```
import pandas as pd
import numpy as np
df=pd.read_csv('/content/drive/MyDrive/Datasets_ML/owid-covid-data (1).csv')
df
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0	NaN	Na
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0	NaN	Na
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0	NaN	Na
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0	NaN	Na
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0	NaN	Na
...	...	...	...	...	...	...	...	..
258742	ZWE	Africa	Zimbabwe	2023-02-15	263642.0	559.0	79.857	5662.
258743	ZWE	Africa	Zimbabwe	2023-02-16	263642.0	NaN	NaN	5662.
258744	ZWE	Africa	Zimbabwe	2023-02-17	263642.0	NaN	NaN	5662.
258745	ZWE	Africa	Zimbabwe	2023-02-18	263642.0	NaN	NaN	5662.
258746	ZWE	Africa	Zimbabwe	2023-02-19	263642.0	NaN	NaN	5662.

258747 rows × 67 columns

```
#First 5 observation display
df.head()
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0	NaN	NaN	
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0	NaN	NaN	
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0	NaN	NaN	
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0	NaN	NaN	
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0	NaN	NaN	

5 rows × 67 columns

```
#last 5 obsevation display
df.tail()
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths

```
#Row and columns
df.shape
```

```
(258747, 67)
```

```
#Column heading print
df.columns
```

```
Index(['iso_code', 'continent', 'location', 'date', 'total_cases', 'new_cases',
       'new_cases_smoothed', 'total_deaths', 'new_deaths',
       'new_deaths_smoothed', 'total_cases_per_million',
       'new_cases_per_million', 'new_cases_smoothed_per_million',
       'total_deaths_per_million', 'new_deaths_per_million',
       'new_deaths_smoothed_per_million', 'reproduction_rate', 'icu_patients',
       'icu_patients_per_million', 'hosp_patients',
       'hosp_patients_per_million', 'weekly_icu_admissions',
       'weekly_icu_admissions_per_million', 'weekly_hosp_admissions',
       'weekly_hosp_admissions_per_million', 'total_tests', 'new_tests',
       'total_tests_per_thousand', 'new_tests_per_thousand',
       'new_tests_smoothed', 'new_tests_smoothed_per_thousand',
       'positive_rate', 'tests_per_case', 'tests_units', 'total_vaccinations',
       'people_vaccinated', 'people_fully_vaccinated', 'total_boosters',
       'new_vaccinations', 'new_vaccinations_smoothed',
       'total_vaccinations_per_hundred', 'people_vaccinated_per_hundred',
       'people_fully_vaccinated_per_hundred', 'total_boosters_per_hundred',
       'new_vaccinations_smoothed_per_million',
       'new_people_vaccinated_smoothed',
       'new_people_vaccinated_smoothed_per_hundred', 'stringency_index',
       'population_density', 'median_age', 'aged_65_older', 'aged_70_older',
       'gdp_per_capita', 'extreme_poverty', 'cardiovasc_death_rate',
       'diabetes_prevalence', 'female_smokers', 'male_smokers',
       'handwashing_facilities', 'hospital_beds_per_thousand',
       'life_expectancy', 'human_development_index', 'population',
       'excess_mortality_cumulative_absolute', 'excess_mortality_cumulative',
       'excess_mortality', 'excess_mortality_cumulative_per_million'],
      dtype='object')
```

```
#To finding missing values
df.isna().sum()
```

```
iso_code                                       0
continent                                  14519
location                                       0
date                                           0
total_cases                                14568
                                           ...
population                                  1109
excess_mortality_cumulative_absolute      250098
excess_mortality_cumulative               250098
excess_mortality                          250098
excess_mortality_cumulative_per_million   250098
Length: 67, dtype: int64
```

```
#Basic Summary of data
df.describe()
```

	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
count	2.441790e+05	2.438290e+05	2.426250e+05	2.245060e+05	224387.000000	223201.000000
mean	5.297678e+06	1.178470e+04	1.183299e+04	7.991636e+04	127.078841	127.652062
std	3.257239e+07	8.204663e+04	7.976279e+04	4.071267e+05	737.116423	683.281768
min	1.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00	0.000000	0.000000
25%	5.754000e+03	0.000000e+00	4.429000e+00	1.250000e+02	0.000000	0.000000
50%	6.213800e+04	3.700000e+01	7.714300e+01	1.361000e+03	0.000000	1.143000
75%	6.443560e+05	8.220000e+02	1.003000e+03	1.100000e+04	11.000000	13.429000
max	6.739415e+08	4.082890e+06	3.436562e+06	6.862848e+06	60900.000000	14860.286000

8 rows × 62 columns

```
#Dropping the column
df.drop(['new_cases_smoothed','new_deaths_smoothed','new_cases_per_million','total_cases_per_million'],axis=1,inplace=True)
df
```

	iso_code	continent	location	date	total_cases	new_cases	total_deaths	new_deaths	new_ca
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0	NaN	NaN	
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0	NaN	NaN	
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0	NaN	NaN	
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0	NaN	NaN	
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0	NaN	NaN	
...	...	...	...	...	...	...	...	...	
258742	ZWE	Africa	Zimbabwe	2023-02-15	263642.0	559.0	5662.0	3.0	
258743	ZWE	Africa	Zimbabwe	2023-02-16	263642.0	NaN	5662.0	0.0	
258744	ZWE	Africa	Zimbabwe	2023-02-17	263642.0	NaN	5662.0	0.0	
258745	ZWE	Africa	Zimbabwe	2023-02-18	263642.0	NaN	5662.0	0.0	
258746	ZWE	Africa	Zimbabwe	2023-02-19	263642.0	NaN	5662.0	0.0	

```
#After dropping shape of column
df.shape
```

```
(258747, 63)
```

```
#Renaming the column name
df.rename(columns={'date':'Date','location':'Country','continent':'Continent','iso_code':'ISO_code'},inplace=True)
df
```

	ISO_code	Continent	Country	Date	total_cases	new_cases	total_deaths	new_deaths	new_ca
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0	NaN	NaN	
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0	NaN	NaN	
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0	NaN	NaN	
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0	NaN	NaN	
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0	NaN	NaN	
...	...	...	...	...	...	...	...	...	
258742	ZWE	Africa	Zimbabwe	2023-02-15	263642.0	559.0	5662.0	3.0	
258743	ZWE	Africa	Zimbabwe	2023-02-16	263642.0	NaN	5662.0	0.0	
258744	ZWE	Africa	Zimbabwe	2023-02-17	263642.0	NaN	5662.0	0.0	
258745	ZWE	Africa	Zimbabwe	2023-02-18	263642.0	NaN	5662.0	0.0	
258746	ZWE	Africa	Zimbabwe	2023-02-19	263642.0	NaN	5662.0	0.0	

258747 rows × 63 columns

```
#List the continent name
continent_unique=list(df.Continent.unique())
continent_unique
```

```
['Asia', nan, 'Europe', 'Africa', 'North America', 'South America', 'Oceania']
```

```
#Fill missing value
from sklearn.impute import SimpleImputer
imputer=SimpleImputer(strategy='constant')
df2=pd.DataFrame(imputer.fit_transform(df),columns=df.columns)
```

```
df2
```

	ISO_code	Continent	Country	Date	total_cases	new_cases	total_deaths	new_deaths	ne
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0	missing_value	missing_value	
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0	missing_value	missing_value	
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0	missing_value	missing_value	
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0	missing_value	missing_value	
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0	missing_value	missing_value	
...	...	...	...	...	...	...	...	...	
258742	ZWE	Africa	Zimbabwe	2023-02-15	263642.0	559.0	5662.0	3.0	
258743	ZWE	Africa	Zimbabwe	2023-02-16	263642.0	missing_value	5662.0	0.0	
258744	ZWE	Africa	Zimbabwe	2023-02-17	263642.0	missing_value	5662.0	0.0	
258745	ZWE	Africa	Zimbabwe	2023-02-18	263642.0	missing_value	5662.0	0.0	
258746	ZWE	Africa	Zimbabwe	2023-02-19	263642.0	missing_value	5662.0	0.0	

258747 rows × 63 columns

```
#Groupby
df3=df2.groupby(['Date','Country',])[['total_cases','total_deaths','total_vaccinations']].sum().reset_index()
df3
```

	Date	Country	total_cases	total_deaths	total_vaccinations
0	2020-01-01	Argentina	missing_value	missing_value	missing_value
1	2020-01-01	Mexico	missing_value	missing_value	missing_value
2	2020-01-02	Argentina	missing_value	missing_value	missing_value
3	2020-01-02	Mexico	missing_value	missing_value	missing_value
4	2020-01-03	Argentina	missing_value	missing_value	missing_value
...	...	...	...	...	...
258742	2023-02-19	Wallis and Futuna	3427.0	7.0	missing_value
258743	2023-02-19	World	673941526.0	6862848.0	13293920837.0
258744	2023-02-19	Yemen	11945.0	2159.0	missing_value
258745	2023-02-19	Zambia	342317.0	4051.0	missing_value
258746	2023-02-19	Zimbabwe	263642.0	5662.0	missing_value

258747 rows × 5 columns

```
#missing values to zero(0)-total_cases column
df3['total_cases'].replace({'missing_value':0},inplace=True)
```

```
#missing values to zero(0)-total_deaths column
df3['total_deaths'].replace({'missing_value':0},inplace=True)
```

```
#missing values to zero(0)-total_vaccinations column
df3['total_vaccinations'].replace({'missing_value':0},inplace=True)
```

```
df3
```

	Date	Country	total_cases	total_deaths	total_vaccinations
0	2020-01-01	Argentina	0.0	0.0	0.000000e+00
1	2020-01-01	Mexico	0.0	0.0	0.000000e+00
2	2020-01-02	Argentina	0.0	0.0	0.000000e+00
3	2020-01-02	Mexico	0.0	0.0	0.000000e+00
4	2020-01-03	Argentina	0.0	0.0	0.000000e+00
...	...	...	...	...	...
258742	2023-02-19	Wallis and Futuna	3427.0	7.0	0.000000e+00
258743	2023-02-19	World	673941526.0	6862848.0	1.329392e+10
258744	2023-02-19	Yemen	11945.0	2159.0	0.000000e+00
258745	2023-02-19	Zimbabwe	263642.0	5662.0	0.000000e+00

Plot subset of specific Data

```
df4=df3[df3['total_deaths']>1000000]
countries=df4['Country'].unique()
len(countries)
```

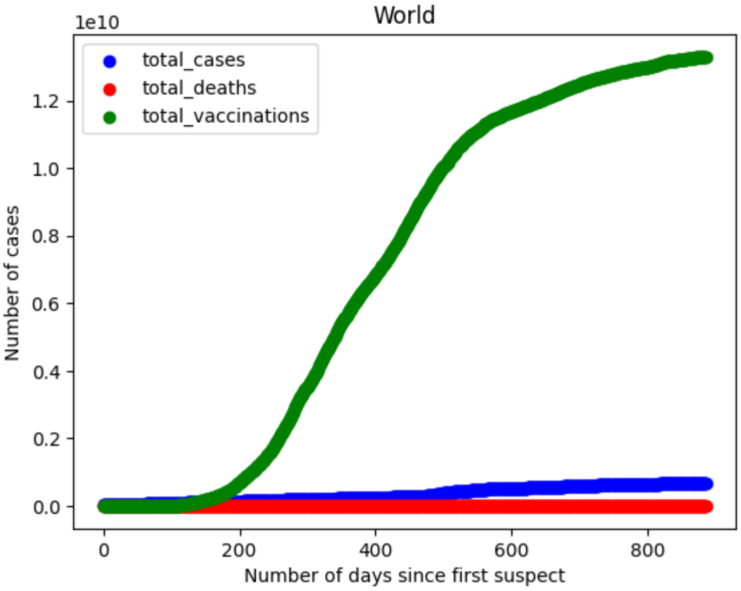
```
10
```

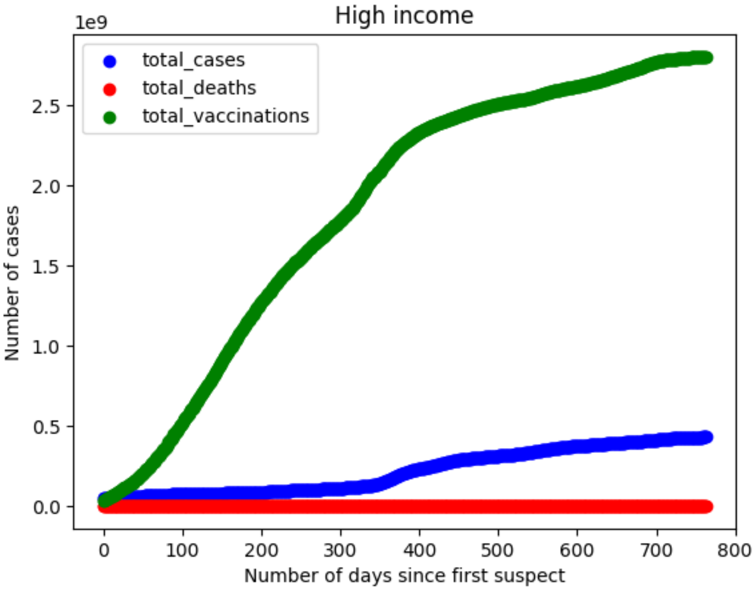
```
country_deaths_greaterthan1000000=list(df4.Country.unique())
country_deaths_greaterthan1000000
```

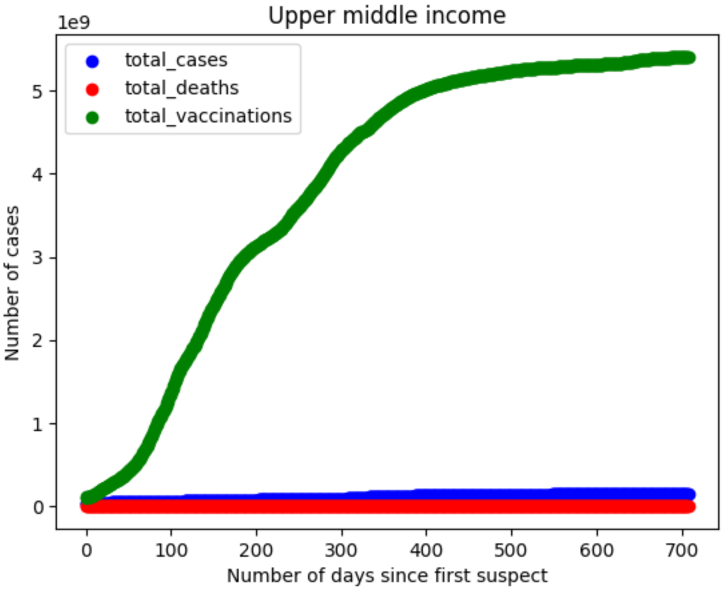
```
['World',
 'High income',
 'Upper middle income',
 'Europe',
 'South America',
 'Asia',
 'Lower middle income',
 'North America',
 'European Union',
 'United States']
```

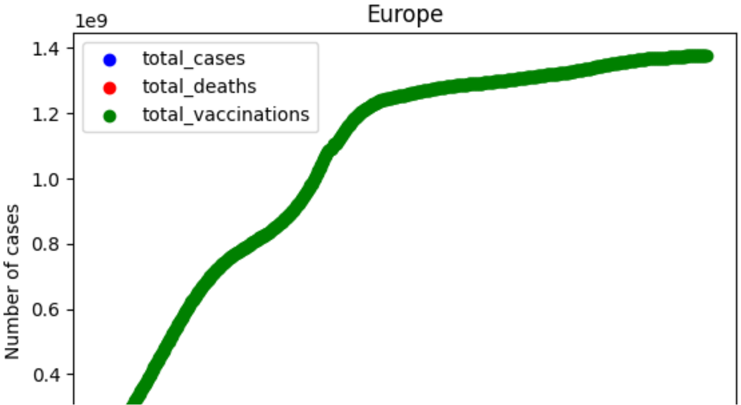
```
import matplotlib.pyplot as plt
```

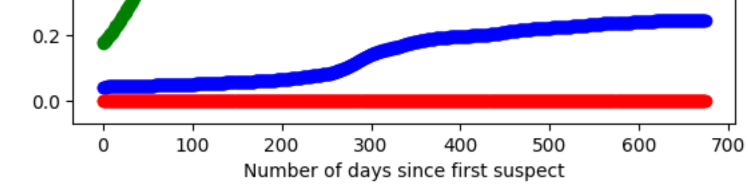
```
for idx in range(0,len(countries)):
  C = df4[df4['Country' ]==countries[idx]].reset_index()
  plt.scatter(np.arange(0,len(C)),C['total_cases'],color="blue",label="total_cases")
  plt.scatter(np.arange(0,len(C)),C['total_deaths'],color="red",label="total_deaths")
  plt.scatter(np.arange(0,len(C)),C['total_vaccinations'],color="green",label="total_vaccinations")
  plt.title(countries[idx])
  plt.xlabel("Number of days since first suspect")
  plt.ylabel("Number of cases")
  plt.legend()
  plt.show()
```

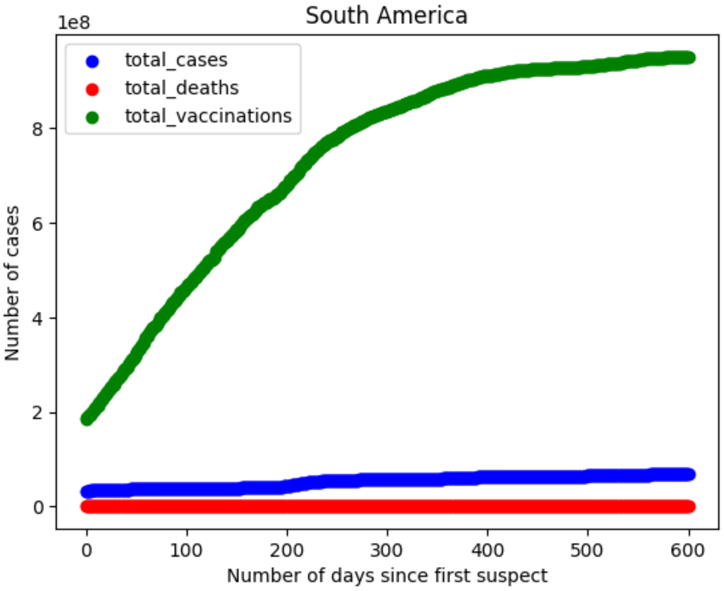

World
1e10
total_cases
total_deaths
total_vaccinations
Number of cases
0.0
0.2
0.4
0.6
0.8
1.0
1.2
0
200
400
600
800
Number of days since first suspect

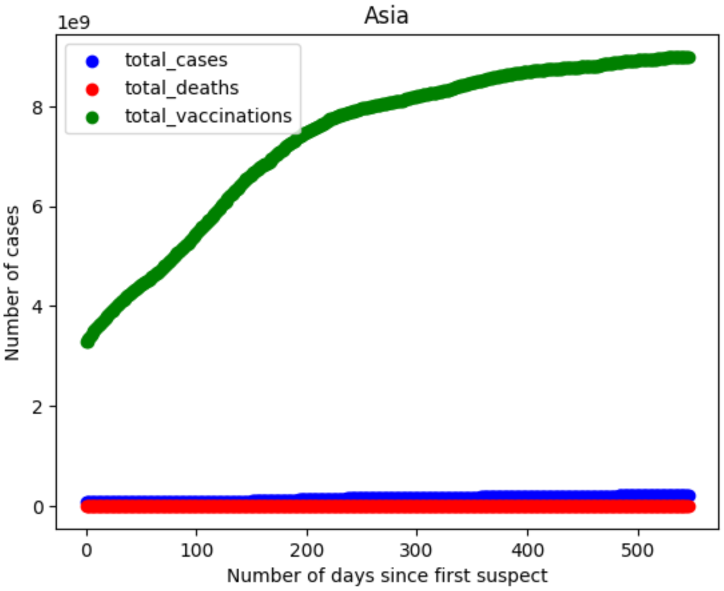

High income
1e9
total_cases
total_deaths
total_vaccinations
Number of cases
0.0
0.5
1.0
1.5
2.0
2.5
0
100
200
300
400
500
600
700
800
Number of days since first suspect

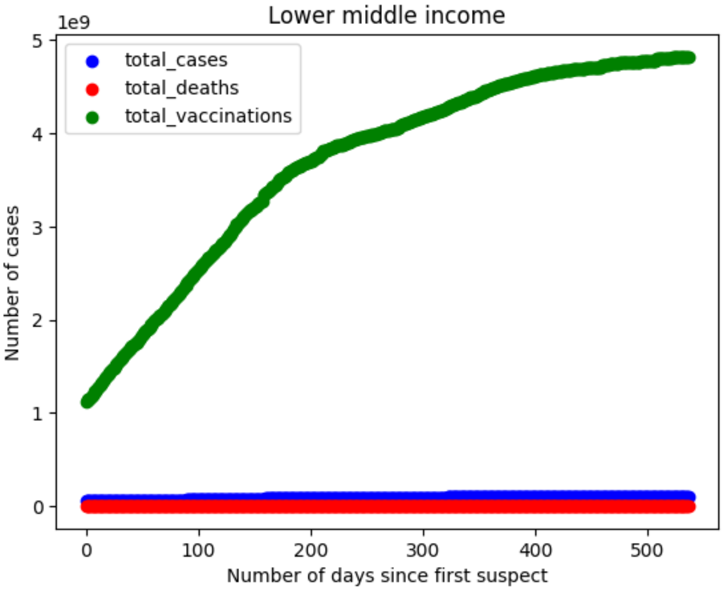

Upper middle income
1e9
total_cases
total_deaths
total_vaccinations
Number of cases
0
1
2
3
4
5
0
100
200
300
400
500
600
700
Number of days since first suspect

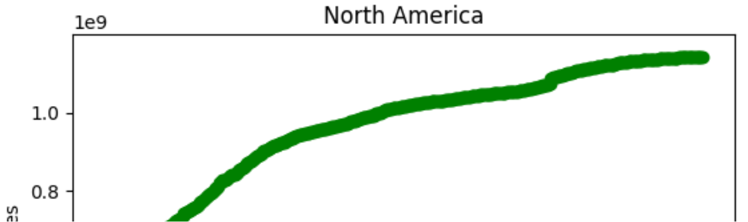

Europe
1e9
total_cases
total_deaths
total_vaccinations
Number of cases
0.4
0.6
0.8
1.0
1.2
1.4

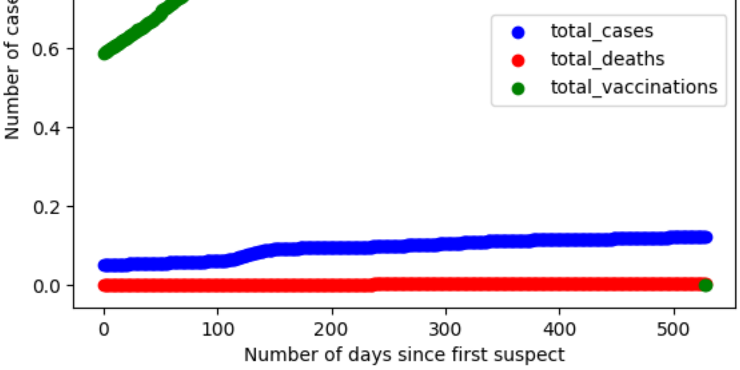

0.2
0.0
0
100
200
300
400
500
600
700
Number of days since first suspect

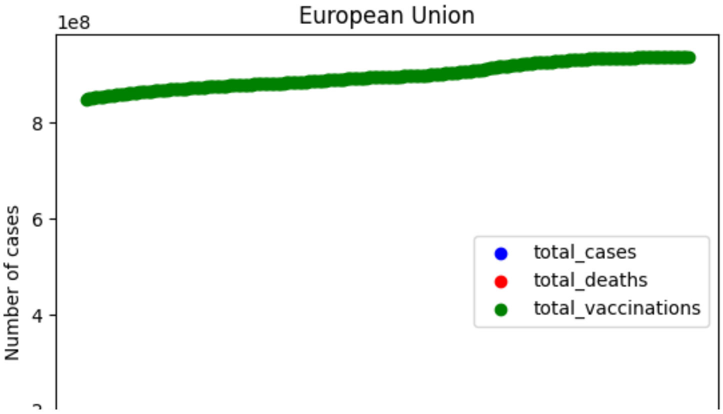

South America
1e8
total_cases
total_deaths
total_vaccinations
8
6
4
2
0
Number of cases
0
100
200
300
400
500
600
Number of days since first suspect


Asia
1e9
total_cases
total_deaths
total_vaccinations
8
6
4
2
0
Number of cases
0
100
200
300
400
500
Number of days since first suspect


Lower middle income
1e9
total_cases
total_deaths
total_vaccinations
5
4
3
2
1
0
Number of cases
0
100
200
300
400
500
Number of days since first suspect

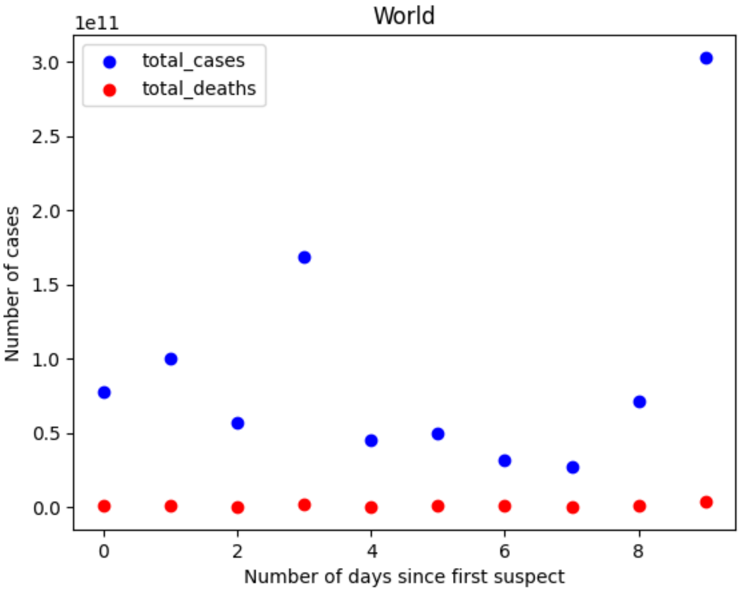

North America
1e9
1.0
0.8





```
df5=df4.groupby(['Country'])[['Country', 'total_cases', 'total_deaths']].sum().reset_index()
C = df5
plt.scatter (np.arange(0,len (C)),C['total_cases'], color="blue", label="total_cases")
plt.scatter (np.arange(0,len (C)),C['total_deaths' ], color="red", label="total_deaths")
plt.title("World")
plt.xlabel("Number of days since first suspect")
plt.ylabel("Number of cases")
plt.legend()
plt.show()
```

```
<ipython-input-22-3f5183088e2c>:1: FutureWarning: The default value of numeric_only in DataFrameGroupB
  df5=df4.groupby(['Country'])[['Country', 'total_cases', 'total_deaths']].sum().reset_index()
```



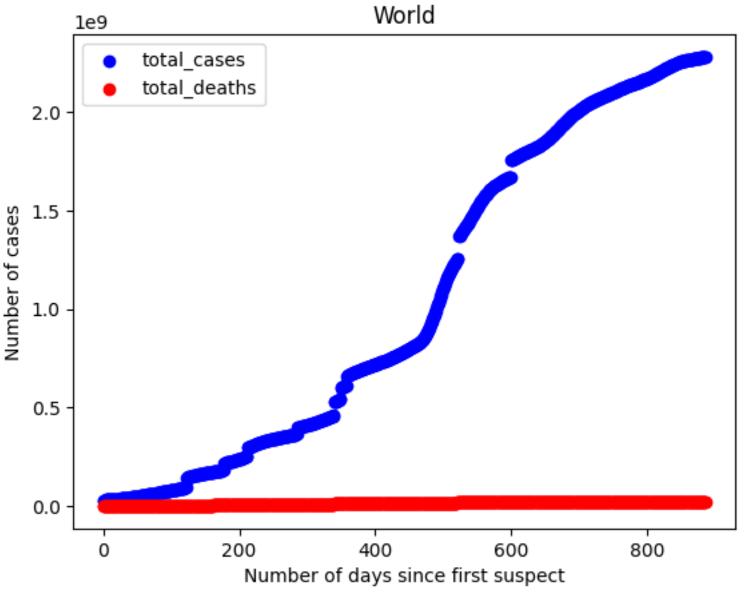
```
date= df4[ 'Date'].unique()
len (date)
```

```
887
```

```
df6=df4.groupby([ 'Date']) [['Date', 'total_cases', 'total_deaths']].sum().reset_index()
```

```
<ipython-input-24-00f930a3b5f8>:1: FutureWarning: The default value of numeric_only in DataFrameGroupBy.sum is deprecated. In a future version, nu
  df6=df4.groupby([ 'Date']) [['Date', 'total_cases', 'total_deaths']].sum().reset_index()
```

```
C=df6
plt.scatter(np.arange(0,len (C)),C['total_cases'],color="blue",label="total_cases")
plt.scatter(np.arange(0,len (C)),C['total_deaths'],color="red",label="total_deaths")
plt.title("World")
plt.xlabel("Number of days since first suspect")
plt.ylabel("Number of cases")
plt.legend()
plt.show()
```



✓ 0s completed at 1:34 PM