In [1]:
```
import os #provides functions for interacting with the operating system
import numpy as np
import pandas as pd
from datetime import datetime
```

In [26]:
```
# loading data right from the source:
#raw_data_deaths = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv')
#raw_data_confirmed = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv')
#raw_data_Recovered = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_recovered_global.csv')
#country_df = pd.read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/web-data/data/cases_country.csv')

raw_data_confirmed = pd.read_csv('D:\\csse_covid_19_time_series\\time_series_covid19_confirmed_global.csv')
raw_data_deaths = pd.read_csv('D:\\csse_covid_19_time_series\\time_series_covid19_deaths_global.csv')
raw_data_Recovered = pd.read_csv('D:\\csse_covid_19_time_series\\time_series_covid19_recovered_global.csv')
```

In [27]:
```
print("The Shape of Conrimred is: ", raw_data_confirmed.shape)
print("The Shape of Conrimred is: ", raw_data_deaths.shape)
print("The Shape of Conrimred is: ", raw_data_Recovered.shape)

raw_data_confirmed.head()
```

```
The Shape of Conrimred is:  (280, 689)
The Shape of Conrimred is:  (280, 689)
The Shape of Conrimred is:  (265, 689)
```

Out[27]:

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	…	11/27/21	11/28/21	11/29/21	11/30/21	12/1/21	12/2/21	12/3/21	12/4/21	12/5/21	12/6/21
0	NaN	Afghanistan	33.9391	67.709953	0	0	0	0	0	0	…	157190	157318	157260	157289	157359	157387	157412	157431	157445	157499
1	NaN	Albania	41.15330	20.168300	0	0	0	0	0	0	…	199137	199555	199750	199945	200173	200639	201045	201402	201730	201902
2	NaN	Algeria	28.03390	1.659600	0	0	0	0	0	0	…	209980	210152	210344	210531	210723	210921	211112	211297	211469	211662
3	NaN	Andorra	42.50630	1.521800	0	0	0	0	0	0	…	16712	16712	16712	17115	17426	17658	18010	18010	18010	18031
4	NaN	Angola	-11.20270	17.873900	0	0	0	0	0	0	…	65139	65144	65155	65168	65183	65208	65223	65244	65259	65259

5 rows × 689 columns

In [28]:
```
# Un-Pivoting the data

raw_data_confirmed2 = pd.melt(raw_data_confirmed, id_vars=['Province/State', 'Country/Region', 'Lat', 'Long'], var_name='Date')
raw_data_deaths2 = pd.melt(raw_data_deaths, id_vars=['Province/State', 'Country/Region', 'Lat', 'Long'], var_name='Date')
raw_data_Recovered2 = pd.melt(raw_data_Recovered, id_vars=['Province/State', 'Country/Region', 'Lat', 'Long'], var_name='Date')

print("The Shape of Conrimred is: ", raw_data_confirmed2.shape)
print("The Shape of Conrimred is: ", raw_data_deaths2.shape)
print("The Shape of Conrimred is: ", raw_data_Recovered2.shape)

raw_data_confirmed2.head()
```

```
The Shape of Conrimred is:  (191800, 6)
The Shape of Conrimred is:  (191800, 6)
The Shape of Conrimred is:  (181525, 6)
```

Out[28]:

	Province/State	Country/Region	Lat	Long	Date	value
0	NaN	Afghanistan	33.93911	67.709953	1/22/20	0
1	NaN	Albania	41.15330	20.168300	1/22/20	0
2	NaN	Algeria	28.03390	1.659600	1/22/20	0
3	NaN	Andorra	42.50630	1.521800	1/22/20	0
4	NaN	Angola	-11.20270	17.873900	1/22/20	0

In [29]:
```
# Converting the new column to dates

raw_data_confirmed2['Date'] = pd.to_datetime(raw_data_confirmed2['Date'])
raw_data_deaths2['Date'] = pd.to_datetime(raw_data_deaths2['Date'])
raw_data_Recovered2['Date'] = pd.to_datetime(raw_data_Recovered2['Date'])
```

In [30]:
```
# Renaming the values
raw_data_confirmed2.columns = raw_data_confirmed2.columns.str.replace('value', 'Confirmed')
raw_data_deaths2.columns = raw_data_deaths2.columns.str.replace('value', 'Deaths')
raw_data_Recovered2.columns = raw_data_Recovered2.columns.str.replace('value', 'Recovered')
```

In [31]:
```
# Investigating the NULL values
raw_data_Recovered2.isnull().sum()
```

Out[31]:
```
Province/State    132890
Country/Region         0
Lat                  685
Long                 685
Date                   0
Recovered              0
dtype: int64
```

In [32]:
```
# Dealing with NULL values

raw_data_confirmed2['Province/State'].fillna(raw_data_confirmed2['Country/Region'], inplace=True)
raw_data_deaths2['Province/State'].fillna(raw_data_deaths2['Country/Region'], inplace=True)
raw_data_Recovered2['Province/State'].fillna(raw_data_Recovered2['Country/Region'], inplace=True)

raw_data_confirmed2.isnull().sum()
```

Out[32]:
```
Province/State       0
Country/Region       0
Lat                  0
Long                 0
Date                 0
Confirmed            0
dtype: int64
```

In [33]:
```
# printing shapes before the join
print("The Shape of Conrimred is: ", raw_data_confirmed2.shape)
print("The Shape of Conrimred is: ", raw_data_deaths2.shape)
print("The Shape of Conrimred is: ", raw_data_Recovered2.shape)
```

```
The Shape of Conrimred is:  (191800, 6)
The Shape of Conrimred is:  (191800, 6)
The Shape of Conrimred is:  (181525, 6)
```

In [34]:
```
raw_data_confirmed2.isnull().sum()
raw_data_deaths2.isnull().sum()
raw_data_Recovered2.isnull().sum()
```

Out[34]:
```
Province/State       0
Country/Region       0
Lat                  685
Long                 685
Date                 0
Recovered            0
dtype: int64
```

In [35]:
```
# Full Joins

# Confirmed with Deaths
full_join = raw_data_confirmed2.merge(raw_data_deaths2[['Province/State','Country/Region','Date','Deaths']],
                                      how = 'left',
                                      left_on = ['Province/State','Country/Region','Date'],
                                      right_on = ['Province/State','Country/Region','Date'])

print("Shape of first join: ", full_join.shape)

# full join with Recovered
full_join = full_join.merge(raw_data_Recovered2[['Province/State','Country/Region','Date','Recovered']],
                            how = 'left',
                            left_on = ['Province/State','Country/Region','Date'],
                            right_on = ['Province/State','Country/Region','Date']]

print("Shape of second join: ", full_join.shape)

full_join.head()
```

```
Shape of first join:  (191800, 7)
Shape of second join:  (191800, 8)
```

Out[35]:

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered
0	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-22	0	0	0.0
1	Albania	Albania	41.15330	20.168300	2020-01-22	0	0	0.0
2	Algeria	Algeria	28.03390	1.659600	2020-01-22	0	0	0.0
3	Andorra	Andorra	42.50630	1.521800	2020-01-22	0	0	0.0
4	Angola	Angola	-11.20270	17.873900	2020-01-22	0	0	0.0

In [36]:
```
full_join.isnull().sum()
```

Out[36]:
```
Province/State       0
Country/Region       0
Lat                  1370
Long                 1370
Date                 0
Confirmed            0
Deaths               0
Recovered            10960
dtype: int64
```

In [37]:
```
# Adding Month and Year as a new Column
full_join['Month-Year'] = full_join['Date'].dt.strftime('%b-%Y')
```

In [38]:
```
full_join.head()
```

Out[38]:

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered	Month-Year
0	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-22	0	0	0.0	Jan-2020
1	Albania	Albania	41.15330	20.168300	2020-01-22	0	0	0.0	Jan-2020
2	Algeria	Algeria	28.03390	1.659600	2020-01-22	0	0	0.0	Jan-2020
3	Andorra	Andorra	42.50630	1.521800	2020-01-22	0	0	0.0	Jan-2020
4	Angola	Angola	-11.20270	17.873900	2020-01-22	0	0	0.0	Jan-2020

In [39]:
```
#################################################################################
###################### Braking the numbers by Day ###############################
#################################################################################

# filtering data to Anhui to give you an example

#creating a new df
test = full_join[full_join['Province/State'] == 'Anhui']

#creating a new df
full_join2 = test.copy()

#creating a new date columns : 1
full_join2['Date - 1'] = full_join2['Date'] + pd.Timedelta(days=1)
full_join2.rename(columns={'Confirmed' : 'Confirmed - 1', 'Deaths' : 'Deaths - 1', 'Recovered': 'Recovered - 1',
                           'Date' : 'Date Minus 1'}, inplace=True)

#Joing on the 2 DFs
full_join3 = test.merge(full_join2[['Province/State', 'Country/Region','Confirmed - 1', 'Deaths - 1',
                                    'Recovered - 1', 'Date - 1', 'Date Minus 1']], how = 'outer',
                        left_on = ['Province/State','Country/Region','Date'],
                        right_on = ['Province/State', 'Country/Region','Date - 1'])

# Additional Calculations
full_join3['Confirmed Daily'] = full_join3['Confirmed'] - full_join3['Confirmed - 1']


test.head()
full_join2.head()
full_join3.head()
```

Out[39]:

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered	Month-Year	Confirmed - 1	Deaths - 1	Recovered - 1	Date - 1	Date Minus 1	Confirmed Daily
0	Anhui	China	31.8257	117.2264	2020-01-22	1.0	0.0	0.0	Jan-2020	NaN	NaN	NaN	NaT	NaT	NaN
1	Anhui	China	31.8257	117.2264	2020-01-23	9.0	0.0	0.0	Jan-2020	1.0	0.0	0.0	2020-01-23	2020-01-22	8.0
2	Anhui	China	31.8257	117.2264	2020-01-24	15.0	0.0	0.0	Jan-2020	9.0	0.0	0.0	2020-01-24	2020-01-23	6.0
3	Anhui	China	31.8257	117.2264	2020-01-25	39.0	0.0	0.0	Jan-2020	15.0	0.0	0.0	2020-01-25	2020-01-24	24.0
4	Anhui	China	31.8257	117.2264	2020-01-26	60.0	0.0	0.0	Jan-2020	39.0	0.0	0.0	2020-01-26	2020-01-25	21.0

In [40]:
```
test.head()
```

Out[40]:

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered	Month-Year
58	Anhui	China	31.8257	117.2264	2020-01-22	1	0	0.0	Jan-2020
338	Anhui	China	31.8257	117.2264	2020-01-23	9	0	0.0	Jan-2020
618	Anhui	China	31.8257	117.2264	2020-01-24	15	0	0.0	Jan-2020
898	Anhui	China	31.8257	117.2264	2020-01-25	39	0	0.0	Jan-2020
1178	Anhui	China	31.8257	117.2264	2020-01-26	60	0	0.0	Jan-2020

In [41]:
```
full_join2.head()
```

Out[41]:

	Province/State	Country/Region	Lat	Long	Date Minus 1	Confirmed - 1	Deaths - 1	Recovered - 1	Month-Year	Date - 1
58	Anhui	China	31.8257	117.2264	2020-01-22	1	0	0.0	Jan-2020	2020-01-23
338	Anhui	China	31.8257	117.2264	2020-01-23	9	0	0.0	Jan-2020	2020-01-24
618	Anhui	China	31.8257	117.2264	2020-01-24	15	0	0.0	Jan-2020	2020-01-25
898	Anhui	China	31.8257	117.2264	2020-01-25	39	0	0.0	Jan-2020	2020-01-26
1178	Anhui	China	31.8257	117.2264	2020-01-26	60	0	0.0	Jan-2020	2020-01-27

In [42]:
```
## Applying it on all dataset

#creating a new df
full_join2 = full_join.copy()

#creating a new date columns : 1
full_join2['Date - 1'] = full_join2['Date'] + pd.Timedelta(days=1)
full_join2.rename(columns={'Confirmed' : 'Confirmed - 1', 'Deaths' : 'Deaths - 1', 'Recovered': 'Recovered - 1',
                           'Date' : 'Date Minus 1'}, inplace=True)

#Joing on the 2 DFs
full_join3 = full_join.merge(full_join2['Province/State', 'Country/Region','Confirmed - 1', 'Deaths - 1',
                                    'Recovered - 1', 'Date - 1', 'Date Minus 1'], how = 'left',
                        left_on = ['Province/State','Country/Region','Date'],
                        right_on = ['Province/State', 'Country/Region','Date - 1'])

#minus_oneDF.rename(columns={'Confirmed' : 'Confirmed - 1', 'Deaths' : 'Deaths - 1', 'Recovered': 'Recovered - 1'}, inplace=True)

full_join3.head()

# Additional Calculations
full_join3['Confirmed Daily'] = full_join3['Confirmed'] - full_join3['Confirmed - 1']
full_join3['Deaths Daily'] = full_join3['Deaths'] - full_join3['Deaths - 1']
full_join3['Recovered Daily'] = full_join3['Recovered'] - full_join3['Recovered - 1']

print(full_join3.shape)
```

```
(191800, 17)
```

In [43]:
```
full_join3.head()
```

Out[43]:

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered	Month-Year	Confirmed - 1	Deaths - 1	Recovered - 1	Date - 1	Date Minus 1	Confirmed Daily	Deaths Daily	Recovered Daily
0	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-22	0	0	0.0	Jan-2020	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN
1	Albania	Albania	41.15330	20.168300	2020-01-22	0	0	0.0	Jan-2020	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN
2	Algeria	Algeria	28.03390	1.659600	2020-01-22	0	0	0.0	Jan-2020	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN
3	Andorra	Andorra	42.50630	1.521800	2020-01-22	0	0	0.0	Jan-2020	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN
4	Angola	Angola	-11.20270	17.873900	2020-01-22	0	0	0.0	Jan-2020	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN

In [44]:
```
full_join3['Confirmed Daily'].loc[full_join3['Date'] == '2020-01-22'] = full_join3['Confirmed']
full_join3['Deaths Daily'].loc[full_join3['Date'] == '2020-01-22'] = full_join3['Deaths']
full_join3['Recovered Daily'].loc[full_join3['Date'] == '2020-01-22'] = full_join3['Recovered']

# deleting columns
del full_join3['Confirmed - 1']
del full_join3['Deaths - 1']
del full_join3['Recovered - 1']
del full_join3['Date - 1']
del full_join3['Date Minus 1']
```

```
c:\users\diya\appdata\local\programs\python\python39\lib\site-packages\pandas\core\indexing.py:1732: 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
  self._setitem_single_block(indexer, value, name)
```

In [45]:
```
# Creating additional slicer for easy of use

full_join3['Hubei Vs Rest of the World'] = 'Rest of the World'
full_join3['Hubei Vs Rest of the World'].loc[full_join3['Province/State'] == 'Hubei'] = 'Hubei - Virus birth'

#full_join3[full_join3['Province/State'] == 'Hubei']
```

In [46]:
```
full_join3.head()
```

Out[46]:

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered	Month-Year	Confirmed Daily	Deaths Daily	Recovered Daily	Hubei Vs Rest of the World
0	Afghanistan	Afghanistan	33.93911	67.709953	2020-01-22	0	0	0.0	Jan-2020	0.0	0.0	0.0	Rest of the World
1	Albania	Albania	41.15330	20.168300	2020-01-22	0	0	0.0	Jan-2020	0.0	0.0	0.0	Rest of the World
2	Algeria	Algeria	28.03390	1.659600	2020-01-22	0	0	0.0	Jan-2020	0.0	0.0	0.0	Rest of the World
3	Andorra	Andorra	42.50630	1.521800	2020-01-22	0	0	0.0	Jan-2020	0.0	0.0	0.0	Rest of the World
4	Angola	Angola	-11.20270	17.873900	2020-01-22	0	0	0.0	Jan-2020	0.0	0.0	0.0	Rest of the World

In [48]:
```
path = "D:\\csse_covid_19_time_series\\"

# Changing my CWD
os.chdir(path)

full_join3.to_csv('CoronaVirus PowerBI Raw', sep='\t')
```

In [ ]:
```

```