New Zealand'},
{'Country/Region': 'Nicaragua'},
{'Country/Region': 'Niger'},
{'Country/Region': 'Nigeria'},
{'Country/Region': 'North Macedonia'},
{'Country/Region': 'Norway'},
{'Country/Region': 'Oman'},
{'Country/Region': 'Pakistan'},
{'Country/Region': 'Panama'},
{'Country/Region': 'Papua New Guinea'},
{'Country/Region': 'Paraguay'},
{'Country/Region': 'Peru'},
{'Country/Region': 'Philippines'},
{'Country/Region': 'Poland'},
{'Country/Region': 'Portugal'},
{'Country/Region': 'Qatar'},
{'Country/Region': 'Romania'},

```
 {'Country/Region': 'Russia'},
 {'Country/Region': 'Rwanda'},
 {'Country/Region': 'Saint Kitts and Nevis'},
 {'Country/Region': 'Saint Lucia'},
 {'Country/Region': 'San Marino'},
 {'Country/Region': 'Saudi Arabia'},
 {'Country/Region': 'Senegal'},
 {'Country/Region': 'Serbia'},
 {'Country/Region': 'Seychelles'},
 {'Country/Region': 'Singapore'},
 {'Country/Region': 'Slovakia'},
 {'Country/Region': 'Slovenia'},
 {'Country/Region': 'Somalia'},
 {'Country/Region': 'South Africa'},
 {'Country/Region': 'South Korea'},
 {'Country/Region': 'Spain'},
 {'Country/Region': 'Sri Lanka'},
 {'Country/Region': 'Sudan'},
 {'Country/Region': 'Suriname'},
 {'Country/Region': 'Sweden'},
 {'Country/Region': 'Switzerland'},
 {'Country/Region': 'Syria'},
 {'Country/Region': 'Taiwan*'},
 {'Country/Region': 'Tanzania'},
 {'Country/Region': 'Thailand'},
 {'Country/Region': 'Timor-Leste'},
 {'Country/Region': 'Togo'},
 {'Country/Region': 'Trinidad and Tobago'},
 {'Country/Region': 'Tunisia'},
 {'Country/Region': 'Turkey'},
 {'Country/Region': 'US'},
 {'Country/Region': 'Uganda'},
 {'Country/Region': 'Ukraine'},
 {'Country/Region': 'United Arab Emirates'},
 {'Country/Region': 'United Kingdom'},
 {'Country/Region': 'Uruguay'},
 {'Country/Region': 'Uzbekistan'},
 {'Country/Region': 'Venezuela'},
 {'Country/Region': 'Vietnam'},
 {'Country/Region': 'West Bank and Gaza'},
 {'Country/Region': 'Zambia'},
 {'Country/Region': 'Zimbabwe'}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(full_data)
pprint(df[(df.Date == "2020-03-30") & (df.Active > 0)]["Country/Region"].
 ↪tolist())
```

```
['Afghanistan',
 'Albania',
 'Algeria',
 'Andorra',
 'Angola',
 'Antigua and Barbuda',
 'Argentina',
 'Armenia',
 'Australia',
 'Austria',
 'Azerbaijan',
 'Bahamas',
 'Bahrain',
 'Bangladesh',
 'Barbados',
 'Belarus',
 'Belgium',
 'Belize',
 'Benin',
 'Bhutan',
 'Bolivia',
 'Bosnia and Herzegovina',
 'Botswana',
 'Brazil',
 'Brunei',
 'Bulgaria',
 'Burkina Faso',
 'Burma',
 'Cabo Verde',
 'Cambodia',
 'Cameroon',
 'Canada',
 'Central African Republic',
 'Chad',
 'Chile',
 'China',
 'Colombia',
 'Congo (Brazzaville)',
 'Congo (Kinshasa)',
 'Costa Rica',
 "Cote d'Ivoire",
 'Croatia',
 'Cuba',
 'Cyprus',
 'Czechia',
 'Denmark',
 'Djibouti',
 'Dominica',
```

```
'Dominican Republic',
'Ecuador',
'Egypt',
'El Salvador',
'Equatorial Guinea',
'Eritrea',
'Estonia',
'Eswatini',
'Ethiopia',
'Fiji',
'Finland',
'France',
'Gabon',
'Gambia',
'Georgia',
'Germany',
'Ghana',
'Greece',
'Greenland',
'Grenada',
'Guatemala',
'Guinea',
'Guinea-Bissau',
'Guyana',
'Haiti',
'Holy See',
'Honduras',
'Hungary',
'Iceland',
'India',
'Indonesia',
'Iran',
'Iraq',
'Ireland',
'Israel',
'Italy',
'Jamaica',
'Japan',
'Jordan',
'Kazakhstan',
'Kenya',
'Kosovo',
'Kuwait',
'Kyrgyzstan',
'Laos',
'Latvia',
'Lebanon',
'Liberia',
```

```
'Libya',
'Liechtenstein',
'Lithuania',
'Luxembourg',
'Madagascar',
'Malaysia',
'Maldives',
'Mali',
'Malta',
'Mauritania',
'Mauritius',
'Mexico',
'Moldova',
'Monaco',
'Mongolia',
'Montenegro',
'Morocco',
'Mozambique',
'Namibia',
'Nepal',
'Netherlands',
'New Zealand',
'Nicaragua',
'Niger',
'Nigeria',
'North Macedonia',
'Norway',
'Oman',
'Pakistan',
'Panama',
'Papua New Guinea',
'Paraguay',
'Peru',
'Philippines',
'Poland',
'Portugal',
'Qatar',
'Romania',
'Russia',
'Rwanda',
'Saint Kitts and Nevis',
'Saint Lucia',
'San Marino',
'Saudi Arabia',
'Senegal',
'Serbia',
'Seychelles',
'Singapore',
```

```
'Slovakia',
'Slovenia',
'Somalia',
'South Africa',
'South Korea',
'Spain',
'Sri Lanka',
'Sudan',
'Suriname',
'Sweden',
'Switzerland',
'Syria',
'Taiwan*',
'Tanzania',
'Thailand',
'Timor-Leste',
'Togo',
'Trinidad and Tobago',
'Tunisia',
'Turkey',
'US',
'Uganda',
'Ukraine',
'United Arab Emirates',
'United Kingdom',
'Uruguay',
'Uzbekistan',
'Venezuela',
'Vietnam',
'West Bank and Gaza',
'Zambia',
'Zimbabwe']
```

4.11 Q11

The latitude and longitude of those countries which are having active cases greater than 100

Similary to Q9, we have chosen the approach of providing average lat and long in our case.

[ ]:
```
qry = covid_coll.aggregate(
    [
        {"$match": {"Active": {"$gt": 100}}},
        {
            "$project": {
                "Lat": {"$ifNull": ["$Lat", 0]},
                "Long": {"$ifNull": ["$Long", 0]},
                "Country/Region": 1,
```

```
            }
        },
        {
            "$group": {
                "_id": "$Country/Region",
                "lat": {"$avg": "$Lat"},
                "long": {"$avg": "$Long"},
            }
        },
        {"$sort": {"_id": 1}},
    ]
)
pprint(list(qry))
```

```
[{'_id': 'Afghanistan', 'lat': 33.93911, 'long': 67.709953},
 {'_id': 'Albania', 'lat': 41.1533, 'long': 20.1683},
 {'_id': 'Algeria', 'lat': 28.0339, 'long': 1.6596},
 {'_id': 'Andorra', 'lat': 42.5063, 'long': 1.5218},
 {'_id': 'Angola', 'lat': -11.2027, 'long': 17.8739},
 {'_id': 'Argentina', 'lat': -38.4161, 'long': -63.6167},
 {'_id': 'Armenia', 'lat': 40.0691, 'long': 45.0382},
 {'_id': 'Australia', 'lat': -34.320312538226304, 'long': 145.4311125382263},
 {'_id': 'Austria', 'lat': 47.5162, 'long': 14.5501},
 {'_id': 'Azerbaijan', 'lat': 40.1431, 'long': 47.5769},
 {'_id': 'Bahamas', 'lat': 25.025885, 'long': -78.035889},
 {'_id': 'Bahrain', 'lat': 26.0275, 'long': 50.55},
 {'_id': 'Bangladesh', 'lat': 23.685, 'long': 90.3563},
 {'_id': 'Belarus', 'lat': 53.7098, 'long': 27.9534},
 {'_id': 'Belgium', 'lat': 50.8333, 'long': 4.469936},
 {'_id': 'Benin', 'lat': 9.3077, 'long': 2.3158},
 {'_id': 'Bolivia', 'lat': -16.2902, 'long': -63.5887},
 {'_id': 'Bosnia and Herzegovina', 'lat': 43.9159, 'long': 17.6791},
 {'_id': 'Botswana', 'lat': -22.3285, 'long': 24.6849},
 {'_id': 'Brazil', 'lat': -14.235, 'long': -51.9253},
 {'_id': 'Brunei', 'lat': 4.5353, 'long': 114.7277},
 {'_id': 'Bulgaria', 'lat': 42.7339, 'long': 25.4858},
 {'_id': 'Burkina Faso', 'lat': 12.2383, 'long': -1.5616},
 {'_id': 'Burma', 'lat': 21.9162, 'long': 95.956},
 {'_id': 'Burundi', 'lat': -3.3730999999999995, 'long': 29.918899999999997},
 {'_id': 'Cabo Verde', 'lat': 16.5388, 'long': -23.041800000000002},
 {'_id': 'Cameroon', 'lat': 3.848, 'long': 11.5021},
 {'_id': 'Canada', 'lat': 51.521756117021276, 'long': -89.00679184397163},
 {'_id': 'Central African Republic', 'lat': 6.6111, 'long': 20.9394},
 {'_id': 'Chad', 'lat': 15.4542, 'long': 18.7322},
 {'_id': 'Chile', 'lat': -35.6751, 'long': -71.543},
 {'_id': 'China', 'lat': 33.631940350877194, 'long': 115.07268755221388},
 {'_id': 'Colombia', 'lat': 4.5709, 'long': -74.2973},
```

{'_id': 'Comoros', 'lat': -11.6455, 'long': 43.3333},
{'_id': 'Congo (Brazzaville)', 'lat': -0.228, 'long': 15.8277},
{'_id': 'Congo (Kinshasa)', 'lat': -4.0383, 'long': 21.7587},
{'_id': 'Costa Rica', 'lat': 9.7489, 'long': -83.7534},
{'_id': "Cote d'Ivoire", 'lat': 7.54, 'long': -5.5471},
{'_id': 'Croatia', 'lat': 45.1, 'long': 15.2},
{'_id': 'Cuba', 'lat': 21.521757, 'long': -77.78116700000002},
{'_id': 'Cyprus', 'lat': 35.1264, 'long': 33.4299},
{'_id': 'Czechia', 'lat': 49.8175, 'long': 15.472999999999999},
{'_id': 'Denmark', 'lat': 56.38198461538462, 'long': 9.157458741258742},
{'_id': 'Djibouti', 'lat': 11.8251, 'long': 42.5903},
{'_id': 'Dominican Republic', 'lat': 18.7357, 'long': -70.1627},
{'_id': 'Ecuador', 'lat': -1.8312, 'long': -78.1834},
{'_id': 'Egypt', 'lat': 26.820553000000004, 'long': 30.802498},
{'_id': 'El Salvador', 'lat': 13.794199999999998, 'long': -88.8965},
{'_id': 'Equatorial Guinea', 'lat': 1.6507999999999998, 'long': 10.2679},
{'_id': 'Eritrea', 'lat': 15.1794, 'long': 39.7823},
{'_id': 'Estonia', 'lat': 58.5953, 'long': 25.0136},
{'_id': 'Eswatini', 'lat': -26.5225, 'long': 31.4659},
{'_id': 'Ethiopia', 'lat': 9.145, 'long': 40.4897},
{'_id': 'Finland', 'lat': 61.92411, 'long': 25.748151},
{'_id': 'France', 'lat': 11.97344885321101, 'long': 4.7488253394495406},
{'_id': 'Gabon', 'lat': -0.8036999999999999, 'long': 11.6094},
{'_id': 'Gambia', 'lat': 13.4432, 'long': -15.3101},
{'_id': 'Georgia', 'lat': 42.3154, 'long': 43.3569},
{'_id': 'Germany', 'lat': 51.165691, 'long': 10.451526},
{'_id': 'Ghana', 'lat': 7.9464999999999995, 'long': -1.0232},
{'_id': 'Greece', 'lat': 39.0742, 'long': 21.8243},
{'_id': 'Guatemala', 'lat': 15.7835, 'long': -90.2308},
{'_id': 'Guinea', 'lat': 9.9456, 'long': -9.6966},
{'_id': 'Guinea-Bissau', 'lat': 11.8037, 'long': -15.1804},
{'_id': 'Guyana', 'lat': 4.860416000000002, 'long': -58.93018000000001},
{'_id': 'Haiti', 'lat': 18.9712, 'long': -72.2852},
{'_id': 'Honduras', 'lat': 15.2, 'long': -86.2419},
{'_id': 'Hungary', 'lat': 47.1625, 'long': 19.5033},
{'_id': 'Iceland', 'lat': 64.9631, 'long': -19.0208},
{'_id': 'India', 'lat': 20.593684, 'long': 78.96288},
{'_id': 'Indonesia', 'lat': -0.7893, 'long': 113.9213},
{'_id': 'Iran', 'lat': 32.427908, 'long': 53.68804599999999},
{'_id': 'Iraq', 'lat': 33.223191, 'long': 43.679291},
{'_id': 'Ireland', 'lat': 53.1424, 'long': -7.692100000000001},
{'_id': 'Israel', 'lat': 31.046050999999995, 'long': 34.851612},
{'_id': 'Italy', 'lat': 41.87194, 'long': 12.56738},
{'_id': 'Jamaica', 'lat': 18.1096, 'long': -77.2975},
{'_id': 'Japan', 'lat': 36.204824, 'long': 138.252924},
{'_id': 'Jordan', 'lat': 31.24, 'long': 36.51},
{'_id': 'Kazakhstan', 'lat': 48.0196, 'long': 66.9237},
{'_id': 'Kenya', 'lat': -0.0236, 'long': 37.9062},

{'_id': 'Kosovo', 'lat': 42.602636, 'long': 20.902977},
{'_id': 'Kuwait', 'lat': 29.31166, 'long': 47.481766},
{'_id': 'Kyrgyzstan', 'lat': 41.20438, 'long': 74.766098},
{'_id': 'Latvia', 'lat': 56.8796, 'long': 24.6032},
{'_id': 'Lebanon', 'lat': 33.8547, 'long': 35.8623},
{'_id': 'Lesotho', 'lat': -29.610000000000003, 'long': 28.2336},
{'_id': 'Liberia', 'lat': 6.428055, 'long': -9.429499},
{'_id': 'Libya', 'lat': 26.3351, 'long': 17.228331},
{'_id': 'Lithuania', 'lat': 55.1694, 'long': 23.8813},
{'_id': 'Luxembourg', 'lat': 49.8153, 'long': 6.1296},
{'_id': 'Madagascar', 'lat': -18.766947, 'long': 46.869107},
{'_id': 'Malawi', 'lat': -13.254300000000002, 'long': 34.3015},
{'_id': 'Malaysia', 'lat': 4.210483999999999, 'long': 101.975766},
{'_id': 'Maldives', 'lat': 3.2027999999999994, 'long': 73.2207},
{'_id': 'Mali', 'lat': 17.570692, 'long': -3.9961660000000014},
{'_id': 'Malta', 'lat': 35.9375, 'long': 14.3754},
{'_id': 'Mauritania', 'lat': 21.0079, 'long': -10.9408},
{'_id': 'Mauritius', 'lat': -20.348404, 'long': 57.552152},
{'_id': 'Mexico', 'lat': 23.6345, 'long': -102.5528},
{'_id': 'Moldova', 'lat': 47.4116, 'long': 28.3699},
{'_id': 'Mongolia', 'lat': 46.8625, 'long': 103.84669999999998},
{'_id': 'Montenegro', 'lat': 42.708678000000006, 'long': 19.37439},
{'_id': 'Morocco', 'lat': 31.7917, 'long': -7.0926},
{'_id': 'Mozambique', 'lat': -18.665695, 'long': 35.529562},
{'_id': 'Namibia', 'lat': -22.9576, 'long': 18.4904},
{'_id': 'Nepal', 'lat': 28.1667, 'long': 84.25},
{'_id': 'Netherlands', 'lat': 52.1326, 'long': 5.2913},
{'_id': 'New Zealand', 'lat': -40.9006, 'long': 174.88600000000002},
{'_id': 'Nicaragua', 'lat': 12.865416, 'long': -85.207229},
{'_id': 'Niger', 'lat': 17.607789, 'long': 8.081666},
{'_id': 'Nigeria', 'lat': 9.082, 'long': 8.6753},
{'_id': 'North Macedonia', 'lat': 41.6086, 'long': 21.7453},
{'_id': 'Norway', 'lat': 60.47200000000001, 'long': 8.4689},
{'_id': 'Oman', 'lat': 21.512583, 'long': 55.92325500000001},
{'_id': 'Pakistan', 'lat': 30.3753, 'long': 69.3451},
{'_id': 'Panama', 'lat': 8.538, 'long': -80.7821},
{'_id': 'Paraguay', 'lat': -23.4425, 'long': -58.4438},
{'_id': 'Peru', 'lat': -9.19, 'long': -75.0152},
{'_id': 'Philippines', 'lat': 12.879721, 'long': 121.774017},
{'_id': 'Poland', 'lat': 51.9194, 'long': 19.1451},
{'_id': 'Portugal', 'lat': 39.3999, 'long': -8.2245},
{'_id': 'Qatar', 'lat': 25.3548, 'long': 51.1839},
{'_id': 'Romania', 'lat': 45.9432, 'long': 24.9668},
{'_id': 'Russia', 'lat': 61.52401, 'long': 105.318756},
{'_id': 'Rwanda', 'lat': -1.9402999999999997, 'long': 29.8739},
{'_id': 'San Marino', 'lat': 43.9424, 'long': 12.4578},
{'_id': 'Sao Tome and Principe', 'lat': 0.1864, 'long': 6.6131},
{'_id': 'Saudi Arabia', 'lat': 23.885942, 'long': 45.079162},

```
{'_id': 'Senegal', 'lat': 14.4974, 'long': -14.4524},
{'_id': 'Serbia', 'lat': 44.0165, 'long': 21.0059},
{'_id': 'Sierra Leone', 'lat': 8.460555000000001, 'long': -11.779888999999999},
{'_id': 'Singapore', 'lat': 1.2833, 'long': 103.8333},
{'_id': 'Slovakia', 'lat': 48.669, 'long': 19.699},
{'_id': 'Slovenia', 'lat': 46.1512, 'long': 14.9955},
{'_id': 'Somalia', 'lat': 5.152149, 'long': 46.199616000000006},
{'_id': 'South Africa', 'lat': -30.5595, 'long': 22.9375},
{'_id': 'South Korea', 'lat': 35.90775700000001, 'long': 127.766922},
{'_id': 'South Sudan', 'lat': 6.877000000000002, 'long': 31.306999999999995},
{'_id': 'Spain', 'lat': 40.463667, 'long': -3.7492200000000007},
{'_id': 'Sri Lanka', 'lat': 7.873054, 'long': 80.77179699999998},
{'_id': 'Sudan', 'lat': 12.862800000000002, 'long': 30.2176},
{'_id': 'Suriname', 'lat': 3.9193, 'long': -56.027800000000006},
{'_id': 'Sweden', 'lat': 60.128161000000006, 'long': 18.643501},
{'_id': 'Switzerland', 'lat': 46.8182, 'long': 8.2275},
{'_id': 'Syria', 'lat': 34.802075, 'long': 38.99681500000001},
{'_id': 'Taiwan*', 'lat': 23.7, 'long': 121.0},
{'_id': 'Tajikistan', 'lat': 38.861, 'long': 71.2761},
{'_id': 'Tanzania', 'lat': -6.369028, 'long': 34.888822},
{'_id': 'Thailand', 'lat': 15.870032000000002, 'long': 100.992541},
{'_id': 'Togo', 'lat': 8.6195, 'long': 0.8248},
{'_id': 'Tunisia', 'lat': 33.886917, 'long': 9.537499},
{'_id': 'Turkey', 'lat': 38.9637, 'long': 35.2433},
{'_id': 'US', 'lat': 40.0, 'long': -100.0},
{'_id': 'Uganda', 'lat': 1.373333, 'long': 32.290275},
{'_id': 'Ukraine', 'lat': 48.3794, 'long': 31.1656},
{'_id': 'United Arab Emirates', 'lat': 23.424076, 'long': 53.847818},
{'_id': 'United Kingdom',
 'lat': 54.280426041666665,
 'long': -3.3155328125000003},
{'_id': 'Uruguay', 'lat': -32.5228, 'long': -55.7658},
{'_id': 'Uzbekistan', 'lat': 41.37749099999999, 'long': 64.585262},
{'_id': 'Venezuela', 'lat': 6.4238, 'long': -66.5897},
{'_id': 'Vietnam', 'lat': 14.058324, 'long': 108.27719899999998},
{'_id': 'West Bank and Gaza', 'lat': 31.9522, 'long': 35.2332},
{'_id': 'Yemen', 'lat': 15.552727, 'long': 48.516388},
{'_id': 'Zambia', 'lat': -13.133897, 'long': 27.849331999999997},
{'_id': 'Zimbabwe', 'lat': -19.015438, 'long': 29.154857}]
```

[ ]:
```
# Verifying using pandas to check if we get the same answer
df = pd.read_csv(covid_data)
df[df["Active"] > 100].groupby(by=["Country/Region"])[["Lat", "Long"]].mean()
```

[ ]:
```
                          Lat        Long
Country/Region
Afghanistan         33.939110   67.709953
```

```
Albania              41.153300   20.168300
Algeria              28.033900    1.659600
Andorra              42.506300    1.521800
Angola              -11.202700   17.873900
…                         …          …
Vietnam              14.058324  108.277199
West Bank and Gaza   31.952200   35.233200
Yemen                15.552727   48.516388
Zambia              -13.133897   27.849332
Zimbabwe            -19.015438   29.154857

[166 rows x 2 columns]
```

4.12 Q12

The countries and respective dates in which maximum increase of active cases occurred.

[ ]:
```
qry = full_coll.aggregate(
    [
        {
            "$setWindowFields": {
                "partitionBy": "$Country/Region",
                "sortBy": {"Date": 1},
                "output": {
                    "PreviousActive": {
                        "$shift": {"output": "$Active", "by": -1, "default": 0}
                    }
                },
            },
        },
        {"$set": {"Delta": {"$subtract": ["$Active", "$PreviousActive"]}}},
        {"$project": {"Delta": 1, "Country/Region": 1, "Date": 1}},
        {"$sort": {"Country/Region": 1, "Delta": -1}},
        {
            "$group": {
                "_id": "$Country/Region",
                "Date": {"$first": "$Date"},
                "Increase": {"$first": "$Delta"},
            }
        },
        {"$sort": {"_id": 1}},
    ]
)

pprint(list(qry))
```

```
[{'Date': '2020-05-30', 'Increase': 819, '_id': 'Afghanistan'},
 {'Date': '2020-07-11', 'Increase': 83, '_id': 'Albania'},
```

{'Date': '2020-07-24', 'Increase': 663, '_id': 'Algeria'},
{'Date': '2020-06-02', 'Increase': 44, '_id': 'Andorra'},
{'Date': '2020-07-10', 'Increase': 61, '_id': 'Angola'},
{'Date': '2020-06-25', 'Increase': 39, '_id': 'Antigua and Barbuda'},
{'Date': '2020-07-17', 'Increase': 3792, '_id': 'Argentina'},
{'Date': '2020-06-04', 'Increase': 677, '_id': 'Armenia'},
{'Date': '2020-03-28', 'Increase': 446, '_id': 'Australia'},
{'Date': '2020-03-26', 'Increase': 1199, '_id': 'Austria'},
{'Date': '2020-06-24', 'Increase': 318, '_id': 'Azerbaijan'},
{'Date': '2020-07-23', 'Increase': 55, '_id': 'Bahamas'},
{'Date': '2020-06-03', 'Increase': 500, '_id': 'Bahrain'},
{'Date': '2020-06-15', 'Increase': 3060, '_id': 'Bangladesh'},
{'Date': '2020-04-02', 'Increase': 12, '_id': 'Barbados'},
{'Date': '2020-04-29', 'Increase': 889, '_id': 'Belarus'},
{'Date': '2020-04-15', 'Increase': 1932, '_id': 'Belgium'},
{'Date': '2020-07-10', 'Increase': 6, '_id': 'Belize'},
{'Date': '2020-06-21', 'Increase': 107, '_id': 'Benin'},
{'Date': '2020-06-07', 'Increase': 11, '_id': 'Bhutan'},
{'Date': '2020-07-18', 'Increase': 1661, '_id': 'Bolivia'},
{'Date': '2020-07-06', 'Increase': 440, '_id': 'Bosnia and Herzegovina'},
{'Date': '2020-07-16', 'Increase': 113, '_id': 'Botswana'},
{'Date': '2020-06-19', 'Increase': 36514, '_id': 'Brazil'},
{'Date': '2020-03-13', 'Increase': 26, '_id': 'Brunei'},
{'Date': '2020-07-09', 'Increase': 264, '_id': 'Bulgaria'},
{'Date': '2020-05-06', 'Increase': 34, '_id': 'Burkina Faso'},
{'Date': '2020-06-18', 'Increase': 22, '_id': 'Burma'},
{'Date': '2020-07-16', 'Increase': 34, '_id': 'Burundi'},
{'Date': '2020-07-03', 'Increase': 81, '_id': 'Cabo Verde'},
{'Date': '2020-03-22', 'Increase': 31, '_id': 'Cambodia'},
{'Date': '2020-07-07', 'Increase': 853, '_id': 'Cameroon'},
{'Date': '2020-04-05', 'Increase': 2737, '_id': 'Canada'},
{'Date': '2020-06-08', 'Increase': 216, '_id': 'Central African Republic'},
{'Date': '2020-05-07', 'Increase': 66, '_id': 'Chad'},
{'Date': '2020-05-30', 'Increase': 11302, '_id': 'Chile'},
{'Date': '2020-02-13', 'Increase': 12305, '_id': 'China'},
{'Date': '2020-07-16', 'Increase': 4892, '_id': 'Colombia'},
{'Date': '2020-05-22', 'Increase': 34, '_id': 'Comoros'},
{'Date': '2020-07-01', 'Increase': 261, '_id': 'Congo (Brazzaville)'},
{'Date': '2020-06-25', 'Increase': 183, '_id': 'Congo (Kinshasa)'},
{'Date': '2020-07-25', 'Increase': 785, '_id': 'Costa Rica'},
{'Date': '2020-06-23', 'Increase': 536, '_id': "Cote d'Ivoire"},
{'Date': '2020-06-25', 'Increase': 91, '_id': 'Croatia'},
{'Date': '2020-04-08', 'Increase': 60, '_id': 'Cuba'},
{'Date': '2020-04-01', 'Increase': 52, '_id': 'Cyprus'},
{'Date': '2020-04-04', 'Increase': 369, '_id': 'Czechia'},
{'Date': '2020-03-31', 'Increase': 267, '_id': 'Denmark'},
{'Date': '2020-05-30', 'Increase': 233, '_id': 'Djibouti'},
{'Date': '2020-03-25', 'Increase': 5, '_id': 'Dominica'},

{'Date': '2020-07-25', 'Increase': 1445, '_id': 'Dominican Republic'},
{'Date': '2020-04-24', 'Increase': 11482, '_id': 'Ecuador'},
{'Date': '2020-06-19', 'Increase': 1295, '_id': 'Egypt'},
{'Date': '2020-07-26', 'Increase': 300, '_id': 'El Salvador'},
{'Date': '2020-07-02', 'Increase': 724, '_id': 'Equatorial Guinea'},
{'Date': '2020-06-14', 'Increase': 31, '_id': 'Eritrea'},
{'Date': '2020-03-26', 'Increase': 134, '_id': 'Estonia'},
{'Date': '2020-07-18', 'Increase': 91, '_id': 'Eswatini'},
{'Date': '2020-07-08', 'Increase': 911, '_id': 'Ethiopia'},
{'Date': '2020-04-04', 'Increase': 5, '_id': 'Fiji'},
{'Date': '2020-04-04', 'Increase': 262, '_id': 'Finland'},
{'Date': '2020-04-12', 'Increase': 25482, '_id': 'France'},
{'Date': '2020-06-15', 'Increase': 256, '_id': 'Gabon'},
{'Date': '2020-07-25', 'Increase': 61, '_id': 'Gambia'},
{'Date': '2020-04-16', 'Increase': 37, '_id': 'Georgia'},
{'Date': '2020-03-27', 'Increase': 5873, '_id': 'Germany'},
{'Date': '2020-05-08', 'Increase': 901, '_id': 'Ghana'},
{'Date': '2020-04-02', 'Increase': 117, '_id': 'Greece'},
{'Date': '2020-03-27', 'Increase': 4, '_id': 'Greenland'},
{'Date': '2020-03-26', 'Increase': 6, '_id': 'Grenada'},
{'Date': '2020-07-04', 'Increase': 1153, '_id': 'Guatemala'},
{'Date': '2020-05-21', 'Increase': 154, '_id': 'Guinea'},
{'Date': '2020-05-04', 'Increase': 156, '_id': 'Guinea-Bissau'},
{'Date': '2020-06-22', 'Increase': 21, '_id': 'Guyana'},
{'Date': '2020-06-06', 'Increase': 332, '_id': 'Haiti'},
{'Date': '2020-03-24', 'Increase': 3, '_id': 'Holy See'},
{'Date': '2020-06-20', 'Increase': 978, '_id': 'Honduras'},
{'Date': '2020-04-10', 'Increase': 183, '_id': 'Hungary'},
{'Date': '2020-03-25', 'Increase': 84, '_id': 'Iceland'},
{'Date': '2020-07-26', 'Increase': 17275, '_id': 'India'},
{'Date': '2020-07-09', 'Increase': 1533, '_id': 'Indonesia'},
{'Date': '2020-03-28', 'Increase': 2391, '_id': 'Iran'},
{'Date': '2020-06-26', 'Increase': 1124, '_id': 'Iraq'},
{'Date': '2020-04-10', 'Increase': 1491, '_id': 'Ireland'},
{'Date': '2020-07-27', 'Increase': 1917, '_id': 'Israel'},
{'Date': '2020-03-20', 'Increase': 5359, '_id': 'Italy'},
{'Date': '2020-04-20', 'Increase': 50, '_id': 'Jamaica'},
{'Date': '2020-07-15', 'Increase': 735, '_id': 'Japan'},
{'Date': '2020-05-28', 'Increase': 97, '_id': 'Jordan'},
{'Date': '2020-07-01', 'Increase': 18256, '_id': 'Kazakhstan'},
{'Date': '2020-07-26', 'Increase': 789, '_id': 'Kenya'},
{'Date': '2020-06-22', 'Increase': 585, '_id': 'Kosovo'},
{'Date': '2020-05-10', 'Increase': 949, '_id': 'Kuwait'},
{'Date': '2020-07-18', 'Increase': 3895, '_id': 'Kyrgyzstan'},
{'Date': '2020-03-26', 'Increase': 3, '_id': 'Laos'},
{'Date': '2020-04-03', 'Increase': 64, '_id': 'Latvia'},
{'Date': '2020-07-25', 'Increase': 169, '_id': 'Lebanon'},
{'Date': '2020-07-25', 'Increase': 57, '_id': 'Lesotho'},

```
{'Date': '2020-06-19', 'Increase': 39, '_id': 'Liberia'},
{'Date': '2020-07-27', 'Increase': 130, '_id': 'Libya'},
{'Date': '2020-03-18', 'Increase': 21, '_id': 'Liechtenstein'},
{'Date': '2020-04-04', 'Increase': 73, '_id': 'Lithuania'},
{'Date': '2020-03-25', 'Increase': 234, '_id': 'Luxembourg'},
{'Date': '2020-07-26', 'Increase': 422, '_id': 'Madagascar'},
{'Date': '2020-07-22', 'Increase': 143, '_id': 'Malawi'},
{'Date': '2020-06-04', 'Increase': 249, '_id': 'Malaysia'},
{'Date': '2020-04-30', 'Increase': 190, '_id': 'Maldives'},
{'Date': '2020-06-10', 'Increase': 62, '_id': 'Mali'},
{'Date': '2020-04-07', 'Increase': 52, '_id': 'Malta'},
{'Date': '2020-06-16', 'Increase': 155, '_id': 'Mauritania'},
{'Date': '2020-04-09', 'Increase': 37, '_id': 'Mauritius'},
{'Date': '2020-07-19', 'Increase': 21313, '_id': 'Mexico'},
{'Date': '2020-06-17', 'Increase': 292, '_id': 'Moldova'},
{'Date': '2020-03-22', 'Increase': 11, '_id': 'Monaco'},
{'Date': '2020-05-14', 'Increase': 56, '_id': 'Mongolia'},
{'Date': '2020-07-17', 'Increase': 649, '_id': 'Montenegro'},
{'Date': '2020-07-25', 'Increase': 623, '_id': 'Morocco'},
{'Date': '2020-07-13', 'Increase': 62, '_id': 'Mozambique'},
{'Date': '2020-07-18', 'Increase': 125, '_id': 'Namibia'},
{'Date': '2020-06-18', 'Increase': 650, '_id': 'Nepal'},
{'Date': '2020-04-10', 'Increase': 1220, '_id': 'Netherlands'},
{'Date': '2020-04-02', 'Increase': 80, '_id': 'New Zealand'},
{'Date': '2020-06-02', 'Increase': 348, '_id': 'Nicaragua'},
{'Date': '2020-04-08', 'Increase': 62, '_id': 'Niger'},
{'Date': '2020-06-11', 'Increase': 533, '_id': 'Nigeria'},
{'Date': '2020-07-18', 'Increase': 181, '_id': 'North Macedonia'},
{'Date': '2020-03-27', 'Increase': 381, '_id': 'Norway'},
{'Date': '2020-07-13', 'Increase': 1003, '_id': 'Oman'},
{'Date': '2020-06-14', 'Increase': 8230, '_id': 'Pakistan'},
{'Date': '2020-06-14', 'Increase': 1344, '_id': 'Panama'},
{'Date': '2020-07-26', 'Increase': 23, '_id': 'Papua New Guinea'},
{'Date': '2020-06-27', 'Increase': 197, '_id': 'Paraguay'},
{'Date': '2020-07-27', 'Increase': 8484, '_id': 'Peru'},
{'Date': '2020-07-08', 'Increase': 2279, '_id': 'Philippines'},
{'Date': '2020-04-19', 'Increase': 473, '_id': 'Poland'},
{'Date': '2020-04-10', 'Increase': 1462, '_id': 'Portugal'},
{'Date': '2020-05-14', 'Increase': 1520, '_id': 'Qatar'},
{'Date': '2020-07-25', 'Increase': 1245, '_id': 'Romania'},
{'Date': '2020-05-04', 'Increase': 9049, '_id': 'Russia'},
{'Date': '2020-06-29', 'Increase': 101, '_id': 'Rwanda'},
{'Date': '2020-03-30', 'Increase': 5, '_id': 'Saint Kitts and Nevis'},
{'Date': '2020-03-29', 'Increase': 6, '_id': 'Saint Lucia'},
{'Date': '2020-05-28',
 'Increase': 7,
 '_id': 'Saint Vincent and the Grenadines'},
{'Date': '2020-04-16', 'Increase': 50, '_id': 'San Marino'},
```

```
{'Date': '2020-05-05', 'Increase': 151, '_id': 'Sao Tome and Principe'},
{'Date': '2020-06-12', 'Increase': 2875, '_id': 'Saudi Arabia'},
{'Date': '2020-05-11', 'Increase': 112, '_id': 'Senegal'},
{'Date': '2020-07-20', 'Increase': 15913, '_id': 'Serbia'},
{'Date': '2020-06-28', 'Increase': 50, '_id': 'Seychelles'},
{'Date': '2020-05-24', 'Increase': 85, '_id': 'Sierra Leone'},
{'Date': '2020-04-20', 'Increase': 1393, '_id': 'Singapore'},
{'Date': '2020-04-08', 'Increase': 98, '_id': 'Slovakia'},
{'Date': '2020-03-27', 'Increase': 67, '_id': 'Slovenia'},
{'Date': '2020-05-12', 'Increase': 76, '_id': 'Somalia'},
{'Date': '2020-07-26', 'Increase': 9096, '_id': 'South Africa'},
{'Date': '2020-03-03', 'Increase': 851, '_id': 'South Korea'},
{'Date': '2020-06-07', 'Increase': 319, '_id': 'South Sudan'},
{'Date': '2020-03-25', 'Increase': 7218, '_id': 'Spain'},
{'Date': '2020-07-10', 'Increase': 299, '_id': 'Sri Lanka'},
{'Date': '2020-05-21', 'Increase': 377, '_id': 'Sudan'},
{'Date': '2020-06-27', 'Increase': 67, '_id': 'Suriname'},
{'Date': '2020-06-29', 'Increase': 2500, '_id': 'Sweden'},
{'Date': '2020-03-23', 'Increase': 1299, '_id': 'Switzerland'},
{'Date': '2020-07-24', 'Increase': 24, '_id': 'Syria'},
{'Date': '2020-03-20', 'Increase': 26, '_id': 'Taiwan*'},
{'Date': '2020-05-20', 'Increase': 375, '_id': 'Tajikistan'},
{'Date': '2020-04-20', 'Increase': 81, '_id': 'Tanzania'},
{'Date': '2020-03-22', 'Increase': 186, '_id': 'Thailand'},
{'Date': '2020-04-16', 'Increase': 10, '_id': 'Timor-Leste'},
{'Date': '2020-05-16', 'Increase': 32, '_id': 'Togo'},
{'Date': '2020-03-21', 'Increase': 39, '_id': 'Trinidad and Tobago'},
{'Date': '2020-03-31', 'Increase': 80, '_id': 'Tunisia'},
{'Date': '2020-04-11', 'Increase': 4501, '_id': 'Turkey'},
{'Date': '2020-07-16', 'Increase': 61549, '_id': 'US'},
{'Date': '2020-07-21', 'Increase': 116, '_id': 'Uganda'},
{'Date': '2020-06-28', 'Increase': 668, '_id': 'Ukraine'},
{'Date': '2020-05-27', 'Increase': 492, '_id': 'United Arab Emirates'},
{'Date': '2020-04-06', 'Increase': 4687, '_id': 'United Kingdom'},
{'Date': '2020-03-27', 'Increase': 36, '_id': 'Uruguay'},
{'Date': '2020-07-21', 'Increase': 593, '_id': 'Uzbekistan'},
{'Date': '2020-07-05', 'Increase': 416, '_id': 'Venezuela'},
{'Date': '2020-03-22', 'Increase': 19, '_id': 'Vietnam'},
{'Date': '2020-07-04', 'Increase': 499, '_id': 'West Bank and Gaza'},
{'Date': '2020-04-05', 'Increase': 4, '_id': 'Western Sahara'},
{'Date': '2020-06-15', 'Increase': 46, '_id': 'Yemen'},
{'Date': '2020-07-17', 'Increase': 810, '_id': 'Zambia'},
{'Date': '2020-07-16', 'Increase': 240, '_id': 'Zimbabwe'}]
```

[ ]:
```
# Verify the veracity of the above query using pandas
groups = df.groupby(by=["Country/Region"])
recs = []
```

```
for s, g in groups:
    g = g.sort_values(by="Date", ascending=True)
    g["Previous"] = g["Active"].shift(periods=1)
    g["Delta"] = g["Active"] - g["Previous"]
    g = g.sort_values(by=["Delta"], ascending=False).head(1)
    recs.append(g[["Date", "Country/Region", "Delta"]])
date_df = pd.concat(recs)
date_df
```

[ ]:

	Date	Country/Region	Delta
33669	2020-05-30	Afghanistan	819.0
44632	2020-07-11	Albania	83.0
48026	2020-07-24	Algeria	663.0
34455	2020-06-02	Andorra	44.0
44374	2020-07-10	Angola	61.0
…	…	…	…
43041	2020-07-04	West Bank and Gaza	499.0
19569	2020-04-05	Western Sahara	4.0
38102	2020-06-15	Yemen	46.0
46424	2020-07-17	Zambia	810.0
46164	2020-07-16	Zimbabwe	240.0

[187 rows x 3 columns]