

Covid - 19 Vaccine Analysis

Project Name	Covid - 19 Vaccine Analysis
Team ID	8934
Date	31/10/2023

Introduction:

In this phase 4, we are going to perform exploratory data analysis, statistical analysis and visualization of COVID-19 Vaccine data.

EXPLORATORY DATA ANALYSIS:

EDA is like detective work for data. It's about looking at data, such as information about COVID-19 vaccines, to find patterns, trends, and important details. Imagine you have a big puzzle, and EDA helps you understand what each piece means and how they fit together.

VISUALIZATION:

Visualization is a way to show data in pictures, like graphs or charts. For COVID-19 vaccine analysis, it might be a graph showing how many people got vaccinated each day or a map that displays where vaccines are being given the most. Visuals make it easier to understand data quickly.

STATISTICAL DATA:

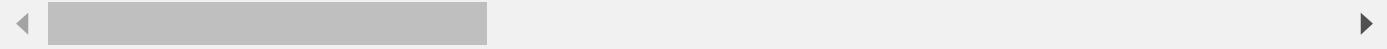
Statistics is all about using math to understand data. In COVID-19 vaccine analysis, statisticians use numbers and formulas to draw conclusions. They might calculate things like the vaccine's effectiveness or the likelihood of side effects happening.

CONCLUSION:

In conclusion, the application of exploratory data analysis (EDA), visualization, and statistical data techniques in the analysis of COVID-19 vaccine data has proven to be invaluable in understanding the development, distribution, and impact of vaccines during the pandemic. Through these approaches, researchers, policymakers, and the general public have gained crucial insights that have shaped the global response to the virus.

```
In [1]: #import the required Libraries  
#import the required dataset  
#view the dataset  
  
import pandas as pd  
import seaborn as sns  
import matplotlib.pyplot as plt  
import plotly.express as px  
%matplotlib inline  
df=pd.read_csv('Documents/country_vaccinations.csv')  
df.head()
```

```
Out[1]:    country iso_code   date  total_vaccinations  people_vaccinated  people_fully_vaccinated  daily_vac  
0  Afghanistan     AFG 2021-02-22             0.0                  0.0                      0.0                 NaN  
1  Afghanistan     AFG 2021-02-23            NaN                  NaN                      NaN                 NaN  
2  Afghanistan     AFG 2021-02-24            NaN                  NaN                      NaN                 NaN  
3  Afghanistan     AFG 2021-02-25            NaN                  NaN                      NaN                 NaN  
4  Afghanistan     AFG 2021-02-26            NaN                  NaN                      NaN                 NaN
```



```
In [3]: df1=df.dropna()  
print(df1)
```

	country	iso_code	date	total_vaccinations	\
94	Afghanistan	AFG	2021-05-27	593313.0	
101	Afghanistan	AFG	2021-06-03	630305.0	
339	Afghanistan	AFG	2022-01-27	5081064.0	
433	Albania	ALB	2021-02-18	3049.0	
515	Albania	ALB	2021-05-11	622507.0	
...
86507	Zimbabwe	ZWE	2022-03-25	8691642.0	
86508	Zimbabwe	ZWE	2022-03-26	8791728.0	
86509	Zimbabwe	ZWE	2022-03-27	8845039.0	
86510	Zimbabwe	ZWE	2022-03-28	8934360.0	
86511	Zimbabwe	ZWE	2022-03-29	9039729.0	
	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	\	
94	479574.0	113739.0	2859.0		
101	481800.0	148505.0	4015.0		
339	4517380.0	3868832.0	6868.0		
433	2438.0	611.0	1348.0		
515	440921.0	181586.0	9548.0		
...
86507	4814582.0	3473523.0	139213.0		
86508	4886242.0	3487962.0	100086.0		
86509	4918147.0	3493763.0	53311.0		
86510	4975433.0	3501493.0	89321.0		
86511	5053114.0	3510256.0	105369.0		
	daily_vaccinations	total_vaccinations_per_hundred	\		
94	6487.0	1.49			
101	5285.0	1.58			
339	9802.0	12.76			
433	254.0	0.11			
515	12160.0	21.67			
...
86507	69579.0	57.59			
86508	83429.0	58.25			
86509	90629.0	58.61			
86510	100614.0	59.20			
86511	103751.0	59.90			
	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	\		
94	1.20	0.29			
101	1.21	0.37			
339	11.34	9.71			
433	0.08	0.02			
515	15.35	6.32			
...
86507	31.90	23.02			
86508	32.38	23.11			
86509	32.59	23.15			
86510	32.97	23.20			
86511	33.48	23.26			
	daily_vaccinations_per_million	\			
94	163.0				
101	133.0				
339	246.0				
433	88.0				
515	4233.0				
...	...				
86507	4610.0				

```
86508          5528.0
86509          6005.0
86510          6667.0
86511          6874.0

                           vaccines \
94    Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
101   Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
339   Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
433   Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, ...
515   Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, ...
...
86507   Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86508   Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86509   Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86510   Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86511   Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

                           source_name \
94    World Health Organization
101   World Health Organization
339   World Health Organization
433   Ministry of Health
515   Ministry of Health
...
86507   Ministry of Health
86508   Ministry of Health
86509   Ministry of Health
86510   Ministry of Health
86511   Ministry of Health

                           source_website
94                  https://covid19.who.int/
101                 https://covid19.who.int/
339                 https://covid19.who.int/
433   https://shendetesia.gov.al/vaksinimi-anticovid...
515   https://shendetesia.gov.al/vaksinimi-anticovid...
...
86507   https://www.arcgis.com/home/webmap/viewer.html...
86508   https://www.arcgis.com/home/webmap/viewer.html...
86509   https://www.arcgis.com/home/webmap/viewer.html...
86510   https://www.arcgis.com/home/webmap/viewer.html...
86511   https://www.arcgis.com/home/webmap/viewer.html...
```

[30847 rows x 15 columns]

```
In [3]: df1=df.reset_index()
print(df1)
```

	index	country	iso_code	date	total_vaccinations	\
0	0	Afghanistan	AFG	2021-02-22	0.0	
1	1	Afghanistan	AFG	2021-02-23	NaN	
2	2	Afghanistan	AFG	2021-02-24	NaN	
3	3	Afghanistan	AFG	2021-02-25	NaN	
4	4	Afghanistan	AFG	2021-02-26	NaN	

86507	86507	Zimbabwe	ZWE	2022-03-25	8691642.0	
86508	86508	Zimbabwe	ZWE	2022-03-26	8791728.0	
86509	86509	Zimbabwe	ZWE	2022-03-27	8845039.0	
86510	86510	Zimbabwe	ZWE	2022-03-28	8934360.0	
86511	86511	Zimbabwe	ZWE	2022-03-29	9039729.0	
	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	\		
0	0.0	NaN	NaN		NaN	
1	NaN	NaN	NaN		NaN	
2	NaN	NaN	NaN		NaN	
3	NaN	NaN	NaN		NaN	
4	NaN	NaN	NaN		NaN	
	
86507	4814582.0	3473523.0	3473523.0		139213.0	
86508	4886242.0	3487962.0	3487962.0		100086.0	
86509	4918147.0	3493763.0	3493763.0		53311.0	
86510	4975433.0	3501493.0	3501493.0		89321.0	
86511	5053114.0	3510256.0	3510256.0		105369.0	
	daily_vaccinations	total_vaccinations_per_hundred	\			
0	NaN	0.00	0.00			
1	1367.0	NaN	NaN			
2	1367.0	NaN	NaN			
3	1367.0	NaN	NaN			
4	1367.0	NaN	NaN			
			
86507	69579.0	57.59	57.59			
86508	83429.0	58.25	58.25			
86509	90629.0	58.61	58.61			
86510	100614.0	59.20	59.20			
86511	103751.0	59.90	59.90			
	people_vaccinated_per_hundred	people_fully_vaccinated_per_hundred	\			
0	0.00	NaN	NaN			
1	NaN	NaN	NaN			
2	NaN	NaN	NaN			
3	NaN	NaN	NaN			
4	NaN	NaN	NaN			
			
86507	31.90	23.02	23.02			
86508	32.38	23.11	23.11			
86509	32.59	23.15	23.15			
86510	32.97	23.20	23.20			
86511	33.48	23.26	23.26			
	daily_vaccinations_per_million	\				
0	NaN					
1	34.0					
2	34.0					
3	34.0					
4	34.0					
				
86507	4610.0					

```
86508          5528.0
86509          6005.0
86510          6667.0
86511          6874.0

                           vaccines \
0    Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
1    Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
2    Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
3    Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
4    Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...
...
86507  Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86508  Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86509  Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86510  Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...
86511  Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

                           source_name \
0    World Health Organization
1    World Health Organization
2    World Health Organization
3    World Health Organization
4    World Health Organization
...
86507      Ministry of Health
86508      Ministry of Health
86509      Ministry of Health
86510      Ministry of Health
86511      Ministry of Health

                           source_website
0                  https://covid19.who.int/
1                  https://covid19.who.int/
2                  https://covid19.who.int/
3                  https://covid19.who.int/
4                  https://covid19.who.int/
...
86507  https://www.arcgis.com/home/webmap/viewer.html...
86508  https://www.arcgis.com/home/webmap/viewer.html...
86509  https://www.arcgis.com/home/webmap/viewer.html...
86510  https://www.arcgis.com/home/webmap/viewer.html...
86511  https://www.arcgis.com/home/webmap/viewer.html...
```

[86512 rows x 16 columns]

In [5]: df.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 [2]: `df.describe()`

Out[2]:

	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vacc
count	4.360700e+04	4.129400e+04	3.880200e+04	3.536200e+04	8.621
mean	4.592964e+07	1.770508e+07	1.413830e+07	2.705996e+05	1.313
std	2.246004e+08	7.078731e+07	5.713920e+07	1.212427e+06	7.682
min	0.000000e+00	0.000000e+00	1.000000e+00	0.000000e+00	0.000
25%	5.264100e+05	3.494642e+05	2.439622e+05	4.668000e+03	9.000
50%	3.590096e+06	2.187310e+06	1.722140e+06	2.530900e+04	7.343
75%	1.701230e+07	9.152520e+06	7.559870e+06	1.234925e+05	4.409
max	3.263129e+09	1.275541e+09	1.240777e+09	2.474100e+07	2.242

In [3]: `df.isnull().sum()`

```
Out[3]: 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
```

```
In [6]: df.fillna(0, inplace = True)
df.drop(df.index[df['iso_code'] == 0], inplace = True)
```

```
In [5]: df.isnull().sum()
```

```
Out[5]: 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
```

```
In [6]: #The date is in the 'object' format. Let us change it to Datetime format for easy handling
df['date'] = pd.to_datetime(df['date'], format='%Y-%m-%d')
```

```
In [7]: df.columns
```

```
Out[7]: Index(['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'],
      dtype='object')
```

```
In [12]: df.drop(["people_fully_vaccinated","daily_vaccinations_raw","people_vaccinated_per_hundred",
       "people_fully_vaccinated_per_hundred","daily_vaccinations_per_million","source_
```

Out[12]:

	country	iso_code	date	total_vaccinations	people_vaccinated	daily_vaccinations	total_vac
0	Afghanistan	AFG	2021-02-22	0.0	0.0	0.0	0.0
1	Afghanistan	AFG	2021-02-23	0.0	0.0	1367.0	
2	Afghanistan	AFG	2021-02-24	0.0	0.0	1367.0	
3	Afghanistan	AFG	2021-02-25	0.0	0.0	1367.0	
4	Afghanistan	AFG	2021-02-26	0.0	0.0	1367.0	
...
86507	Zimbabwe	ZWE	2022-03-25	8691642.0	4814582.0	69579.0	
86508	Zimbabwe	ZWE	2022-03-26	8791728.0	4886242.0	83429.0	
86509	Zimbabwe	ZWE	2022-03-27	8845039.0	4918147.0	90629.0	
86510	Zimbabwe	ZWE	2022-03-28	8934360.0	4975433.0	100614.0	
86511	Zimbabwe	ZWE	2022-03-29	9039729.0	5053114.0	103751.0	

86512 rows × 8 columns



In [15]:

```
#BANGLADESH
df_BGD = df[df["iso_code"] == 'BGD'].copy()
df_BGD
```

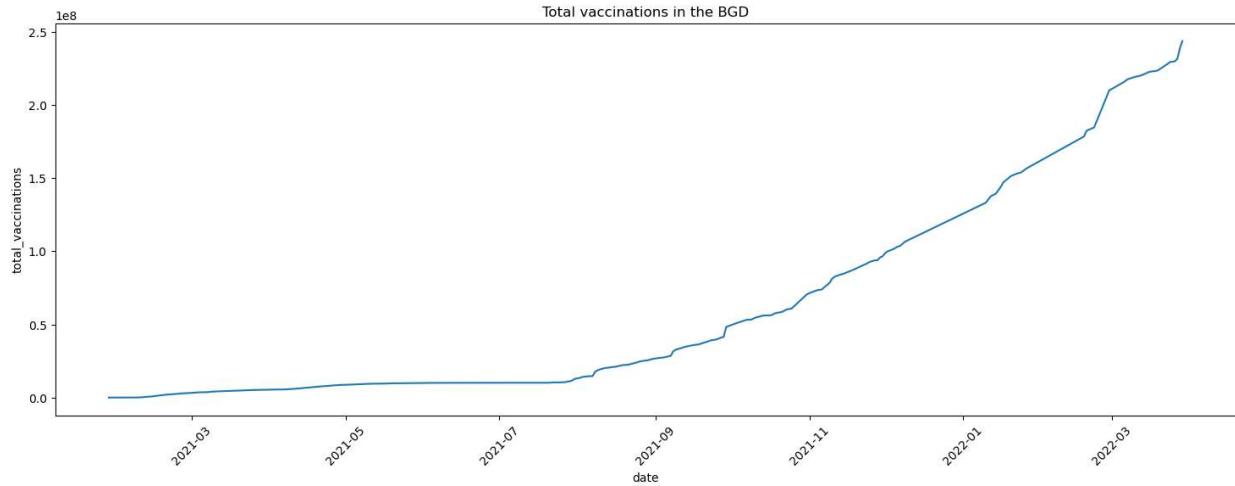
Out[15]:

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily
6133	Bangladesh	BGD	2021-01-26	0.0	0.0	0.0	0.0
6134	Bangladesh	BGD	2021-01-27	26.0	26.0	26.0	0.0
6135	Bangladesh	BGD	2021-01-28	567.0	567.0	567.0	0.0
6136	Bangladesh	BGD	2021-01-29	0.0	0.0	0.0	0.0
6137	Bangladesh	BGD	2021-01-30	0.0	0.0	0.0	0.0
...
6556	Bangladesh	BGD	2022-03-25	0.0	0.0	0.0	0.0
6557	Bangladesh	BGD	2022-03-26	229789298.0	127084404.0	95424031.0	
6558	Bangladesh	BGD	2022-03-27	231420654.0	127169172.0	96671169.0	
6559	Bangladesh	BGD	2022-03-28	238459012.0	127365973.0	103130478.0	
6560	Bangladesh	BGD	2022-03-29	243642749.0	127544055.0	107712737.0	

428 rows × 15 columns

In [16]: #Drop the dates with missing values, previously NaN that we filled with 0.
df_BGD.drop(df_BGD.index[df_BGD['total_vaccinations'] == 0], inplace = True)

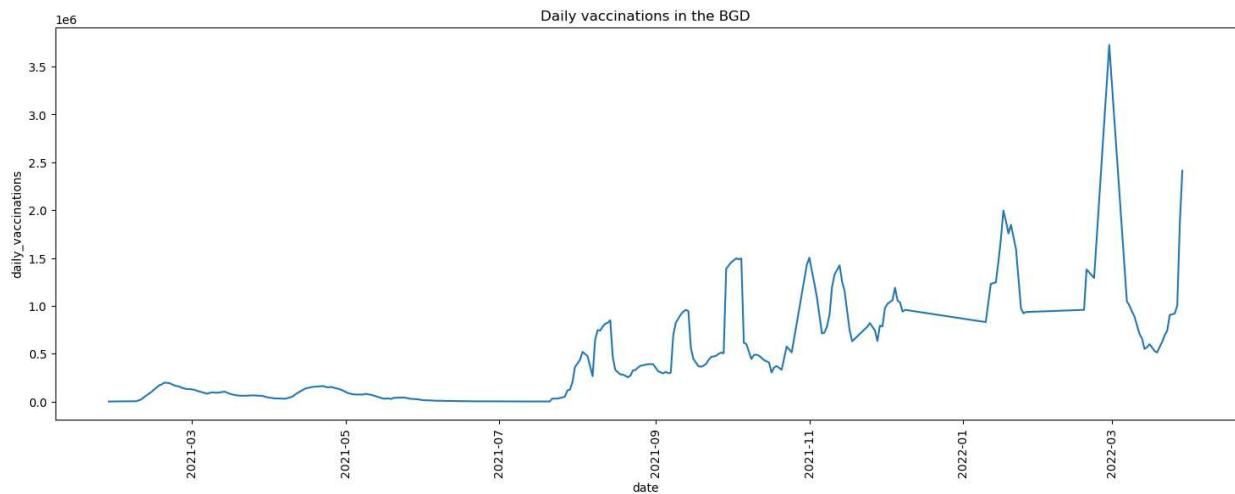
```
In [19]: #Plot total vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_BGD, x="date", y="total_vaccinations")
plt.title("Total vaccinations in the BGD")
plt.xticks(rotation=45)
plt.show()
```



```
In [20]: #Plot daily vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_BGD, x="date", y="daily_vaccinations")

plt.xticks(rotation=90)
plt.title("Daily vaccinations in the BGD")
```

Out[20]: Text(0.5, 1.0, 'Daily vaccinations in the BGD')



```
In [35]: #INDIA
df_IND = df[df["iso_code"] == 'IND'].copy()
df_IND=df_IND.head(10)
df_IND
```

Out[35]:

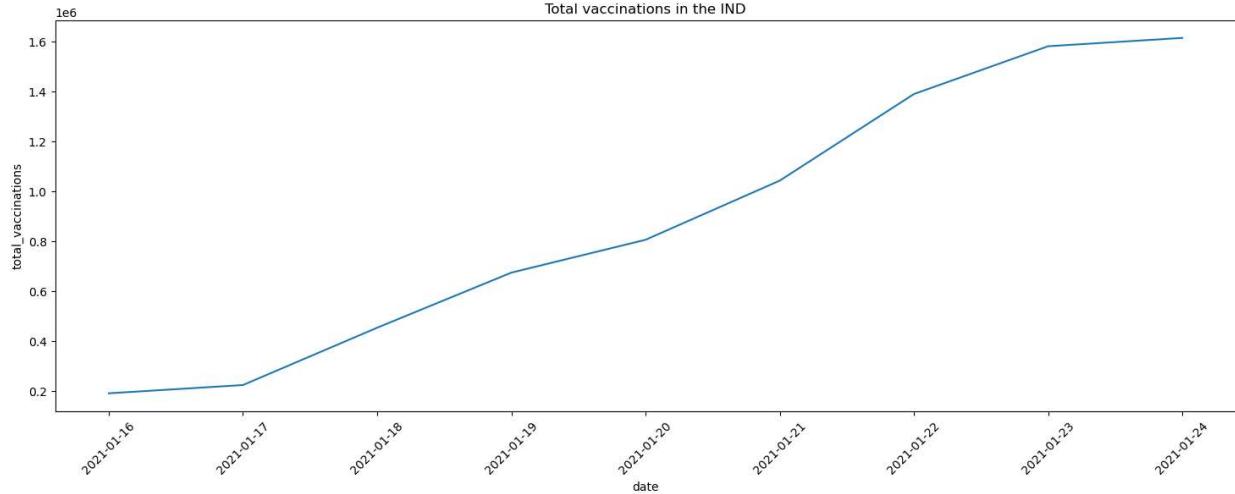
	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations
35372	India	IND	2021-01-15	0.0	0.0	0.0	NaN
35373	India	IND	2021-01-16	191181.0	191181.0	191181.0	NaN
35374	India	IND	2021-01-17	224301.0	224301.0	224301.0	NaN
35375	India	IND	2021-01-18	454049.0	454049.0	454049.0	NaN
35376	India	IND	2021-01-19	674835.0	674835.0	674835.0	NaN
35377	India	IND	2021-01-20	806484.0	806484.0	806484.0	NaN
35378	India	IND	2021-01-21	1043534.0	1043534.0	1043534.0	NaN
35379	India	IND	2021-01-22	1390592.0	1390592.0	1390592.0	NaN
35380	India	IND	2021-01-23	1582201.0	1582201.0	1582201.0	NaN
35381	India	IND	2021-01-24	1615504.0	1615504.0	1615504.0	NaN

In [36]:

```
#Drop the dates with missing values, previously NaN that we filled with 0.
df_IND.drop(df_IND.index[df_IND['total_vaccinations'] == 0], inplace = True)
```

In [37]:

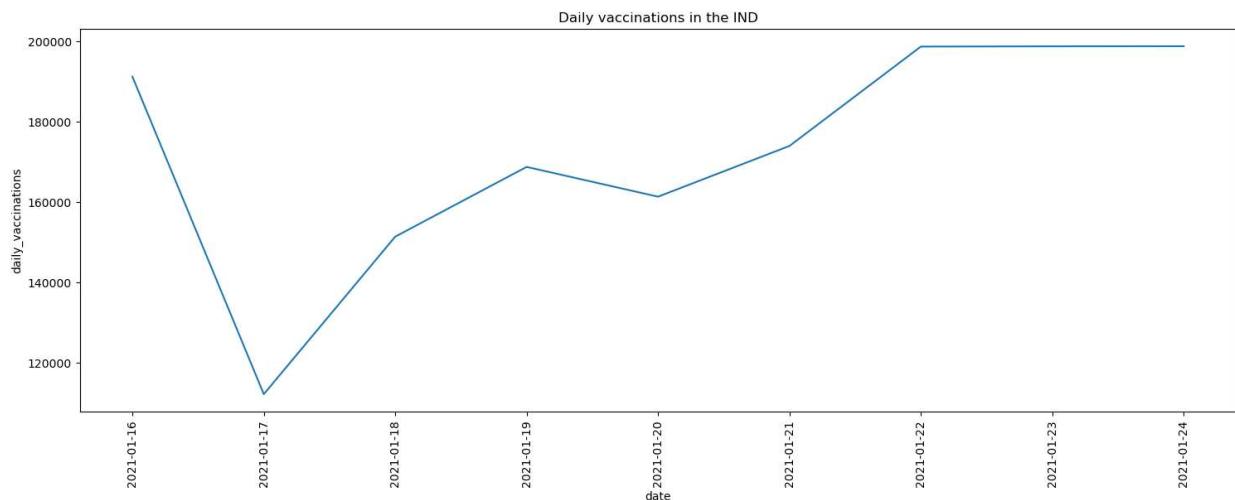
```
#Plot total vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_IND, x="date", y="total_vaccinations")
plt.title("Total vaccinations in the IND")
plt.xticks(rotation=45)
plt.show()
```



```
In [38]: #Plot daily vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_IND, x="date", y="daily_vaccinations")

plt.xticks(rotation=90)
plt.title("Daily vaccinations in the IND")
```

Out[38]: Text(0.5, 1.0, 'Daily vaccinations in the IND')



```
In [42]: #CHINA
df_CHN = df[df["iso_code"] == 'CHN'].copy()
df_CHN=df_CHN.head(20)
df_CHN
```

Out[42]:

		country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations
15756	China	CHN	2020-12-15		1500000.0	NaN	NaN	NaN
15757	China	CHN	2020-12-16		NaN	NaN	NaN	NaN
15758	China	CHN	2020-12-17		NaN	NaN	NaN	NaN
15759	China	CHN	2020-12-18		NaN	NaN	NaN	NaN
15760	China	CHN	2020-12-19		NaN	NaN	NaN	NaN
15761	China	CHN	2020-12-20		NaN	NaN	NaN	NaN
15762	China	CHN	2020-12-21		NaN	NaN	NaN	NaN
15763	China	CHN	2020-12-22		NaN	NaN	NaN	NaN
15764	China	CHN	2020-12-23		NaN	NaN	NaN	NaN
15765	China	CHN	2020-12-24		NaN	NaN	NaN	NaN
15766	China	CHN	2020-12-25		NaN	NaN	NaN	NaN
15767	China	CHN	2020-12-26		NaN	NaN	NaN	NaN
15768	China	CHN	2020-12-27		NaN	NaN	NaN	NaN

country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations
---------	----------	------	--------------------	-------------------	-------------------------	--------------------

15769	China	CHN	2020-12-28	NaN	NaN	NaN
-------	-------	-----	------------	-----	-----	-----

15770	China	CHN	2020-12-29	NaN	NaN	NaN
-------	-------	-----	------------	-----	-----	-----

15771	China	CHN	2020-12-30	NaN	NaN	NaN
-------	-------	-----	------------	-----	-----	-----

15772	China	CHN	2020-12-31	4500000.0	NaN	NaN
-------	-------	-----	------------	-----------	-----	-----

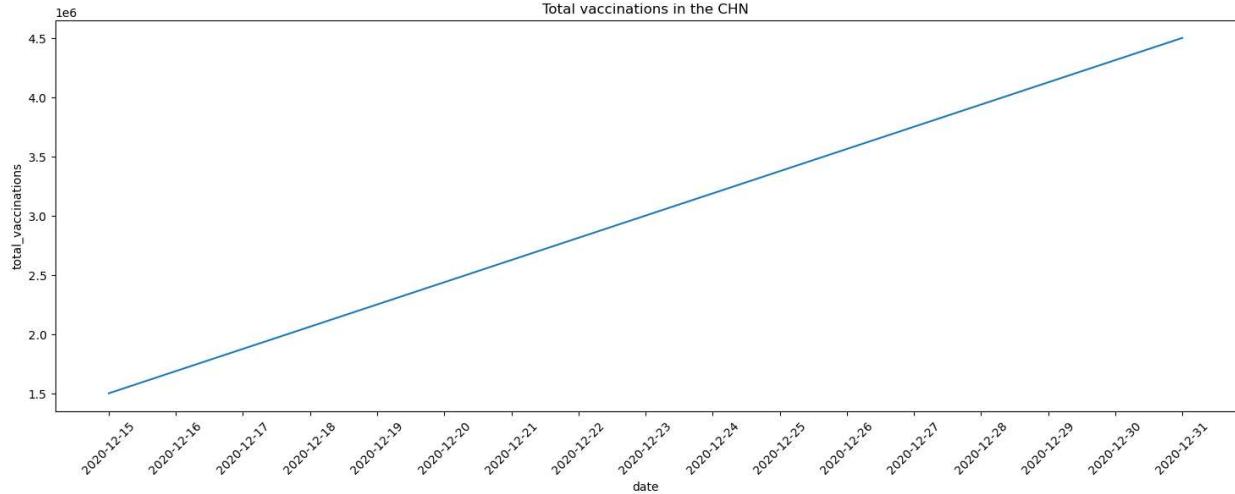
15773	China	CHN	2021-01-01	NaN	NaN	NaN
-------	-------	-----	------------	-----	-----	-----

15774	China	CHN	2021-01-02	NaN	NaN	NaN
-------	-------	-----	------------	-----	-----	-----

15775	China	CHN	2021-01-03	NaN	NaN	NaN
-------	-------	-----	------------	-----	-----	-----

In [43]: `#Drop the dates with missing values, previously NaN that we filled with 0.
df_CHN.drop(df_CHN.index[df_CHN['total_vaccinations'] == 0], inplace = True)`

In [44]: `#Plot total vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_CHN, x="date", y="total_vaccinations")
plt.title("Total vaccinations in the CHN")
plt.xticks(rotation=45)
plt.show()`

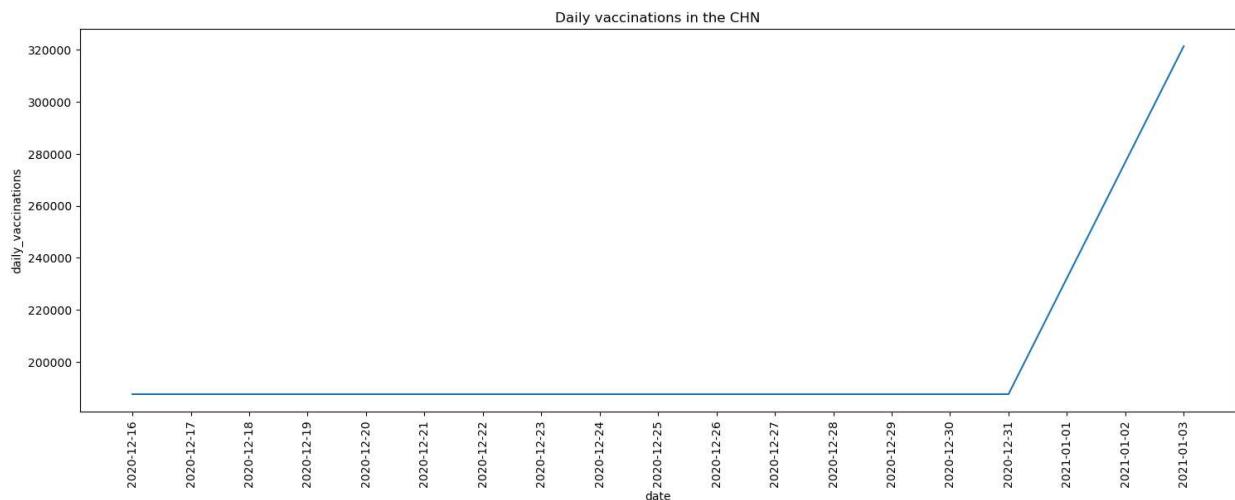


In [45]: *#Plot daily vaccinations as a function of date*

```
plt.figure(figsize=(18,6))
sns.lineplot(data=df_CHN, x="date", y="daily_vaccinations")

plt.xticks(rotation=90)
plt.title("Daily vaccinations in the CHN")
```

Out[45]:



In [49]: *#PAKISTAN*

```
df_PAK = df[df["iso_code"] == 'PAK'].copy()
df_PAK=df_PAK.head(20)
df_PAK
```

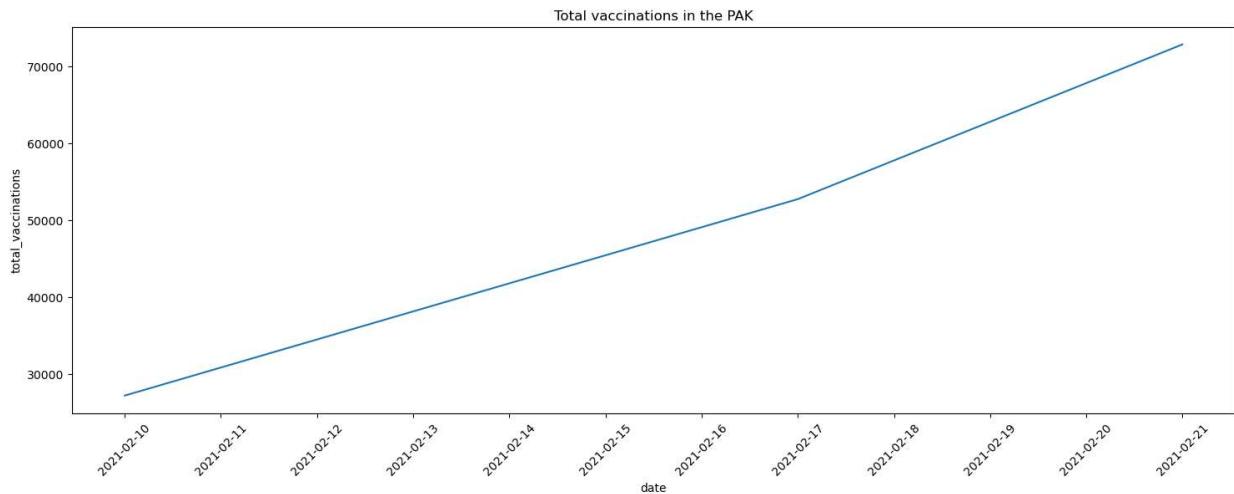
Out[49]:

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations
59449	Pakistan	PAK	2021-02-02	0.0	0.0		NaN
59450	Pakistan	PAK	2021-02-03	NaN	NaN		NaN
59451	Pakistan	PAK	2021-02-04	NaN	NaN		NaN
59452	Pakistan	PAK	2021-02-05	NaN	NaN		NaN
59453	Pakistan	PAK	2021-02-06	NaN	NaN		NaN
59454	Pakistan	PAK	2021-02-07	NaN	NaN		NaN
59455	Pakistan	PAK	2021-02-08	NaN	NaN		NaN
59456	Pakistan	PAK	2021-02-09	NaN	NaN		NaN
59457	Pakistan	PAK	2021-02-10	27228.0	27228.0		NaN
59458	Pakistan	PAK	2021-02-11	NaN	NaN		NaN

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations
59459	Pakistan	PAK	2021-02-12	NaN	NaN	NaN	NaN
59460	Pakistan	PAK	2021-02-13	NaN	NaN	NaN	NaN
59461	Pakistan	PAK	2021-02-14	NaN	NaN	NaN	NaN
59462	Pakistan	PAK	2021-02-15	NaN	NaN	NaN	NaN
59463	Pakistan	PAK	2021-02-16	NaN	NaN	NaN	NaN
59464	Pakistan	PAK	2021-02-17	52768.0	52768.0	52768.0	NaN
59465	Pakistan	PAK	2021-02-18	NaN	NaN	NaN	NaN
59466	Pakistan	PAK	2021-02-19	NaN	NaN	NaN	NaN
59467	Pakistan	PAK	2021-02-20	NaN	NaN	NaN	NaN
59468	Pakistan	PAK	2021-02-21	72882.0	72882.0	72882.0	NaN

```
In [50]: #Drop the dates with missing values, previously NaN that we filled with 0.
df_PAK.drop(df_PAK.index[df_PAK['total_vaccinations'] == 0], inplace = True)
```

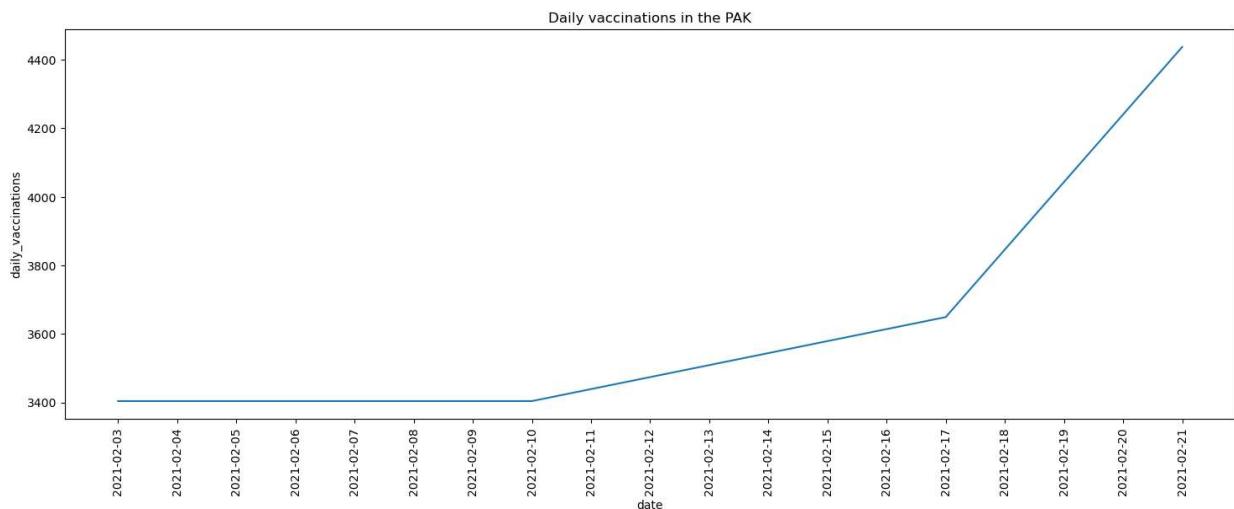
```
In [51]: #Plot total vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_PAK, x="date", y="total_vaccinations")
plt.title("Total vaccinations in the PAK")
plt.xticks(rotation=45)
plt.show()
```



```
In [52]: #Plot daily vaccinations as a function of date
plt.figure(figsize=(18,6))
sns.lineplot(data=df_PAK, x="date", y="daily_vaccinations")

plt.xticks(rotation=90)
plt.title("Daily vaccinations in the PAK")
```

Out[52]: Text(0.5, 1.0, 'Daily vaccinations in the PAK')



```
In [53]: #Group by total vaccinations given by country and sort descending to identify the top
vacc_by_country = df.groupby('country').max().sort_values('total_vaccinations', ascending=False)
vacc_by_country = vacc_by_country.iloc[:10]
vacc_by_country
```

Out[53]:

	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vacci
country						

country						
China	CHN	2022-03-29	3.263129e+09	1.275541e+09	1.240777e+09	
India	IND	2022-03-29	1.834501e+09	9.848381e+08	8.282295e+08	
United States	USA	2022-03-28	5.601818e+08	2.553624e+08	2.174990e+08	
Brazil	BRA	2022-03-29	4.135596e+08	1.810781e+08	1.602729e+08	
Indonesia	IDN	2022-03-29	3.771089e+08	1.962409e+08	1.588305e+08	
Japan	JPN	2022-03-29	2.543456e+08	1.024675e+08	1.006337e+08	
Bangladesh	BGD	2022-03-29	2.436427e+08	1.275441e+08	1.077127e+08	
Pakistan	PAK	2022-03-10	2.193686e+08	1.280741e+08	1.018812e+08	
Vietnam	VNM	2022-03-22	2.031444e+08	7.994719e+07	7.775411e+07	
Mexico	MEX	2022-03-29	1.919079e+08	8.558029e+07	7.971176e+07	

< In [54]:

```
#Now sort by total vaccinations per 100
vacc_by_country = vacc_by_country.sort_values('total_vaccinations_per_hundred', ascending=False)
```



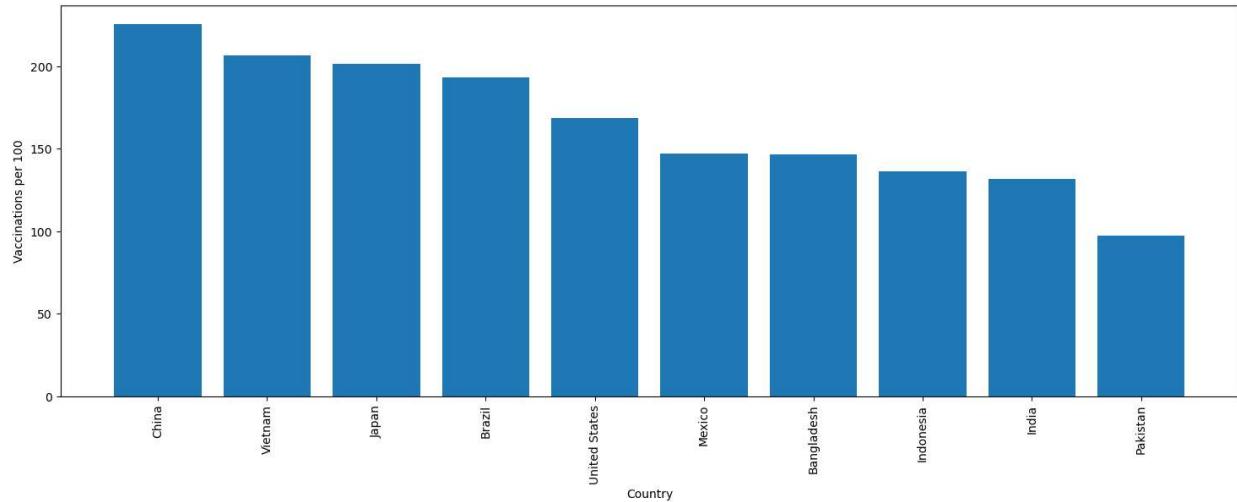
Out[54]:

	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vacci
country						
China	CHN	2022-03-29	3.263129e+09	1.275541e+09	1.240777e+09	
Vietnam	VNM	2022-03-22	2.031444e+08	7.994719e+07	7.775411e+07	
Japan	JPN	2022-03-29	2.543456e+08	1.024675e+08	1.006337e+08	
Brazil	BRA	2022-03-29	4.135596e+08	1.810781e+08	1.602729e+08	
United States	USA	2022-03-28	5.601818e+08	2.553624e+08	2.174990e+08	
Mexico	MEX	2022-03-29	1.919079e+08	8.558029e+07	7.971176e+07	
Bangladesh	BGD	2022-03-29	2.436427e+08	1.275441e+08	1.077127e+08	
Indonesia	IDN	2022-03-29	3.771089e+08	1.962409e+08	1.588305e+08	
India	IND	2022-03-29	1.834501e+09	9.848381e+08	8.282295e+08	
Pakistan	PAK	2022-03-10	2.193686e+08	1.280741e+08	1.018812e+08	

In [55]:

```
plt.figure(figsize=(18, 6))
plt.bar(vacc_by_country.index, vacc_by_country.total_vaccinations_per_hundred)

plt.xticks(rotation = 90)
plt.ylabel('Vaccinations per 100')
plt.xlabel('Country')
plt.show()
```



```
In [56]: total_vacc_by_country = df.groupby('country').max().sort_values('total_vaccinations', ascending=False).iloc[:10]
```

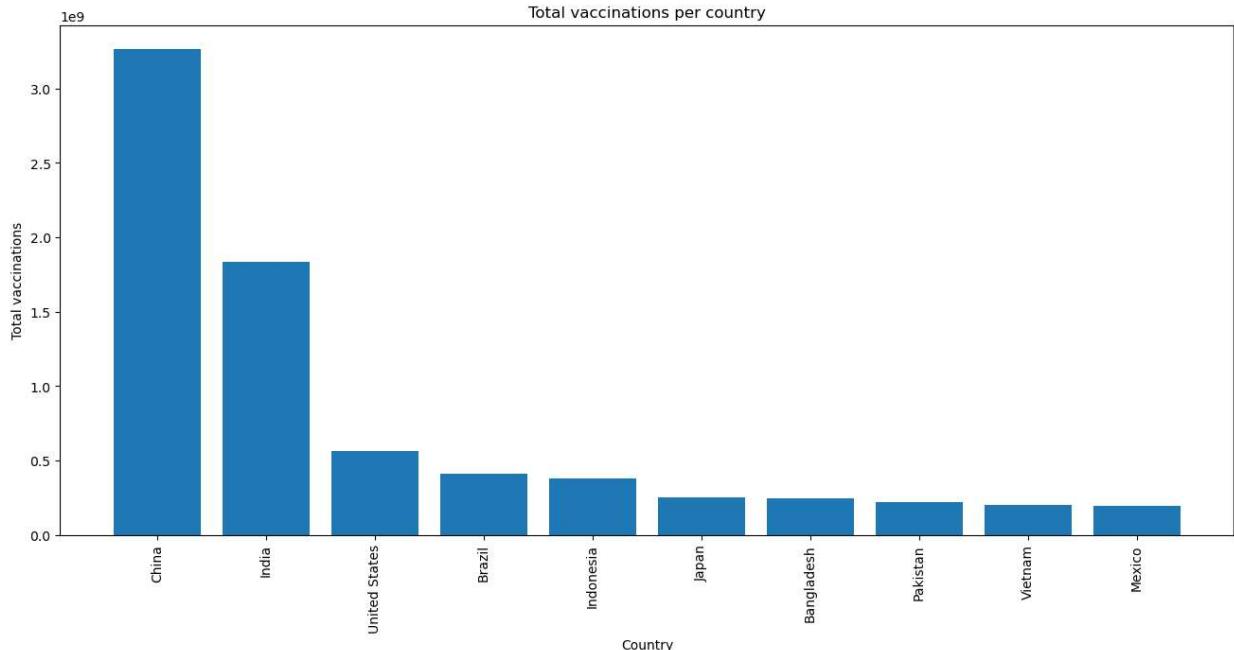
Out[56]:

	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vacci
country						
China	CHN	2022-03-29	3.263129e+09	1.275541e+09		1.240777e+09
India	IND	2022-03-29	1.834501e+09	9.848381e+08		8.282295e+08
United States	USA	2022-03-28	5.601818e+08	2.553624e+08		2.174990e+08
Brazil	BRA	2022-03-29	4.135596e+08	1.810781e+08		1.602729e+08
Indonesia	IDN	2022-03-29	3.771089e+08	1.962409e+08		1.588305e+08
Japan	JPN	2022-03-29	2.543456e+08	1.024675e+08		1.006337e+08
Bangladesh	BGD	2022-03-29	2.436427e+08	1.275441e+08		1.077127e+08
Pakistan	PAK	2022-03-10	2.193686e+08	1.280741e+08		1.018812e+08
Vietnam	VNM	2022-03-22	2.031444e+08	7.994719e+07		7.775411e+07
Mexico	MEX	2022-03-29	1.919079e+08	8.558029e+07		7.971176e+07

In [57]:

```
plt.figure(figsize=(16, 7))
plt.bar(total_vacc_by_country.index, total_vacc_by_country.total_vaccinations)

plt.title('Total vaccinations per country')
plt.xticks(rotation = 90)
plt.ylabel('Total vaccinations')
plt.xlabel('Country')
plt.show()
```



In [58]: `#Sort by total vaccinations delivered by countries and group by vaccines.`
`vacc_names_by_country = df.groupby('vaccines').max().sort_values('total_vaccinations',`
`vacc_names_by_country.head()`

Out[58]:

vaccines	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated
CanSino, Sinopharm/Beijing, Sinopharm/Wuhan, Sinovac, ZF2001	China	CHN	2022-03-29	3.263129e+09	1.275541e+09	1.2401e+09
Covaxin, Oxford/AstraZeneca, Sputnik V	India	IND	2022-03-29	1.834501e+09	9.848381e+08	8.2821e+08
Johnson&Johnson, Moderna, Pfizer/BioNTech	United States	USA	2022-03-28	5.601818e+08	2.553624e+08	2.1741e+08
Johnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac	Brazil	BRA	2022-03-29	4.135596e+08	1.810781e+08	1.6021e+08
Johnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac	Indonesia	IDN	2022-03-29	3.771089e+08	1.962409e+08	1.5881e+08

In [59]: `#Get the top 10 vaccines by country for easy plotting`
`vacc_names_by_country = vacc_names_by_country.iloc[:10]`
`vacc_names_by_country`

Out[59]:

vaccines	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_v
CanSino, Sinopharm/Beijing, Sinopharm/Wuhan, Sinovac, ZF2001	China	CHN	2022-03-29	3.263129e+09	1.275541e+09	1.24
Covaxin, Oxford/AstraZeneca, Sputnik V	India	IND	2022-03-29	1.834501e+09	9.848381e+08	8.28
Johnson&Johnson, Moderna, Pfizer/BioNTech	United States	USA	2022-03-28	5.601818e+08	2.553624e+08	2.17
Johnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac	Brazil	BRA	2022-03-29	4.135596e+08	1.810781e+08	1.60
Johnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac	Indonesia	IDN	2022-03-29	3.771089e+08	1.962409e+08	1.58
Moderna, Oxford/AstraZeneca, Pfizer/BioNTech	Wales	SXM	2022-03-29	2.543456e+08	1.024675e+08	1.00
Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac	Bangladesh	BGD	2022-03-29	2.436427e+08	1.275441e+08	1.07
CanSino, Covaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sinovac, Sputnik V	Pakistan	PAK	2022-03-10	2.193686e+08	1.280741e+08	1.01
Abdala, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing, Sputnik V	Vietnam	VNM	2022-03-22	2.031444e+08	7.994719e+07	7.77
CanSino, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac, Sputnik V	Mexico	MEX	2022-03-29	1.919079e+08	8.558029e+07	7.97

```
In [60]: #Reset index to move vaccines from being index to a column.
#This makes it easy for us to plot using Seaborn, especially if we want to sort by country
vacc_names_by_country=vacc_names_by_country.reset_index()
vacc_names_by_country
```

Out[60]:

	vaccines	country	iso_code	date	total_vaccinations	people_vaccinated	people_full
0	CanSino, Sinopharm/Beijing, Sinopharm/Wuhan, S...	China	CHN	2022-03-29	3.263129e+09	1.275541e+09	-
1	Covaxin, Oxford/AstraZeneca, Sputnik V	India	IND	2022-03-29	1.834501e+09	9.848381e+08	-
2	Johnson&Johnson, Moderna, Pfizer/BioNTech	United States	USA	2022-03-28	5.601818e+08	2.553624e+08	-
3	Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...	Brazil	BRA	2022-03-29	4.135596e+08	1.810781e+08	-
4	Johnson&Johnson, Moderna, Novavax, Oxford/Astr...	Indonesia	IDN	2022-03-29	3.771089e+08	1.962409e+08	-
5	Moderna, Oxford/AstraZeneca, Pfizer/BioNTech	Wales	SXM	2022-03-29	2.543456e+08	1.024675e+08	-
6	Johnson&Johnson, Moderna, Oxford/AstraZeneca,	Bangladesh	BGD	2022-03-29	2.436427e+08	1.275441e+08	-
...							
7	CanSino, Covaxin, Moderna, Oxford/AstraZeneca,...	Pakistan	PAK	2022-03-10	2.193686e+08	1.280741e+08	-
8	Abdala, Moderna, Oxford/AstraZeneca, Pfizer/Bi...	Vietnam	VNM	2022-03-22	2.031444e+08	7.994719e+07	-
9	CanSino, Johnson&Johnson, Moderna, Oxford/Astr...	Mexico	MEX	2022-03-29	1.919079e+08	8.558029e+07	-

```
In [61]: plt.figure(figsize=(12,8))

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

```
Out[61]: (array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),  
 [Text(0, 0, 'CanSino, Sinopharm/Beijing, Sinopharm/Wuhan, Sinovac, ZF2001'),  
  Text(1, 0, 'Covaxin, Oxford/AstraZeneca, Sputnik V'),  
  Text(2, 0, 'Johnson&Johnson, Moderna, Pfizer/BioNTech'),  
  Text(3, 0, 'Johnson&Johnson, Oxford/AstraZeneca, Pfizer/BioNTech, Sinovac'),  
  Text(4, 0, 'Johnson&Johnson, Moderna, Novavax, Oxford/AstraZeneca, Pfizer/BioNTech,  
  Sinopharm/Beijing, Sinovac'),  
  Text(5, 0, 'Moderna, Oxford/AstraZeneca, Pfizer/BioNTech'),  
  Text(6, 0, 'Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinophar  
  m/Beijing, Sinovac'),  
  Text(7, 0, 'CanSino, Covaxin, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopha  
  rm/Beijing, Sinovac, Sputnik V'),  
  Text(8, 0, 'Abdala, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijin  
  g, Sputnik V'),  
  Text(9, 0, 'CanSino, Johnson&Johnson, Moderna, Oxford/AstraZeneca, Pfizer/BioNTech,  
  Sinovac, Sputnik V')])
```

