

```
In [421]: import pandas as pd
import matplotlib.pyplot as plt

#first download csv file from URL
#local path of csv file
USVaccURL="https://raw.githubusercontent.com/owid/covid-19-data/master/public/data/usa_vaccinations.csv"
#read CSV file to pandas dataframe
dfUSVacc=pd.read_csv(USVaccURL)
#print dataframe
dfUSVacc
```

Out[421]:

	location	date	vaccine	source_url	total_vaccinations	people_vaccinated
0	United States	2020-12-13	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	29326	24
1	United States	2020-12-14	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	33886	28
2	United States	2020-12-15	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	83579	76
3	United States	2020-12-16	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	243356	230
4	United States	2020-12-17	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	515783	495
...
449	United States	2022-03-07	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556258145	254356
450	United States	2022-03-08	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556451963	254407
451	United States	2022-03-09	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556631018	254455
452	United States	2022-03-10	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556756401	254486
453	United States	2022-03-11	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556777608	254492

454 rows × 8 columns

```
In [422]: dfUSVacc.columns
```

Out[422]:

```
Index(['location', 'date', 'vaccine', 'source_url', 'total_vaccination
s',
      'people_vaccinated', 'people_fully_vaccinated', 'total_boosters
...
```

In [423]: *#Inspect data*

```
dfUSVacc.head(10)
```

Out[423]:

	location	date	vaccine	source_url	total_vaccinations	people_vaccinated
0	United States	2020-12-13	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	29326	24448
1	United States	2020-12-14	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	33886	28849
2	United States	2020-12-15	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	83579	76213
3	United States	2020-12-16	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	243356	230601
4	United States	2020-12-17	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	515783	495911
5	United States	2020-12-18	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	931898	903477
6	United States	2020-12-19	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	1113670	1081326
7	United States	2020-12-20	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	1218868	1184589
8	United States	2020-12-21	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	1600537	1559731
9	United States	2020-12-22	Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	2047964	1998672

In [424]: dfUSVacc.tail(10)

Out[424]:

	location	date	vaccine	source_url	total_vaccinations	people_vaccinated
444	United States	2022-03-02	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	555288308	254093
445	United States	2022-03-03	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	555520897	254154

	location	date	vaccine	source_url	total_vaccinations	people_vaccina
446	United States	2022-03-04	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	555812647	254228
447	United States	2022-03-05	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	555981876	254276
448	United States	2022-03-06	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556064969	254302
449	United States	2022-03-07	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556258145	254356
450	United States	2022-03-08	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556451963	254407
451	United States	2022-03-09	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556631018	254455
452	United States	2022-03-10	Johnson&Johnson, Moderna, Pfizer/BioNTech	https://data.cdc.gov/Vaccinations/COVID-19-Vac...	556756401	254486

```
In [425]: #to get information about column of dataset
dfUSVacc.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 454 entries, 0 to 453
Data columns (total 8 columns):
 #   Column                                Non-Null Count  Dtype
---  -
 0   location                             454 non-null    object
 1   date                                 454 non-null    object
 2   vaccine                             454 non-null    object
 3   source_url                           454 non-null    object
 4   total_vaccinations                   454 non-null    int64
 5   people_vaccinated                   454 non-null    int64
 6   people_fully_vaccinated              454 non-null    int64
 7   total_boosters                       454 non-null    int64
dtypes: int64(4), object(4)
memory usage: 28.5+ KB
```

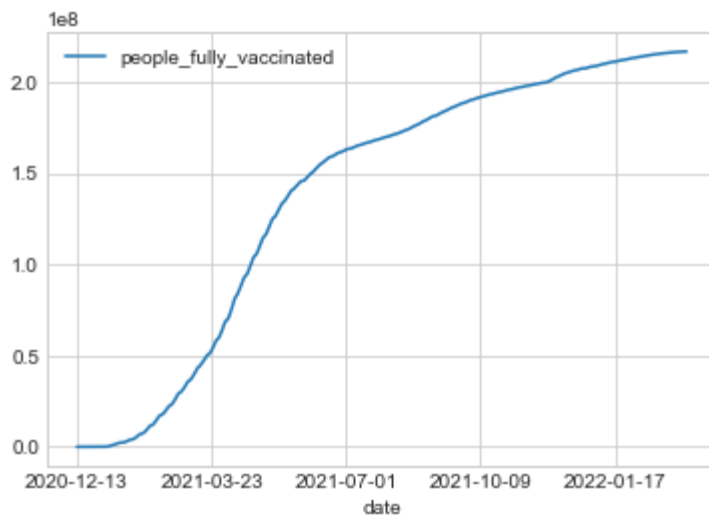
```
In [426]: #removed unnecessary columns (vaccine,source_url)
dfUSVacc.drop(['location','vaccine','source_url','total_vaccinations','to
```

```
In [427]: #Columns after removed unnecessary columns
dfUSVacc.columns
```

```
Out[427]: Index(['date', 'people_fully_vaccinated'], dtype='object')
```

```
In [428]: plt.style.use('seaborn-whitegrid')
dfUSVacc.plot.line(x='date', y=['people_fully_vaccinated'])
```

```
Out[428]: <AxesSubplot:xlabel='date'>
```



```
In [429]: #Columns information after removed unnecessary columns
dfUSVacc.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 454 entries, 0 to 453
Data columns (total 2 columns):
 #   Column                      Non-Null Count  Dtype
---  -
 0   date                        454 non-null   object
 1   people_fully_vaccinated     454 non-null   int64
dtypes: int64(1), object(1)
memory usage: 7.2+ KB
```

```
In [430]: dfUSVacc
```

```
Out[430]:
```

	date	people_fully_vaccinated
0	2020-12-13	5621
1	2020-12-14	5740
2	2020-12-15	6000
3	2020-12-16	6472
4	2020-12-17	7216
...
449	2022-03-07	216443742
450	2022-03-08	216496541
451	2022-03-09	216546240
452	2022-03-10	216581385
453	2022-03-11	216587984

454 rows × 2 columns

In [431]: dfUSVacc.describe()

Out[431]:

	people_fully_vaccinated
count	4.540000e+02
mean	1.385327e+08
std	7.509866e+07
min	5.621000e+03
25%	7.185233e+07
50%	1.689962e+08
75%	1.985548e+08
max	2.165880e+08

```
In [432]: #Monthly grouping
dfUSVacc['date'] = pd.to_datetime(dfUSVacc['date'])
dfUSVacc = dfUSVacc.sort_values(by='date')
dfUSVacc=dfUSVacc.groupby(pd.DatetimeIndex(dfUSVacc.date).to_period('M'))
dfUSVacc.set_index('date', inplace=True)
#dfUSVacc.to_csv('dfUSVacc.csv')
#dfUSVacc
```

In [433]: dfUSVacc

Out[433]:

	people_fully_vaccinated
date	
2020-12-31	40563
2021-01-31	7348534
2021-02-28	29856080
2021-03-31	64151041
2021-04-30	114113909
2021-05-31	145930115
2021-06-30	162437860
2021-07-31	169848427
2021-08-31	179605883
2021-09-30	189456150
2021-10-31	195708071
2021-11-30	200716297
2021-12-31	208368699
2022-01-31	213083575

people_fully_vaccinated**date**

```
In [434]: #load covid19 death,confirmed and recovered data
urlCovid19='https://raw.githubusercontent.com/datasets/covid-19/master/data/countries.csv'
dfUSCovid19Data=pd.read_csv(urlCovid19)
dfUSCovid19Data.rename(columns={'Date': 'date'}, inplace=True)
# Print Initial dataset of covid-19
dfUSCovid19Data
```

Out[434]:

	date	Country	Confirmed	Recovered	Deaths
0	2020-01-22	Afghanistan	0	0	0
1	2020-01-23	Afghanistan	0	0	0
2	2020-01-24	Afghanistan	0	0	0
3	2020-01-25	Afghanistan	0	0	0
4	2020-01-26	Afghanistan	0	0	0
...
154633	2022-03-08	Zimbabwe	240343	0	5400
154634	2022-03-09	Zimbabwe	240343	0	5400
154635	2022-03-10	Zimbabwe	241548	0	5408
154636	2022-03-11	Zimbabwe	241548	0	5408
154637	2022-03-12	Zimbabwe	242069	0	5412

154638 rows × 5 columns

```
In [435]: #Initial total columns of dfUSCovid19Data dataset
dfUSCovid19Data.columns
```

```
Out[435]: Index(['date', 'Country', 'Confirmed', 'Recovered', 'Deaths'], dtype='object')
```

```
In [436]: #Initial columns info of dfUSCovid19Data dataset
dfUSCovid19Data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 154638 entries, 0 to 154637
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  -
0   date        154638 non-null object
1   Country     154638 non-null object
2   Confirmed   154638 non-null int64
3   Recovered   154638 non-null int64
4   Deaths      154638 non-null int64
dtypes: int64(3), object(2)
memory usage: 5.9+ MB
```

```
In [437]: #filtering data for considering only USA data
dfUSCovid19Data=dfUSCovid19Data.loc[(dfUSCovid19Data['Country'] == 'US')
```

```
In [438]: # Printing Covid-19 USA Data
dfUSCovid19Data
```

Out[438]:

	date	Country	Confirmed	Recovered	Deaths
142933	2020-02-01	US	8	0	0
142934	2020-02-02	US	8	0	0
142935	2020-02-03	US	11	0	0
142936	2020-02-04	US	11	0	0
142937	2020-02-05	US	11	0	0
...
143687	2022-02-24	US	78812640	0	946099
143688	2022-02-25	US	78887236	0	948130
143689	2022-02-26	US	78934671	0	948826
143690	2022-02-27	US	78950518	0	949018
143691	2022-02-28	US	79047371	0	951114

759 rows × 5 columns

```
In [439]: dfUSCovid19Data.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 759 entries, 142933 to 143691
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype
---  -
 0   date        759 non-null    object
 1   Country     759 non-null    object
 2   Confirmed   759 non-null    int64
 3   Recovered   759 non-null    int64
 4   Deaths     759 non-null    int64
dtypes: int64(3), object(2)
memory usage: 35.6+ KB
```

```
In [440]: #removed unnecessary columns, calculating death rate, and grouping
dfUSCovid19Data.drop(['Recovered'],axis=1,inplace=True)
dfUSCovid19Data['Deaths'] = (dfUSCovid19Data['Deaths']/dfUSCovid19Data['Confirmed'])
dfUSCovid19Data['date'] = pd.to_datetime(dfUSCovid19Data['date'])
dfUSCovid19Data=dfUSCovid19Data.groupby(pd.Grouper(key='date', axis=0, freq='D'))
```

```
C:\Users\Taslina Akter\anaconda3\lib\site-packages\pandas\core\frame.p
y:4906: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
return super().drop(
C:\Users\TASLIM~1\AppData\Local\Temp\ipykernel_19284\1664610843.py:3:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
dfUSCovid19Data['Deaths'] = (dfUSCovid19Data['Deaths']/dfUSCovid19D
ata['Confirmed'])*1
C:\Users\TASLIM~1\AppData\Local\Temp\ipykernel_19284\1664610843.py:4:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
In [441]: # Printing covid-19 Data of USA after removing unnecessary columns and gr
dfUSCovid19Data
```

Out[441]:

	Confirmed	Deaths
date		
2020-02-29	402	0.040000
2020-03-31	1121455	1.073644
2020-04-30	19835424	1.508744
2020-05-31	45294659	1.917086
2020-06-30	64822529	1.663059
2020-07-31	111086834	1.225709
2020-08-31	166531654	0.981105
2020-09-30	199608857	0.884212
2020-10-31	251226672	0.839544
2020-11-30	338160262	0.671164
2020-12-31	527988498	0.570083
2021-01-31	733787758	0.524889
2021-02-28	776848931	0.490191
2021-03-31	919661679	0.559992
2021-04-30	947668934	0.536053

	Confirmed	Deaths
date		
2021-05-31	1023439204	0.549538
2021-06-30	1007947426	0.535629
2021-07-31	1061524357	0.550731
2021-08-31	1151033303	0.521806
2021-09-30	1250238382	0.482108
2021-10-31	1392850378	0.500566
2021-11-30	1419741886	0.485748
2021-12-31	1578107825	0.491237

```
In [442]: #before vaccination in the year 2020 , death rate
dfUSCovid19Data.drop(['Confirmed'],axis=1,inplace=True)
print(dfUSCovid19Data)
#dfUSCovid19Data.to_csv('dfUSCovid19Data.csv')
#dfUSCovid19Data.rename(columns={'Confirmed': 'covid_postive'}, inplace=T
```

	Deaths
date	
2020-02-29	0.040000
2020-03-31	1.073644
2020-04-30	1.508744
2020-05-31	1.917086
2020-06-30	1.663059
2020-07-31	1.225709
2020-08-31	0.981105
2020-09-30	0.884212
2020-10-31	0.839544
2020-11-30	0.671164
2020-12-31	0.570083
2021-01-31	0.524889
2021-02-28	0.490191
2021-03-31	0.559992
2021-04-30	0.536053
2021-05-31	0.549538
2021-06-30	0.535629
2021-07-31	0.550731
2021-08-31	0.521806
2021-09-30	0.482108
2021-10-31	0.500566
2021-11-30	0.485748
2021-12-31	0.491237
2022-01-31	0.406118
2022-02-28	0.333173

```
In [443]: dfUSCovid19Data.describe()
```

Out[443]:

	Deaths
count	25.000000

Deaths

mean	0.733685
std	0.440583
min	0.040000
25%	0.491237
50%	0.549538
75%	0.884212

```
In [447]: plt.style.use('seaborn-whitegrid')
dfUSCovid19Data.plot.bar()
```

```
Out[447]: <AxesSubplot:xlabel='date'>
```

