COVİD 19 WORLD VACCINATION PROGRESS

The data contains the following information:

- **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 vaccin ations** - 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;

Content:

- [Missing Data](#)
- [Data Visualization](#)

In [1]:
```python
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the /python Docker image: https://github.com/docker-python
# For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

```python
# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list all f
iles under the input directory

import matplotlib.pyplot as plt

# plotly
# import plotly.plotly as py
from plotly.offline import init_notebook_mode, iplot, plot
import plotly.express as px
import plotly as py
init_notebook_mode(connected=True)
import plotly.graph_objs as go

from pandas_profiling import ProfileReport
import scipy

# seaborn library
import seaborn as sns

# word cloud library
from wordcloud import WordCloud

import os
for dirname, _, filenames in os.walk('/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/working/) that gets preserved a
s output when you create a version using "Save & Run All"
# You can also write temporary files to /temp/, but they won't be saved outside of th
e current session
```

/input/covid-world-vaccination-progress/country_vaccinations_by_manufacturer.csv
/input/covid-world-vaccination-progress/country_vaccinations.csv

In [2]:
```python
data = pd.read_csv("/input/covid-world-vaccination-progress/country_vaccinations.csv")
data.head()
```

Out[2]:

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
0	Afghanist	AF	2021	0.0	0.0	NaN	NaN	NaN	0.0	0.0	NaN	NaN	Johnson &Johnso	World He	https://covid19.who.i

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
	an	G	-02-22										n, Oxford/AstraZeneca, Pfizer/Bi...	alth Organization	nt/
1	Afghanistan	AFG	2021-02-23	NaN	NaN	NaN	NaN	1367.0	NaN	NaN	NaN	34.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi...	World Health Organization	https://covid19.who.int/
2	Afghanistan	AFG	2021-02-24	NaN	NaN	NaN	NaN	1367.0	NaN	NaN	NaN	34.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi...	World Health Organization	https://covid19.who.int/
3	Afghanist	AFG	2021-	NaN	NaN	NaN	NaN	1367.0	NaN	NaN	NaN	34.0	Johnson &Johnson,	World Health	https://covid19.who.i

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
	an		02-25										Oxford/AstraZeneca, Pfizer/Bi...	h Organization	nt/
4	Afghanistan	AFG	2021-02-26	NaN	NaN	NaN	NaN	1367.0	NaN	NaN	NaN	34.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi...	World Health Organization	https://covid19.who.int/

```
In [3]:
report = ProfileReport(data)
report

Summarize dataset: 100%
106/106 [00:26<00:00, 3.82it/s, Completed]
Generate report structure: 100%
1/1 [00:07<00:00, 7.25s/it]
Render HTML: 100%
1/1 [00:03<00:00, 3.96s/it]
Out[3]:
```

Missing Data

We will fix some shortcomings to make data visualization easier and more understandable.

```
In [4]:
data.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
In [5]:
data.shape

Out[5]:
(86512, 15)
In [6]:
data.isna().sum()

Out[6]:
country                                 0
iso_code                                0
date                                    0
total_vaccinations                  42905
people_vaccinated                   45218
people_fully_vaccinated             47710
daily_vaccinations_raw              51150
daily_vaccinations                    299
total_vaccinations_per_hundred      42905
people_vaccinated_per_hundred       45218
people_fully_vaccinated_per_hundred 47710
daily_vaccinations_per_million        299
vaccines                                0
source_name                             0
source_website                          0
dtype: int64
```

As can be seen, there is quite much missing data.

Drop the total_vaccinations column from these deficiencies first

```
In [7]:
data = data.drop(data[data.total_vaccinations.isna()].index)
```

```
In [8]:
data.isna().sum()
```

```
Out[8]:
country                                  0
iso_code                                 0
date                                     0
total_vaccinations                       0
people_vaccinated                     2717
people_fully_vaccinated               5097
daily_vaccinations_raw                8245
daily_vaccinations                     223
total_vaccinations_per_hundred           0
people_vaccinated_per_hundred         2717
people_fully_vaccinated_per_hundred   5097
daily_vaccinations_per_million         223
vaccines                                 0
source_name                              0
source_website                           0
dtype: int64
```

As you can see the missing data in the total_vaccinations column has been removed.

Now let's remove the missing data from the people_vaccinated column

```
In [9]:
check_data = data.drop(data[data.people_vaccinated.isna()].index)
```

```
In [10]:
check_data.head()
```

Out[10]:

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
0	Afghanistan	AFG	2021-02-22	0.0	0.0	NaN	NaN	NaN	0.00	0.00	NaN	NaN	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi..	World Health Organization	https://covid19.who.int/

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
													.		
6	Afghanistan	AFG	2021-02-28	8200.0	8200.0	NaN	NaN	1367.0	0.02	0.02	NaN	34.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi..	World Health Organization	https://covid19.who.int/
22	Afghanistan	AFG	2021-03-16	54000.0	54000.0	NaN	NaN	2862.0	0.14	0.14	NaN	72.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi..	World Health Organization	https://covid19.who.int/
44	Afghanistan	AFG	2021-04-07	120000.0	120000.0	NaN	NaN	3000.0	0.30	0.30	NaN	75.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi..	World Health Organization	https://covid19.who.int/

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
													.		
59	Afghanistan	AFG	2021-04-22	240000.0	240000.0	NaN	NaN	8000.0	0.60	0.60	NaN	201.0	Johnson &Johnson, Oxford/AstraZeneca, Pfizer/Bi..	World Health Organization	https://covid19.who.int/

As you can see the missing data in the people_vaccinated column has been removed.

Let's look at the values between the columns by looking at the correlation map

```
In [11]:
plt.subplots(figsize = (10,10))
sns.heatmap(data.corr(), annot = True, square = True)
plt.show()
```

people_vaccinated and people_vaccinated_per_hundred

The data of the total_vaccinations column and the people_vaccinated column look almost the same.

As can bee seen from the heatmap, these features have almost ideal correlation.

```
In [12]:
diff = check_data.total_vaccinations.mean() - check_data.people_vaccinated.mean()
diff_per_hundred = check_data.total_vaccinations_per_hundred.mean() - check_data.people_vaccinated_per_hundred.mean()

data.people_vaccinated = data.people_vaccinated.fillna(data.total_vaccinations - diff)
data.people_vaccinated_per_hundred = data.people_vaccinated_per_hundred.fillna(data.total_vaccinations_per_hundred - diff_per_hundred)
```

Let's check if everything ok.

```
In [13]:
```

```
data.isna().sum()
```

```
Out[13]:
country                                   0
iso_code                                  0
date                                      0
total_vaccinations                        0
people_vaccinated                         0
people_fully_vaccinated                5097
daily_vaccinations_raw                 8245
daily_vaccinations                      223
total_vaccinations_per_hundred            0
people_vaccinated_per_hundred             0
people_fully_vaccinated_per_hundred    5097
daily_vaccinations_per_million          223
vaccines                                  0
source_name                               0
source_website                            0
dtype: int64
```

daily_vaccinations_raw and daily_vaccinations

The data of the daily_vaccinations column and the daily_vaccinations_raw column look almost the same.

As can bee seen from the heatmap, these features have almost ideal correlation.

```
In [14]:
diff = check_data.daily_vaccinations.mean() - check_data.daily_vaccinations_raw.mean()

data.daily_vaccinations_raw = data.daily_vaccinations_raw.fillna(data.daily_vaccinati
ons - diff)
```

```
In [15]:
data.isna().sum()
```

```
Out[15]:
country                                   0
iso_code                                  0
date                                      0
total_vaccinations                        0
people_vaccinated                         0
people_fully_vaccinated                5097
daily_vaccinations_raw                  223
daily_vaccinations                      223
total_vaccinations_per_hundred            0
people_vaccinated_per_hundred             0
people_fully_vaccinated_per_hundred    5097
daily_vaccinations_per_million          223
vaccines                                  0
source_name                               0
source_website                            0
dtype: int64
```

```
In [16]:
data.daily_vaccinations = data.daily_vaccinations.fillna(0)
```

```
data.daily_vaccinations_raw = data.daily_vaccinations_raw.fillna(0)
```

In [17]:
```
data.isna().sum()
```

Out[17]:
```
country                                  0
iso_code                                 0
date                                     0
total_vaccinations                       0
people_vaccinated                        0
people_fully_vaccinated               5097
daily_vaccinations_raw                   0
daily_vaccinations                       0
total_vaccinations_per_hundred           0
people_vaccinated_per_hundred            0
people_fully_vaccinated_per_hundred   5097
daily_vaccinations_per_million         223
vaccines                                 0
source_name                              0
source_website                           0
dtype: int64
```
The data of the total_vaccinations column and the people_fully_vaccinated column look almost the same.

As can bee seen from the heatmap, these features have almost ideal correlation.

people_fully_vaccinated

In [18]:
```
diff = check_data.total_vaccinations.mean() - check_data.people_fully_vaccinated.mean
()

data.people_fully_vaccinated = data.people_fully_vaccinated.fillna(data.total_vaccina
tions - diff)
```

In [19]:
```
data.isna().sum()
```

Out[19]:
```
country                                  0
iso_code                                 0
date                                     0
total_vaccinations                       0
people_vaccinated                        0
people_fully_vaccinated                  0
daily_vaccinations_raw                   0
daily_vaccinations                       0
total_vaccinations_per_hundred           0
people_vaccinated_per_hundred            0
people_fully_vaccinated_per_hundred   5097
daily_vaccinations_per_million         223
vaccines                                 0
source_name                              0
```

```
source_website                         0
dtype: int64
```

people_fully_vaccinated_per_hundred

The data of the total_vaccinations_per_hundred column and the people_fully_vaccinated_per_hundred column look almost the same.

As can bee seen from the heatmap, these features have almost ideal correlation.

```
In [20]:
diff = check_data.total_vaccinations_per_hundred.mean() - check_data.people_fully_vac
cinated_per_hundred.mean()

data.people_fully_vaccinated_per_hundred = data.people_fully_vaccinated_per_hundred.f
illna(data.total_vaccinations_per_hundred - diff)

In [21]:
data.isna().sum()

Out[21]:
country                                0
iso_code                               0
date                                   0
total_vaccinations                     0
people_vaccinated                      0
people_fully_vaccinated                0
daily_vaccinations_raw                 0
daily_vaccinations                     0
total_vaccinations_per_hundred         0
people_vaccinated_per_hundred          0
people_fully_vaccinated_per_hundred    0
daily_vaccinations_per_million       223
vaccines                               0
source_name                            0
source_website                         0
dtype: int64
```

Since there is not much similarity between them and the others in the daily_vaccinations_per_million correlation map, we will assign the value 0 instead of the missing data.

```
In [22]:
data.daily_vaccinations_per_million = data.daily_vaccinations_per_million.fillna(0)

In [23]:
data.isna().sum()

Out[23]:
country                    0
iso_code                   0
date                       0
total_vaccinations         0
people_vaccinated          0
people_fully_vaccinated    0
daily_vaccinations_raw     0
```

```
daily_vaccinations                        0
total_vaccinations_per_hundred            0
people_vaccinated_per_hundred             0
people_fully_vaccinated_per_hundred       0
daily_vaccinations_per_million            0
vaccines                                  0
source_name                               0
source_website                            0
dtype: int64
```

There is no missing data in our columns.

Our missing data has been cleared.

Everything worked fine

iso_code

Let's see if there is any missing data in iso_code

```
In [24]:
data[data.iso_code.isna()].country.unique()
```

```
Out[24]:
array([], dtype=object)
```

Thats the iso-codes which are used for these countries : GB-ENG for England, NC for Northern Cyprus, GB-NIR for Northern Ireland, GB-SCT for Scotland, GB-WLS for Wales.

We will fill missing iso-codes with appropriate ones.

```
In [25]:
data[data.country == 'England'] = data[data.country == 'England'].fillna('GB-ENG')
data[data.country == 'Northern Ireland'] == data[data.country == 'Northern Ireland'].
fillna('GB-NIR')
data[data.country == 'Scotland'] = data[data.country == 'Scotland'].fillna('GB-SCT')
data[data.country == 'Wales'] = data[data.country == 'Wales'].fillna('GB-WLS')
data = data.fillna('NC')
```

```
In [26]:
data.isna().sum()
```

```
Out[26]:
country                                   0
iso_code                                  0
date                                      0
total_vaccinations                        0
people_vaccinated                         0
people_fully_vaccinated                   0
daily_vaccinations_raw                    0
daily_vaccinations                        0
total_vaccinations_per_hundred            0
people_vaccinated_per_hundred             0
people_fully_vaccinated_per_hundred       0
daily_vaccinations_per_million            0
vaccines                                  0
```

```
source_name                    0
source_website                 0
dtype: int64
```
Finally we managed to organize the lost data

Lets transform date column

```
In [27]:
data["date"] = pd.to_datetime(data["date"])
data = data.sort_values("date", ascending = True )
data["date"] = data["date"].dt.strftime("%Y-%m-%d")

In [28]:
unique_dates = data["date"].unique()

In [29]:
data.head()

Out[29]:
```

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
58517	Norway	NOR	2020-12-02	0.0	0.0	-1.834694e+07	0.0	0.0	0.0	0.0	-44.719168	0.0	Moderna, Pfizer/BioNTech	Norwegian Institute of Public Health	https://github.com/folkehelseinstituttet/surve...
58518	Norway	NOR	2020-12-0	0.0	0.0	-1.834694e+07	0.0	0.0	0.0	0.0	-44.719168	0.0	Moderna, Pfizer/BioNTech	Norwegian Institute	https://github.com/folkehelseinstituttet/surve...

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
			3											of Public Health	
43117	Latvia	LVA	2020-12-04	1.0	1.0	-1.834693e+07	0.0	0.0	0.0	0.0	-44.719168	0.0	Johnson&Johnson, Moderna, Novavax, Pfizer/BioN...	National Health Service	https://data.gov.lv/dati/eng/dataset/covid19-v...
58519	Norway	NOR	2020-12-04	0.0	0.0	-1.834694e+07	0.0	0.0	0.0	0.0	-44.719168	0.0	Moderna, Pfizer/BioNTech	Norwegian Institute of Public Health	https://github.com/folkehelseinstituttet/surve...

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	daily_vaccinations_per_million	vaccines	source_name	source_website
585820	Norway	NOR	2020-12-05	0.0	0.0	-1.834694e+07	0.0	0.0	0.0	0.0	-44.719168	0.0	Moderna, Pfizer/BioNTech	Norwegian Institute of Public Health	https://github.com/folkehelseinstituttet/surve...

our dates are listed

Data Visualization

First, let's watch total_vaccinations and daily_vaccinations animatedly on the world map.

Let's take a look at the rate of vaccination in countries

```
In [30]:
fig = px.choropleth(
    data,
    locations="iso_code",
    color="total_vaccinations",
    title='Number of people vaccinated',
    color_continuous_scale='viridis',
    animation_frame="date",
    projection = "natural earth",
    range_color = [0,5000000],

)
```
date=2020-12-022020-12-022021-03-292021-07-242021-11-182022-03-1501M2M3M4M5Mtotal_vaccinationsNumber of people vaccinated▶■
Animated world map by date of total vaccinations by country

In [31]:
```
fig = px.choropleth(
    data,
    locations= "iso_code",
    color = "daily_vaccinations",
    animation_frame = "date",
    color_continuous_scale= "viridis",
    projection= "natural earth",
    range_color= [0,1000000] ,
    title = "Number of daily vaccinations"
)
fig.show()
```
date=2020-12-022020-12-022021-03-292021-07-242021-11-182022-03-
1500.2M0.4M0.6M0.8M1Mdaily_vaccinationsNumber of daily vaccinations▶■

Animated world map by date of daily vaccinations by country

First we will create a new table by selecting the columns we will use.

In [32]:
```
columns = ["country","iso_code","total_vaccinations","people_vaccinated","total_vacci
nations_per_hundred","vaccines","daily_vaccinations"]
vacc_data = data[columns].groupby("country").max().sort_values("total_vaccinations",
ascending = True)
```

In [33]:
```
vacc_data.head()
```

Out[33]:

	iso_code	total_vaccinations	people_vaccinated	total_vaccinations_per_hundred	vaccines	daily_vaccinations
country						
Pitcairn	PCN	94.0	47.0	200.00	Oxford/AstraZeneca	1.0
Tokelau	TKL	1936.0	968.0	141.52	Pfizer/BioNTech	23.0
Niue	NIU	4161.0	1650.0	257.81	Pfizer/BioNTech	87.0
Montserrat	MSR	4211.0	1897.0	84.54	Oxford/AstraZeneca	53.0

	iso_code	total_vaccinations	people_vaccinated	total_vaccinations_per_hundred	vaccines	daily_vaccinations
country						
Falkland Islands	FLK	4407.0	2632.0	124.91	Oxford/AstraZeneca	189.0

```
In [34]:
vacc_data.info()

<class 'pandas.core.frame.DataFrame'>
Index: 223 entries, Pitcairn to China
Data columns (total 6 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   iso_code                        223 non-null    object
 1   total_vaccinations              223 non-null    float64
 2   people_vaccinated               223 non-null    float64
 3   total_vaccinations_per_hundred  223 non-null    float64
 4   vaccines                        223 non-null    object
 5   daily_vaccinations              223 non-null    float64
dtypes: float64(4), object(2)
memory usage: 12.2+ KB
```

```
In [35]:
fig = px.choropleth(
    vacc_data,
    locations= "iso_code",
    color = "total_vaccinations_per_hundred",
    title = "Number of total vaccinations per hunderd",
    color_continuous_scale= "rainbow"
)
fig.show("notebook")
```

50100150200250300total_vaccinations_per_hundredNumber of total vaccinations per hunderd

As can be seen on the map, countries have vaccination percentages.

Let's draw a map to see which countries these vaccines are used in.

```
In [36]:
fig = px.choropleth(
    locations = vacc_data.iso_code,
    color = vacc_data.vaccines,
    title = "name of the vaccine",
    color_continuous_scale= "rainbow"
)
```

colorOxford/AstraZenecaPfizer/BioNTechSinopharm/BeijingModernaOxford/AstraZeneca, Pfizer/BioNTechModerna, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZenecaModerna, Oxford/AstraZeneca, Pfizer/BioNTechOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingPfizer/BioNTech, Sputnik VJohnson&Johnson, Moderna, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTechOxford/AstraZeneca, Pfizer/BioNTech, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VOxford/AstraZeneca, Sinopharm/Beijing, Sputnik VJohnson&Johnson, ModernaJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/BeijingJohnson&Johnson, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingPfizer/BioNTech, Sinopharm/Beijing, Sputnik VOxford/AstraZeneca, Sinopharm/BeijingOxford/AstraZeneca, Pfizer/BioNTech, SinovacCovaxin, Oxford/AstraZeneca, Sinopharm/BeijingOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, SinovacModerna, Oxford/AstraZeneca, Sinopharm/Beijing, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VCovaxin, Oxford/AstraZenecaJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingPfizer/BioNTech, Sinopharm/BeijingOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VModerna, Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, SinovacCovaxin, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Moderna, Novavax, Pfizer/BioNTechSinopharm/Beijing, Sputnik VJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VJohnson&Johnson, Pfizer/BioNTech, Sinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VEpiVacCorona, Oxford/AstraZeneca, QazVac, Sinopharm/Beijing, Sputnik V, ZF2001Covaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VOxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik VAbdala, Johnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Soberana02, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VOxford/AstraZeneca, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VModerna, Pfizer/BioNTech, Sinopharm/Beijing, SinovacPfizer/BioNTech, SinovacModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacQazVac, Sinopharm/Beijing, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinopharm/Wuhan, Sputnik

VCovaxin, Johnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, SinovacCanSino, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacAbdala, Soberana Plus, Soberana02Abdala, Sinopharm/Beijing, Sinovac, Soberana02, Sputnik Light, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik Light, Sputnik V, ZF2001Medigen, Moderna, Oxford/AstraZeneca, Pfizer/BioNTechCanSino, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCanSino, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCOVIran Barekat, Covaxin, FAKHRAVAC, Oxford/AstraZeneca, Razi Cov Pars, Sinopharm/Beijing, Soberana02, SpikoGen, Sputnik VPfizer/BioNTech, Sinovac, TurkovacEpiVacCorona, Sputnik VCanSino, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VAbdala, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VCanSino, Covaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCovaxin, Oxford/AstraZeneca, Sputnik VCanSino, Sinopharm/Beijing, Sinopharm/Wuhan, Sinovac, ZF2001name of the vaccine

Vaccine types can be seen on the sides according to the colors of the countries. By looking at this map, it can be seen which country has which vaccine.

How many people have been vaccinated

First, let's look at the statistics of the countries, then let's show these countries on the world map.

```
In [37]:
vacc_country = data.groupby(["country", "iso_code", "vaccines"])['total_vaccinations',
                                                      'total_vaccina
tions_per_hundred',
                                                      'daily_vaccinat
ions',
                                                      'daily_vaccinat
ions_per_million',
                                                      'people_vaccina
ted',
                                                      'people_vaccina
ted_per_hundred',
                                                       'people_fully_
vaccinated', 'people_fully_vaccinated_per_hundred'
                                                    ].max().reset_i
ndex()
/opt/conda/lib/python3.7/site-packages/ipykernel_launcher.py:1: FutureWarning:

Indexing with multiple keys (implicitly converted to a tuple of keys) will be dep
recated, use a list instead.


In [38]:
def trace_bar(data, feature, title, xlab, ylab,color):
    data = data.sort_values(feature, ascending=False)
    trace = go.Bar(
```

```python
        x = data['country'],
        y = data[feature],
        marker=dict(color=color),
        text=data['country']
    )
data = [trace]

layout = dict(title = title,
        xaxis = dict(
            title = xlab,
            showticklabels=True,
            tickangle=45,
            zeroline=True,
            zerolinewidth=1,
            zerolinecolor='grey',
            showline=True,
            linewidth=2,
            linecolor='black',
            mirror=True,
            tickfont=dict(
                    size=10,
                    color='black'),),


        yaxis = dict(
            title = ylab,
            gridcolor='lightgrey',
            zeroline=True,
            zerolinewidth=1,
            zerolinecolor='grey',
            showline=True,
            linewidth=2,
            linecolor='black',
            mirror=True),

        plot_bgcolor = 'rgba(0, 0, 0, 0)',
        paper_bgcolor = 'rgba(0, 0, 0, 0)',
        hovermode = 'closest'
            )
fig = dict(data = data, layout = layout)
iplot(fig)
```

In [39]:
```python
trace_bar(vacc_country, 'total_vaccinations', 'Vaccination total per country', 'Count
ry', 'Vaccination total', "purple" )
```

ChinaIndiaUnited
StatesBrazilIndonesiaJapanBangladeshPakistanVietnamMexicoGermanyRussiaPhilippines
TurkeyIranFranceUnited KingdomItalyThailandSouth
KoreaEnglandArgentinaSpainCanadaColombiaEgyptMalaysiaPeruSaudi
ArabiaAustraliaMoroccoPolandChileTaiwanMyanmarUzbekistanNepalSri
LankaVenezuelaCambodiaCubaNetherlandsSouth
AfricaEcuadorUkraineNigeriaEthiopiaMozambiqueBelgiumUnited Arab
EmiratesPortugalSwedenKazakhstanGreeceRwandaAustriaIsraelIraqAngolaCzechiaKeny
aUgandaRomaniaHungaryGuatemalaSwitzerlandDominican RepublicHong

KongSingaporeAlgeriaAzerbaijanDenmarkTunisiaBoliviaGhanaScotlandHondurasFinlandBelarusNorwayNew ZealandIrelandTajikistanEl SalvadorCote d'IvoireCosta RicaLaosJordanNicaraguaZimbabweSerbiaParaguayUruguayPanamaKuwaitTurkmenistan SlovakiaOmanWalesQatarSudanGuineaAfghanistanMongoliaLebanonCroatiaTanzaniaLithuaniaBulgariaNorthern IrelandPalestineBahrainLibyaZambiaSyriaBeninSloveniaKyrgyzstanLatviaGeorgiaTogoAlbaniaNigerMauritaniaBotswanaSenegalMauritiusSomaliaBurkina FasoSierra LeoneMoldovaArmeniaMalawiEstoniaBosnia and HerzegovinaMaliNorth MacedoniaKosovoBhutanCyprusCameroonTrinidad and TobagoJamaicaFijiMadagascarTimorLuxembourgMaltaMacaoLiberiaBruneiCentral African RepublicDemocratic Republic of CongoMaldivesLesothoGuyanaNamibiaCongoYemenIcelandCape VerdeMontenegroComorosNorthern CyprusSouth SudanPapua New GuineaGuinea-BissauGabonEswatiniSurinameEquatorial GuineaBelizeChadNew CaledoniaFrench PolynesiaGambiaSolomon IslandsBahamasBarbadosSamoaHaitiCuracaoJerseySao Tome and PrincipeSeychellesVanuatuIsle of ManArubaDjiboutiAndorraGuernseyTongaCayman IslandsKiribatiBermudaAntigua and BarbudaSaint LuciaGibraltarFaeroe IslandsGrenadaGreenlandLiechtensteinSaint Vincent and the GrenadinesTurks and Caicos IslandsSan MarinoChinaBangladeshPhilippinesThailandColombiaMoroccoNepalSouth AfricaBelgiumRwandaKenyaDominican RepublicTunisiaBelarusCote d'IvoireSerbiaSlovakiaAfghanistanBulgariaSyriaTogoMauritiusMalawiBhutanMadagascarBruneiNamibiaComorosEswatiniFrench PolynesiaHaitiIsle of ManCayman IslandsFaeroe IslandsSan MarinoBonaire Sint Eustatius and SabaTuvaluPitcairn00.5B1B1.5B2B2.5B3B Vaccination total per countryCountryVaccination total

In [40]:

```
trace_bar(vacc_country, 'daily_vaccinations', 'Daily vaccinations per country', 'Country', 'Daily vaccinations', "red" )
```

ChinaIndiaBangladeshUnited StatesPakistanJapanBrazilIndonesiaVietnamMexicoPhilippinesEthiopiaTurkeyIranGermanyRussiaUnited KingdomSouth KoreaThailandEnglandFranceSpainItalyEgyptTaiwanMalaysiaCanadaSri LankaVenezuelaArgentinaGhanaIraqUzbekistanColombiaSaudi ArabiaMoroccoPeruNepalMyanmarEcuadorKazakhstanPolandMozambiqueCubaRwandaNicaraguaChileAustraliaUgandaCambodiaNetherlandsNigeriaAlgeriaUkraineKenyaSouth AfricaIsraelDominican RepublicAngolaGuineaTunisiaSudanUnited Arab EmiratesBelgiumHondurasPortugalLaosBotswanaDenmarkRomaniaHungaryMongoliaCote d'IvoireAustriaSwedenGreeceBoliviaZimbabweCzechiaGuatemalaSwitzerlandBhutanIrelandParaguaySyriaJordanCosta RicaHong KongAzerbaijanNigerTajikistanNew ZealandSingaporeBelarusEl SalvadorFinlandPanamaScotlandAfghanistanNorwayMauritaniaSerbiaWalesOmanBeninSenegalMaliLebanonUruguayCameroonPalestineTurkmenistanKuwaitBurkina FasoLiberiaTogoTanzaniaQatarZambiaSlovakiaCroatiaSomaliaSierra LeoneGeorgiaLithuaniaBulgariaDemocratic Republic of CongoLibyaNorthern IrelandKyrgyzstanBahrainMalawiMauritiusKosovoSloveniaCentral African RepublicArmeniaTrinidad and TobagoAlbaniaGuinea-BissauLatviaNamibiaMoldovaNorth MacedoniaLesothoFijiBosnia and HerzegovinaJamaicaCyprusMontenegroTimorEswatiniMadagascarYemenEstoniaSouth SudanGuyanaMaltaBruneiMaldivesIcelandLuxembourgGabonCongoMacaoCape VerdeSurinameIsle of ManNorthern CyprusComorosPapua New GuineaNew

CaledoniaEquatorial GuineaBelizeBarbadosHaitiChadGambiaCuracaoBahamasSamoaFrench PolynesiaArubaSao Tome and PrincipeSeychellesSolomon IslandsTongaVanuatuSaint LuciaAndorraGuernseyKiribatiAntigua and BarbudaJerseyBermudaGreenlandGibraltarCayman IslandsFaeroe IslandsSint Maarten (Dutch part)DominicaSaint Vincent and the GrenadinesCook IslandsDjiboutiSaint Kitts and NevisLiechtensteinSan MarinoGrenadaNauruTurks and Caicos IslandsChinaBrazilTurkeyThailandTaiwanGhanaPeruMozambiqueUgandaKenyaTunisiaLaosCote d'IvoireCzechiaSyriaTajikistanPanamaWalesUruguayLiberiaCroatiaDemocratic Republic of CongoMauritiusAlbaniaLesothoTimorGuyanaGabonNorthern CyprusBarbadosSamoaTongaAntigua and BarbudaFaeroe IslandsSaint Kitts and NevisBurundiBonaire Sint Eustatius and SabaPitcairn05M10M15M20M Daily vaccinations per countryCountryDaily vaccinations

In [41]:

```
trace_bar(vacc_country, 'daily_vaccinations_per_million', 'Daily vaccinations per million per country', 'Country','Daily vaccinations per million', "magenta" )
```

BhutanIsle of ManBotswanaNiueFalkland IslandsNauruNicaraguaCook IslandsMongoliaGibraltarWallis and FutunaCubaGuernseySaint HelenaArubaTaiwanSeychellesRwandaDenmarkAndorraBangladeshSri LankaAnguillaPitcairnCuracaoSan MarinoSint Maarten (Dutch part)IcelandFaeroe IslandsTuvaluIsraelGreenlandEcuadorBermudaLaosWalesLiechtensteinTongaKazakhstan IrelandMontenegroBruneiNew CaledoniaSouth KoreaVietnamCambodiaCosta RicaDominican RepublicTokelauMalaysiaPanamaSamoaVenezuelaMauritiusNew ZealandMaltaJapanFijiNetherlandsBarbadosNorthern CyprusChileUnited Arab EmiratesChinaCayman IslandsHondurasSaint Kitts and NevisTurkeyPortugalEnglandPhilippinesMaldivesIranMauritaniaSingaporeSpainCanadaUnited KingdomUruguayTunisiaCyprusGhanaGermanyBelgiumTurks and Caicos IslandsNepalHungaryScotlandFinlandAntigua and BarbudaGuineaTrinidad and TobagoMexicoNorwayBahrainQatarUzbekistanParaguayNorthern IrelandAustriaKiribatiJerseyThailandCape VerdeEthiopiaDominicaOmanPeruGuyanaAustraliaHong KongSaudi ArabiaFrench PolynesiaLuxembourgBelizeBritish Virgin IslandsEl SalvadorItalySao Tome and PrincipeSwedenFranceKosovoMoroccoMontserratSwitzerlandUnited StatesSurinameGreeceIraqSaint LuciaMacaoMozambiqueLithuaniaArgentinaPakistanEswatiniKuwaitSerbiaSloveniaCzechiaLatviaBoliviaBrazilPolandGuinea-BissauCroatiaJordanTajikistanAzerbaijanBahamasTimorLebanonBelarusColombiaPalestineMonacoEstoniaLiberiaSaint Vincent and the GrenadinesNorth MacedoniaBonaire Sint Eustatius and SabaGeorgiaIndiaRussiaRomaniaZimbabweIndonesiaTurkmenistanLesothoSlovakiaMyanmarNamibiaVanuatuUgandaComorosAlbaniaArmeniaEgyptUkraineAlgeriaAngolaGuatemalaGrenadaSyriaBeninJamaicaTogoCote d'IvoireKenyaBosnia and HerzegovinaMoldovaSierra LeoneBulgariaCentral African RepublicSouth AfricaSudanEquatorial GuineaKyrgyzstanSenegalLibyaNigerSolomon IslandsGabonMaliSomaliaZambiaBurkina FasoAfghanistanCameroonGambiaCongoNigeriaMalawiSouth SudanDjiboutiTanzaniaPapua New GuineaHaitiMadagascarYemenDemocratic Republic of CongoChadBurundiBhutanNicaraguaGuernseyDenmarkCuracaoIsraelLiechtensteinNew CaledoniaTokelauNew ZealandNorthern CyprusSaint Kitts and NevisIranUruguayTurks and Caicos

IslandsGuineaUzbekistanThailandGuyanaBelizeFranceSurinameLithuaniaSloveniaGuinea-BissauTimorEstoniaIndiaLesothoComorosAngolaTogoBulgariaSenegalSomaliaCongoPapua New GuineaBurundi020k40k60k80k100k120k

Daily vaccinations per million per countryCountryDaily vaccinations per million

In [42]:
```python
trace_bar(vacc_country, 'people_vaccinated', 'People vaccinated per country', 'Country','People vaccinated', "lightblue" )
```

ChinaIndiaBrazilUnited StatesIndonesiaJapanVietnamRussiaPhilippinesPakistanBangladeshMexicoSpainIranGermanyTurkeyThailandColombiaFranceUnited KingdomItalySouth KoreaEgyptEnglandArgentinaCanadaPolandPeruMalaysiaMyanmarSaudi ArabiaUzbekistanMoroccoNepalEthiopiaAustraliaVenezuelaNigeriaSouth AfricaCubaTaiwanChileSri LankaUkraineEcuadorCambodiaMozambiqueUgandaNetherlandsKenyaAngolaIraqUnited Arab EmiratesPortugalKazakhstanBelgiumRwandaGhanaGreeceSwedenGuatemalaAlgeriaDominican RepublicTunisiaCote d'IvoireBoliviaCzechiaAustriaIsraelHong KongHungarySwitzerlandBelarusLaosNicaraguaRomaniaHondurasAzerbaijanSudanTajikistanAfghanistanZimbabweSingaporeDenmarkJordanEl SalvadorFinlandScotlandTurkmenistanCosta RicaNorwayNew ZealandIrelandTanzaniaParaguayGuineaPanamaKuwaitSerbiaOmanUruguayBeninSlovakiaWalesLebanonZambiaQatarCroatiaSyriaMongoliaLibyaNigerBurkina FasoPalestineLithuaniaSomaliaBulgariaSierra LeoneGeorgiaMalawiTogoMauritaniaKyrgyzstanSenegalBotswanaNorthern IrelandCameroonLatviaMaliAlbaniaSloveniaBahrainArmeniaMadagascarLiberiaMauritiusMoldovaBosnia and HerzegovinaCentral African RepublicKosovoEstoniaDemocratic Republic of CongoNorth MacedoniaLesothoJamaicaTrinidad and TobagoTimorCyprusBhutanCongoFijiYemenMacaoSouth SudanGuinea-BissauLuxembourgMaltaGuyanaNamibiaBruneiMaldivesEswatiniCape VerdeComorosGambiaPapua New GuineaIcelandGabonMontenegroNorthern CyprusChadSurinameEquatorial GuineaBelizeSolomon IslandsNew CaledoniaFrench PolynesiaSamoaBahamasHaitiBarbadosDjiboutiVanuatuSao Tome and PrincipeCuracaoArubaSeychellesJerseyKiribatiTongaIsle of ManChinaVietnamSpainFranceArgentinaSaudi ArabiaVenezuelaSri LankaNetherlandsKazakhstanGuatemalaCzechiaBelarusSudanJordanNorwayPanamaSlovakiaSyriaLithuaniaTogoCameroonArmeniaCentral African RepublicJamaicaFijiMaltaCape VerdeMontenegroSolomon IslandsBarbadosSeychellesCayman IslandsGibraltarSaint Kitts and NevisBonaire Sint Eustatius and SabaTuvaluPitcairn00.5B1B1.5B2B2.5B3B

People vaccinated per countryCountryPeople vaccinated

In [43]:
```python
trace_bar(vacc_country, 'people_vaccinated_per_hundred', 'People vaccinated per hundred per country', 'Country','People vaccinated per hundred', "orange" )
```

GibraltarCubaUnited Arab EmiratesGuernseyChinaQatarPortugalFinlandSpainVietnamBermudaCyprusJapanIsle of ManKuwaitBrazilSan MarinoNiuePitcairnColombiaPhilippinesPanamaNepalChileBruneiSingaporeMaltaCayman IslandsBhutanArgentinaMacaoSouth KoreaCambodiaSamoaAustraliaHong KongCanadaCook IslandsUruguaySeychellesPeruFaeroe IslandsCosta

RicaNauruItalyIcelandMalaysiaNew ZealandDenmarkPolandEcuadorNicaraguaArubaJerseyIrelandScotlandTaiwanAntigua and BarbudaTurks and Caicos IslandsFranceWalesCroatiaBelgiumNorwayThailandSri LankaUzbekistanWallis and FutunaMauritiusEnglandNetherlandsHondurasUnited KingdomVenezuelaSwedenUnited StatesBangladeshGreeceGermanyLaosLuxembourgFijiAustriaIranNorthern IrelandAndorraFalkland IslandsOmanNorthern CyprusSaudi ArabiaMaldivesMonacoGreenlandLithuaniaBonaire Sint Eustatius and SabaIsraelTongaLatviaRussiaSaint HelenaTurkmenistanIndonesiaTokelauIndiaBahrainEl SalvadorAnguillaLiechtensteinSwitzerlandMongoliaTurkeyBelarusRwandaBritish Virgin IslandsMoroccoHungaryFrench PolynesiaMexicoDominican RepublicCuracaoNew CaledoniaEstoniaCzechiaKiribatiSint Maarten (Dutch part)Cape VerdeSloveniaBotswanaTunisiaSaint Kitts and NevisGuyanaBoliviaKosovoBelizePakistanBarbadosAzerbaijanParaguayTrinidad and TobagoTuvaluTajikistanTimorSlovakiaSao Tome and PrincipeKazakhstanSerbiaRomaniaMyanmarMontenegroJordanSurinameDominicaAlbania MozambiqueEgyptGuatemalaBahamasNorth MacedoniaGeorgiaComorosGrenadaPalestineMontserratArmeniaLebanonVanuatuLesotho UkraineSouth AfricaSolomon IslandsZimbabweAngolaEswatiniMauritaniaSaint Vincent and the GrenadinesLibyaSaint LuciaUgandaBosnia and HerzegovinaGhanaBulgariaCote d'IvoireJamaicaGuinea-BissauGuineaIraqBeninMoldovaKyrgyzstanKenyaSierra LeoneEthiopiaLiberiaCentral African RepublicTogoEquatorial GuineaNamibiaAlgeriaDjiboutiGambiaZambiaGabonAfghanistanSyriaCongoSomaliaSudan NigeriaBurkina FasoNigerSenegalMalawiTanzaniaMaliCameroonSouth SudanMadagascarPapua New GuineaYemenChadHaitiDemocratic Republic of CongoBurundiGibraltarPortugalJapanPitcairnBruneiMacaoCanadaCosta RicaDenmarkIrelandWalesUzbekistanUnited KingdomGermanyNorthern IrelandMaldivesTongaTokelauSwitzerlandMoroccoNew CaledoniaSloveniaKosovoTrinidad and TobagoKazakhstanSurinameBahamasMontserratSouth AfricaSaint Vincent and the GrenadinesBulgariaBeninLiberiaDjiboutiCongoSenegalMadagascarBurundi050100150200 250300

People vaccinated per hundred per countryCountryPeople vaccinated per hundred

In [44]:
```python
def plot_scatter(data, x, y, size, color, hover_name, title):
    fig = px.scatter(data, x=x, y=y, size=size, color=color, hover_name=hover_name, title=title, size_max=80)
    fig.update_layout({"legend_orientation":"h"})
    fig.update_layout(legend = dict(yanchor ="top", y = -0.2))
    fig.update_layout({"legend_title":"Vaccine scheme"})
    fig.update_layout({"plot_bgcolor":"rgba(0,0,0,0)","paper_bgcolor":"rgba(0,0,0,0)"})
    fig.update_xaxes(showline=True, linewidth=2, linecolor='black', mirror=True)
    fig.update_yaxes(showline=True, linewidth=2, linecolor='black', mirror=True)
    fig.update_xaxes(zeroline=True, zerolinewidth=1, zerolinecolor='grey')
    fig.update_yaxes(zeroline=True, zerolinewidth=1, zerolinecolor='grey')
    fig.update_xaxes(showgrid=True, gridwidth=1, gridcolor='lightgrey')
    fig.update_yaxes(showgrid=True, gridwidth=1, gridcolor='lightgrey')
    fig.show()
```

In [45]:
```python
plot_scatter(vacc_country, x = "total_vaccinations", y = "daily_vaccinations",
             size = "total_vaccinations", color="vaccines",
```

01B2B3B4B050M

Vaccine schemeJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingOxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VOxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTechOxford/AstraZenecaOxford/AstraZeneca, Pfizer/BioNTechOxford/AstraZeneca, Pfizer/BioNTech, Sputnik VCanSino, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VModerna, Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik VPfizer/BioNTechJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingSinopharm/Beijing, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VModerna, Pfizer/BioNTechCovaxin, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacJohnson&Johnson, Oxford/AstraZenecaJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/BeijingSinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, SinovacCovaxin, Oxford/AstraZenecaCanSino, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacCanSino, Sinopharm/Beijing, Sinopharm/Wuhan, Sinovac, ZF2001Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacCovaxin, Oxford/AstraZeneca, Sinopharm/BeijingModerna, Oxford/AstraZeneca, Sinopharm/Beijing, Sputnik VAbdala, Soberana Plus, Soberana02Johnson&Johnson, Moderna, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCovaxin, Johnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, SinovacJohnson&Johnson, Pfizer/BioNTechPfizer/BioNTech, Sinopharm/Beijing, Sputnik VOxford/AstraZeneca, Sputnik VModernaModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VOxford/AstraZeneca, Sinopharm/BeijingModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VJohnson&Johnson, ModernaJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VPfizer/BioNTech, SinovacJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VCovaxin, Oxford/AstraZeneca, Sputnik VJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCOVIran Barekat, Covaxin, FAKHRAVAC, Oxford/AstraZeneca, Razi Cov Pars, Sinopharm/Beijing, Soberana02, SpikoGen, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VQazVac, Sinopharm/Beijing, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Novavax, Pfizer/BioNTechOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VPfizer/BioNTech, Sinopharm/BeijingCanSino, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCanSino, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VAbdala, Johnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech,

Soberana02, Sputnik Light, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, SinovacCanSino, Covaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VCovaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VEpiVacCorona, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VPfizer/BioNTech, Sputnik VOxford/AstraZeneca, Sinopharm/Beijing, Sputnik VModerna, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VMedigen, Moderna, Oxford/AstraZeneca, Pfizer/BioNTechModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VJohnson&Johnson, Pfizer/BioNTech, Sinopharm/BeijingModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacPfizer/BioNTech, Sinovac, TurkovacEpiVacCorona, Oxford/AstraZeneca, QazVac, Sinopharm/Beijing, Sputnik V, ZF2001Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinopharm/Wuhan, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik Light, Sputnik V, ZF2001Abdala, Sinopharm/Beijing, Sinovac, Soberana02, Sputnik Light, Sputnik VAbdala, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, SinovacVaccinations (Total vs. Daily) grouped per country and vaccinestotal_vaccinationsdaily_vaccinations

In [46]:
```python
plot_scatter(vacc_country,x = "people_vaccinated", y = "daily_vaccinations_per_millio
n", size = "total_vaccinations", color = "vaccines", hover_name = "country",title = "
Vaccinations (daily / million vs. iso_code) grouped per country and vaccines")
```

01B2B3B4B−100k0100k
Vaccine schemeJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingOxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VOxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTechOxford/AstraZenecaOxford/AstraZeneca, Pfizer/BioNTechOxford/AstraZeneca, Pfizer/BioNTech, Sputnik VCanSino, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VModerna, Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik VPfizer/BioNTechJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingSinopharm/Beijing, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VModerna, Pfizer/BioNTechCovaxin, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacJohnson&Johnson, Oxford/AstraZenecaJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/BeijingSinopharm/BeijingJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, SinovacCovaxin, Oxford/AstraZenecaCanSino, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacCanSino, Sinopharm/Beijing, Sinopharm/Wuhan, Sinovac,

ZF2001Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, SinovacCovaxin, Oxford/AstraZeneca, Sinopharm/BeijingModerna, Oxford/AstraZeneca, Sinopharm/Beijing, Sputnik VAbdala, Soberana Plus, Soberana02Johnson&Johnson, Moderna, Pfizer/BioNTechJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCovaxin, Johnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, SinovacJohnson&Johnson, Pfizer/BioNTechPfizer/BioNTech, Sinopharm/Beijing, Sputnik VOxford/AstraZeneca, Sputnik VModernaModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VOxford/AstraZeneca, Sinopharm/BeijingModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VJohnson&Johnson, ModernaJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VPfizer/BioNTech, SinovacJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VCovaxin, Oxford/AstraZeneca, Sputnik VJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCOVIran Barekat, Covaxin, FAKHRAVAC, Oxford/AstraZeneca, Razi Cov Pars, Sinopharm/Beijing, Soberana02, SpikoGen, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VQazVac, Sinopharm/Beijing, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VJohnson&Johnson, Moderna, Novavax, Pfizer/BioNTechOxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VPfizer/BioNTech, Sinopharm/BeijingCanSino, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacCanSino, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VAbdala, Johnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Soberana02, Sputnik Light, Sputnik VOxford/AstraZeneca, Pfizer/BioNTech, SinovacCanSino, Covaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VCovaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VEpiVacCorona, Sputnik VJohnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik VPfizer/BioNTech, Sputnik VOxford/AstraZeneca, Sinopharm/Beijing, Sputnik VModerna, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacJohnson&Johnson, Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac, Sputnik Light, Sputnik VMedigen, Moderna, Oxford/AstraZeneca, Pfizer/BioNTechModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik VJohnson&Johnson, Pfizer/BioNTech, Sinopharm/BeijingModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, SinovacPfizer/BioNTech, Sinovac, TurkovacEpiVacCorona, Oxford/AstraZeneca, QazVac, Sinopharm/Beijing, Sputnik V, ZF2001Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinopharm/Wuhan, Sputnik VModerna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik Light, Sputnik V, ZF2001Abdala, Sinopharm/Beijing, Sinovac, Soberana02, Sputnik Light, Sputnik VAbdala, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik VJohnson&Johnson, Oxford/AstraZeneca, SinovacVaccinations (daily / million vs. iso_code) grouped per country and vaccinespeople_vaccinateddaily_vaccinations_per_million

In [47]:

```python
trace = go.Choropleth(
            locations = vacc_country['country'],
```

```python
            locationmode='country names',
            z = vacc_country['total_vaccinations'],
            text = vacc_country['country'],
            autocolorscale =False,
            reversescale = True,
            colorscale = 'viridis',
            marker = dict(
                line = dict(
                    color = 'rgb(0,0,0)',
                    width = 0.5)
            ),
            colorbar = dict(
                title = 'Total vaccinations',
                tickprefix = '')
        )

data = [trace]
layout = go.Layout(
    title = 'Total vaccinations per country',
    geo = dict(
        showframe = True,
        showlakes = False,
        showcoastlines = True,
        projection = dict(
            type = 'natural earth'
        )
    )
)

fig = dict( data=data, layout=layout )
```
00.5B1B1.5B2B2.5B3BTotal vaccinationsTotal vaccinations per country
in the total vaccination of countries;

China is the country with the most vaccines overall.

The country with the least vaccines is Chad.

In [48]:
```python
trace = go.Choropleth(
            locations = vacc_country['country'],
            locationmode='country names',
            z = vacc_country['daily_vaccinations'],
            text = vacc_country['country'],
            autocolorscale =False,
            reversescale = True,
            colorscale = 'viridis',
            marker = dict(
                line = dict(
                    color = 'rgb(0,0,0)',
                    width = 0.5)
            ),
            colorbar = dict(
                title = 'Daily vaccinations',
                tickprefix = '')
        )
```

```
data = [trace]
layout = go.Layout(
    title = 'Daily vaccinations per country',
    geo = dict(
        showframe = True,
        showlakes = False,
        showcoastlines = True,
        projection = dict(
            type = 'natural earth'
        )
    )
)

fig = dict( data=data, layout=layout )
```

05M10M15M20MDaily vaccinationsDaily vaccinations per country

in the daily vaccination of countries;

China is the country with the most daily vaccinations.

The country with the least daily vaccination is Chad.

```
In [49]:
trace = go.Choropleth(
            locations = vacc_country['country'],
            locationmode='country names',
            z = vacc_country['daily_vaccinations_per_million'],
            text = vacc_country['country'],
            autocolorscale =False,
            reversescale = True,
            colorscale = 'viridis',
            marker = dict(
                line = dict(
                    color = 'rgb(0,0,0)',
                    width = 0.5)
            ),
            colorbar = dict(
                title = 'Daily vaccinations per million',
                tickprefix = '')
        )

data = [trace]
layout = go.Layout(
    title = 'Daily vaccinations per million per country',
    geo = dict(
        showframe = True,
        showlakes = False,
        showcoastlines = True,
        projection = dict(
            type = 'natural earth'
        )
    )
)

fig = dict( data=data, layout=layout )
```

20k40k60k80k100kDaily vaccinations per millionDaily vaccinations per million per country

Number of daily vaccines per million;

Bhudan is the country with the most daily vaccinations per million.

The country with the lowest daily vaccination per million is Chad.

In [50]:
```python
trace = go.Choropleth(
            locations = vacc_country['country'],
            locationmode='country names',
            z = vacc_country['people_vaccinated'],
            text = vacc_country['country'],
            autocolorscale =False,
            reversescale = True,
            colorscale = 'viridis',
            marker = dict(
                line = dict(
                    color = 'rgb(0,0,0)',
                    width = 0.5)
            ),
            colorbar = dict(
                title = 'People vaccinated',
                tickprefix = '')
        )

data = [trace]
layout = go.Layout(
    title = 'People vaccinated per country',
    geo = dict(
        showframe = True,
        showlakes = False,
        showcoastlines = True,
        projection = dict(
            type = 'natural earth'
        )
    )
)

fig = dict( data=data, layout=layout )
```
00.5B1B1.5B2B2.5B3BPeople vaccinatedPeople vaccinated per country

people vaccinated in countries;

China is the country with the most people vaccinated.

Chad is the country with the least number of people vaccinated.

In [51]:
```python
linkcode
trace = go.Choropleth(
            locations = vacc_country['country'],
            locationmode='country names',
            z = vacc_country['people_vaccinated_per_hundred'],
```

```python
            text = vacc_country['country'],
            autocolorscale =False,
            reversescale = True,
            colorscale = 'viridis',
            marker = dict(
                line = dict(
                    color = 'rgb(0,0,0)',
                    width = 0.5)
            ),
            colorbar = dict(
                title = 'People vaccinated per hundred',
                tickprefix = '')
        )

data = [trace]
layout = go.Layout(
    title = "People vaccinated per hundred per country",
    geo = dict(
        showframe = True,
        showlakes = False,
        showcoastlines = True,
        projection = dict(
            type = 'natural earth'
        )
    )
)

fig = dict( data=data, layout=layout )
```

50100150200250300People vaccinated per hundredPeople vaccinated per hundred per country

percentage of people vaccinated in countries;

China has the highest percentage of people vaccinated.

democratic republic of congo has the lowest percentage of people vaccinated.