VELLORE INSTITUTE OF TECHNOLOGY

SCHOOL OF ADVANCED SCIENCES

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

A PROJECT REPORT

ON COVID VACCINATION ANALYSIS ON WORLD DATASET

SUBMITTEED BY

PARVINDER KAUR (21MDT0131)

COURSE CODE: CSE5007

COURSE TITLE: Exploratory Data Analysis

Under the guidance of

Dr B RUSHI KUMAR

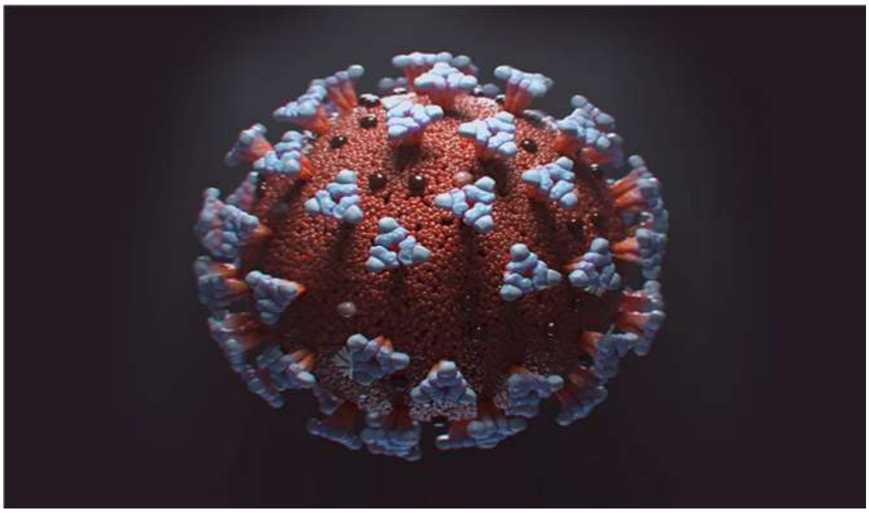
Associate Professor , SAS

VELLORE INSTITUTE OF TECHNOLOGY, VELLORE

INTRODUCTION

What is coronavirus?

Coronaviruses are a family of   viruses that can cause respiratory illness in humans. They are called “corona” because of crown-like spikes on the surface of the virus. Severe acute respiratory syndrome (SARS), Middle East respiratory syndrome (MERS) and the common cold are examples of coronaviruses that cause illness in humans.



SARS-CoV-2, the virus that causes COVID-19, enters your body through your mouth, nose or eyes (directly from the airborne droplets or from the transfer of the virus from your hands to your face). It then travels to the back of your nasal passages and mucous membrane in the back of your throat. It attaches to cells there, begins to multiply and moves into lung tissue. From there, the virus can spread to other body tissues.

The virus travels in respiratory droplets released into the air when an infected person coughs, sneezes, talks near you.

BACKGROUND

In December 2019, China reported an outbreak of pneumonia of unknown causes in Wuhan, the capital city of Hubei province. Most of the early cases were epidemiologically linked to the Huanan seafood wholesale market where aquatic animals and live animals were sold. Using unbiased next-generation sequencing, an unknown betacoronavirus was discovered from lower respiratory tract samples of these patients. Human airway epithelial cells were used to isolate the virus that was named 2019–novel Coronavirus. The virus when observed under electron microscope had a diameter of 60 to 140 nm with characteristic spikes of 9 to 12 nm, similar to the Coronoviridae family. The World Health Organization (WHO) named the resultant disease as Coronavirus disease (COVID-19). On March 11, 2020, WHO, after assessing the situation across the globe, declared COVID-19 as a pandemic.

DATA OVERVIEW

This dataset includes information about:

- **Country** - this is the country for which the vaccination information is provided;
- **Country ISO Code** - ISO code for the country;
- **Date**- date for the data entry; for some of the dates we have only the daily vaccinations, for others, only the (cumulative) total;
- **Total number of vaccinations** - this is the absolute number of total immunizations in the country;
- **Total number of people vaccinated** - a person, depending on the immunization scheme, will receive one or more

(typically 2) vaccines; at a certain moment, the number of vaccination might be larger than the number of people;

- **Total number of people fully vaccinated** - this is the number of people that received the entire set of immunization according to the immunization scheme (typically 2); at a certain moment in time, there might be a certain number of people that received one vaccine and another number (smaller) of people that received all vaccines in the scheme;
- **Daily vaccinations (raw)** - for a certain data entry, the number of vaccination for that date/country;
- **Daily vaccinations** - for a certain data entry, the number of vaccination for that date/country;
- **Total vaccinations per hundred** - ratio (in percent) between vaccination number and total population up to the date in the country;
- **Total number of people vaccinated per hundred** - ratio (in percent) between population immunized and total population up to the date in the country;
- **Total number of people fully vaccinated per hundred** - ratio (in percent) between population fully immunized and total population up to the date in the country;
- **Number of vaccinations per day** - number of daily vaccination for that day and country;
- **Daily vaccinations per million** - ratio (in ppm) between vaccination number and total population for the current date in the country;
- **Vaccines used in the country** - total number of vaccines used in the country (up to date);
- **Source name** - source of the information (national authority, international organization, local organization etc.);
- **Source website** - website of the source of information;

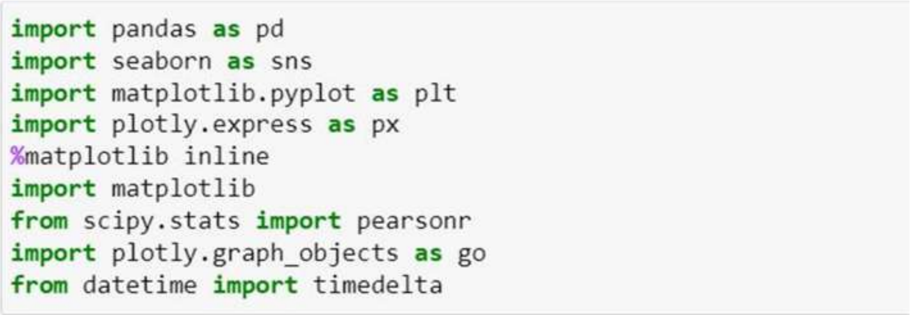
There is a second file added with the following columns:

- **Location** - country;
- **Date** - date;
- **Vaccine** - vaccine type;

- **Total number of vaccinations** - total number of vaccinations / current time and vaccine type.

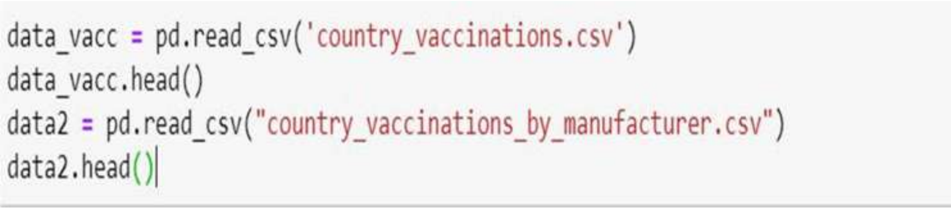
1. SETTING ENVIRONMENT

1.1 IMPORTING LIBRARIES



```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
%matplotlib inline
import matplotlib
from scipy.stats import pearsonr
import plotly.graph_objects as go
from datetime import timedelta
```

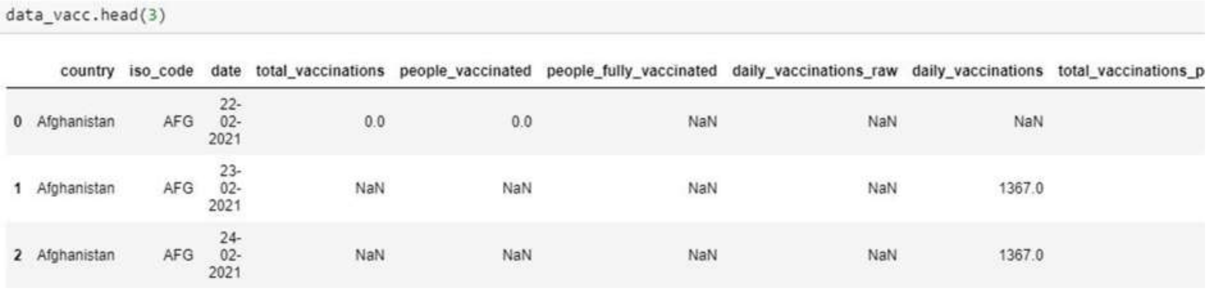
1.2 Reading files



```
data_vacc = pd.read_csv('country_vaccinations.csv')
data_vacc.head()
data2 = pd.read_csv("country_vaccinations_by_manufacturer.csv")
data2.head()
```

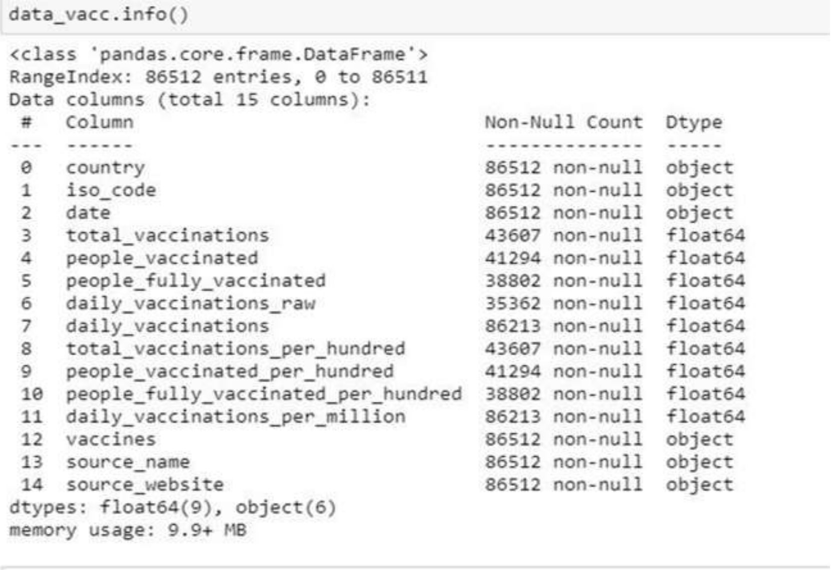
2. DATA CHECKING AND CLEANING

2.1 COUNTRY VACCINATION



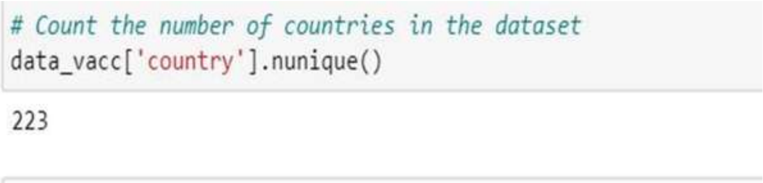
```
data_vacc.head(3)
```

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_p
0	Afghanistan	AFG	22-02-2021	0.0	0.0	NaN	NaN	NaN	
1	Afghanistan	AFG	23-02-2021	NaN	NaN	NaN	NaN	1367.0	
2	Afghanistan	AFG	24-02-2021	NaN	NaN	NaN	NaN	1367.0	



```
data_vacc.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 86512 entries, 0 to 86511
Data columns (total 15 columns):
 #   Column                               Non-Null Count  Dtype  
---  ------                               --------------  -----  
 0   country                              86512 non-null  object 
 1   iso_code                             86512 non-null  object 
 2   date                                 86512 non-null  object 
 3   total_vaccinations                   43607 non-null  float64
 4   people_vaccinated                    41294 non-null  float64
 5   people_fully_vaccinated              38802 non-null  float64
 6   daily_vaccinations_raw               35362 non-null  float64
 7   daily_vaccinations                   86213 non-null  float64
 8   total_vaccinations_per_hundred       43607 non-null  float64
 9   people_vaccinated_per_hundred        41294 non-null  float64
 10  people_fully_vaccinated_per_hundred  38802 non-null  float64
 11  daily_vaccinations_per_million       86213 non-null  float64
 12  vaccines                             86512 non-null  object 
 13  source_name                          86512 non-null  object 
 14  source_website                       86512 non-null  object 
dtypes: float64(9), object(6)
memory usage: 9.9+ MB
```



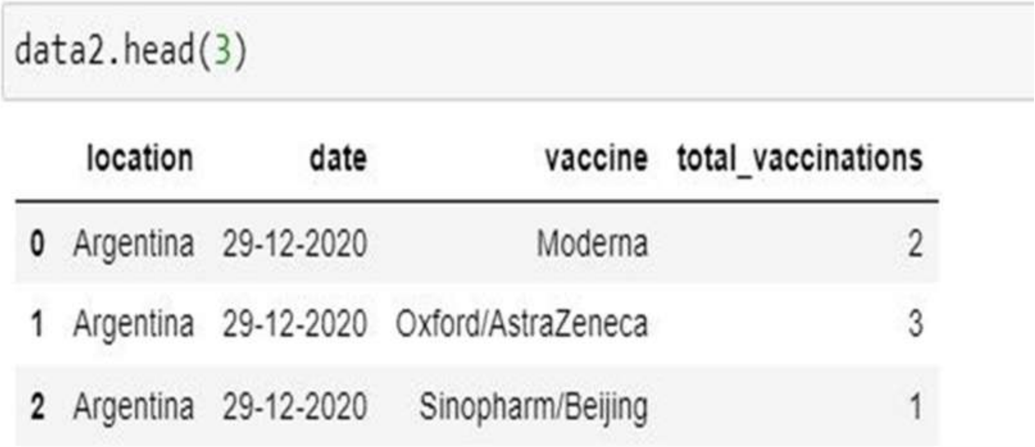
```
# Count the number of countries in the dataset
data_vacc['country'].nunique()
```

```
223
```

There are 223 countries in the first dataset.

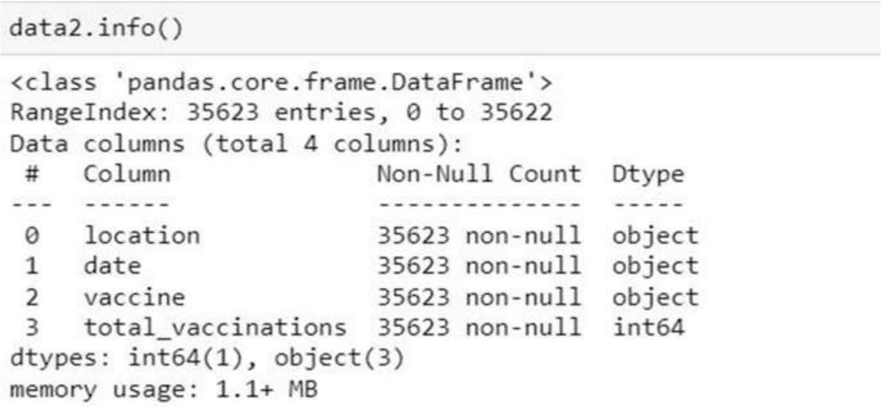
Since it looks like that the null values do not affect our calculation seen the values at the total_vaccionation columns are cummulative. Therefore, we did not need to clean our data.

2.2 COUNTRY VACCINATION BY MANUFACTURER



```
data2.head(3)
```

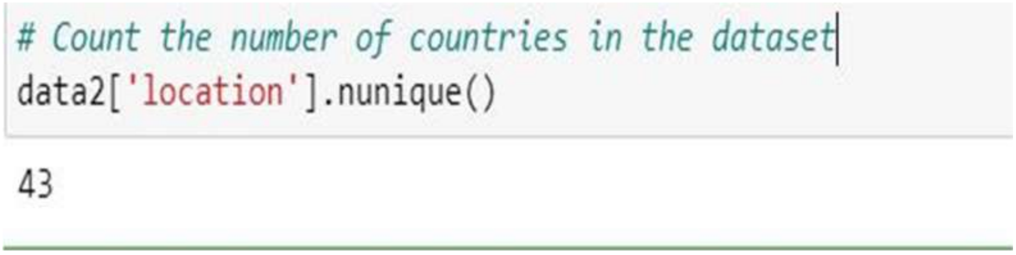
	location	date	vaccine	total_vaccinations
0	Argentina	29-12-2020	Moderna	2
1	Argentina	29-12-2020	Oxford/AstraZeneca	3
2	Argentina	29-12-2020	Sinopharm/Beijing	1



```
data2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35623 entries, 0 to 35622
Data columns (total 4 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   location            35623 non-null  object
 1   date                35623 non-null  object
 2   vaccine             35623 non-null  object
 3   total_vaccinations  35623 non-null  int64
dtypes: int64(1), object(3)
memory usage: 1.1+ MB
```

This dataset looks perfect!



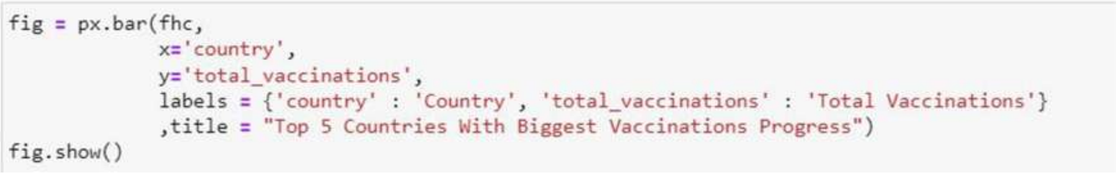
```
# Count the number of countries in the dataset
data2['location'].nunique()
```

```
43
```

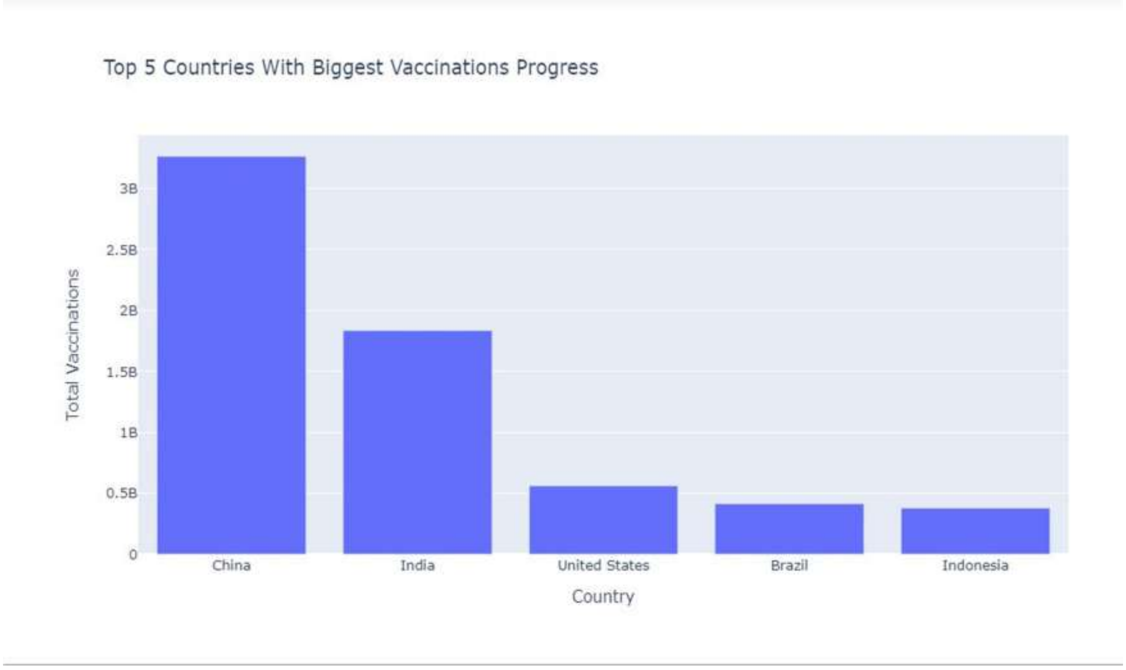
Meanwhile , in the country vaccinations by manufacturer , only 43 countries are listed in the data.

3. EXPLORATORY DATA ANALYSIS

3.1 What are the top 5 countries with biggest vaccination progress?

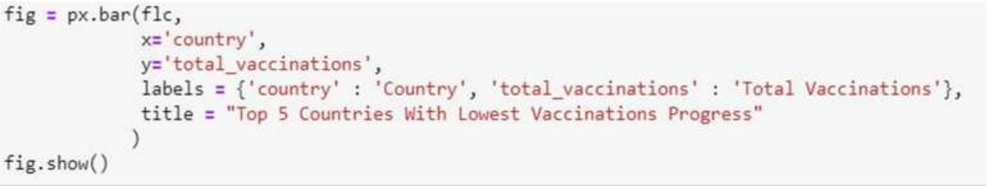


```
fig = px.bar(fhc,
             x='country',
             y='total_vaccinations',
             labels = {'country' : 'Country', 'total_vaccinations' : 'Total Vaccinations'}
             ,title = "Top 5 Countries With Biggest Vaccinations Progress")
fig.show()
```

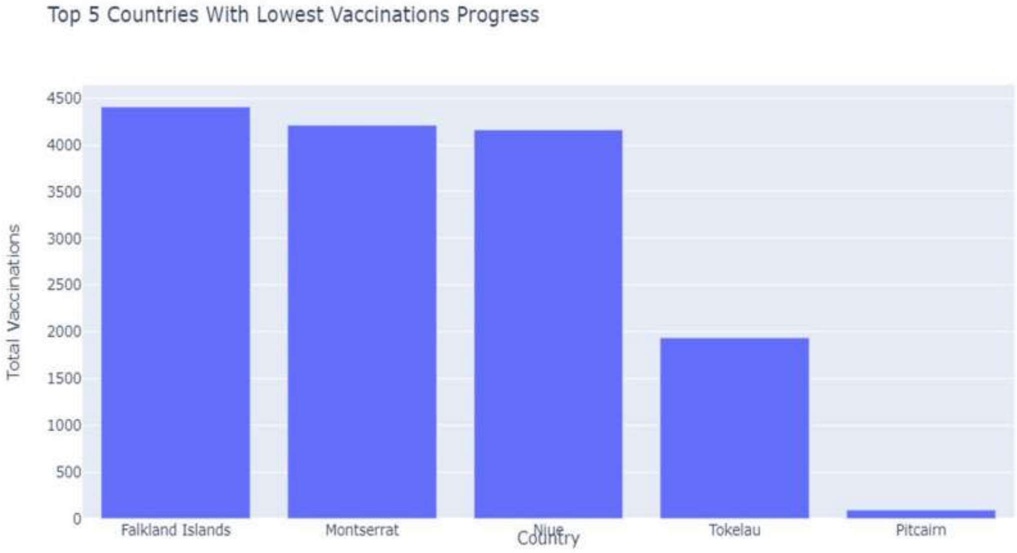


China is dominating in the number of vaccine used.

3.2 What are the top 5 smallest Countries with vaccination progress?

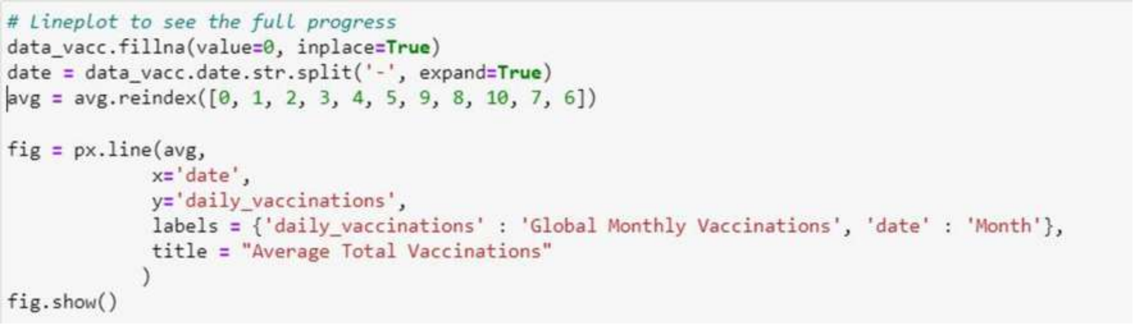


```
fig = px.bar(flc,
             x='country',
             y='total_vaccinations',
             labels = {'country' : 'Country', 'total_vaccinations' : 'Total Vaccinations'},
             title = "Top 5 Countries With Lowest Vaccinations Progress"
            )
fig.show()
```

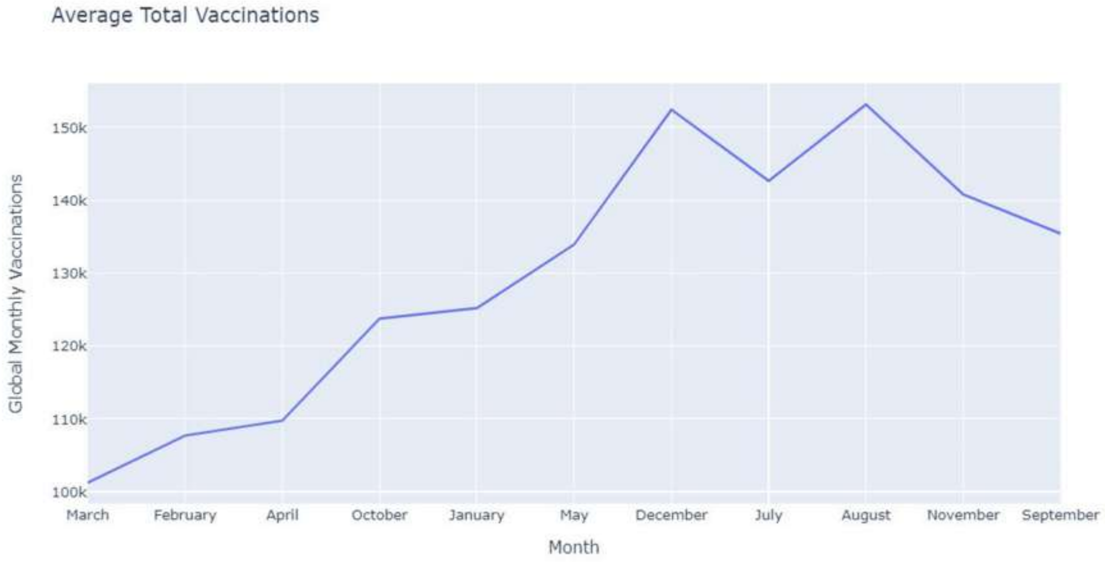


Pitcairn is an island whose sorvereign state is United Kingdom and it has lowest vaccination progress.

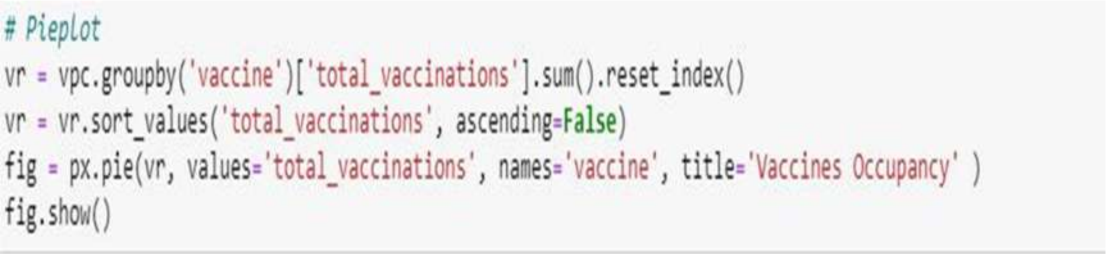
3.3 What is the global average vaccinations by month ?



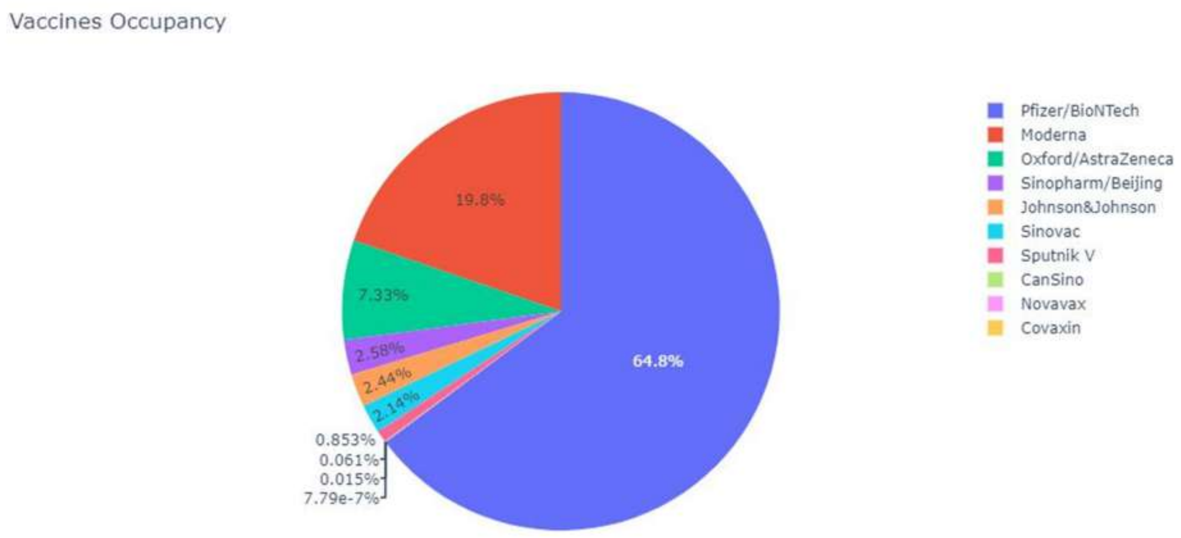
```
# Lineplot to see the full progress
data_vacc.fillna(value=0, inplace=True)
date = data_vacc.date.str.split('-', expand=True)
avg = avg.reindex([0, 1, 2, 3, 4, 5, 9, 8, 10, 7, 6])

fig = px.line(avg,
             x='date',
             y='daily_vaccinations',
             labels = {'daily_vaccinations' : 'Global Monthly Vaccinations', 'date' : 'Month'},
             title = "Average Total Vaccinations"
            )
fig.show()
```



3.4 What vaccine is most common used and least common used ?

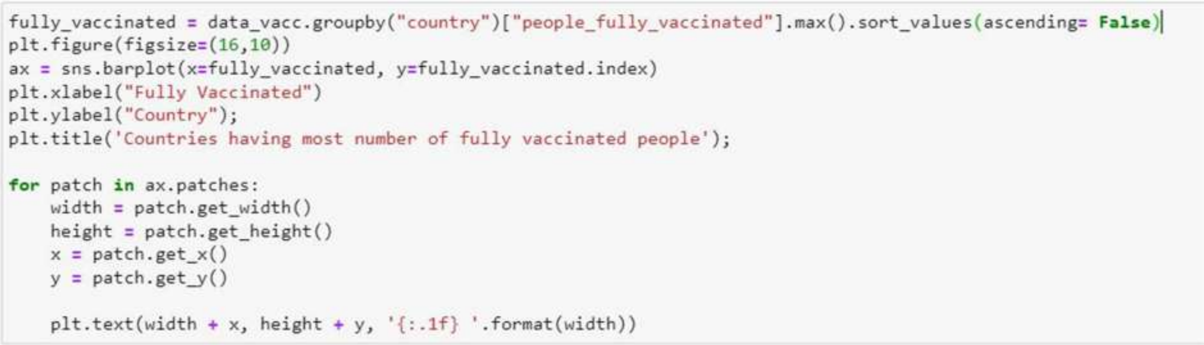


```
# Pieplot
vr = vpc.groupby('vaccine')['total_vaccinations'].sum().reset_index()
vr = vr.sort_values('total_vaccinations', ascending=False)
fig = px.pie(vr, values='total_vaccinations', names='vaccine', title='Vaccines Occupancy' )
fig.show()
```

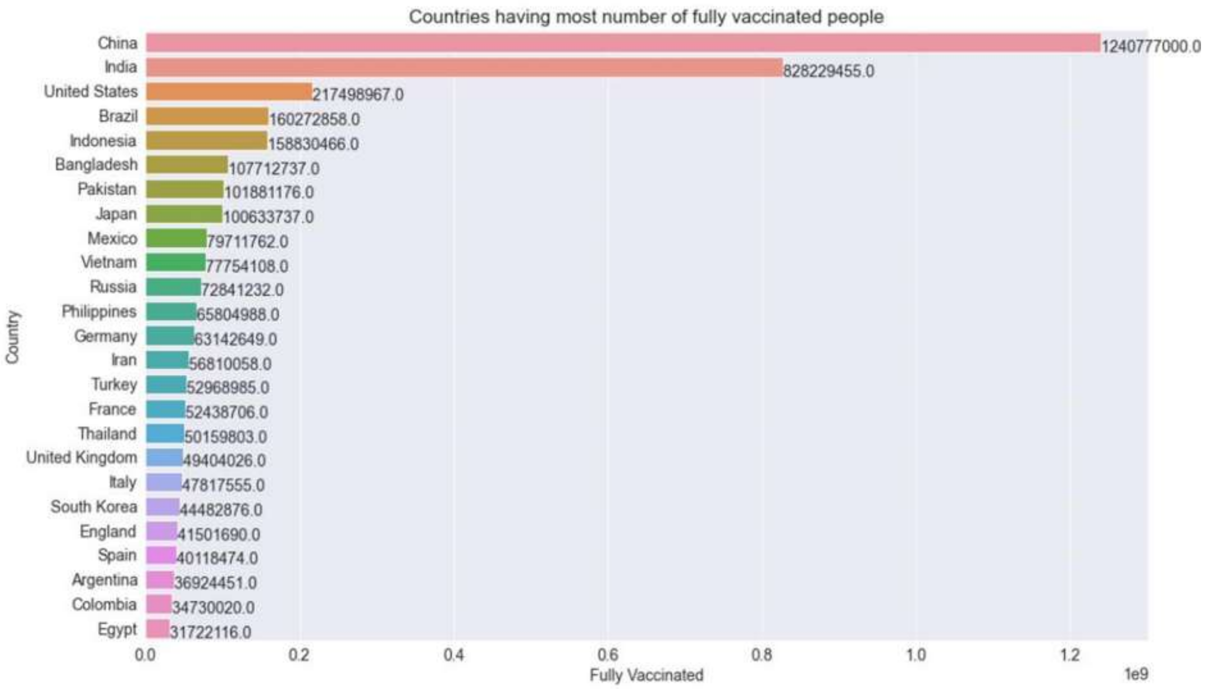


BioNTech is mostly used while covaxin is the least used.

3.5 Which country has most number of fully vaccinated people?

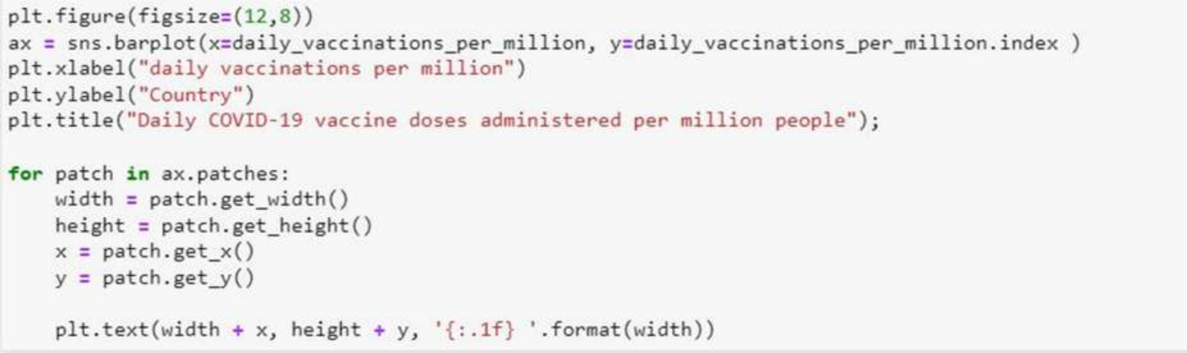


```
fully_vaccinated = data_vacc.groupby("country")["people_fully_vaccinated"].max().sort_values(ascending= False)
plt.figure(figsize=(16,10))
ax = sns.barplot(x=fully_vaccinated, y=fully_vaccinated.index)
plt.xlabel("Fully Vaccinated")
plt.ylabel("Country");
plt.title('Countries having most number of fully vaccinated people');

for patch in ax.patches:
    width = patch.get_width()
    height = patch.get_height()
    x = patch.get_x()
    y = patch.get_y()

    plt.text(width + x, height + y, '{:.1f} '.format(width))
```

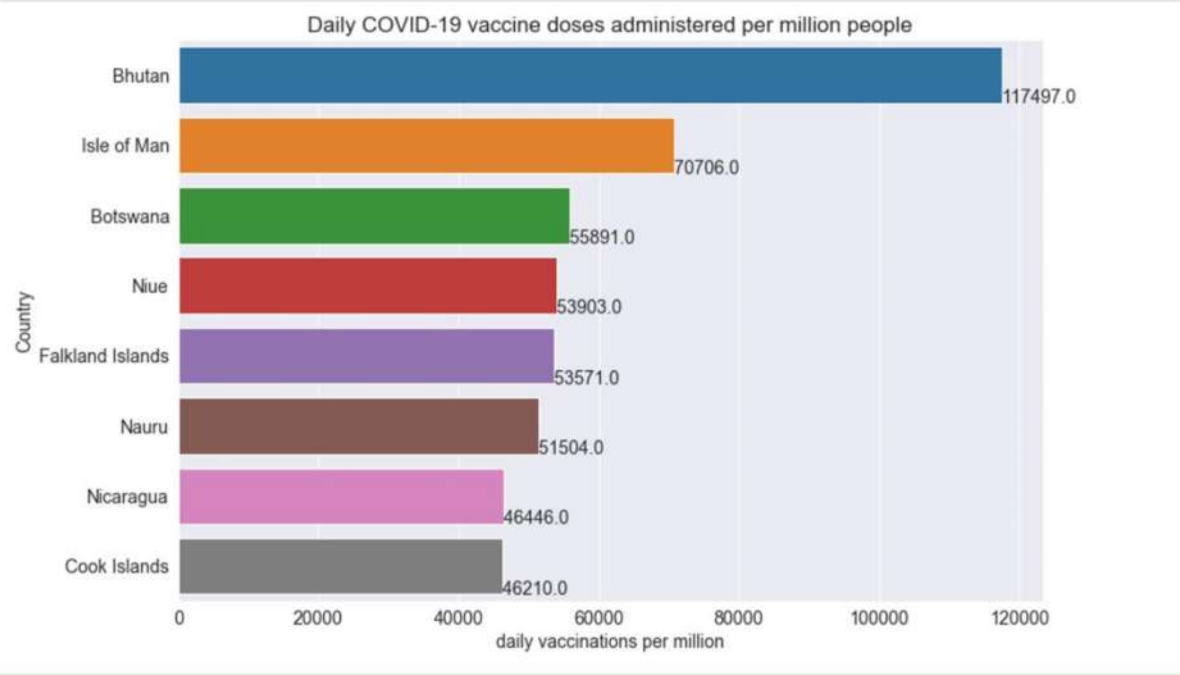


As we can see from plot, China has most number of fully vaccinated people.

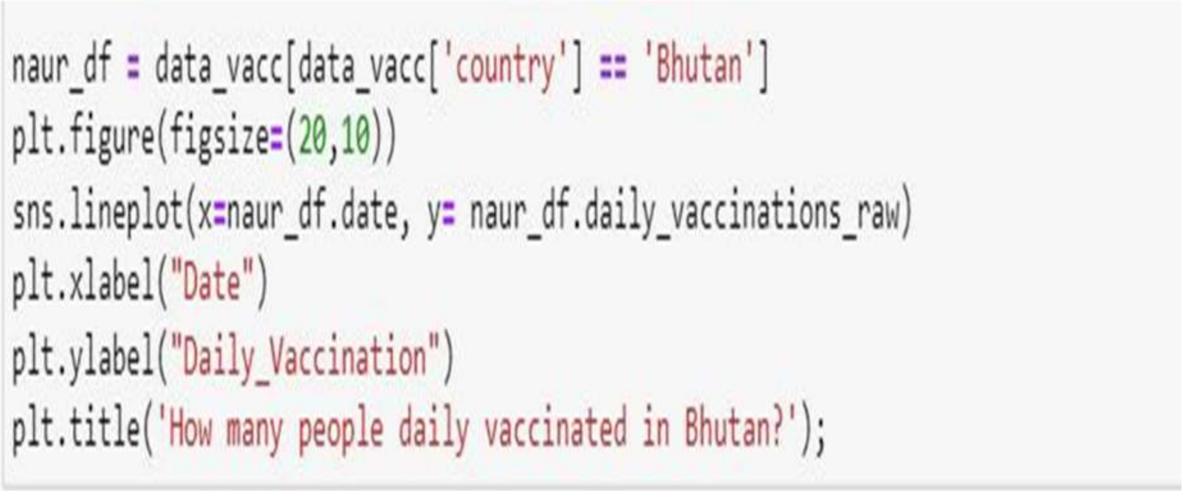
3.6 Explore the daily covid vaccine doses administered per million people?



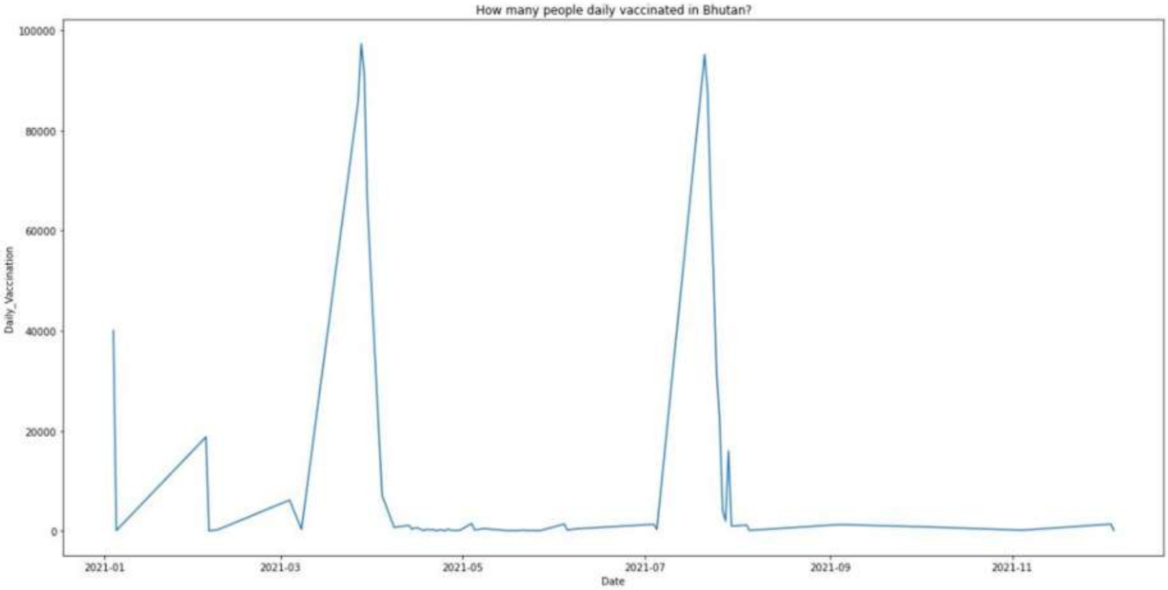
```
plt.figure(figsize=(12,8))
ax = sns.barplot(x=daily_vaccinations_per_million, y=daily_vaccinations_per_million.index )
plt.xlabel("daily vaccinations per million")
plt.ylabel("Country")
plt.title("Daily COVID-19 vaccine doses administered per million people");

for patch in ax.patches:
    width = patch.get_width()
    height = patch.get_height()
    x = patch.get_x()
    y = patch.get_y()

    plt.text(width + x, height + y, '{:.1f} '.format(width))
```



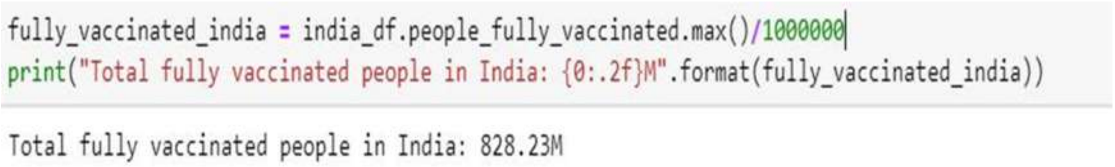
3.7 How many people daily vaccinated in Bhutan?



```
naur_df = data_vacc[data_vacc['country'] == 'Bhutan']
plt.figure(figsize=(20,10))
sns.lineplot(x=naur_df.date, y= naur_df.daily_vaccinations_raw)
plt.xlabel("Date")
plt.ylabel("Daily_Vaccination")
plt.title('How many people daily vaccinated in Bhutan?');
```



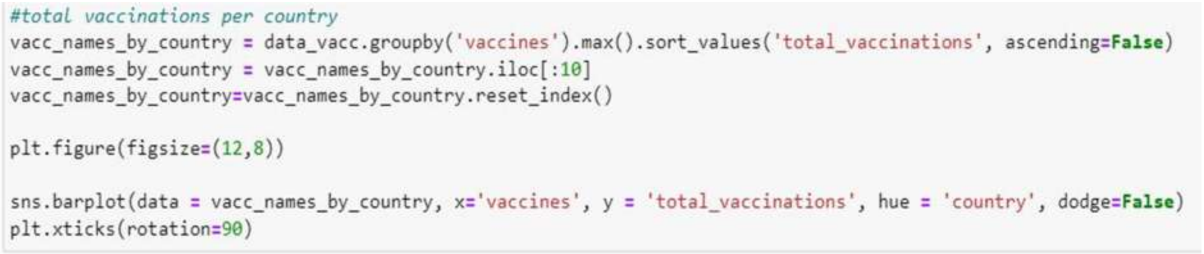
3.8 How many people are fully vaccinated in India?



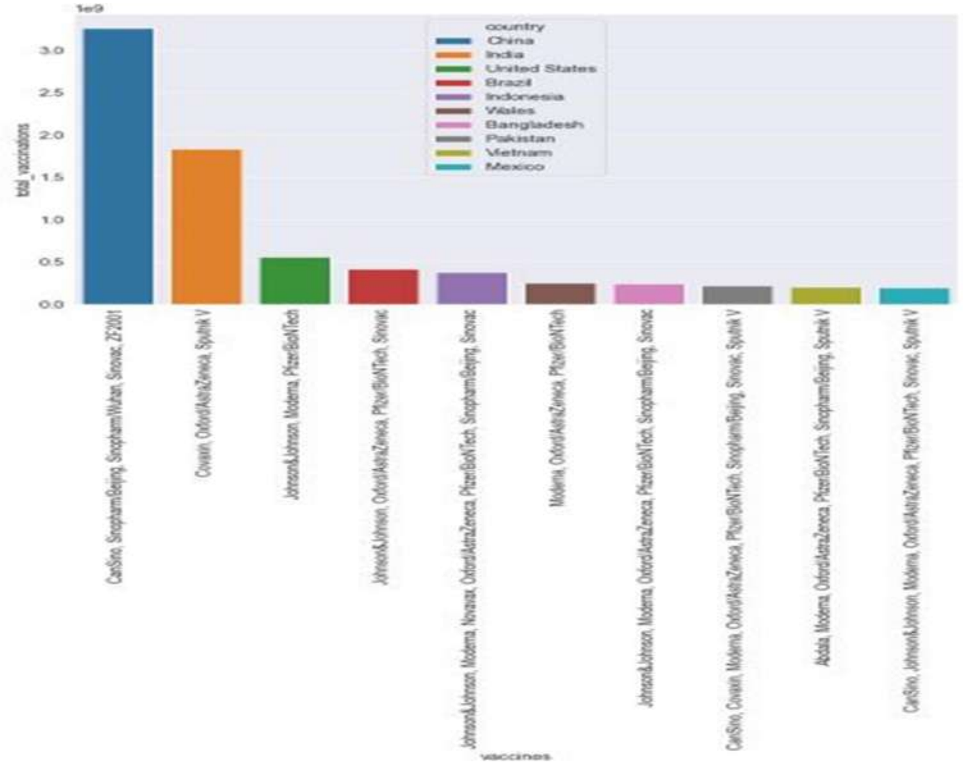
```
fully_vaccinated_india = india_df.people_fully_vaccinated.max()/1000000
print("Total fully vaccinated people in India: {0:.2f}M".format(fully_vaccinated_india))
```

```
Total fully vaccinated people in India: 828.23M
```

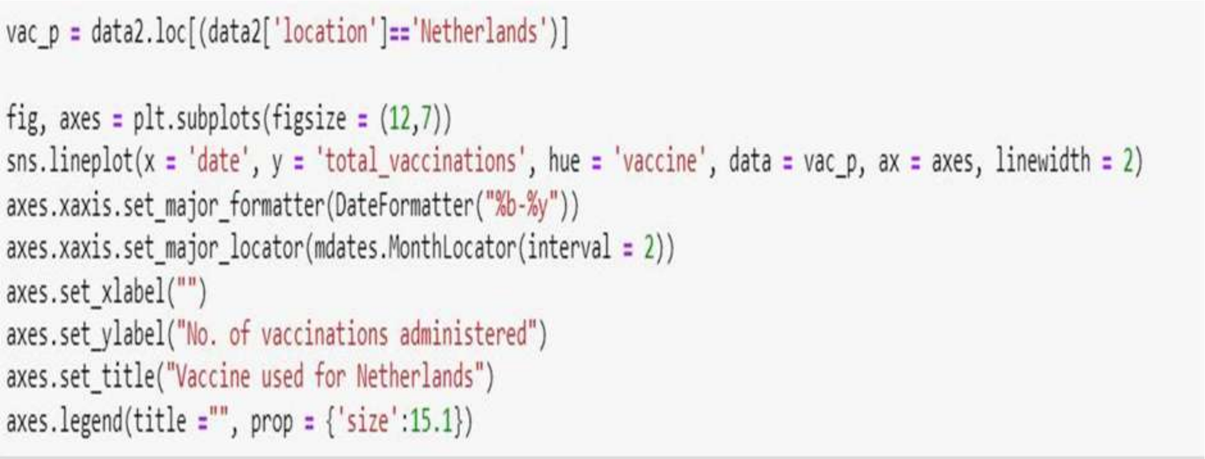
3.9 Explore the total vaccinations per country?



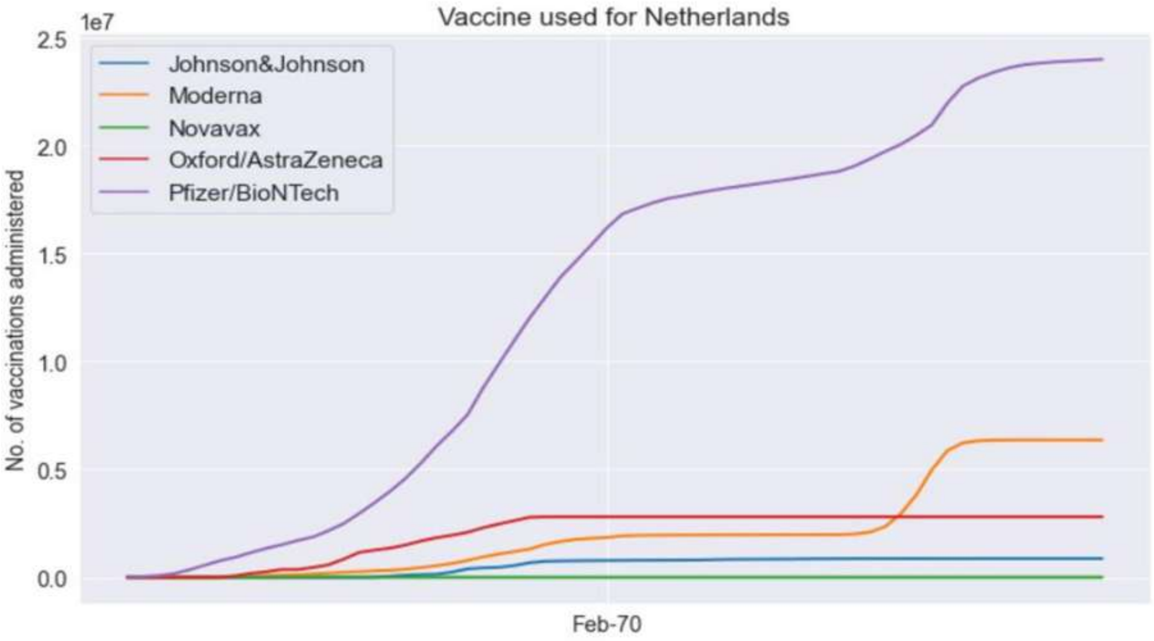
```
#total vaccinations per country
vacc_names_by_country = data_vacc.groupby('vaccines').max().sort_values('total_vaccinations', ascending=False)
vacc_names_by_country = vacc_names_by_country.iloc[:10]
vacc_names_by_country=vacc_names_by_country.reset_index()

plt.figure(figsize=(12,8))

sns.barplot(data = vacc_names_by_country, x='vaccines', y = 'total_vaccinations', hue = 'country', dodge=False)
plt.xticks(rotation=90)
```



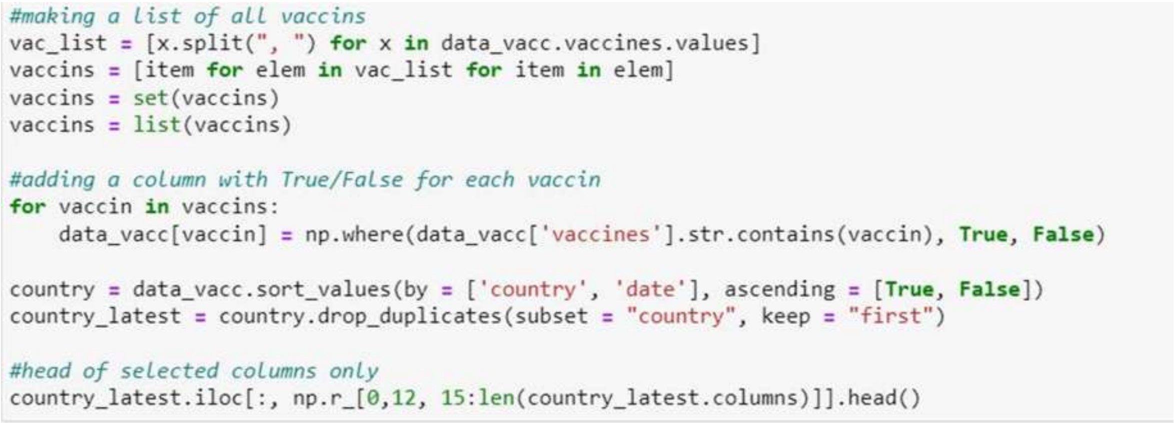
3.10 What are the vaccines used for Netherlands?



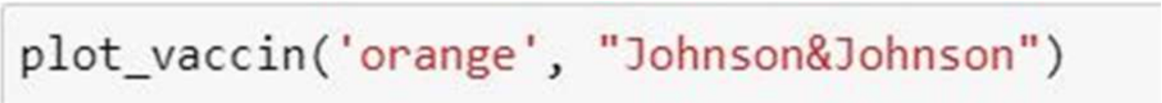
```
vac_p = data2.loc[(data2['location']=='Netherlands')]

fig, axes = plt.subplots(figsize = (12,7))
sns.lineplot(x = 'date', y = 'total_vaccinations', hue = 'vaccine', data = vac_p, ax = axes, linewidth = 2)
axes.xaxis.set_major_formatter(DateFormatter("%b-%y"))
axes.xaxis.set_major_locator(mdates.MonthLocator(interval = 2))
axes.set_xlabel("")
axes.set_ylabel("No. of vaccinations administered")
axes.set_title("Vaccine used for Netherlands")
axes.legend(title ="", prop = {'size':15.1})
```



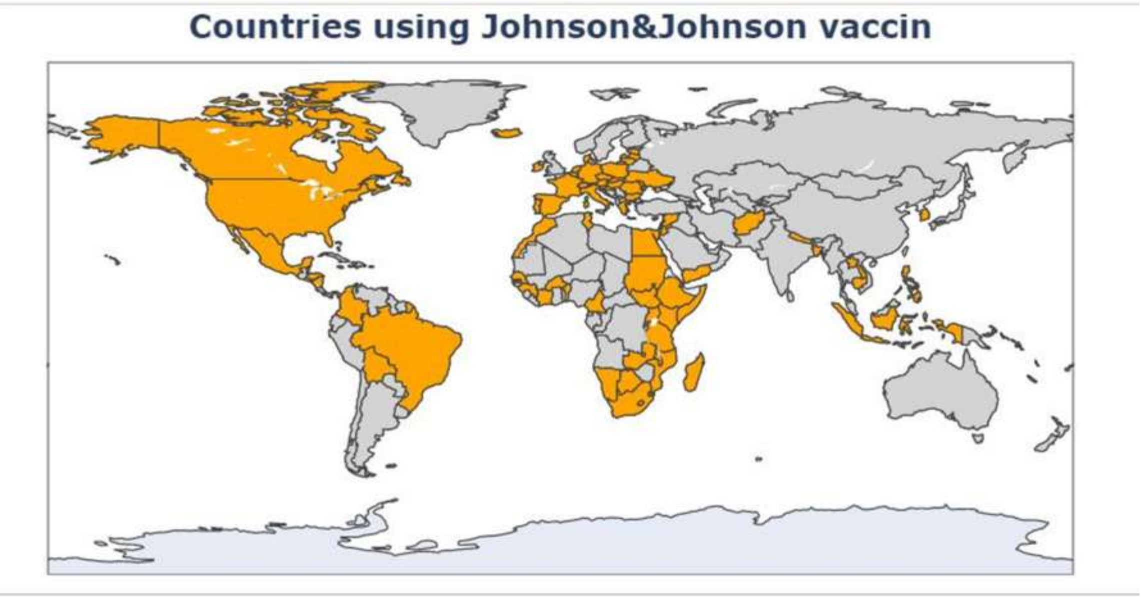
3.11 Which Countries use Johnson&Johnson vaccine?



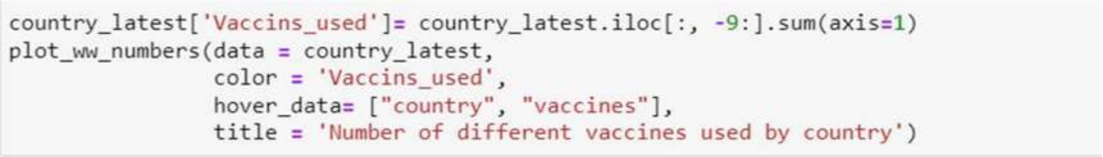
```
#making a list of all vaccins
vac_list = [x.split(", ") for x in data_vacc.vaccines.values]
vaccins = [item for elem in vac_list for item in elem]
vaccins = set(vaccins)
vaccins = list(vaccins)

#adding a column with True/False for each vaccin
for vaccin in vaccins:
    data_vacc[vaccin] = np.where(data_vacc['vaccines'].str.contains(vaccin), True, False)

country = data_vacc.sort_values(by = ['country', 'date'], ascending = [True, False])
country_latest = country.drop_duplicates(subset = "country", keep = "first")

#head of selected columns only
country_latest.iloc[:, np.r_[0,12, 15:len(country_latest.columns)]].head()
```



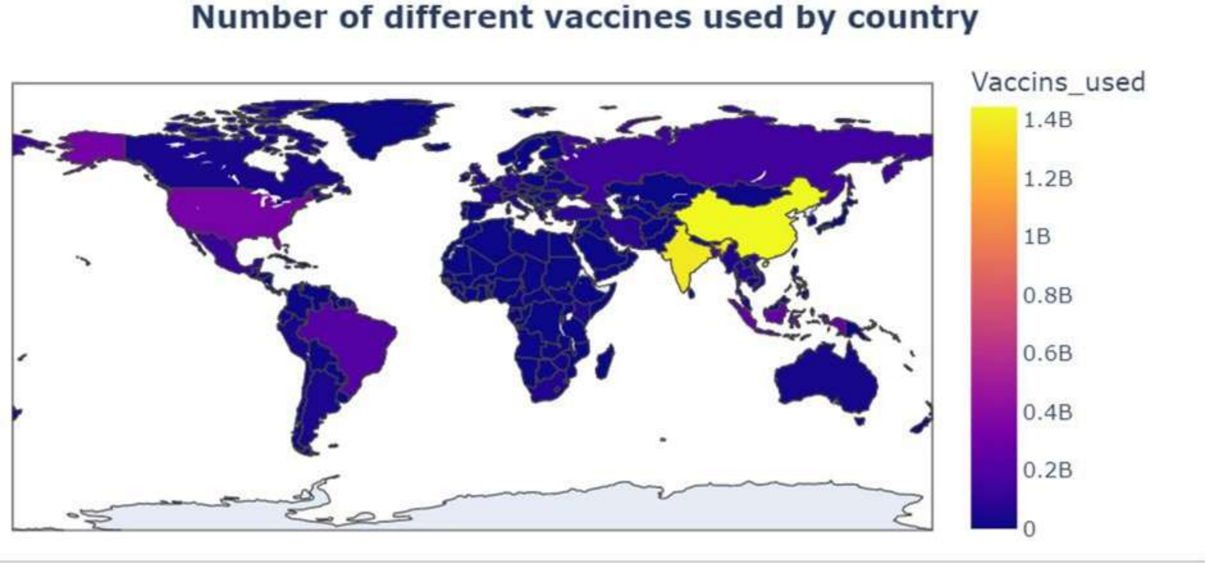
```
plot_vaccin('orange', "Johnson&Johnson")
```



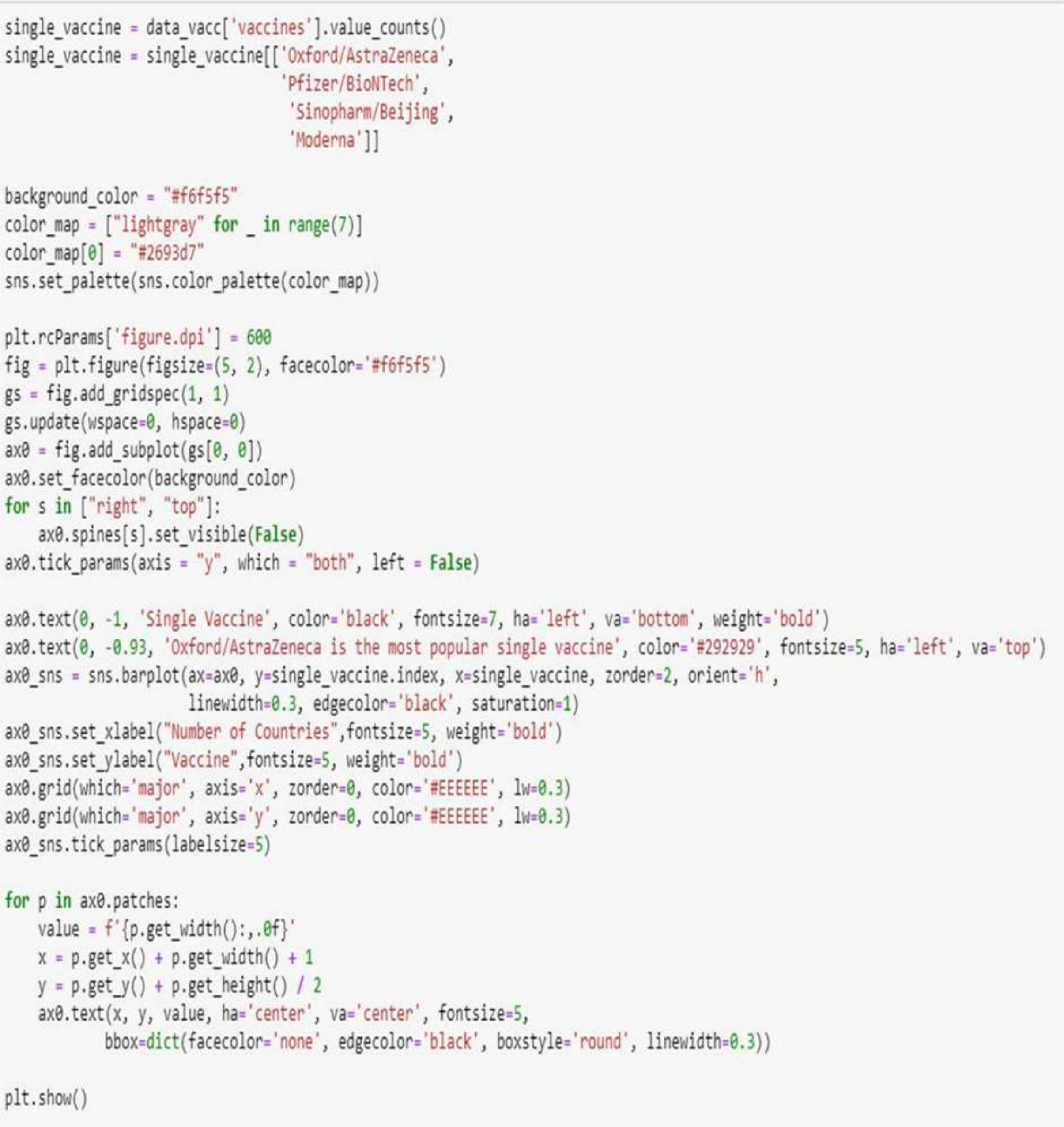
3.12 Plot the number of vaccines used by countries?



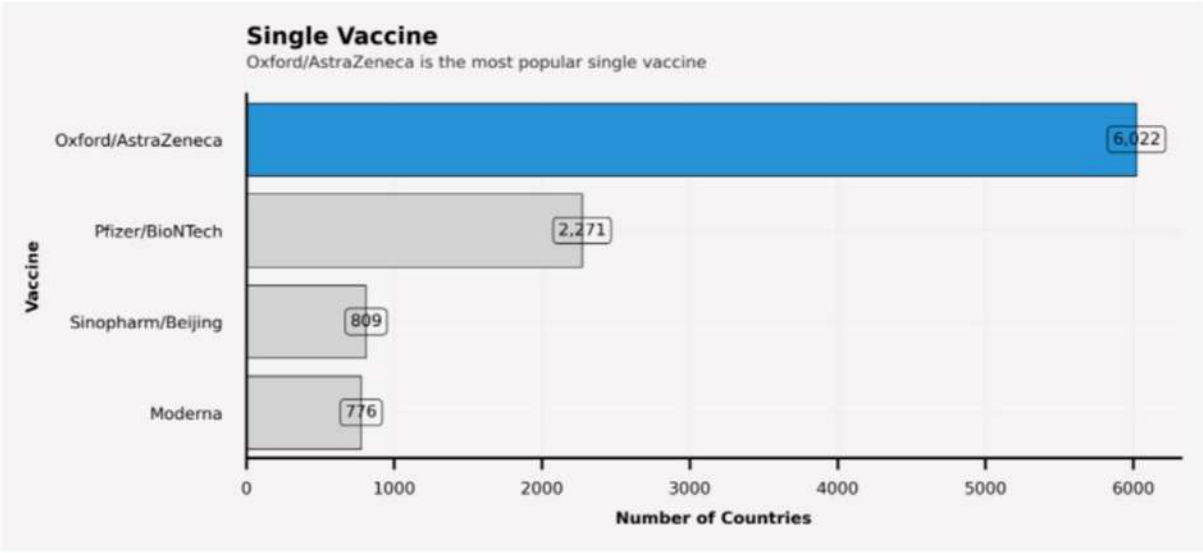
```
country_latest['Vaccins_used']= country_latest.iloc[:, -9:].sum(axis=1)
plot_ww_numbers(data = country_latest,
                color = 'Vaccins_used',
                hover_data= ["country", "vaccines"],
                title = 'Number of different vaccines used by country')
```



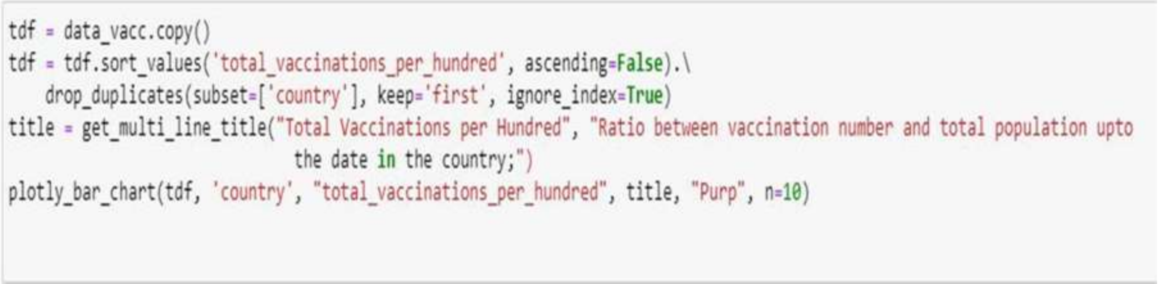
3.13 Explore the countries which use only single vaccine?



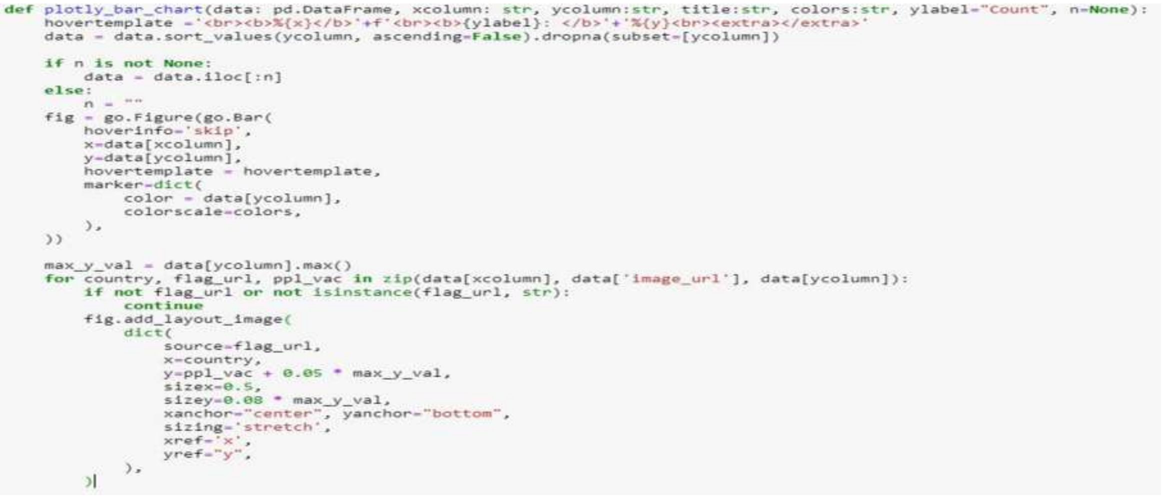
```
single_vaccine = data_vacc['vaccines'].value_counts()
single_vaccine = single_vaccine[['Oxford/AstraZeneca',
                                'Pfizer/BioNTech',
                                 'Sinopharm/Beijing',
                                 'Moderna']]

background_color = "#f6f5f5"
color_map = ["lightgray" for _ in range(7)]
color_map[0] = "#2693d7"
sns.set_palette(sns.color_palette(color_map))

plt.rcParams['figure.dpi'] = 600
fig = plt.figure(figsize=(5, 2), facecolor='#f6f5f5')
gs = fig.add_gridspec(1, 1)
gs.update(wspace=0, hspace=0)
ax0 = fig.add_subplot(gs[0, 0])
ax0.set_facecolor(background_color)
for s in ["right", "top"]:
    ax0.spines[s].set_visible(False)
ax0.tick_params(axis = "y", which = "both", left = False)

ax0.text(0, -1, 'Single Vaccine', color='black', fontsize=7, ha='left', va='bottom', weight='bold')
ax0.text(0, -0.93, 'Oxford/AstraZeneca is the most popular single vaccine', color='#292929', fontsize=5, ha='left', va='top')
ax0_sns = sns.barplot(ax=ax0, y=single_vaccine.index, x=single_vaccine, zorder=2, orient='h',
                      linewidth=0.3, edgecolor='black', saturation=1)
ax0_sns.set_xlabel("Number of Countries",fontsize=5, weight='bold')
ax0_sns.set_ylabel("Vaccine",fontsize=5, weight='bold')
ax0.grid(which='major', axis='x', zorder=0, color='#EEEEEE', lw=0.3)
ax0.grid(which='major', axis='y', zorder=0, color='#EEEEEE', lw=0.3)
ax0_sns.tick_params(labelsize=5)

for p in ax0.patches:
    value = f'{p.get_width():,.0f}'
    x = p.get_x() + p.get_width() + 1
    y = p.get_y() + p.get_height() / 2
    ax0.text(x, y, value, ha='center', va='center', fontsize=5,
            bbox=dict(facecolor='none', edgecolor='black', boxstyle='round', linewidth=0.3))

plt.show()
```



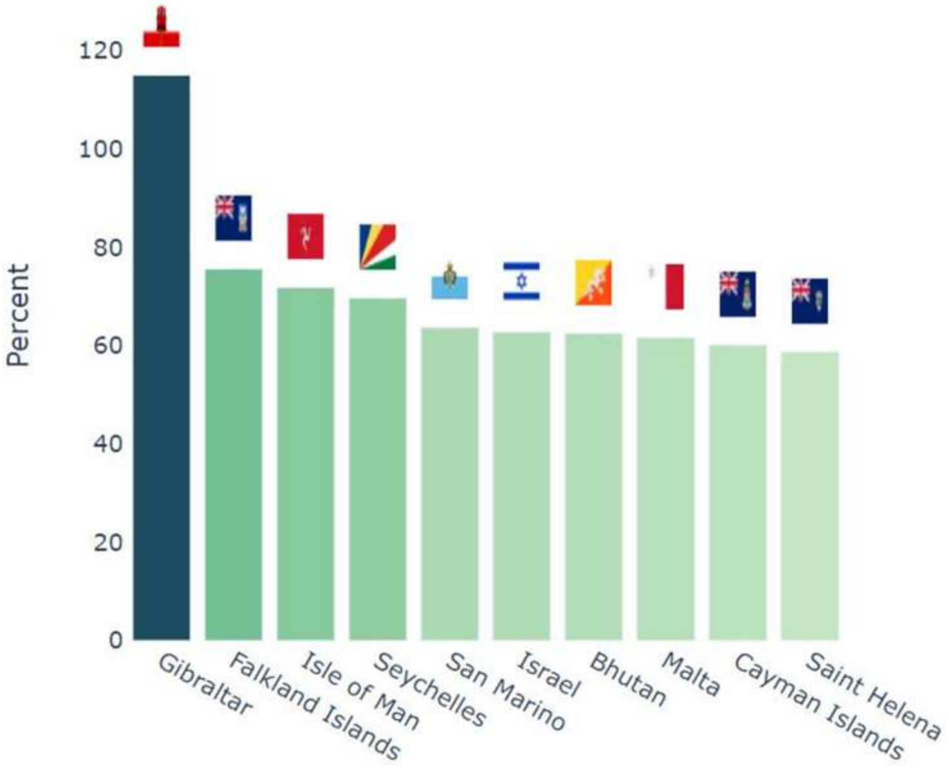
3.14 How many percentage of individuals vaccinated?



```
tdf = data_vacc.copy()
tdf = tdf.sort_values('total_vaccinations_per_hundred', ascending=False).\
    drop_duplicates(subset=['country'], keep='first', ignore_index=True)
title = get_multi_line_title("Total Vaccinations per Hundred", "Ratio between vaccination number and total population upto
                             the date in the country;")
plotly_bar_chart(tdf, 'country', "total_vaccinations_per_hundred", title, "Purp", n=10)
```



```
def plotly_bar_chart(data: pd.DataFrame, xcolumn: str, ycolumn:str, title:str, colors:str, ylabel="Count", n=None):
    hovertemplate ='<br><b>%{x}</b>'+f'<br><b>{ylabel}: </b>'+'%{y}<br><extra></extra>'
    data = data.sort_values(ycolumn, ascending=False).dropna(subset=[ycolumn])

    if n is not None:
        data = data.iloc[:n]
    else:
        n = ""
    fig = go.Figure(go.Bar(
        hoverinfo='skip',
        x=data[xcolumn],
        y=data[ycolumn],
        hovertemplate = hovertemplate,
        marker=dict(
            color = data[ycolumn],
            colorscale=colors,
        ),
    ))

    max_y_val = data[ycolumn].max()
    for country, flag_url, ppl_vac in zip(data[xcolumn], data['image_url'], data[ycolumn]):
        if not flag_url or not isinstance(flag_url, str):
            continue
        fig.add_layout_image(
            dict(
                source=flag_url,
                x=country,
                y=ppl_vac + 0.05 * max_y_val,
                sizex=0.5,
                sizey=0.08 * max_y_val,
                xanchor="center", yanchor="bottom",
                sizing='stretch',
                xref='x',
                yref="y",
            ),
        )
```



4. CONCLUSION

The above graphs shows how slowly but surely, the vaccines are being administered in increasingly large numbers each day. If we look carefully, we can also identify a slight downward trend in the number of new cases each day, as the vaccinations progress. Humanity is on its way to victory!

Humanity today is actively confronting the global pandemic. Despite the active increase in infection in early 2020, mankind has managed to develop a weapon against the virus and reduce the number of cases and deaths around the world.

The vaccine has slowed the spread of the infection, reducing the trend of an increase in the number of cases, but the virus continues to resist. Thus, it is too early to talk about the end of the battle, since no one knows what new blow the darkness will inflict.

COVID-19 has taken a heavy toll on mankind. We have lost far too many people and suffered too much for too long. Now is the time to fight back. Regardless of what people might say, always wear a mask when out in public and maintain social distancing. DO NOT give in hearsay!

5. REFRENCES

- https://ourworldindata.org/covid-vaccinations
- https://github.com/owid/covid-19-data/blob/master/public/data/vaccinations/vaccinations.csv
- https://www.analyticsvidhya.com/blog/2021/08/understanding-bar-plots-in-python-beginners-guide-to-data-visualization/
- https://www.geeksforgeeks.org/plotting-world-map-using-pygal-in-python/
- https://www.geeksforgeeks.org/plot-a-pie-chart-in-python-using-matplotlib/
- https://seaborn.pydata.org/generated/seaborn.lineplot.html