

In [1]:

```
!pip install bs4
```

Defaulting to user installation because normal site-packages is not writeable

Requirement already satisfied: bs4 in c:\users\darshan\appdata\roaming\python\python39\site-packages (0.0.1)

Requirement already satisfied: beautifulsoup4 in c:\programdata\anaconda3\lib\site-packages (from bs4) (4.11.1)

Requirement already satisfied: soupsieve>1.2 in c:\programdata\anaconda3\lib\site-packages (from beautifulsoup4->bs4) (2.3.1)

In [2]:

```
from bs4 import BeautifulSoup
```

In [3]:

```
import requests
```

In [4]:

```
!pip install requests
```

Defaulting to user installation because normal site-packages is not writeable

Requirement already satisfied: requests in c:\programdata\anaconda3\lib\site-packages (2.28.1)

Requirement already satisfied: certifi>=2017.4.17 in c:\programdata\anaconda3\lib\site-packages (from requests) (2022.9.14)

Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\programdata\anaconda3\lib\site-packages (from requests) (1.26.11)

Requirement already satisfied: charset-normalizer<3,>=2 in c:\programdata\anaconda3\lib\site-packages (from requests) (2.0.4)

Requirement already satisfied: idna<4,>=2.5 in c:\programdata\anaconda3\lib\site-packages (from requests) (3.3)

In [5]:

```
page=requests.get('https://www.worldometers.info/coronavirus/')
```

In [6]:

```
page
```

Out[6]:

```
<Response [200]>
```


In [13]:



```
name=a[177::22]  
name
```

...

In [14]:



```
total_case=a[178::22]  
total_case
```

...

In [15]:



```
new_case=a[179::22]  
new_case
```

...

In [16]:



```
total_death=a[180::22]  
total_death
```

...

In [17]:



```
new_death=a[181::22]  
new_death
```

...

In [18]:



```
total_recover=a[182::22]  
total_recover
```

...

In [19]:



```
new_recover=a[183::22]  
new_recover
```

...

In [20]:



```
active_case=a[184::22]  
active_case
```

...

In [21]:



```
serious=a[185::22]  
serious
```

...

In [22]:

```
top_case=a[186::22]  
top_case
```

...

In [23]:

```
death_perm=a[187::22]  
death_perm
```

...

In [24]:

```
total_test=a[188::22]  
total_test
```

...

In [25]:

```
population=a[190::22]  
population
```

...

In [26]:

```
num=no[0:231]  
num
```

...

In [27]:

```
names=name[0:231]  
names
```

Out[27]:

```
['USA',  
 'India',  
 'France',  
 'Germany',  
 'Brazil',  
 'Japan',  
 'S. Korea',  
 'Italy',  
 'UK',  
 'Russia',  
 'Turkey',  
 'Spain',  
 'Vietnam',  
 'Australia',  
 'Taiwan',  
 'Argentina',  
 'Netherlands',
```



```
total_cases = total_case[0:231]
total_cases
```



1



```
new_cases=new_case[0:231]
new_cases
```


1

In [30]:

```
deaths=total_death[0:231]
deaths
```

Out[30]:

```
['1,157,462',
 '531,016',
 '165,916',
 '171,748',
 '700,811',
 '74,164',
 '34,361',
 '189,262',
 '209,396',
 '397,681',
 '102,174',
 '120,426',
 '43,186',
 '19,933',
 '19,005',
 '130,472',
 '22,992']
```

In [31]:

```
total_deaths=[]
for i in deaths:
    b=i.strip()
    total_deaths.append(b)
total_deaths
```

Out[31]:

```
['1,157,462',
 '531,016',
 '165,916',
 '171,748',
 '700,811',
 '74,164',
 '34,361',
 '189,262',
 '209,396',
 '397,681',
 '102,174',
 '120,426',
 '43,186',
 '19,933',
 '19,005',
 '130,472',
 '22,992',
```


1



1

In [37]:

```
top_cases_per_M=top_case[0:231]
```

In [38]:

```
top_death_per_M=death_perm[0:231]
```

In [39]:

```
total_tests=total_test[0:231]
```

In [40]:

```
population_=population[0:231]
```

In [41]:

```
import pandas as pd
```

In [42]:

```
Covid_report=pd.DataFrame({})
Covid_report['sr.no']=num
Covid_report['Country Name']=names
Covid_report['Total Cases']=total_cases
Covid_report['New Cases']=new_cases
Covid_report['Total Deaths']=total_deaths
Covid_report['New Deaths']=new_deaths
Covid_report['Total Recovered']=total_recovered
Covid_report['New Recovered']=new_recovered
Covid_report['Serious Cond']=serious
Covid_report['Top cases per Million']=top_cases_per_M
Covid_report['Top death per million']=top_death_per_M
Covid_report['Total Tests']=total_tests
Covid_report['Total Population']=population_
```

In [43]:

Covid_report

Out[43]:

	sr.no	Country Name	Total Cases	New Cases	Total Deaths	New Deaths	Total Recovered	New Recovered	Serious Cont
0	1	USA	106,418,595		1,157,462		104,260,123		1,726
1	2	India	44,776,002		531,016		44,204,771		N/A
2	3	France	39,867,463		165,916		39,569,363		866
3	4	Germany	38,377,656		171,748		38,143,900	+5,100	N/A
4	5	Brazil	37,358,092		700,811		36,249,161		N/A
...
226	227	Vatican City	29				29		
227	228	Western Sahara	10		1		9		
228	229	MS Zaandam	9		2		7		
229	230	Tokelau	5						
230	231	China	503,302		5,272		379,053		N/A

231 rows × 13 columns

In [44]:

Covid_report.to_csv('Covidreport')

In [45]:

import pymysql as mycon

In [47]:

con1=mycon.connect(host='localhost',user='root',database='covid')

In [48]:

```
#create cursor
mycursor=con1.cursor()
```

In [50]:

```
mycursor.execute("create table covidreports(sr_no int,Country_name varchar(50),Total_cas
```

Out[50]:

0

In [51]:

```
for i in range(len(num)):
    mycursor.execute("insert into covidreports values('%s','%s','%s','%s','%s','%s','%s'
    con1.commit()
```

In [52]:

```
!pip install mysql-connector-python
```

In [53]:

```
import mysql.connector as mycon
```

In [56]:

```
conn=mycon.connect(host='localhost',user='root',database='covid',password='')
```

In [58]:

```
if conn.is_connected():
    mycursor=conn.cursor()
    mycursor.execute("create table if not exists covidreport(sr_no int,Country_name varc
    print("Table created")
    for i,row in Covid_report.iterrows():
        sql="insert into covidreport values(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)"
        mycursor.execute(sql,tuple(row))
    conn.commit()
```

Table created

In []: