

```
In [1]: import pandas as pd  
import matplotlib.pyplot as plt  
import numpy as np
```

```
In [2]: try:  
    covid_deaths_raw = pd.read_excel('covid_19_dataset.xlsx',sheet_name ='covid_19_deaths_v1')  
    covid_confirmed_raw = pd.read_excel('covid_19_dataset.xlsx',sheet_name = 'covid_19_confirmed_v1')  
    covid_recovered_raw = pd.read_excel('covid_19_dataset.xlsx',sheet_name ='covid_19_recovered_v1')  
except FileNotFoundError as e:  
    print("Error :",e)
```

```
In [3]: covid_deaths_raw.head(3)
```

```
Out[3]:
```

| | Province/State | Country/Region | Lat | Long | 1/22/20 | 1/23/20 | 1/24/20 | 1/25/20 | 1/26/20 | 1/27/20 | ... | 5/20/21 | 5/21/21 | 5/22/21 | 5/23/21 |
|---|----------------|----------------|----------|-----------|---------|---------|---------|---------|---------|---------|-----|---------|---------|---------|---------|
| 0 | NaN | Afghanistan | 33.93911 | 67.709953 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2772 | 2782 | 2792 | 280 |
| 1 | NaN | Albania | 41.15330 | 20.168300 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2440 | 2441 | 2442 | 244 |
| 2 | NaN | Algeria | 28.03390 | 1.659600 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3401 | 3405 | 3411 | 341 |

3 rows × 498 columns

```
In [4]: covid_confirmed_raw.head(3)
```

```
Out[4]:
```

| | Province/State | Country/Region | Lat | Long | 1/22/20 | 1/23/20 | 1/24/20 | 1/25/20 | 1/26/20 | 1/27/20 | ... | 5/20/21 | 5/21/21 | 5/22/21 | 5/23/21 |
|---|----------------|----------------|----------|-----------|---------|---------|---------|---------|---------|---------|-----|---------|---------|---------|---------|
| 0 | NaN | Afghanistan | 33.93911 | 67.709953 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 64575 | 65080 | 65486 | 6572 |
| 1 | NaN | Albania | 41.15330 | 20.168300 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 132118 | 132153 | 132176 | 13220 |
| 2 | NaN | Algeria | 28.03390 | 1.659600 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 126156 | 126434 | 126651 | 12686 |

3 rows × 498 columns

```
In [5]: covid_recovered_raw.head(3)
```

Out[5]:

| | Province/State | Country/Region | Lat | Long | 1/22/20 | 1/23/20 | 1/24/20 | 1/25/20 | 1/26/20 | 1/27/20 | ... | 5/20/21 | 5/21/21 | 5/22/21 | 5/23/22 |
|---|----------------|----------------|----------|-----------|---------|---------|---------|---------|---------|---------|-----|---------|---------|---------|---------|
| 0 | NaN | Afghanistan | 33.93911 | 67.709953 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 55687 | 55790 | 55889 | 5603 |
| 1 | NaN | Albania | 41.15330 | 20.168300 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 127869 | 128425 | 128601 | 12873 |
| 2 | NaN | Algeria | 28.03390 | 1.659600 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 87902 | 88066 | 88208 | 8834 |

3 rows × 498 columns

In [6]:

```
# Unpivot death table
idvars=covid_deaths_raw.loc[:,['Province/State','Country/Region','Lat','Long']].columns.tolist()
valvars=covid_deaths_raw.iloc[:,4:].columns.tolist()

covid_deaths=pd.melt(covid_deaths_raw,id_vars=idvars,value_vars=valvars,var_name='Date',value_name='Deaths')
covid_deaths
```

Out[6]:

| | Province/State | Country/Region | Lat | Long | Date | Deaths |
|--------|----------------|--------------------|------------|------------|---------|--------|
| 0 | NaN | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0.0 |
| 1 | NaN | Albania | 41.153300 | 20.168300 | 1/22/20 | 0.0 |
| 2 | NaN | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0.0 |
| 3 | NaN | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0.0 |
| 4 | NaN | Angola | -11.202700 | 17.873900 | 1/22/20 | 0.0 |
| ... | ... | ... | ... | ... | ... | ... |
| 136339 | NaN | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 47.0 |
| 136340 | NaN | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 3492.0 |
| 136341 | NaN | Yemen | 15.552727 | 48.516388 | 5/29/21 | 1319.0 |
| 136342 | NaN | Zambia | -13.133897 | 27.849332 | 5/29/21 | 1276.0 |
| 136343 | NaN | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 1594.0 |

136344 rows × 6 columns

In [7]:

```
# Unpivot confirmed table
idvars=covid_confirmed_raw.loc[:,['Province/State','Country/Region','Lat','Long']].columns.tolist()
```

```
valvars=covid_confirmed_raw.iloc[:,4: ].select_dtypes(include=['int']).columns.tolist()

covid_confirmed = pd.melt(covid_confirmed_raw,idvars,valvars,var_name='Date',value_name='Confirmed')
covid_confirmed
```

Out[7]:

| | Province/State | Country/Region | Lat | Long | Date | Confirmed |
|--------|----------------|--------------------|------------|------------|---------|-----------|
| 0 | NaN | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0 |
| 1 | NaN | Albania | 41.153300 | 20.168300 | 1/22/20 | 0 |
| 2 | NaN | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0 |
| 3 | NaN | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0 |
| 4 | NaN | Angola | -11.202700 | 17.873900 | 1/22/20 | 0 |
| ... | ... | ... | ... | ... | ... | ... |
| 136339 | NaN | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 6908 |
| 136340 | NaN | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 307838 |
| 136341 | NaN | Yemen | 15.552727 | 48.516388 | 5/29/21 | 6731 |
| 136342 | NaN | Zambia | -13.133897 | 27.849332 | 5/29/21 | 94751 |
| 136343 | NaN | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 38933 |

136344 rows × 6 columns

In [8]:

```
# Unpivot recovered table
rcols=covid_recovered_raw.columns.tolist()
idvars = ['Province/State','Country/Region','Lat','Long']
valvars = [i for i in rcols if i not in idvars]

covid_recovered = pd.melt(covid_recovered_raw,idvars,valvars,'Date','Recovered')
covid_recovered
```

Out[8]:

| | Province/State | Country/Region | Lat | Long | Date | Recovered |
|---------------|----------------|--------------------|------------|------------|---------|-----------|
| 0 | NaN | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0.0 |
| 1 | NaN | Albania | 41.153300 | 20.168300 | 1/22/20 | 0.0 |
| 2 | NaN | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0.0 |
| 3 | NaN | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0.0 |
| 4 | NaN | Angola | -11.202700 | 17.873900 | 1/22/20 | 0.0 |
| ... | ... | ... | ... | ... | ... | ... |
| 128929 | NaN | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 2896.0 |
| 128930 | NaN | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 300524.0 |
| 128931 | NaN | Yemen | 15.552727 | 48.516388 | 5/29/21 | 3399.0 |
| 128932 | NaN | Zambia | -13.133897 | 27.849332 | 5/29/21 | 91594.0 |
| 128933 | NaN | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 36578.0 |

128934 rows × 6 columns

In [9]:

```
# Structure of dataset in terms of rows, columns and dtypes
covid_deaths.info()
covid_deaths.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 136344 entries, 0 to 136343
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   Province/State  42484 non-null   object 
 1   Country/Region  136344 non-null   object 
 2   Lat              135356 non-null   float64
 3   Long             135356 non-null   float64
 4   Date             136344 non-null   object 
 5   Deaths           136343 non-null   float64
dtypes: float64(3), object(3)
memory usage: 6.2+ MB
```

```
Out[9]: Province/State    93860  
Country/Region         0  
Lat                  988  
Long                 988  
Date                  0  
Deaths                 1  
dtype: int64
```

```
In [10]: covid_confirmed.info()  
covid_confirmed.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 136344 entries, 0 to 136343  
Data columns (total 6 columns):  
 #   Column           Non-Null Count  Dtype     
---  --    
 0   Province/State  42484 non-null   object    
 1   Country/Region  136344 non-null   object    
 2   Lat              135356 non-null   float64   
 3   Long             135356 non-null   float64   
 4   Date             136344 non-null   object    
 5   Confirmed        136344 non-null   int64    
 dtypes: float64(2), int64(1), object(3)  
 memory usage: 6.2+ MB
```

```
Out[10]: Province/State    93860  
Country/Region         0  
Lat                  988  
Long                 988  
Date                  0  
Confirmed             0  
dtype: int64
```

```
In [11]: covid_recovered.info()  
covid_recovered.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128934 entries, 0 to 128933
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Province/State  34580 non-null    object  
 1   Country/Region  128934 non-null    object  
 2   Lat              128440 non-null    float64 
 3   Long             128440 non-null    float64 
 4   Date             128934 non-null    object  
 5   Recovered        128933 non-null    float64 
dtypes: float64(3), object(3)
memory usage: 5.9+ MB
```

```
Out[11]: Province/State    94354
Country/Region      0
Lat                 494
Long                494
Date                0
Recovered           1
dtype: int64
```

Data Cleaning & Handling Missing values

```
In [12]: # Replace blank values in province column with "All Provinces" and clean the dataset (covid_deaths)
covid_deaths['Province/State']=covid_deaths['Province/State'].fillna('All Provinces')
covid_deaths.head()
```

```
Out[12]:
```

| | Province/State | Country/Region | Lat | Long | Date | Deaths |
|---|----------------|----------------|-----------|-----------|---------|--------|
| 0 | All Provinces | Afghanistan | 33.93911 | 67.709953 | 1/22/20 | 0.0 |
| 1 | All Provinces | Albania | 41.15330 | 20.168300 | 1/22/20 | 0.0 |
| 2 | All Provinces | Algeria | 28.03390 | 1.659600 | 1/22/20 | 0.0 |
| 3 | All Provinces | Andorra | 42.50630 | 1.521800 | 1/22/20 | 0.0 |
| 4 | All Provinces | Angola | -11.20270 | 17.873900 | 1/22/20 | 0.0 |

```
In [13]: covid_deaths[['Lat','Long']] = covid_deaths.groupby('Country/Region')[['Lat','Long']].transform(lambda x: x.fillna(method='ffill')).fillna(method='ffill')
```

```
C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\700309625.py:1: FutureWarning: Series.fillna with 'method' is deprecated and will raise in  
a future version. Use obj.ffill() or obj.bfill() instead.  
covid_deaths[['Lat','Long']] = covid_deaths.groupby('Country/Region')[['Lat','Long']].transform(lambda x: x.fillna(method='ffill').fillna(me  
thod='bfill'))  
C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\700309625.py:1: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise  
in a future version. Use obj.ffill() or obj.bfill() instead.  
covid_deaths[['Lat','Long']] = covid_deaths.groupby('Country/Region')[['Lat','Long']].transform(lambda x: x.fillna(method='ffill').fillna(me  
thod='bfill'))
```

Out[13]:

| | Province/State | Country/Region | Lat | Long | Date | Deaths |
|--------|----------------|--------------------|------------|------------|---------|--------|
| 0 | All Provinces | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0.0 |
| 1 | All Provinces | Albania | 41.153300 | 20.168300 | 1/22/20 | 0.0 |
| 2 | All Provinces | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0.0 |
| 3 | All Provinces | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0.0 |
| 4 | All Provinces | Angola | -11.202700 | 17.873900 | 1/22/20 | 0.0 |
| ... | ... | ... | ... | ... | ... | ... |
| 136339 | All Provinces | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 47.0 |
| 136340 | All Provinces | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 3492.0 |
| 136341 | All Provinces | Yemen | 15.552727 | 48.516388 | 5/29/21 | 1319.0 |
| 136342 | All Provinces | Zambia | -13.133897 | 27.849332 | 5/29/21 | 1276.0 |
| 136343 | All Provinces | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 1594.0 |

136344 rows × 6 columns

In [14]:

```
covid_deaths.isnull().sum()
```

Out[14]:

```
Province/State    0  
Country/Region   0  
Lat              0  
Long             0  
Date             0  
Deaths           1  
dtype: int64
```

In [15]:

```
# Filling deaths missing value  
print(covid_deaths[covid_deaths['Deaths'].isnull()])  
ar=covid_deaths[(covid_deaths['Country/Region']=='Algeria')]
```

```
m=ar['Deaths'].mean().round(2)
md=ar['Deaths'].median()
s=ar['Deaths'].std()
print(f"Mean: {m}")
print(f"Median: {md}")
print(f"Std: {s}")
print(f"Diff: {(s/m)*100} %")
```

| | Province/State | Country/Region | Lat | Long | Date | Deaths |
|-------|----------------|----------------|---------|--------|---------|--------|
| 24566 | All Provinces | Algeria | 28.0339 | 1.6596 | 4/20/20 | NaN |

Mean: 1691.85
Median: 1707.0
Std: 1157.611130316425
Diff: 68.42279932124154 %

```
In [16]: covid_deaths.loc[24566, 'Deaths']=covid_deaths.loc[24566, 'Deaths']=md #use Median
covid_deaths.loc[24566,:]
```

```
Out[16]: Province/State      All Provinces
Country/Region          Algeria
Lat                      28.0339
Long                     1.6596
Date                      4/20/20
Deaths                   1707.0
Name: 24566, dtype: object
```

```
In [17]: covid_deaths.isnull().sum()
```

```
Out[17]: Province/State      0
Country/Region          0
Lat                      0
Long                     0
Date                      0
Deaths                   0
dtype: int64
```

```
In [18]: # Replace blank values in province column with "All Provinces" and clean the dataset (covid_confirmed)
covid_confirmed['Province/State']=covid_confirmed['Province/State'].fillna('All Provinces')
covid_confirmed
```

Out[18]:

| | Province/State | Country/Region | Lat | Long | Date | Confirmed |
|--------|----------------|--------------------|------------|------------|---------|-----------|
| 0 | All Provinces | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0 |
| 1 | All Provinces | Albania | 41.153300 | 20.168300 | 1/22/20 | 0 |
| 2 | All Provinces | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0 |
| 3 | All Provinces | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0 |
| 4 | All Provinces | Angola | -11.202700 | 17.873900 | 1/22/20 | 0 |
| ... | ... | ... | ... | ... | ... | ... |
| 136339 | All Provinces | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 6908 |
| 136340 | All Provinces | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 307838 |
| 136341 | All Provinces | Yemen | 15.552727 | 48.516388 | 5/29/21 | 6731 |
| 136342 | All Provinces | Zambia | -13.133897 | 27.849332 | 5/29/21 | 94751 |
| 136343 | All Provinces | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 38933 |

136344 rows × 6 columns

```
In [19]: covid_confirmed[['Lat','Long']] = covid_confirmed.groupby('Country/Region')[['Lat','Long']].transform(lambda x:x.fillna(method='ffill')).fillna(method='bfill')
```

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\3678558340.py:1: FutureWarning: Series.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.

```
covid_confirmed[['Lat','Long']] = covid_confirmed.groupby('Country/Region')[['Lat','Long']].transform(lambda x:x.fillna(method='ffill').fillna(method='bfill'))
```

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\3678558340.py:1: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.

```
covid_confirmed[['Lat','Long']] = covid_confirmed.groupby('Country/Region')[['Lat','Long']].transform(lambda x:x.fillna(method='ffill').fillna(method='bfill'))
```

Out[19]:

| | Province/State | Country/Region | Lat | Long | Date | Confirmed |
|---------------|----------------|--------------------|------------|------------|---------|-----------|
| 0 | All Provinces | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0 |
| 1 | All Provinces | Albania | 41.153300 | 20.168300 | 1/22/20 | 0 |
| 2 | All Provinces | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0 |
| 3 | All Provinces | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0 |
| 4 | All Provinces | Angola | -11.202700 | 17.873900 | 1/22/20 | 0 |
| ... | ... | ... | ... | ... | ... | ... |
| 136339 | All Provinces | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 6908 |
| 136340 | All Provinces | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 307838 |
| 136341 | All Provinces | Yemen | 15.552727 | 48.516388 | 5/29/21 | 6731 |
| 136342 | All Provinces | Zambia | -13.133897 | 27.849332 | 5/29/21 | 94751 |
| 136343 | All Provinces | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 38933 |

136344 rows × 6 columns

In [20]:

`covid_confirmed.isnull().sum()`

Out[20]:

| | |
|----------------|---|
| Province/State | 0 |
| Country/Region | 0 |
| Lat | 0 |
| Long | 0 |
| Date | 0 |
| Confirmed | 0 |
| dtype: int64 | |

In [21]:

```
# Replace blank values in province column with "All Provinces" and clean the dataset (covid_recovered)
covid_recovered['Province/State']=covid_recovered['Province/State'].fillna('All Provinces')
covid_recovered
```

Out[21]:

| | Province/State | Country/Region | Lat | Long | Date | Recovered |
|---------------|----------------|--------------------|------------|------------|---------|-----------|
| 0 | All Provinces | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0.0 |
| 1 | All Provinces | Albania | 41.153300 | 20.168300 | 1/22/20 | 0.0 |
| 2 | All Provinces | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0.0 |
| 3 | All Provinces | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0.0 |
| 4 | All Provinces | Angola | -11.202700 | 17.873900 | 1/22/20 | 0.0 |
| ... | ... | ... | ... | ... | ... | ... |
| 128929 | All Provinces | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 2896.0 |
| 128930 | All Provinces | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 300524.0 |
| 128931 | All Provinces | Yemen | 15.552727 | 48.516388 | 5/29/21 | 3399.0 |
| 128932 | All Provinces | Zambia | -13.133897 | 27.849332 | 5/29/21 | 91594.0 |
| 128933 | All Provinces | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 36578.0 |

128934 rows × 6 columns

```
In [22]: covid_recovered[['Lat','Long']] = covid_recovered.groupby('Country/Region')[['Lat','Long']].transform(lambda x:x.fillna(method='ffill')).fillna(method='bfill')
```

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\1861019684.py:1: FutureWarning: Series.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.

```
covid_recovered[['Lat','Long']] = covid_recovered.groupby('Country/Region')[['Lat','Long']].transform(lambda x:x.fillna(method='ffill').fillna(method='bfill'))
```

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\1861019684.py:1: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.

```
covid_recovered[['Lat','Long']] = covid_recovered.groupby('Country/Region')[['Lat','Long']].transform(lambda x:x.fillna(method='ffill').fillna(method='bfill'))
```

Out[22]:

| | Province/State | Country/Region | Lat | Long | Date | Recovered |
|---------------|----------------|--------------------|------------|------------|---------|-----------|
| 0 | All Provinces | Afghanistan | 33.939110 | 67.709953 | 1/22/20 | 0.0 |
| 1 | All Provinces | Albania | 41.153300 | 20.168300 | 1/22/20 | 0.0 |
| 2 | All Provinces | Algeria | 28.033900 | 1.659600 | 1/22/20 | 0.0 |
| 3 | All Provinces | Andorra | 42.506300 | 1.521800 | 1/22/20 | 0.0 |
| 4 | All Provinces | Angola | -11.202700 | 17.873900 | 1/22/20 | 0.0 |
| ... | ... | ... | ... | ... | ... | ... |
| 128929 | All Provinces | Vietnam | 14.058324 | 108.277199 | 5/29/21 | 2896.0 |
| 128930 | All Provinces | West Bank and Gaza | 31.952200 | 35.233200 | 5/29/21 | 300524.0 |
| 128931 | All Provinces | Yemen | 15.552727 | 48.516388 | 5/29/21 | 3399.0 |
| 128932 | All Provinces | Zambia | -13.133897 | 27.849332 | 5/29/21 | 91594.0 |
| 128933 | All Provinces | Zimbabwe | -19.015438 | 29.154857 | 5/29/21 | 36578.0 |

128934 rows × 6 columns

In [23]: covid_recovered.isnull().sum()

```
Out[23]: Province/State      0
Country/Region     0
Lat                  0
Long                  0
Date                  0
Recovered             1
dtype: int64
```

In [24]: print(covid_recovered[covid_recovered['Recovered'].isnull()])

```
ca=covid_recovered[covid_recovered['Country/Region']=='Argentina']
m=ca['Recovered'].mean()
md=ca['Recovered'].median()
s=ca['Recovered'].std()
print(f"Mean: {m}")
print(f"Median: {md}")
print(f"Std: {s}")
print(f"Diff: {(s/m)*100} %")
```

```
Province/State Country/Region      Lat      Long      Date Recovered
23235  All Provinces      Argentina -38.4161 -63.6167  4/20/20        NaN
Mean: 889958.4381338742
Median: 546924.0
Std: 948619.1314853535
Diff: 106.5913969504557 %
```

```
In [25]: covid_recovered.loc[23235, 'Recovered']=covid_recovered.loc[23235, 'Recovered']=md #use median
covid_recovered.loc[23235,:]
```

```
Out[25]: Province/State      All Provinces
Country/Region      Argentina
Lat                  -38.4161
Long                 -63.6167
Date                4/20/20
Recovered           546924.0
Name: 23235, dtype: object
```

```
In [26]: covid_recovered.isnull().sum()
```

```
Out[26]: Province/State      0
Country/Region      0
Lat                  0
Long                 0
Date                0
Recovered           0
dtype: int64
```

```
# Dtypes of all tables
covid_deaths['Date']=pd.to_datetime(covid_deaths['Date'])
covid_confirmed['Date']=pd.to_datetime(covid_confirmed['Date'])
covid_recovered['Date']=pd.to_datetime(covid_recovered['Date'])
print(covid_deaths.info())
print('*'*100)
print(covid_confirmed.info())
print('*'*100)
print(covid_recovered.info())
```

```
C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\4294271968.py:2: UserWarning: Could not infer format, so each element will be parsed individually, falling back to `dateutil`. To ensure parsing is consistent and as-expected, please specify a format.  
covid_deaths['Date']=pd.to_datetime(covid_deaths['Date'])  
C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\4294271968.py:3: UserWarning: Could not infer format, so each element will be parsed individually, falling back to `dateutil`. To ensure parsing is consistent and as-expected, please specify a format.  
covid_confirmed['Date']=pd.to_datetime(covid_confirmed['Date'])  
C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\4294271968.py:4: UserWarning: Could not infer format, so each element will be parsed individually, falling back to `dateutil`. To ensure parsing is consistent and as-expected, please specify a format.  
covid_recovered['Date']=pd.to_datetime(covid_recovered['Date'])
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 136344 entries, 0 to 136343
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Province/State  136344 non-null   object  
 1   Country/Region  136344 non-null   object  
 2   Lat              136344 non-null   float64 
 3   Long             136344 non-null   float64 
 4   Date             136344 non-null   datetime64[ns]
 5   Deaths           136344 non-null   float64 
dtypes: datetime64[ns](1), float64(3), object(2)
memory usage: 6.2+ MB
None
*****
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 136344 entries, 0 to 136343
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Province/State  136344 non-null   object  
 1   Country/Region  136344 non-null   object  
 2   Lat              136344 non-null   float64 
 3   Long             136344 non-null   float64 
 4   Date             136344 non-null   datetime64[ns]
 5   Confirmed        136344 non-null   int64  
dtypes: datetime64[ns](1), float64(2), int64(1), object(2)
memory usage: 6.2+ MB
None
*****
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 128934 entries, 0 to 128933
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Province/State  128934 non-null   object  
 1   Country/Region  128934 non-null   object  
 2   Lat              128934 non-null   float64 
 3   Long             128934 non-null   float64 
 4   Date             128934 non-null   datetime64[ns]
 5   Recovered        128934 non-null   float64 
dtypes: datetime64[ns](1), float64(3), object(2)
memory usage: 5.9+ MB
None
```

In [28]: covid_deaths.head(2)

Out[28]:

| | Province/State | Country/Region | Lat | Long | Date | Deaths |
|---|----------------|----------------|----------|-----------|------------|--------|
| 0 | All Provinces | Afghanistan | 33.93911 | 67.709953 | 2020-01-22 | 0.0 |
| 1 | All Provinces | Albania | 41.15330 | 20.168300 | 2020-01-22 | 0.0 |

In [29]: covid_confirmed.head(2)

Out[29]:

| | Province/State | Country/Region | Lat | Long | Date | Confirmed |
|---|----------------|----------------|----------|-----------|------------|-----------|
| 0 | All Provinces | Afghanistan | 33.93911 | 67.709953 | 2020-01-22 | 0 |
| 1 | All Provinces | Albania | 41.15330 | 20.168300 | 2020-01-22 | 0 |

In [30]: covid_recovered.head(2)

Out[30]:

| | Province/State | Country/Region | Lat | Long | Date | Recovered |
|---|----------------|----------------|----------|-----------|------------|-----------|
| 0 | All Provinces | Afghanistan | 33.93911 | 67.709953 | 2020-01-22 | 0.0 |
| 1 | All Provinces | Albania | 41.15330 | 20.168300 | 2020-01-22 | 0.0 |

In [31]: # Merging the Cleaned Datasets for Analysis

```
# Converting all the tables at level of data ('Country/Region & Date')
death_agg = covid_deaths.groupby(['Country/Region', 'Date'])['Deaths'].sum().reset_index()
confirmed_agg = covid_confirmed.groupby(['Country/Region', 'Date'])['Confirmed'].sum().reset_index()
recovered_agg = covid_recovered.groupby(['Country/Region', 'Date'])['Recovered'].sum().reset_index()

#Merging tables
covid=(death_agg
      .merge(confirmed_agg,on=['Country/Region', 'Date'],how='inner')
      .merge(recovered_agg,on=['Country/Region', 'Date'],how='inner')
     ).sort_values(by=['Country/Region', 'Date'])
covid
```

Out[31]:

| | Country/Region | Date | Deaths | Confirmed | Recovered |
|-------|----------------|------------|--------|-----------|-----------|
| 0 | Afghanistan | 2020-01-02 | 0.0 | 0 | 0.0 |
| 1 | Afghanistan | 2020-01-03 | 0.0 | 1 | 0.0 |
| 2 | Afghanistan | 2020-01-04 | 4.0 | 197 | 5.0 |
| 3 | Afghanistan | 2020-01-05 | 68.0 | 2291 | 310.0 |
| 4 | Afghanistan | 2020-01-06 | 266.0 | 15753 | 1428.0 |
| ... | ... | ... | ... | ... | ... |
| 95337 | Zimbabwe | 2021-12-01 | 551.0 | 23239 | 13396.0 |
| 95338 | Zimbabwe | 2021-12-02 | 1393.0 | 35045 | 29959.0 |
| 95339 | Zimbabwe | 2021-12-03 | 1496.0 | 36423 | 33996.0 |
| 95340 | Zimbabwe | 2021-12-04 | 1542.0 | 37307 | 34901.0 |
| 95341 | Zimbabwe | 2021-12-05 | 1579.0 | 38466 | 36277.0 |

95342 rows × 5 columns

Exploratory Data Analysis

Q1: Generate plots of confirmed cases over time for top countries.

```
In [32]: #covid['Year-month']=covid['Date'].map(lambda x:x.strftime('%Y-%b'))
contries=confirmed_agg.loc[confirmed_agg.groupby(['Country/Region'])['Date'].idxmax()].nlargest(3,columns='Confirmed')
contries
```

Out[32]:

| | Country/Region | Date | Confirmed |
|-------|----------------|------------|-----------|
| 88919 | US | 2021-12-05 | 32814784 |
| 39519 | India | 2021-12-05 | 23703665 |
| 11855 | Brazil | 2021-12-05 | 15359397 |

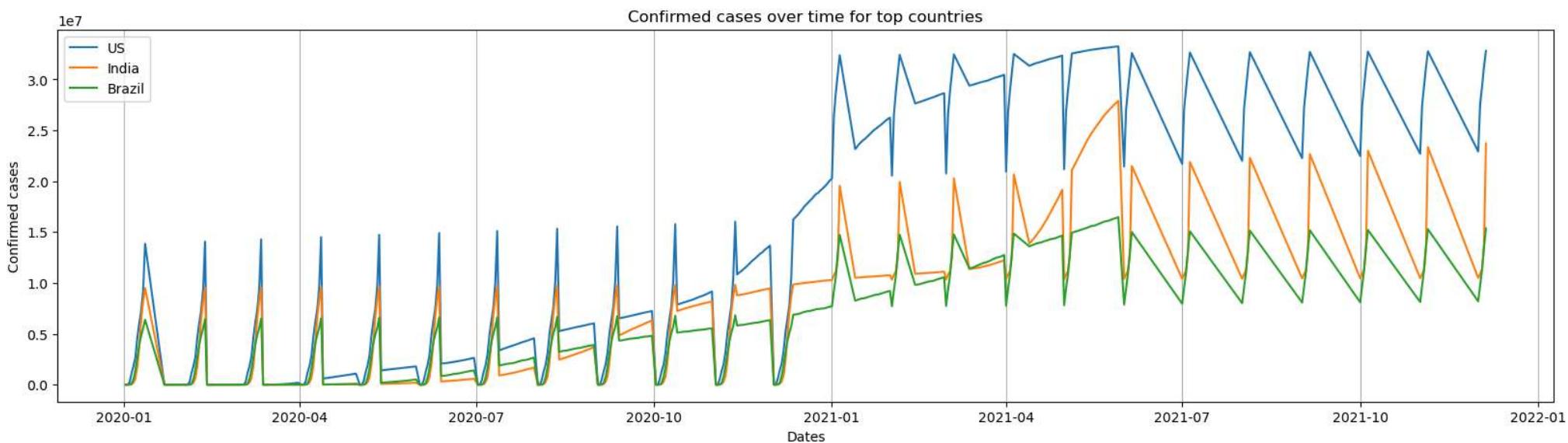
In [33]:

```
topctry=contries['Country/Region'].tolist()
topctry
```

```
Out[33]: ['US', 'India', 'Brazil']
```

```
In [34]: plt.figure(figsize=(20,5))
```

```
for i in topctry:  
    tc=covid[covid['Country/Region']==i].sort_values('Date')  
    plt.plot(tc['Date'],tc['Confirmed'],label=i)  
plt.title('Confirmed cases over time for top countries')  
plt.xlabel('Dates')  
plt.ylabel('Confirmed cases')  
plt.legend()  
plt.grid(axis='x')  
plt.show()
```



Q2: Generate Confirmed cases overtime (monthly) for China

```
In [35]:
```

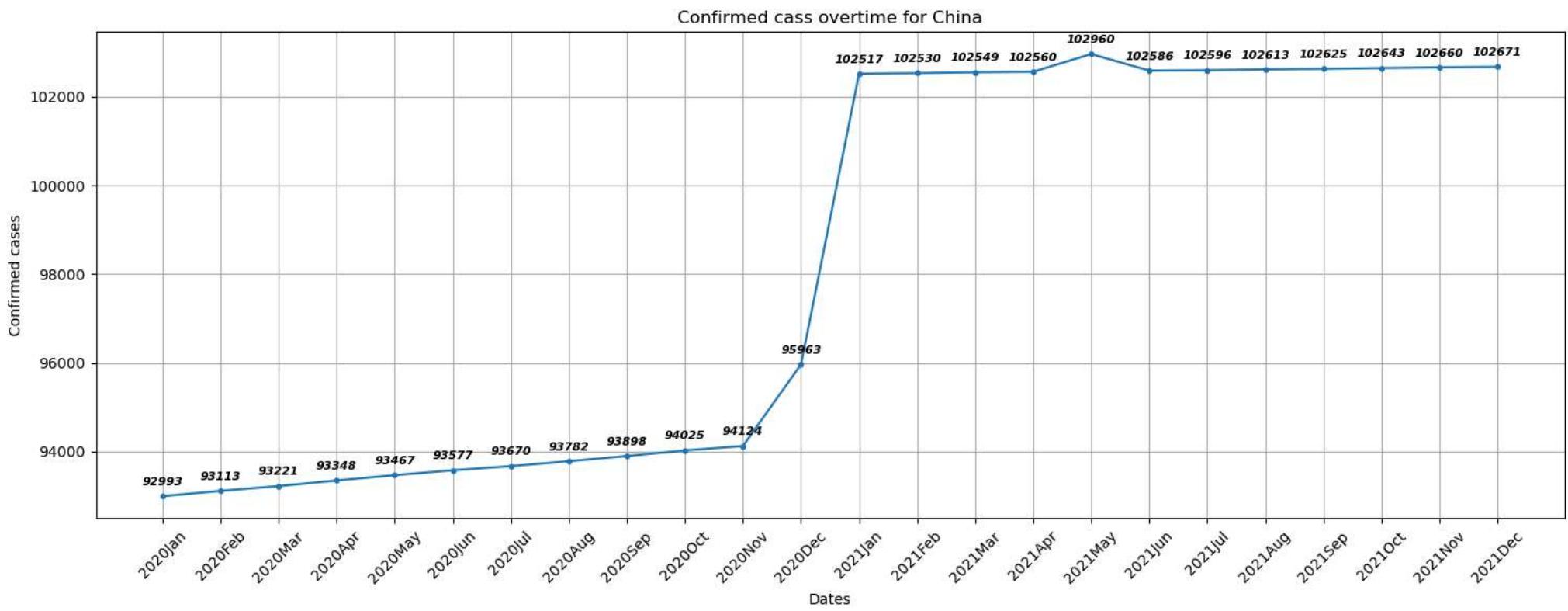
```
covid['Year-month']=covid['Date'].map(lambda x: x.strftime('%Y%b'))  
covid['Year']=covid['Date'].map(lambda x: x.strftime('%Y')).astype(int)  
covid['Month']=covid['Date'].map(lambda x: x.strftime('%m')).astype(int)  
covid
```

Out[35]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month |
|-------|----------------|------------|--------|-----------|-----------|------------|------|-------|
| 0 | Afghanistan | 2020-01-02 | 0.0 | 0 | 0.0 | 2020Jan | 2020 | 1 |
| 1 | Afghanistan | 2020-01-03 | 0.0 | 1 | 0.0 | 2020Jan | 2020 | 1 |
| 2 | Afghanistan | 2020-01-04 | 4.0 | 197 | 5.0 | 2020Jan | 2020 | 1 |
| 3 | Afghanistan | 2020-01-05 | 68.0 | 2291 | 310.0 | 2020Jan | 2020 | 1 |
| 4 | Afghanistan | 2020-01-06 | 266.0 | 15753 | 1428.0 | 2020Jan | 2020 | 1 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 95337 | Zimbabwe | 2021-12-01 | 551.0 | 23239 | 13396.0 | 2021Dec | 2021 | 12 |
| 95338 | Zimbabwe | 2021-12-02 | 1393.0 | 35045 | 29959.0 | 2021Dec | 2021 | 12 |
| 95339 | Zimbabwe | 2021-12-03 | 1496.0 | 36423 | 33996.0 | 2021Dec | 2021 | 12 |
| 95340 | Zimbabwe | 2021-12-04 | 1542.0 | 37307 | 34901.0 | 2021Dec | 2021 | 12 |
| 95341 | Zimbabwe | 2021-12-05 | 1579.0 | 38466 | 36277.0 | 2021Dec | 2021 | 12 |

95342 rows × 8 columns

```
In [36]: china=(covid[covid['Country/Region']=='China'].groupby(['Country/Region','Year-month','Year','Month'])[['Confirmed']].max()).reset_index().sort_values('Year-month', ascending=True)
plt.figure(figsize=(18,6))
plt.plot(china['Year-month'],china['Confirmed'],label='China',marker='.')
for x, y in zip(china['Year-month'], china['Confirmed']):
    plt.text(x, y+200, str(int(y)), fontsize=8, ha='center', va='bottom',fontstyle='oblique',fontweight='bold')
plt.title('Confirmed cass overtime for China')
plt.xlabel('Dates')
plt.xticks(rotation=45)
plt.ylabel('Confirmed cases')
plt.grid()
plt.show()
```



Q3: Analyse the peak number of daily new cases in Germany, France, and Italy. Which country experienced the highest single-day surge, and when did it occur?

```
In [37]: c=['Germany', 'France', 'Italy']
ac=covid[covid['Country/Region'].isin(c)]
ac
```

Out[37]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month |
|-------|----------------|------------|----------|-----------|-----------|------------|------|-------|
| 30628 | France | 2020-01-02 | 0.0 | 6 | 0.0 | 2020Jan | 2020 | 1 |
| 30629 | France | 2020-01-03 | 2.0 | 130 | 12.0 | 2020Jan | 2020 | 1 |
| 30630 | France | 2020-01-04 | 4778.0 | 57122 | 11053.0 | 2020Jan | 2020 | 1 |
| 30631 | France | 2020-01-05 | 24566.0 | 169387 | 50212.0 | 2020Jan | 2020 | 1 |
| 30632 | France | 2020-01-06 | 28837.0 | 191382 | 68558.0 | 2020Jan | 2020 | 1 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 42479 | Italy | 2021-12-01 | 79819.0 | 2303263 | 1653404.0 | 2021Dec | 2021 | 12 |
| 42480 | Italy | 2021-12-02 | 93045.0 | 2697296 | 2202077.0 | 2021Dec | 2021 | 12 |
| 42481 | Italy | 2021-12-03 | 101564.0 | 3175807 | 2564926.0 | 2021Dec | 2021 | 12 |
| 42482 | Italy | 2021-12-04 | 114612.0 | 3779594 | 3140565.0 | 2021Dec | 2021 | 12 |
| 42483 | Italy | 2021-12-05 | 123544.0 | 4131078 | 3655112.0 | 2021Dec | 2021 | 12 |

1482 rows × 8 columns

In [38]:

```
ac['Lag_con']=ac.groupby('Country/Region')['Confirmed'].shift(1).fillna(0)
ac
```

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\3248965288.py:1: 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
ac['Lag_con']=ac.groupby('Country/Region')['Confirmed'].shift(1).fillna(0)

Out[38]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Lag_con |
|--------------|----------------|------------|----------|-----------|-----------|------------|------|-------|-----------|
| 30628 | France | 2020-01-02 | 0.0 | 6 | 0.0 | 2020Jan | 2020 | 1 | 0.0 |
| 30629 | France | 2020-01-03 | 2.0 | 130 | 12.0 | 2020Jan | 2020 | 1 | 6.0 |
| 30630 | France | 2020-01-04 | 4778.0 | 57122 | 11053.0 | 2020Jan | 2020 | 1 | 130.0 |
| 30631 | France | 2020-01-05 | 24566.0 | 169387 | 50212.0 | 2020Jan | 2020 | 1 | 57122.0 |
| 30632 | France | 2020-01-06 | 28837.0 | 191382 | 68558.0 | 2020Jan | 2020 | 1 | 169387.0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 42479 | Italy | 2021-12-01 | 79819.0 | 2303263 | 1653404.0 | 2021Dec | 2021 | 12 | 4123230.0 |
| 42480 | Italy | 2021-12-02 | 93045.0 | 2697296 | 2202077.0 | 2021Dec | 2021 | 12 | 2303263.0 |
| 42481 | Italy | 2021-12-03 | 101564.0 | 3175807 | 2564926.0 | 2021Dec | 2021 | 12 | 2697296.0 |
| 42482 | Italy | 2021-12-04 | 114612.0 | 3779594 | 3140565.0 | 2021Dec | 2021 | 12 | 3175807.0 |
| 42483 | Italy | 2021-12-05 | 123544.0 | 4131078 | 3655112.0 | 2021Dec | 2021 | 12 | 3779594.0 |

1482 rows × 9 columns

In [39]: `#ac.to_excel('LCC.xlsx')`In [40]: `ac['DailyNC_Con']=ac['Confirmed']-ac['Lag_con']
ac`

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\21586084.py:1: 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
`ac['DailyNC_Con']=ac['Confirmed']-ac['Lag_con']`

Out[40]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Lag_con | DailyNC_Con |
|-------|----------------|------------|----------|-----------|-----------|------------|------|-------|-----------|-------------|
| 30628 | France | 2020-01-02 | 0.0 | 6 | 0.0 | 2020Jan | 2020 | 1 | 0.0 | 6.0 |
| 30629 | France | 2020-01-03 | 2.0 | 130 | 12.0 | 2020Jan | 2020 | 1 | 6.0 | 124.0 |
| 30630 | France | 2020-01-04 | 4778.0 | 57122 | 11053.0 | 2020Jan | 2020 | 1 | 130.0 | 56992.0 |
| 30631 | France | 2020-01-05 | 24566.0 | 169387 | 50212.0 | 2020Jan | 2020 | 1 | 57122.0 | 112265.0 |
| 30632 | France | 2020-01-06 | 28837.0 | 191382 | 68558.0 | 2020Jan | 2020 | 1 | 169387.0 | 21995.0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 42479 | Italy | 2021-12-01 | 79819.0 | 2303263 | 1653404.0 | 2021Dec | 2021 | 12 | 4123230.0 | -1819967.0 |
| 42480 | Italy | 2021-12-02 | 93045.0 | 2697296 | 2202077.0 | 2021Dec | 2021 | 12 | 2303263.0 | 394033.0 |
| 42481 | Italy | 2021-12-03 | 101564.0 | 3175807 | 2564926.0 | 2021Dec | 2021 | 12 | 2697296.0 | 478511.0 |
| 42482 | Italy | 2021-12-04 | 114612.0 | 3779594 | 3140565.0 | 2021Dec | 2021 | 12 | 3175807.0 | 603787.0 |
| 42483 | Italy | 2021-12-05 | 123544.0 | 4131078 | 3655112.0 | 2021Dec | 2021 | 12 | 3779594.0 | 351484.0 |

1482 rows × 10 columns

In [41]:

```
maxval=ac.groupby(['Country/Region'])['DailyNC_Con'].idxmax()
print(maxval)
mdnc = ac.loc[maxval,['Country/Region','Date','DailyNC_Con']]
mdnc
```

Country/Region
France 30952
Germany 33067
Italy 42000
Name: DailyNC_Con, dtype: int64

Out[41]:

| | Country/Region | Date | DailyNC_Con |
|-------|----------------|------------|-------------|
| 30952 | France | 2020-12-11 | 1163615.0 |
| 33067 | Germany | 2021-06-05 | 592770.0 |
| 42000 | Italy | 2020-01-12 | 911566.0 |

Q4: Compare the recovery rates (recoveries/confirmed cases) between Canada and Australia as of December 31, 2020. Which country showed better management of the pandemic according to this metric?

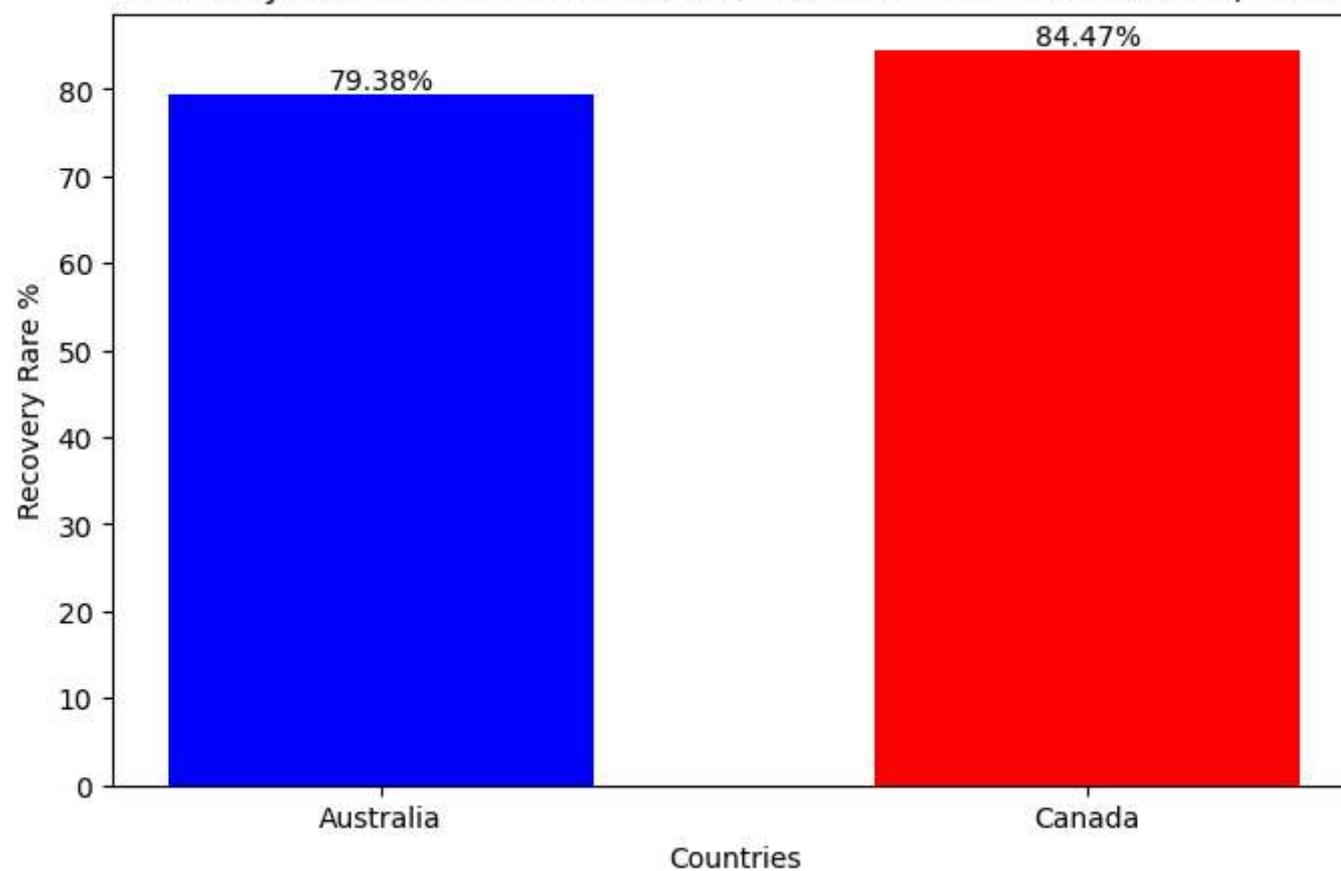
```
In [42]: fc = ['Canada', 'Australia']
covid['Recovered_Rate %']=((covid['Recovered']/covid['Confirmed'])*100).round(2).fillna(0).replace(np.inf,0)
c=covid[(covid['Country/Region'].isin(fc)) & (covid['Date']=='2020-12-31')]
c
```

```
Out[42]:
```

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Recovered_Rate % |
|-------|----------------|------------|---------|-----------|-----------|------------|------|-------|------------------|
| 4296 | Australia | 2020-12-31 | 909.0 | 28425 | 22565.0 | 2020Dec | 2020 | 12 | 79.38 |
| 16152 | Canada | 2020-12-31 | 15762.0 | 584409 | 493638.0 | 2020Dec | 2020 | 12 | 84.47 |

```
In [43]: plt.figure(figsize=(8,5))
bars=plt.bar(c['Country/Region'],c['Recovered_Rate %'],width=0.6,color=['b','r'])
plt.title('Recovery rates between Canada and Australia as of December 31, 2020')
plt.xlabel('Countries')
plt.ylabel('Recovery Rate %')
plt.bar_label(bars, fmt='%.2f%%')
plt.show()
```

Recovery rates between Canada and Australia as of December 31, 2020



Q5: What is the distribution of death rates (deaths/confirmed cases) among provinces in Canada? Identify the province with the highest and lowest death rate as of the latest data point.

```
In [44]: covid['Death_Rate %']=((covid['Deaths']/covid['Confirmed'])*100).round(2).fillna(0).replace(np.inf,0)  
covid
```

Out[44]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Recovered_Rate % | Death_Rate % |
|-------|----------------|------------|--------|-----------|-----------|------------|------|-------|------------------|--------------|
| 0 | Afghanistan | 2020-01-02 | 0.0 | 0 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 |
| 1 | Afghanistan | 2020-01-03 | 0.0 | 1 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 |
| 2 | Afghanistan | 2020-01-04 | 4.0 | 197 | 5.0 | 2020Jan | 2020 | 1 | 2.54 | 2.03 |
| 3 | Afghanistan | 2020-01-05 | 68.0 | 2291 | 310.0 | 2020Jan | 2020 | 1 | 13.53 | 2.97 |
| 4 | Afghanistan | 2020-01-06 | 266.0 | 15753 | 1428.0 | 2020Jan | 2020 | 1 | 9.06 | 1.69 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 95337 | Zimbabwe | 2021-12-01 | 551.0 | 23239 | 13396.0 | 2021Dec | 2021 | 12 | 57.64 | 2.37 |
| 95338 | Zimbabwe | 2021-12-02 | 1393.0 | 35045 | 29959.0 | 2021Dec | 2021 | 12 | 85.49 | 3.97 |
| 95339 | Zimbabwe | 2021-12-03 | 1496.0 | 36423 | 33996.0 | 2021Dec | 2021 | 12 | 93.34 | 4.11 |
| 95340 | Zimbabwe | 2021-12-04 | 1542.0 | 37307 | 34901.0 | 2021Dec | 2021 | 12 | 93.55 | 4.13 |
| 95341 | Zimbabwe | 2021-12-05 | 1579.0 | 38466 | 36277.0 | 2021Dec | 2021 | 12 | 94.31 | 4.10 |

95342 rows × 10 columns

```
In [45]: ucc=(covid_confirmed[covid_confirmed['Country/Region']=='Canada'].sort_values('Date').groupby('Province/State').tail(1)
      [['Province/State','Confirmed']].set_index('Province/State')
     )
ucc
```

Out[45]:

Confirmed

| Province/State | Confirmed |
|---------------------------|-----------|
| Saskatchewan | 43673 |
| Alberta | 213635 |
| British Columbia | 137223 |
| Diamond Princess | 0 |
| Grand Princess | 13 |
| Manitoba | 43143 |
| New Brunswick | 2024 |
| Newfoundland and Labrador | 1166 |
| Nova Scotia | 4301 |
| Nunavut | 594 |
| Ontario | 508043 |
| Prince Edward Island | 187 |
| Quebec | 360201 |
| Repatriated Travellers | 13 |
| Yukon | 84 |
| Northwest Territories | 120 |

```
In [46]: ucd=(covid_deaths[covid_deaths['Country/Region']=='Canada'].sort_values('Date').groupby('Province/State').tail(1)
      [['Province/State','Deaths']]).set_index('Province/State')
      )
ucd
```

Out[46]:

Deaths

| Province/State | |
|----------------------------------|---------|
| Saskatchewan | 508.0 |
| Alberta | 2123.0 |
| British Columbia | 1627.0 |
| Diamond Princess | 1.0 |
| Grand Princess | 0.0 |
| Manitoba | 1000.0 |
| New Brunswick | 41.0 |
| Newfoundland and Labrador | 6.0 |
| Nova Scotia | 71.0 |
| Nunavut | 4.0 |
| Ontario | 8360.0 |
| Prince Edward Island | 0.0 |
| Quebec | 11012.0 |
| Repatriated Travellers | 0.0 |
| Yukon | 2.0 |
| Northwest Territories | 0.0 |

In [47]:

```
cpdr=pd.merge(ucc,ucd,on='Province/State',how='inner')
cpdr['Death_Rate %']=((cpdr['Deaths']/cpdr['Confirmed'].replace(0,pd.NA))*100).fillna(0).round(2)
cpdr.sort_values('Death_Rate %',ascending=False,inplace=True)
cpdr
```

C:\Users\Krish\AppData\Local\Temp\ipykernel_7500\3734990589.py:2: FutureWarning: Downcasting object dtype arrays on .fillna, .ffill, .bfill is deprecated and will change in a future version. Call result.infer_objects(copy=False) instead. To opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
` cpdr['Death_Rate %']=((cpdr['Deaths']/cpdr['Confirmed'].replace(0,pd.NA))*100).fillna(0).round(2)

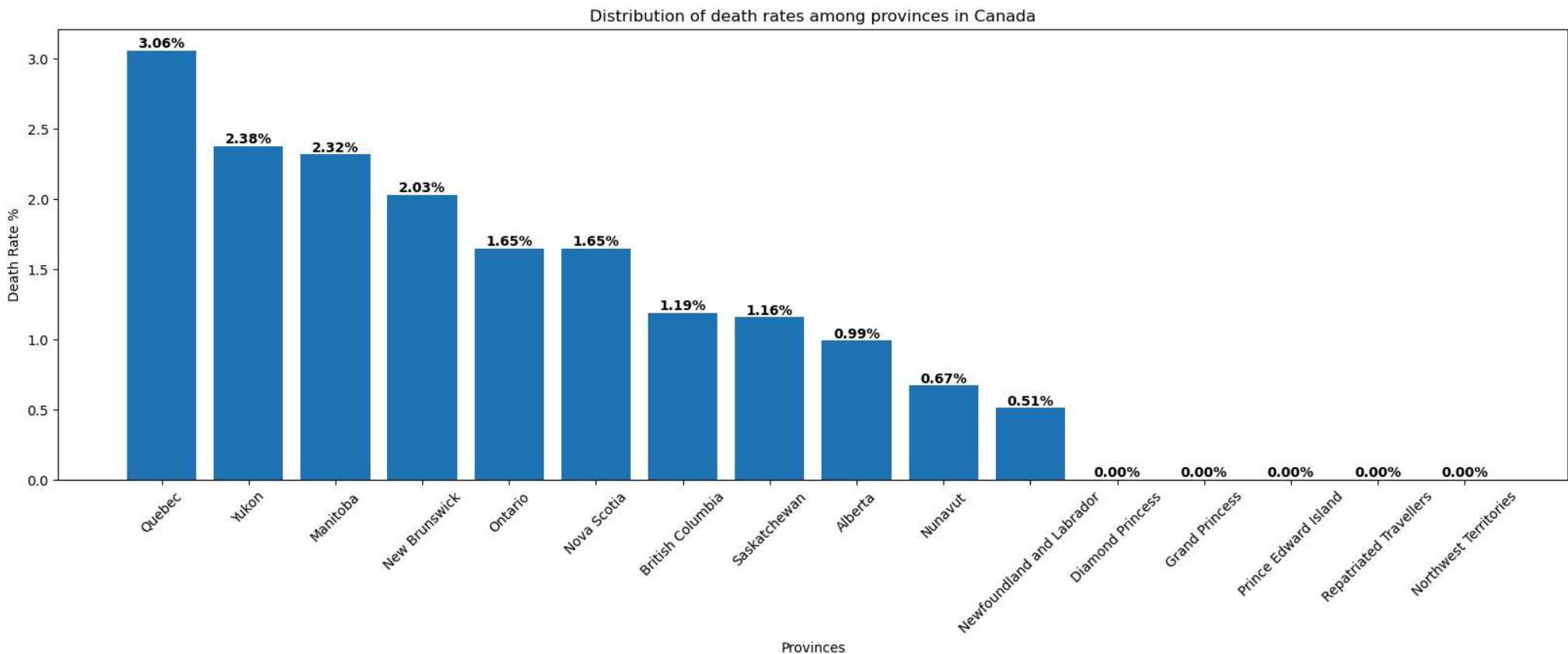
Out[47]:

Confirmed Deaths Death_Rate %

| Province/State | Confirmed | Deaths | Death_Rate % |
|----------------------------------|-----------|---------|--------------|
| Quebec | 360201 | 11012.0 | 3.06 |
| Yukon | 84 | 2.0 | 2.38 |
| Manitoba | 43143 | 1000.0 | 2.32 |
| New Brunswick | 2024 | 41.0 | 2.03 |
| Ontario | 508043 | 8360.0 | 1.65 |
| Nova Scotia | 4301 | 71.0 | 1.65 |
| British Columbia | 137223 | 1627.0 | 1.19 |
| Saskatchewan | 43673 | 508.0 | 1.16 |
| Alberta | 213635 | 2123.0 | 0.99 |
| Nunavut | 594 | 4.0 | 0.67 |
| Newfoundland and Labrador | 1166 | 6.0 | 0.51 |
| Diamond Princess | 0 | 1.0 | 0.00 |
| Grand Princess | 13 | 0.0 | 0.00 |
| Prince Edward Island | 187 | 0.0 | 0.00 |
| Repatriated Travellers | 13 | 0.0 | 0.00 |
| Northwest Territories | 120 | 0.0 | 0.00 |

In [48]:

```
plt.figure(figsize=(20,6))
bars=plt.bar(cpdr.index,cpdr['Death_Rate %'])
plt.title('Distribution of death rates among provinces in Canada')
plt.xlabel('Provinces')
plt.ylabel('Death Rate %')
plt.xticks(rotation=45)
plt.bar_label(bars, fmt='%.2f%%',fontweight='bold')
plt.show()
```



Q6: What is the total number of deaths reported for top 10 countries up to the current date?

```
In [49]: tcpc=covid.loc[covid.groupby(['Country/Region'])['Date'].idxmax()][['Country/Region', 'Deaths']].nlargest(10,columns='Deaths')  
tcpc
```

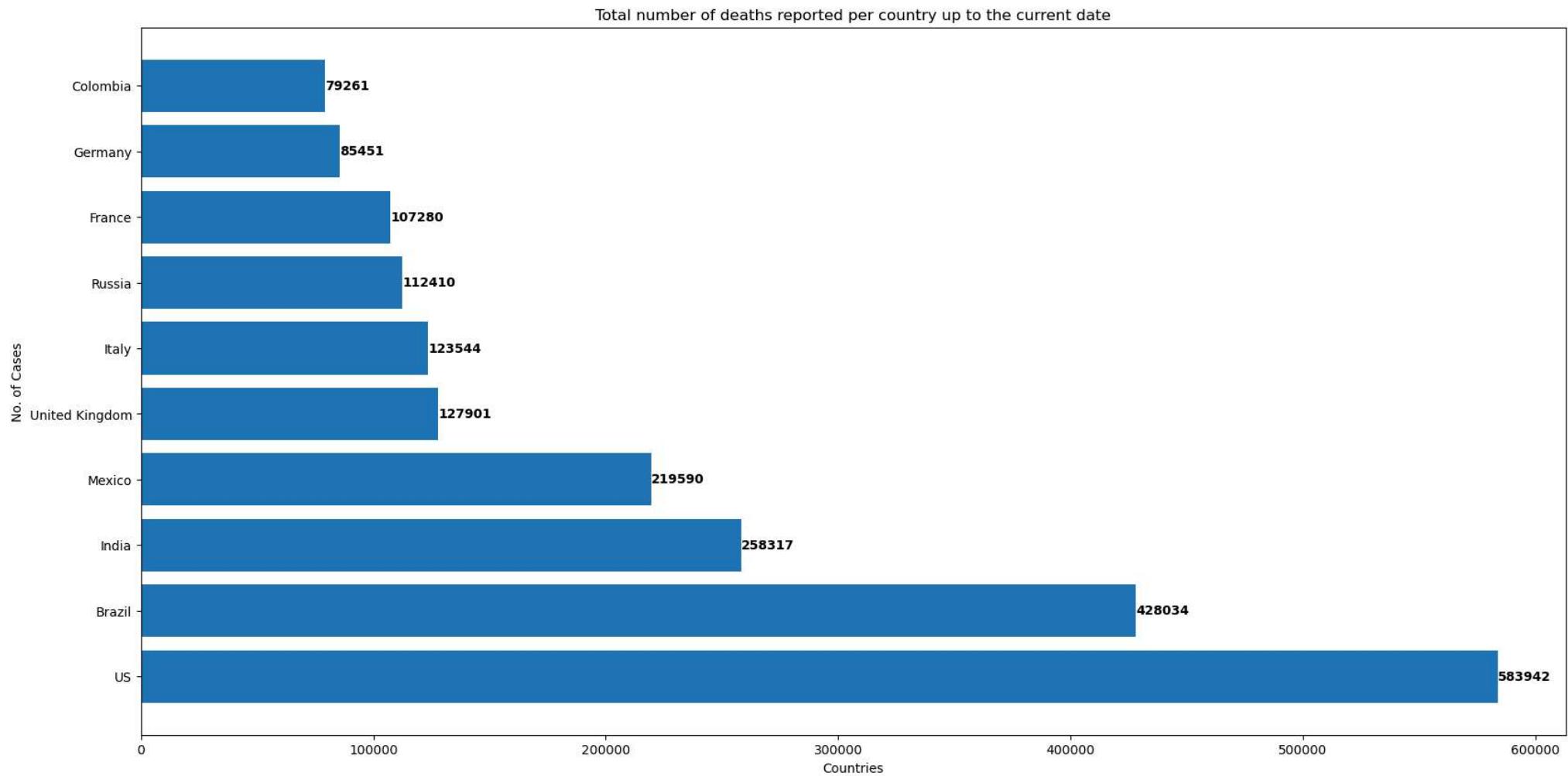
Out[49]:

Country/Region Deaths

| | | |
|--------------|----------------|----------|
| 88919 | US | 583942.0 |
| 11855 | Brazil | 428034.0 |
| 39519 | India | 258317.0 |
| 57303 | Mexico | 219590.0 |
| 90895 | United Kingdom | 127901.0 |
| 42483 | Italy | 123544.0 |
| 71135 | Russia | 112410.0 |
| 31121 | France | 107280.0 |
| 33097 | Germany | 85451.0 |
| 18771 | Colombia | 79261.0 |

In [50]:

```
plt.figure(figsize=(20,10))
bars=plt.barh(tcpc['Country/Region'],tcpc['Deaths'])
plt.title('Total number of deaths reported per country up to the current date')
plt.xlabel('Countries')
plt.ylabel('No. of Cases')
plt.bar_label(bars,fontweight='bold')
#plt.xticks(rotation=90)
plt.show()
```



Q7: What are the top 5 countries with the highest average daily deaths?

```
In [51]: covid['DailyDeaths']=covid.groupby('Country/Region')['Deaths'].diff().fillna(0).clip(lower=0)  
covid
```

Out[51]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Recovered_Rate % | Death_Rate % | DailyDeaths |
|-------|----------------|------------|--------|-----------|-----------|------------|------|-------|------------------|--------------|-------------|
| 0 | Afghanistan | 2020-01-02 | 0.0 | 0 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 | 0.0 |
| 1 | Afghanistan | 2020-01-03 | 0.0 | 1 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 | 0.0 |
| 2 | Afghanistan | 2020-01-04 | 4.0 | 197 | 5.0 | 2020Jan | 2020 | 1 | 2.54 | 2.03 | 4.0 |
| 3 | Afghanistan | 2020-01-05 | 68.0 | 2291 | 310.0 | 2020Jan | 2020 | 1 | 13.53 | 2.97 | 64.0 |
| 4 | Afghanistan | 2020-01-06 | 266.0 | 15753 | 1428.0 | 2020Jan | 2020 | 1 | 9.06 | 1.69 | 198.0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 95337 | Zimbabwe | 2021-12-01 | 551.0 | 23239 | 13396.0 | 2021Dec | 2021 | 12 | 57.64 | 2.37 | 0.0 |
| 95338 | Zimbabwe | 2021-12-02 | 1393.0 | 35045 | 29959.0 | 2021Dec | 2021 | 12 | 85.49 | 3.97 | 842.0 |
| 95339 | Zimbabwe | 2021-12-03 | 1496.0 | 36423 | 33996.0 | 2021Dec | 2021 | 12 | 93.34 | 4.11 | 103.0 |
| 95340 | Zimbabwe | 2021-12-04 | 1542.0 | 37307 | 34901.0 | 2021Dec | 2021 | 12 | 93.55 | 4.13 | 46.0 |
| 95341 | Zimbabwe | 2021-12-05 | 1579.0 | 38466 | 36277.0 | 2021Dec | 2021 | 12 | 94.31 | 4.10 | 37.0 |

95342 rows × 11 columns

In [52]:

```
adc=covid.groupby('Country/Region')['DailyDeaths'].mean().round(0).reset_index().nlargest(5,columns='DailyDeaths')
adc
```

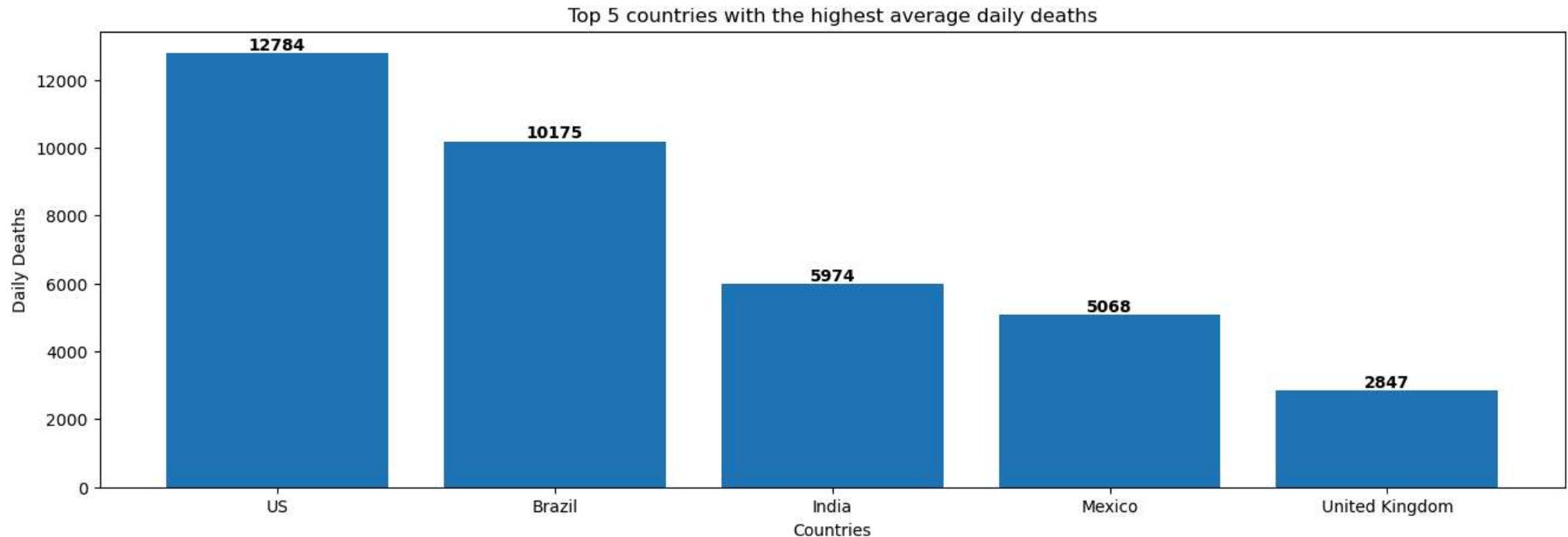
Out[52]:

| | Country/Region | DailyDeaths |
|-----|----------------|-------------|
| 179 | US | 12784.0 |
| 23 | Brazil | 10175.0 |
| 79 | India | 5974.0 |
| 115 | Mexico | 5068.0 |
| 183 | United Kingdom | 2847.0 |

In [53]:

```
plt.figure(figsize=(16,5))
bars=plt.bar(adc['Country/Region'],adc['DailyDeaths'])
plt.title('Top 5 countries with the highest average daily deaths')
plt.xlabel('Countries')
plt.ylabel('Daily Deaths')
```

```
plt.bar_label(bars,fontweight='bold')
plt.show()
```



Q8: How have the total deaths evolved over time in the United States?

```
In [54]: us=(covid[covid['Country/Region']=='US'].groupby(['Country/Region','Date'])['Deaths'].sum().reset_index())
us
```

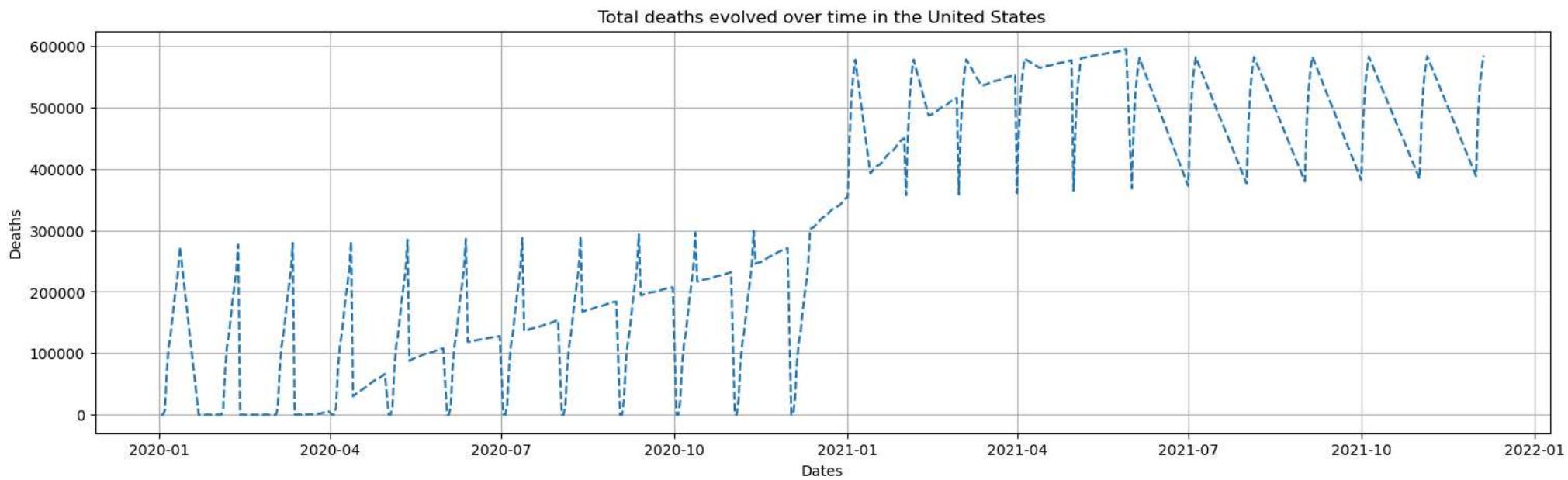
Out[54]:

| | Country/Region | Date | Deaths |
|-----|----------------|------------|----------|
| 0 | US | 2020-01-02 | 0.0 |
| 1 | US | 2020-01-03 | 1.0 |
| 2 | US | 2020-01-04 | 6653.0 |
| 3 | US | 2020-01-05 | 68144.0 |
| 4 | US | 2020-01-06 | 108616.0 |
| ... | ... | ... | ... |
| 489 | US | 2021-12-01 | 388001.0 |
| 490 | US | 2021-12-02 | 484494.0 |
| 491 | US | 2021-12-03 | 534234.0 |
| 492 | US | 2021-12-04 | 563103.0 |
| 493 | US | 2021-12-05 | 583942.0 |

494 rows × 3 columns

In [55]:

```
plt.figure(figsize=(18,5))
plt.plot(us['Date'],us['Deaths'],linestyle='--')
plt.title('Total deaths evolved over time in the United States')
plt.xlabel('Dates')
plt.ylabel('Deaths')
plt.grid()
plt.show()
```



Q9: Analyze the monthly sum of confirmed cases, deaths, and recoveries for 'US', 'Brazil', 'Italy' & 'India' to understand the progression of the pandemic.[From the merged dataset]

In [56]:

```
covid['DailyConfirmed']=covid.groupby('Country/Region')['Confirmed'].diff().fillna(0).clip(lower=0)
covid['DailyRecovered']=covid.groupby('Country/Region')['Recovered'].diff().fillna(0).clip(lower=0)
covid
```

Out[56]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Recovered_Rate % | Death_Rate % | DailyDeaths | DailyConfirmed | DailyRecovered |
|-------|----------------|------------|--------|-----------|-----------|------------|------|-------|------------------|--------------|-------------|----------------|----------------|
| 0 | Afghanistan | 2020-01-02 | 0.0 | 0 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 | 0.0 | 0.0 | 0.0 |
| 1 | Afghanistan | 2020-01-03 | 0.0 | 1 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 | 0.0 | 1.0 | 1.0 |
| 2 | Afghanistan | 2020-01-04 | 4.0 | 197 | 5.0 | 2020Jan | 2020 | 1 | 2.54 | 2.03 | 4.0 | 196.0 | 196.0 |
| 3 | Afghanistan | 2020-01-05 | 68.0 | 2291 | 310.0 | 2020Jan | 2020 | 1 | 13.53 | 2.97 | 64.0 | 2094.0 | 2094.0 |
| 4 | Afghanistan | 2020-01-06 | 266.0 | 15753 | 1428.0 | 2020Jan | 2020 | 1 | 9.06 | 1.69 | 198.0 | 13462.0 | 13462.0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 95337 | Zimbabwe | 2021-12-01 | 551.0 | 23239 | 13396.0 | 2021Dec | 2021 | 12 | 57.64 | 2.37 | 0.0 | 0.0 | 0.0 |
| 95338 | Zimbabwe | 2021-12-02 | 1393.0 | 35045 | 29959.0 | 2021Dec | 2021 | 12 | 85.49 | 3.97 | 842.0 | 11806.0 | 11806.0 |
| 95339 | Zimbabwe | 2021-12-03 | 1496.0 | 36423 | 33996.0 | 2021Dec | 2021 | 12 | 93.34 | 4.11 | 103.0 | 1378.0 | 1378.0 |
| 95340 | Zimbabwe | 2021-12-04 | 1542.0 | 37307 | 34901.0 | 2021Dec | 2021 | 12 | 93.55 | 4.13 | 46.0 | 884.0 | 884.0 |
| 95341 | Zimbabwe | 2021-12-05 | 1579.0 | 38466 | 36277.0 | 2021Dec | 2021 | 12 | 94.31 | 4.10 | 37.0 | 1159.0 | 1159.0 |

95342 rows × 13 columns

In [57]:

```
monthly_data=(covid.groupby(['Country/Region','Year-month','Year','Month'])[['DailyDeaths','DailyConfirmed','DailyRecovered']].sum()
            .rename(columns={
                'DailyDeaths':'MonthlyDeaths',
                'DailyConfirmed':'MonthlyConfirmed',
                'DailyRecovered':'MonthlyRecovered'
            })
            .reset_index().sort_values(['Country/Region','Year','Month'])
monthly_data
```

Out[57]:

| | Country/Region | Year-month | Year | Month | MonthlyDeaths | MonthlyConfirmed | MonthlyRecovered |
|------|----------------|------------|------|-------|---------------|------------------|------------------|
| 4 | Afghanistan | 2020Jan | 2020 | 1 | 1822.0 | 46516.0 | 36946.0 |
| 3 | Afghanistan | 2020Feb | 2020 | 2 | 1841.0 | 46719.0 | 37218.0 |
| 7 | Afghanistan | 2020Mar | 2020 | 3 | 1850.0 | 46999.0 | 37265.0 |
| 0 | Afghanistan | 2020Apr | 2020 | 4 | 1888.0 | 48298.0 | 37488.0 |
| 8 | Afghanistan | 2020May | 2020 | 5 | 1990.0 | 57053.0 | 38073.0 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 4621 | Zimbabwe | 2021Aug | 2021 | 8 | 1108.0 | 18754.0 | 23868.0 |
| 4631 | Zimbabwe | 2021Sep | 2021 | 9 | 1093.0 | 17920.0 | 23624.0 |
| 4630 | Zimbabwe | 2021Oct | 2021 | 10 | 1069.0 | 16956.0 | 23626.0 |
| 4629 | Zimbabwe | 2021Nov | 2021 | 11 | 1051.0 | 16151.0 | 23008.0 |
| 4622 | Zimbabwe | 2021Dec | 2021 | 12 | 1028.0 | 15227.0 | 22881.0 |

4632 rows × 7 columns

In [58]:

```
mu=monthly_data[monthly_data['Country/Region']=='US'].sort_values(['Year','Month'])
mu.head()
```

Out[58]:

| | Country/Region | Year-month | Year | Month | MonthlyDeaths | MonthlyConfirmed | MonthlyRecovered |
|------|----------------|------------|------|-------|---------------|------------------|------------------|
| 4300 | US | 2020Jan | 2020 | 1 | 273566.0 | 13858550.0 | 5226581.0 |
| 4299 | US | 2020Feb | 2020 | 2 | 276382.0 | 14061111.0 | 5322132.0 |
| 4303 | US | 2020Mar | 2020 | 3 | 284656.0 | 14474792.0 | 5411030.0 |
| 4296 | US | 2020Apr | 2020 | 4 | 318584.0 | 15000135.0 | 5580854.0 |
| 4304 | US | 2020May | 2020 | 5 | 304723.0 | 15130106.0 | 5777354.0 |

In [59]:

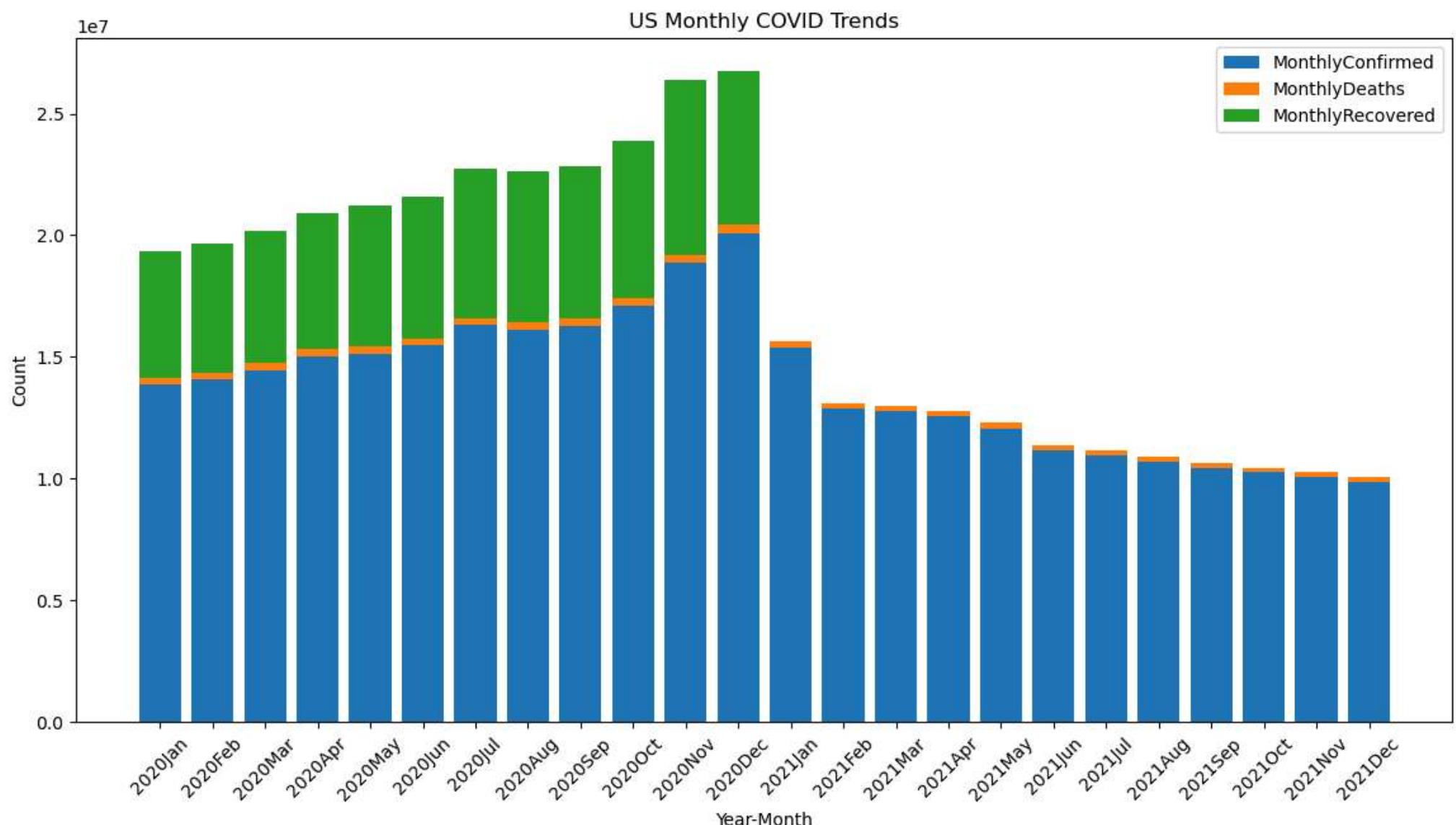
```
plt.figure(figsize=(14,7))

plt.bar(mu['Year-month'],
        mu['MonthlyConfirmed'],
        label='MonthlyConfirmed')
```

```
plt.bar(mu[ 'Year-month' ],
        mu[ 'MonthlyDeaths' ],
        bottom=mu[ 'MonthlyConfirmed' ],
        label='MonthlyDeaths')

plt.bar(mu[ 'Year-month' ],
        mu[ 'MonthlyRecovered' ],
        bottom=mu[ 'MonthlyConfirmed' ] + mu[ 'MonthlyDeaths' ],
        label='MonthlyRecovered')

plt.xticks(rotation=45)
plt.xlabel("Year-Month")
plt.ylabel("Count")
plt.title("US Monthly COVID Trends")
plt.legend()
plt.show()
```



```
In [60]: mb=monthly_data[monthly_data['Country/Region']=='Brazil'].sort_values(['Year','Month'])
mb.head()
```

Out[60]:

| | Country/Region | Year-month | Year | Month | MonthlyDeaths | MonthlyConfirmed | MonthlyRecovered |
|-----|----------------|------------|------|-------|---------------|------------------|------------------|
| 556 | Brazil | 2020Jan | 2020 | 1 | 173817.0 | 6386787.0 | 5712305.0 |
| 555 | Brazil | 2020Feb | 2020 | 2 | 174515.0 | 6436652.0 | 5759294.0 |
| 559 | Brazil | 2020Mar | 2020 | 3 | 175471.0 | 6492650.0 | 5792666.0 |
| 552 | Brazil | 2020Apr | 2020 | 4 | 180642.0 | 6597725.0 | 5861507.0 |
| 560 | Brazil | 2020May | 2020 | 5 | 192702.0 | 6901889.0 | 5992969.0 |

In [61]:

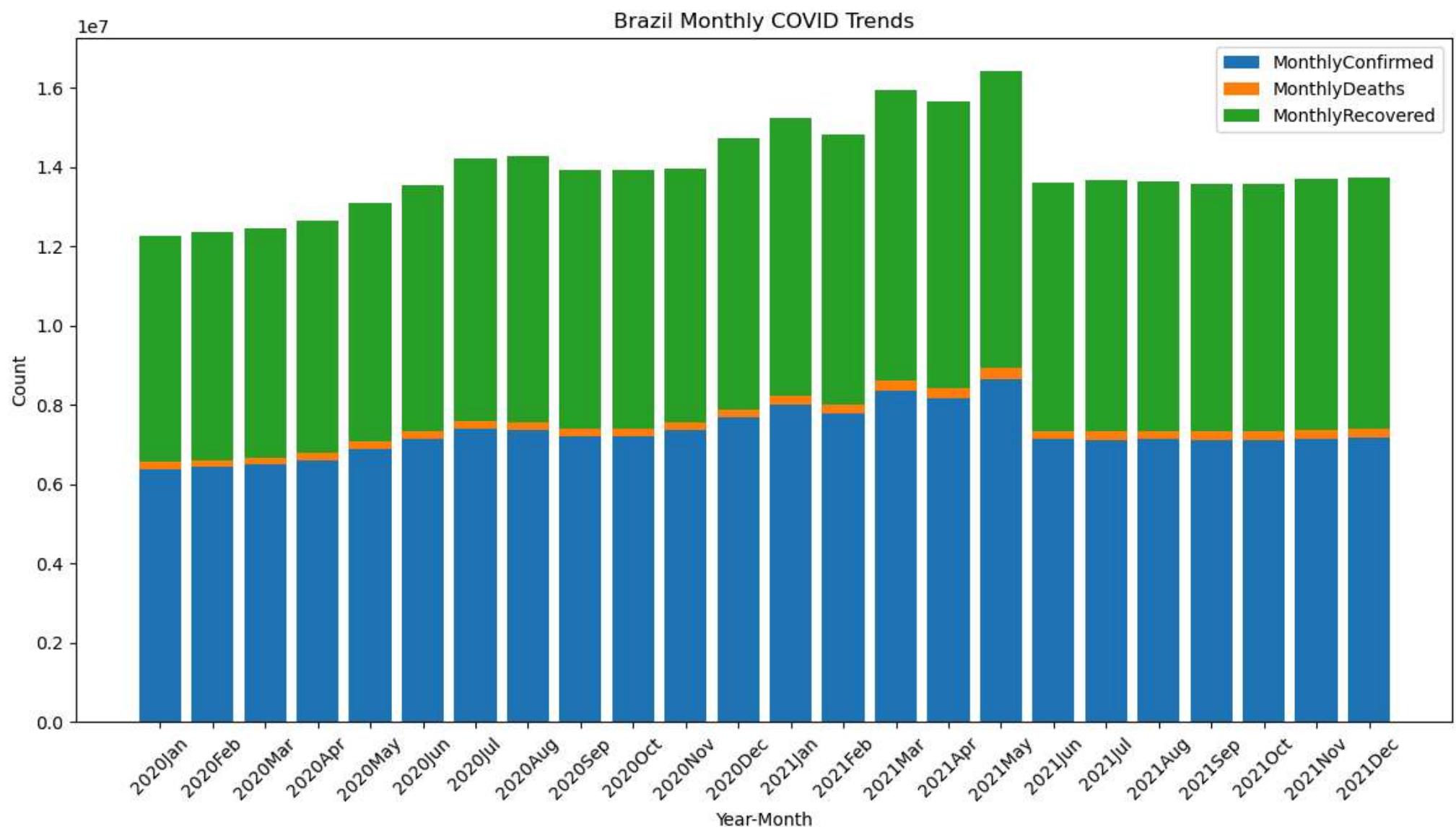
```
plt.figure(figsize=(14,7))

plt.bar(mb['Year-month'],
        mb['MonthlyConfirmed'],
        label='MonthlyConfirmed')

plt.bar(mb['Year-month'],
        mb['MonthlyDeaths'],
        bottom=mb['MonthlyConfirmed'],
        label='MonthlyDeaths')

plt.bar(mb['Year-month'],
        mb['MonthlyRecovered'],
        bottom=mb['MonthlyConfirmed'] + mb['MonthlyDeaths'],
        label='MonthlyRecovered')

plt.xticks(rotation=45)
plt.xlabel("Year-Month")
plt.ylabel("Count")
plt.title("Brazil Monthly COVID Trends")
plt.legend()
plt.show()
```



```
In [62]: mi=monthly_data[monthly_data['Country/Region']=='Italy'].sort_values(['Year','Month'])
mi.head()
```

Out[62]:

| | Country/Region | Year-month | Year | Month | MonthlyDeaths | MonthlyConfirmed | MonthlyRecovered |
|-------------|----------------|------------|------|-------|---------------|------------------|------------------|
| 2044 | Italy | 2020Jan | 2020 | 1 | 56361.0 | 1620901.0 | 784595.0 |
| 2043 | Italy | 2020Feb | 2020 | 2 | 57074.0 | 1642733.0 | 823382.0 |
| 2047 | Italy | 2020Mar | 2020 | 3 | 69200.0 | 1752959.0 | 861099.0 |
| 2040 | Italy | 2020Apr | 2020 | 4 | 66354.0 | 1734884.0 | 912895.0 |
| 2048 | Italy | 2020May | 2020 | 5 | 61823.0 | 1720882.0 | 941274.0 |

In [63]:

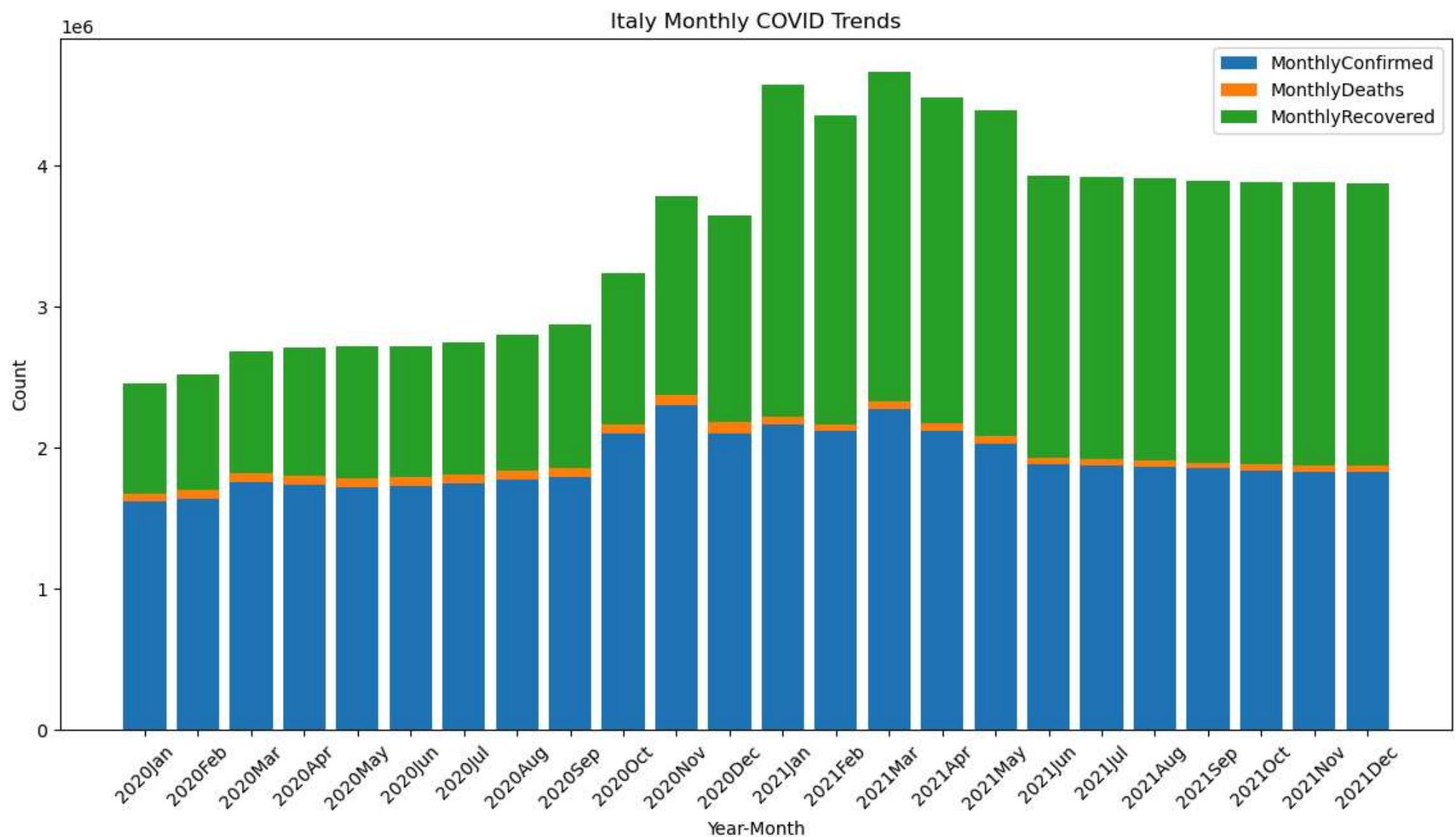
```
plt.figure(figsize=(14,7))

plt.bar(mi['Year-month'],
        mi['MonthlyConfirmed'],
        label='MonthlyConfirmed')

plt.bar(mi['Year-month'],
        mi['MonthlyDeaths'],
        bottom=mi['MonthlyConfirmed'],
        label='MonthlyDeaths')

plt.bar(mi['Year-month'],
        mi['MonthlyRecovered'],
        bottom=mi['MonthlyConfirmed'] + mi['MonthlyDeaths'],
        label='MonthlyRecovered')

plt.xticks(rotation=45)
plt.xlabel("Year-Month")
plt.ylabel("Count")
plt.title("Italy Monthly COVID Trends")
plt.legend()
plt.show()
```



```
In [64]: mh=monthly_data[monthly_data['Country/Region']=='India'].sort_values(['Year','Month'])
mh.head()
```

Out[64]:

| | Country/Region | Year-month | Year | Month | MonthlyDeaths | MonthlyConfirmed | MonthlyRecovered |
|-------------|----------------|------------|------|-------|---------------|------------------|------------------|
| 1900 | India | 2020Jan | 2020 | 1 | 138122.0 | 9499413.0 | 8932647.0 |
| 1899 | India | 2020Feb | 2020 | 2 | 138648.0 | 9534963.0 | 8973376.0 |
| 1903 | India | 2020Mar | 2020 | 3 | 139222.0 | 9572871.0 | 9016408.0 |
| 1896 | India | 2020Apr | 2020 | 4 | 140496.0 | 9632618.0 | 9066709.0 |
| 1904 | India | 2020May | 2020 | 5 | 143039.0 | 9756773.0 | 9166244.0 |

In [65]:

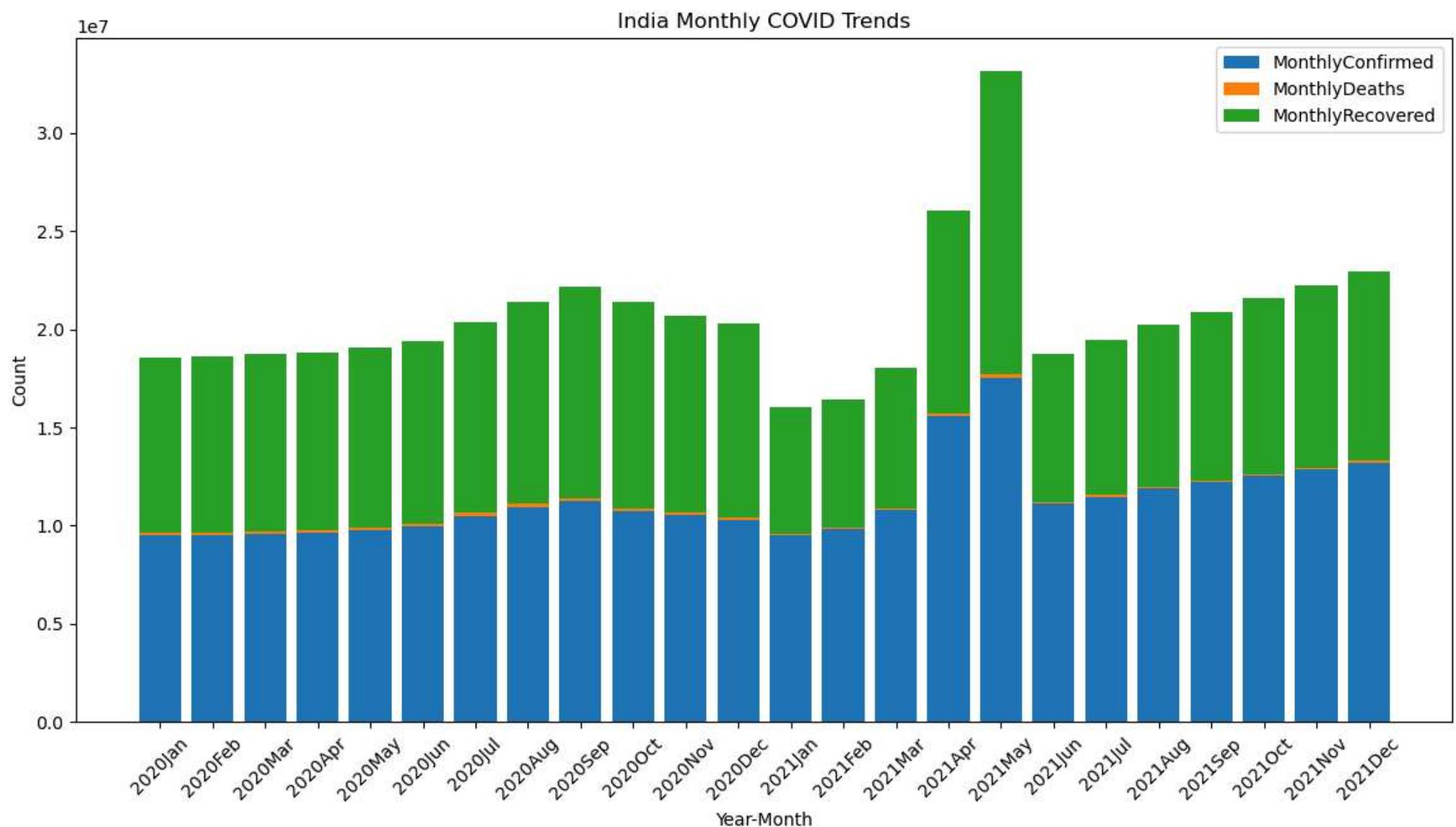
```
plt.figure(figsize=(14,7))

plt.bar(mh[ 'Year-month' ],
        mh[ 'MonthlyConfirmed' ],
        label='MonthlyConfirmed')

plt.bar(mh[ 'Year-month' ],
        mh[ 'MonthlyDeaths' ],
        bottom=mh[ 'MonthlyConfirmed' ],
        label='MonthlyDeaths')

plt.bar(mh[ 'Year-month' ],
        mh[ 'MonthlyRecovered' ],
        bottom=mh[ 'MonthlyConfirmed' ] + mh[ 'MonthlyDeaths' ],
        label='MonthlyRecovered')

plt.xticks(rotation=45)
plt.xlabel("Year-Month")
plt.ylabel("Count")
plt.title("India Monthly COVID Trends")
plt.legend()
plt.show()
```



Q10: For the combined dataset, identify the three countries with the highest average death rates (deaths/confirmed cases) throughout 2020. What might this indicate about the pandemic's impact in these countries?

```
In [66]: hdr=covid[covid['Year']==2020]
hdr.groupby(['Country/Region'])[['Death_Rate %']].mean().round(2).reset_index().nlargest(3,columns='Death_Rate %')
```

Out[66]:

| | Country/Region | Death_Rate % |
|-----|----------------|--------------|
| 190 | Yemen | 19.24 |
| 105 | MS Zaandam | 17.71 |
| 85 | Italy | 9.46 |

Q11: Using the merged dataset, compare the total number of recoveries to the total number of deaths in South Africa. What can this tell us about the outcomes of COVID-19 cases in the country?

In [67]:

```
sa=covid[covid['Country/Region']=='South Africa'].sort_values(['Year','Month'])  
sa
```

Out[67]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Recovered_Rate % | Death_Rate % | DailyDeaths | DailyConfirmed | Daily |
|-------|----------------|------------|---------|-----------|-----------|------------|------|-------|------------------|--------------|-------------|----------------|-------|
| 79534 | South Africa | 2020-01-02 | 0.0 | 0 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 | 0.0 | 0.0 | 0.0 |
| 79535 | South Africa | 2020-01-03 | 0.0 | 0 | 0.0 | 2020Jan | 2020 | 1 | 0.00 | 0.00 | 0.0 | 0.0 | 0.0 |
| 79536 | South Africa | 2020-01-04 | 5.0 | 1380 | 50.0 | 2020Jan | 2020 | 1 | 3.62 | 0.36 | 5.0 | 1380.0 | |
| 79537 | South Africa | 2020-01-05 | 116.0 | 5951 | 2382.0 | 2020Jan | 2020 | 1 | 40.03 | 1.95 | 111.0 | 4571.0 | |
| 79538 | South Africa | 2020-01-06 | 705.0 | 34357 | 17291.0 | 2020Jan | 2020 | 1 | 50.33 | 2.05 | 589.0 | 28406.0 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 80023 | South Africa | 2021-12-01 | 34334.0 | 1259748 | 1019123.0 | 2021Dec | 2021 | 12 | 80.90 | 2.73 | 0.0 | 0.0 | |
| 80024 | South Africa | 2021-12-02 | 47670.0 | 1487681 | 1383626.0 | 2021Dec | 2021 | 12 | 93.01 | 3.20 | 13336.0 | 227933.0 | |
| 80025 | South Africa | 2021-12-03 | 51179.0 | 1526873 | 1449654.0 | 2021Dec | 2021 | 12 | 94.94 | 3.35 | 3509.0 | 39192.0 | |
| 80026 | South Africa | 2021-12-04 | 53356.0 | 1559113 | 1484356.0 | 2021Dec | 2021 | 12 | 95.21 | 3.42 | 2177.0 | 32240.0 | |
| 80027 | South Africa | 2021-12-05 | 54968.0 | 1602031 | 1519734.0 | 2021Dec | 2021 | 12 | 94.86 | 3.43 | 1612.0 | 42918.0 | |

494 rows × 13 columns



In [68]:

```
saaa=sa.groupby('Country/Region')[['Deaths','Recovered']].max()
saaa
```

Out[68]:

Deaths Recovered

Country/Region

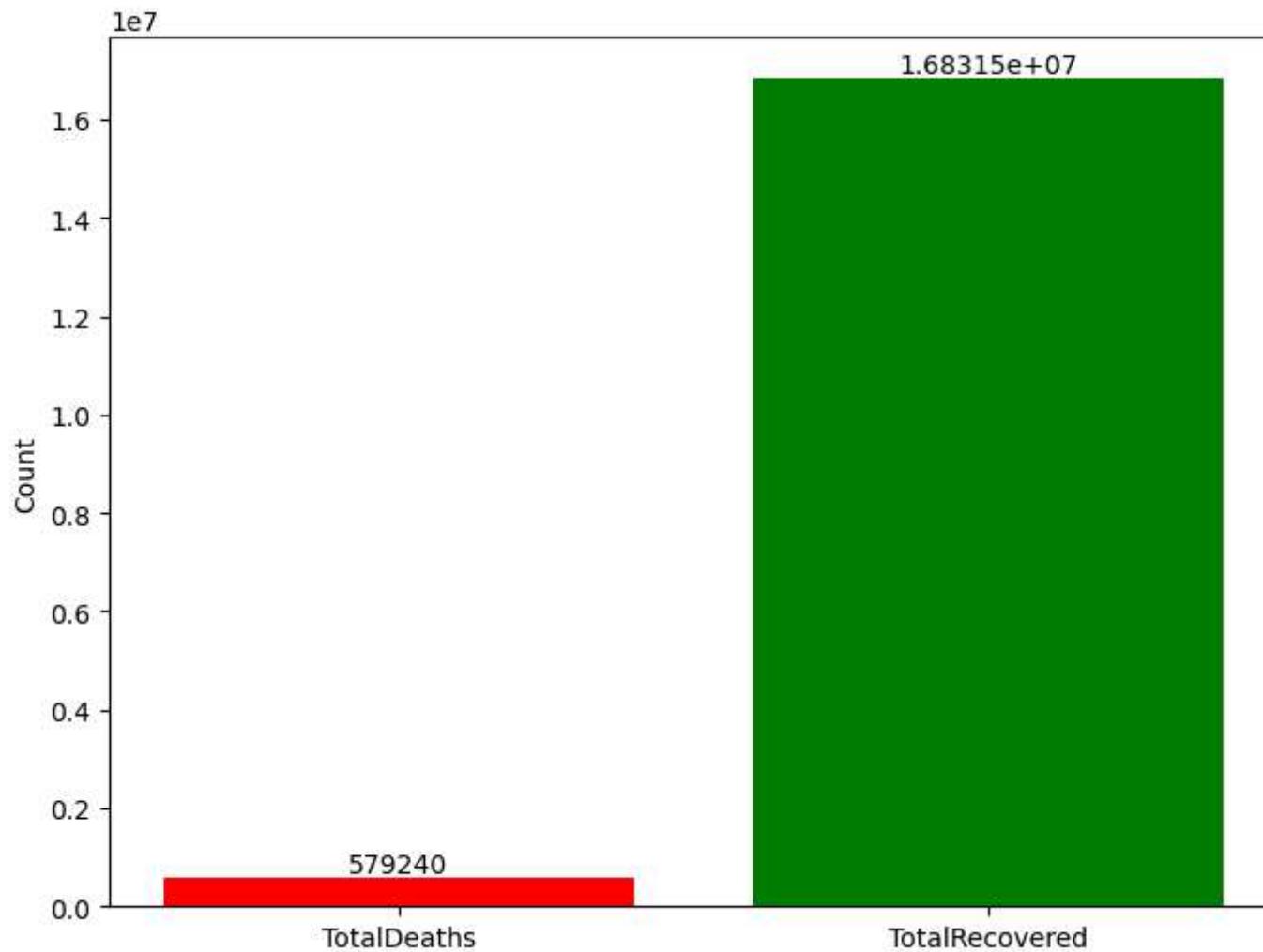
| Country/Region | Deaths | Recovered |
|----------------|---------|-----------|
| South Africa | 56363.0 | 1554184.0 |

```
In [69]: saa=saa.groupby('Country/Region')[['DailyDeaths','DailyRecovered']].sum()  
saa
```

```
Out[69]:          DailyDeaths  DailyRecovered
```

| Country/Region | DailyDeaths | DailyRecovered |
|----------------|-------------|----------------|
| South Africa | 579240.0 | 16831548.0 |

```
In [70]: plt.figure(figsize=(8,6))  
values = saa.iloc[0] # first row (South Africa)  
labels = ['TotalDeaths', 'TotalRecovered']  
counts = [values['DailyDeaths'], values['DailyRecovered']]  
bars=plt.bar(labels, counts,color=['#FF0000','#008000'])  
plt.ylabel('Count')  
plt.bar_label(bars)  
plt.show()
```



Q12: Analyze the ratio of recoveries to confirmed cases for the United States on a monthly basis from March 2020 to May 2021. Which month experienced the highest recovery ratio, and what could be the potential reasons?

```
In [71]: usda=covid[(covid['Country/Region']=='US') & (covid['Date'].between('2020-03-01','2021-05-31'))]  
usda
```

Out[71]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | Recovered_Rate % | Death_Rate % | DailyDeaths | DailyConfirmed | Dai |
|-------|----------------|------------|----------|-----------|-----------|------------|------|-------|------------------|--------------|-------------|----------------|-----|
| 88475 | US | 2020-03-02 | 0.0 | 11 | 0.0 | 2020Mar | 2020 | 3 | 0.00 | 0.00 | 0.0 | 0.0 | 0.0 |
| 88476 | US | 2020-03-03 | 7.0 | 74 | 7.0 | 2020Mar | 2020 | 3 | 9.46 | 9.46 | 7.0 | 63.0 | |
| 88477 | US | 2020-03-04 | 9596.0 | 289087 | 9707.0 | 2020Mar | 2020 | 3 | 3.36 | 3.32 | 9589.0 | 289013.0 | |
| 88478 | US | 2020-03-05 | 71065.0 | 1167593 | 180152.0 | 2020Mar | 2020 | 3 | 15.43 | 6.09 | 61469.0 | 878506.0 | |
| 88479 | US | 2020-03-06 | 110607.0 | 1857511 | 479258.0 | 2020Mar | 2020 | 3 | 25.80 | 5.95 | 39542.0 | 689918.0 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 88880 | US | 2021-05-25 | 590941.0 | 33166418 | 0.0 | 2021May | 2021 | 5 | 0.00 | 1.78 | 621.0 | 22756.0 | |
| 88881 | US | 2021-05-26 | 591950.0 | 33190470 | 0.0 | 2021May | 2021 | 5 | 0.00 | 1.78 | 1009.0 | 24052.0 | |
| 88882 | US | 2021-05-27 | 593288.0 | 33217995 | 0.0 | 2021May | 2021 | 5 | 0.00 | 1.79 | 1338.0 | 27525.0 | |
| 88883 | US | 2021-05-28 | 593963.0 | 33239963 | 0.0 | 2021May | 2021 | 5 | 0.00 | 1.79 | 675.0 | 21968.0 | |
| 88884 | US | 2021-05-29 | 594306.0 | 33251939 | 0.0 | 2021May | 2021 | 5 | 0.00 | 1.79 | 343.0 | 11976.0 | |

410 rows × 13 columns

In [72]:

```

usra=(usda.loc[usda.groupby(['Year-month','Year','Month'])['Date'].idxmax()]
[[ 'Country/Region', 'Date', 'Deaths', 'Confirmed', 'Recovered', 'Year-month', 'Year', 'Month']]
).sort_values(['Year', 'Month'])

usra['MRR%']=((usra['Recovered']/usra['Confirmed'])*100).round(2)
usra

```

Out[72]:

| | Country/Region | Date | Deaths | Confirmed | Recovered | Year-month | Year | Month | MRR% |
|-------|----------------|------------|----------|-----------|-----------|------------|------|-------|-------|
| 88504 | US | 2020-03-31 | 5370.0 | 192301 | 7024.0 | 2020Mar | 2020 | 3 | 3.65 |
| 88533 | US | 2020-04-30 | 66235.0 | 1081020 | 153947.0 | 2020Apr | 2020 | 4 | 14.24 |
| 88563 | US | 2020-05-31 | 107840.0 | 1798718 | 444758.0 | 2020May | 2020 | 5 | 24.73 |
| 88592 | US | 2020-06-30 | 127607.0 | 2642174 | 720631.0 | 2020Jun | 2020 | 6 | 27.27 |
| 88622 | US | 2020-07-31 | 154157.0 | 4567420 | 1438160.0 | 2020Jul | 2020 | 7 | 31.49 |
| 88652 | US | 2020-08-31 | 183818.0 | 6026895 | 2184825.0 | 2020Aug | 2020 | 8 | 36.25 |
| 88681 | US | 2020-09-30 | 207217.0 | 7235428 | 2840688.0 | 2020Sep | 2020 | 9 | 39.26 |
| 88711 | US | 2020-10-31 | 231749.0 | 9165619 | 3612478.0 | 2020Oct | 2020 | 10 | 39.41 |
| 88740 | US | 2020-11-30 | 271020.0 | 13670332 | 5146319.0 | 2020Nov | 2020 | 11 | 37.65 |
| 88770 | US | 2020-12-31 | 352163.0 | 20099363 | 0.0 | 2020Dec | 2020 | 12 | 0.00 |
| 88794 | US | 2021-01-31 | 449448.0 | 26247053 | 0.0 | 2021Jan | 2021 | 1 | 0.00 |
| 88815 | US | 2021-02-28 | 515084.0 | 28648744 | 0.0 | 2021Feb | 2021 | 2 | 0.00 |
| 88839 | US | 2021-03-31 | 552661.0 | 30462210 | 0.0 | 2021Mar | 2021 | 3 | 0.00 |
| 88862 | US | 2021-04-30 | 576397.0 | 32346971 | 0.0 | 2021Apr | 2021 | 4 | 0.00 |
| 88884 | US | 2021-05-29 | 594306.0 | 33251939 | 0.0 | 2021May | 2021 | 5 | 0.00 |

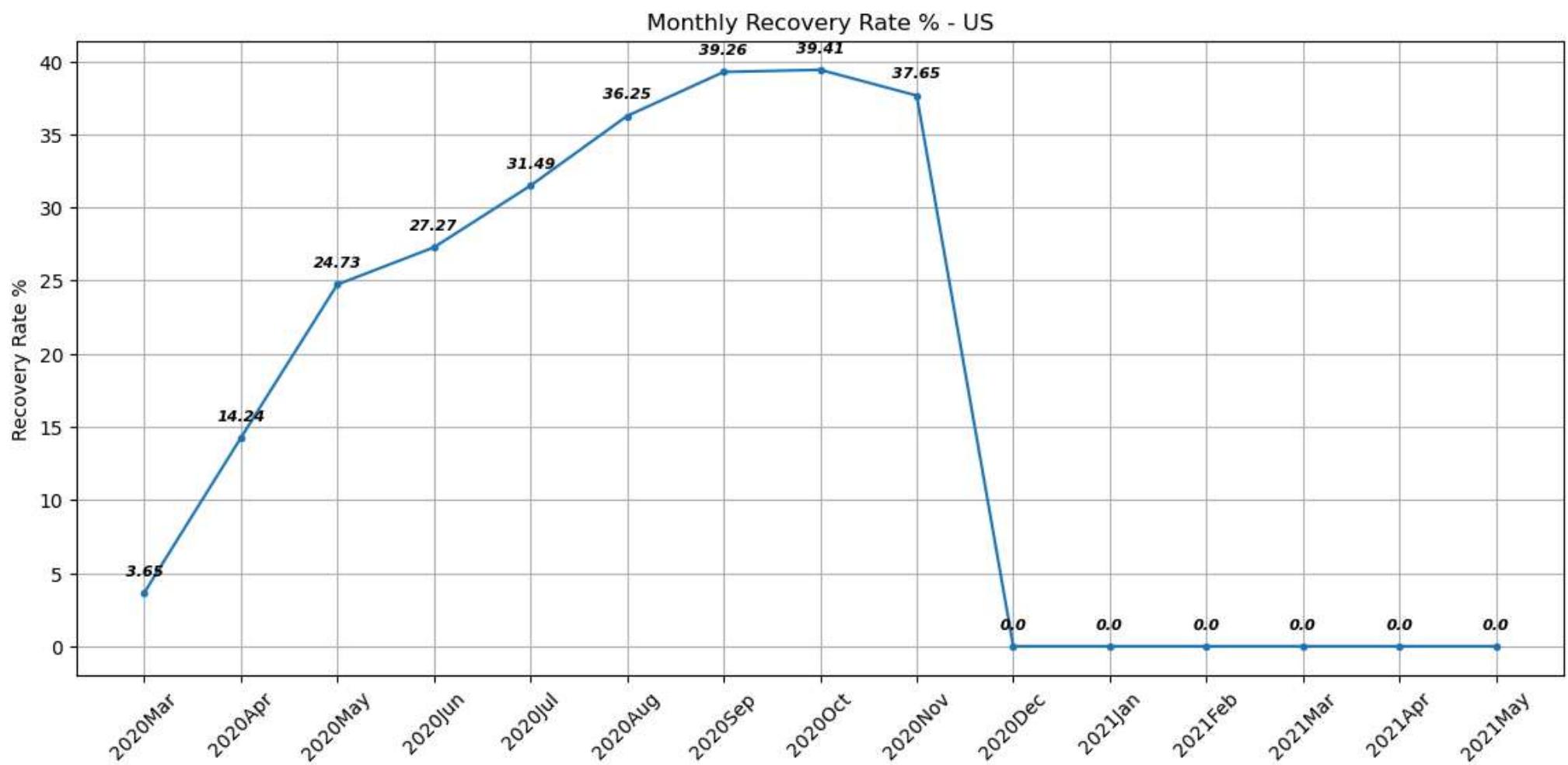
In [73]:

```

plt.figure(figsize=(14,6))
plt.plot(usra['Year-month'], usra['MRR%'], marker='.')
for x, y in zip(usra['Year-month'], usra['MRR%']):
    plt.text(x, y+1, str(y), fontsize=8, ha='center', va='bottom', fontstyle='oblique', fontweight='bold')

plt.title("Monthly Recovery Rate % - US")
plt.ylabel("Recovery Rate %")
plt.xticks(rotation=45)
plt.grid()
plt.show()

```



In []: