Introduction

Covid-19 22 January 2020 to 11 April 2020 data analysis

Data Information

- Number of rows 25353
- Number of columns 6
  - column_Name(Id) Data_Type(int64) Records(25353)
  - column_Name(Province_State) Data_Type(object) Records(10773)
  - column_Name(Country_Region) Data_Type(object) Records(25353)
  - column_Name(Date) Data_Type(datetime64[ns]) Records(25353)
  - column_Name(ConfirmedCases) Data_Type(float64) Records(25353)
  - column_Name(Fatalities) Data_Type(float64) Records(25353)

Necessery Liberary Import

In [1]:
```
import pandas as pd # Load data
import numpy as np # Scientific Computing
import seaborn as sns # Data Visualization
import matplotlib.pyplot as plt # Data Visualization
import warnings # Ignore Warnings
warnings.filterwarnings("ignore")
sns.set() # Set Graphs Background
```

Load Data

In [2]:
```
data = pd.read_csv('data/train (3).csv')
data.head()
```

Out[2]:

	Id	Province_State	Country_Region	Date	ConfirmedCases	Fatalities
0	1	NaN	Afghanistan	2020-01-22	0.0	0.0
1	2	NaN	Afghanistan	2020-01-23	0.0	0.0
2	3	NaN	Afghanistan	2020-01-24	0.0	0.0
3	4	NaN	Afghanistan	2020-01-25	0.0	0.0
4	5	NaN	Afghanistan	2020-01-26	0.0	0.0

Information Of Data

In [3]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25353 entries, 0 to 25352
Data columns (total 6 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Id              25353 non-null  int64  
 1   Province_State  10773 non-null  object 
 2   Country_Region  25353 non-null  object 
 3   Date            25353 non-null  object 
 4   ConfirmedCases  25353 non-null  float64
 5   Fatalities      25353 non-null  float64
dtypes: float64(2), int64(1), object(3)
memory usage: 1.2+ MB
```

Unique Country

```
In [4]: data['Country_Region'].unique()
```

```
Out[4]: array(['Afghanistan', 'Albania', 'Algeria', 'Andorra', 'Angola',
               'Antigua and Barbuda', 'Argentina', 'Armenia', 'Australia',
               'Austria', 'Azerbaijan', 'Bahamas', 'Bahrain', 'Bangladesh',
               'Barbados', 'Belarus', 'Belgium', 'Belize', 'Benin', 'Bhutan',
               'Bolivia', 'Bosnia and Herzegovina', 'Botswana', 'Brazil',
               'Brunei', 'Bulgaria', 'Burkina Faso', 'Burma', 'Burundi',
               'Cabo Verde', 'Cambodia', 'Cameroon', 'Canada',
               'Central African Republic', 'Chad', 'Chile', 'China', 'Colombia',
               'Congo (Brazzaville)', 'Congo (Kinshasa)', 'Costa Rica',
               "Cote d'Ivoire", 'Croatia', 'Cuba', 'Cyprus', 'Czechia', 'Denmark',
               'Diamond Princess', 'Djibouti', 'Dominica', 'Dominican Republic',
               'Ecuador', 'Egypt', 'El Salvador', 'Equatorial Guinea', 'Eritrea',
               'Estonia', 'Eswatini', 'Ethiopia', 'Fiji', 'Finland', 'France',
               'Gabon', 'Gambia', 'Georgia', 'Germany', 'Ghana', 'Greece',
               'Grenada', 'Guatemala', 'Guinea', 'Guinea-Bissau', 'Guyana',
               'Haiti', 'Holy See', 'Honduras', 'Hungary', 'Iceland', 'India',
               'Indonesia', 'Iran', 'Iraq', 'Ireland', 'Israel', 'Italy',
               'Jamaica', 'Japan', 'Jordan', 'Kazakhstan', 'Kenya',
               'Korea, South', 'Kosovo', 'Kuwait', 'Kyrgyzstan', 'Laos', 'Latvia',
               'Lebanon', 'Liberia', 'Libya', 'Liechtenstein', 'Lithuania',
               'Luxembourg', 'MS Zaandam', 'Madagascar', 'Malawi', 'Malaysia',
               'Maldives', 'Mali', 'Malta', 'Mauritania', 'Mauritius', 'Mexico',
               'Moldova', 'Monaco', 'Mongolia', 'Montenegro', 'Morocco',
               'Mozambique', 'Namibia', 'Nepal', 'Netherlands', 'New Zealand',
               'Nicaragua', 'Niger', 'Nigeria', 'North Macedonia', 'Norway',
               'Oman', 'Pakistan', 'Panama', 'Papua New Guinea', 'Paraguay',
               'Peru', 'Philippines', 'Poland', 'Portugal', 'Qatar', 'Romania',
               'Russia', 'Rwanda', 'Saint Kitts and Nevis', 'Saint Lucia',
               'Saint Vincent and the Grenadines', 'San Marino',
               'Sao Tome and Principe', 'Saudi Arabia', 'Senegal', 'Serbia',
               'Seychelles', 'Sierra Leone', 'Singapore', 'Slovakia', 'Slovenia',
               'Somalia', 'South Africa', 'South Sudan', 'Spain', 'Sri Lanka',
               'Sudan', 'Suriname', 'Sweden', 'Switzerland', 'Syria', 'Taiwan*',
               'Tanzania', 'Thailand', 'Timor-Leste', 'Togo',
               'Trinidad and Tobago', 'Tunisia', 'Turkey', 'US', 'Uganda',
               'Ukraine', 'United Arab Emirates', 'United Kingdom', 'Uruguay',
               'Uzbekistan', 'Venezuela', 'Vietnam', 'West Bank and Gaza',
               'Western Sahara', 'Zambia', 'Zimbabwe'], dtype=object)
```

Unique country Count

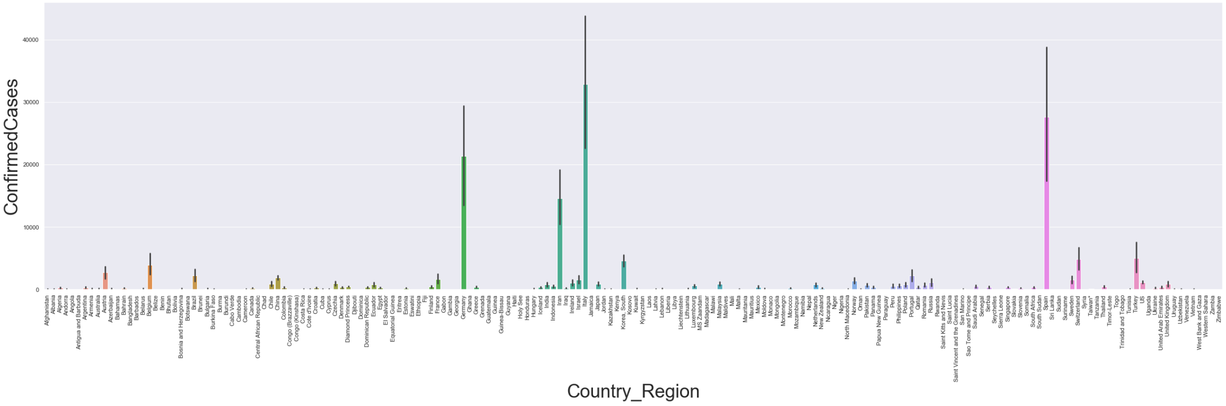
```
In [5]: data['Country_Region'].nunique()
```

```
Out[5]: 184
```

There have 184 Unique Country present

Barplot For All Data

In [6]:
```
plt.figure(figsize=(40,10))  # For Figure Resize
sns.barplot(x='Country_Region',y='ConfirmedCases', data=data)
plt.xlabel('Country_Region',fontsize = 35)
plt.ylabel('ConfirmedCases',fontsize = 35)
plt.xticks(rotation=90)  #For X label Value_Name rotation
plt.show() # Show The Plotfontsize = 25
```

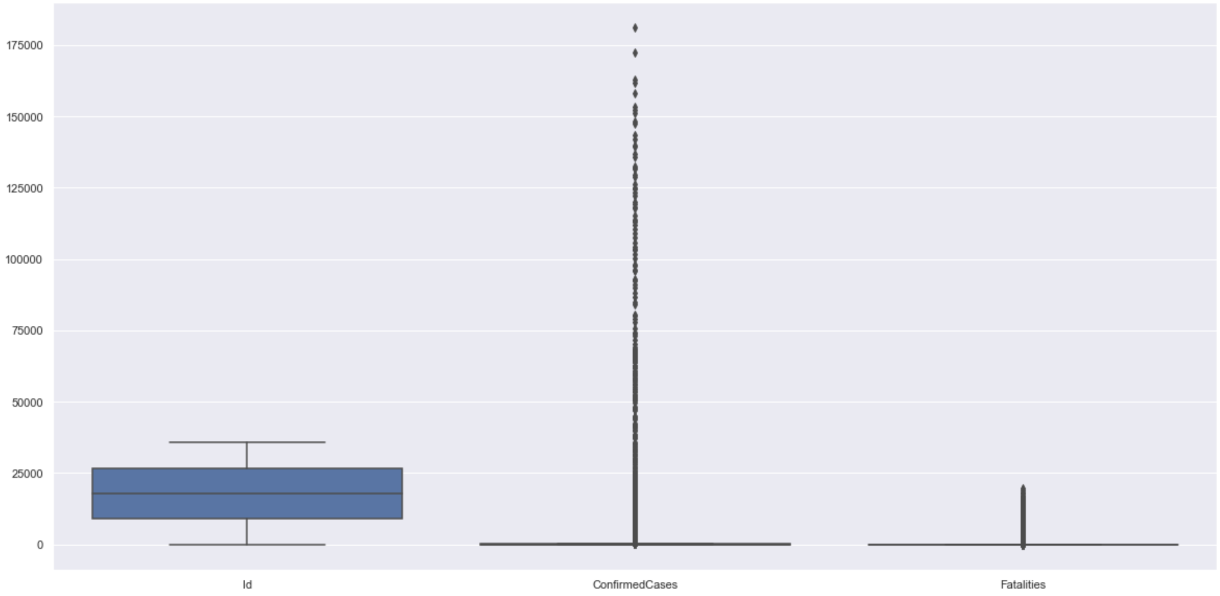


Insights of Barplot For All Data

- This barplot represents Country_Region vs ConfirmedCases
- Italy First number of ConfirmedCases
- Spain Second number of ConfirmedCases
- Germany Third number of ConfirmedCases
- Then Iran,Korea, switzerland, Turkey also

Box Plot

In [7]:
```
plt.figure(figsize=(20,10))
sns.boxplot(data=data)
plt.show()
```

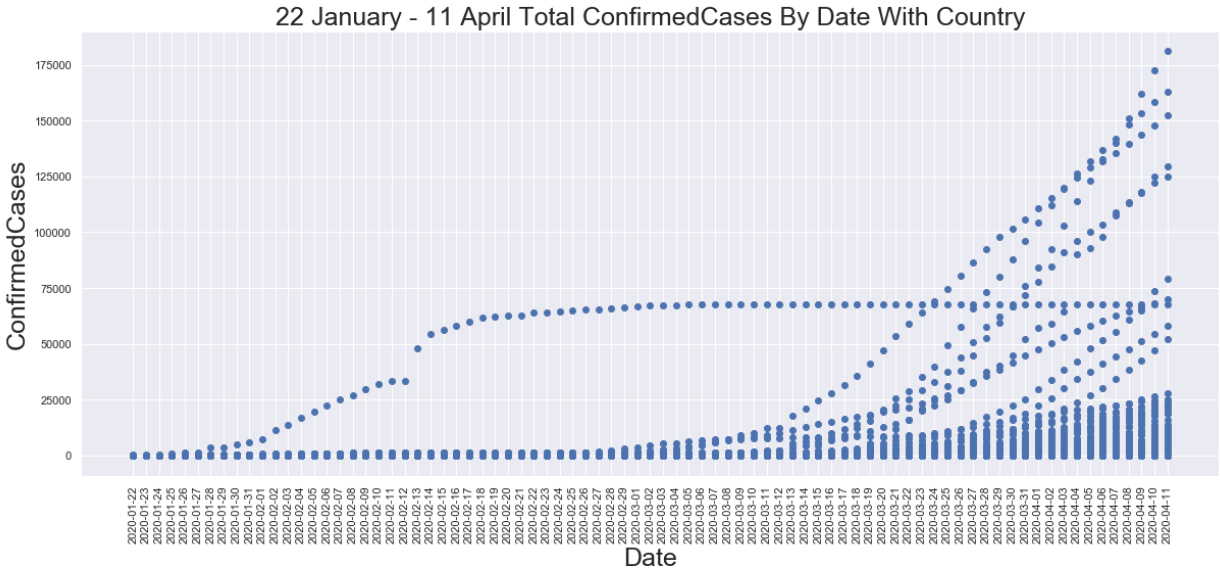


- The Maximum ConfirmedCases & Fatalities Data Out of 75% quantile.

Scatter Plot Date VS ConfirmedCases with Country

In [8]:

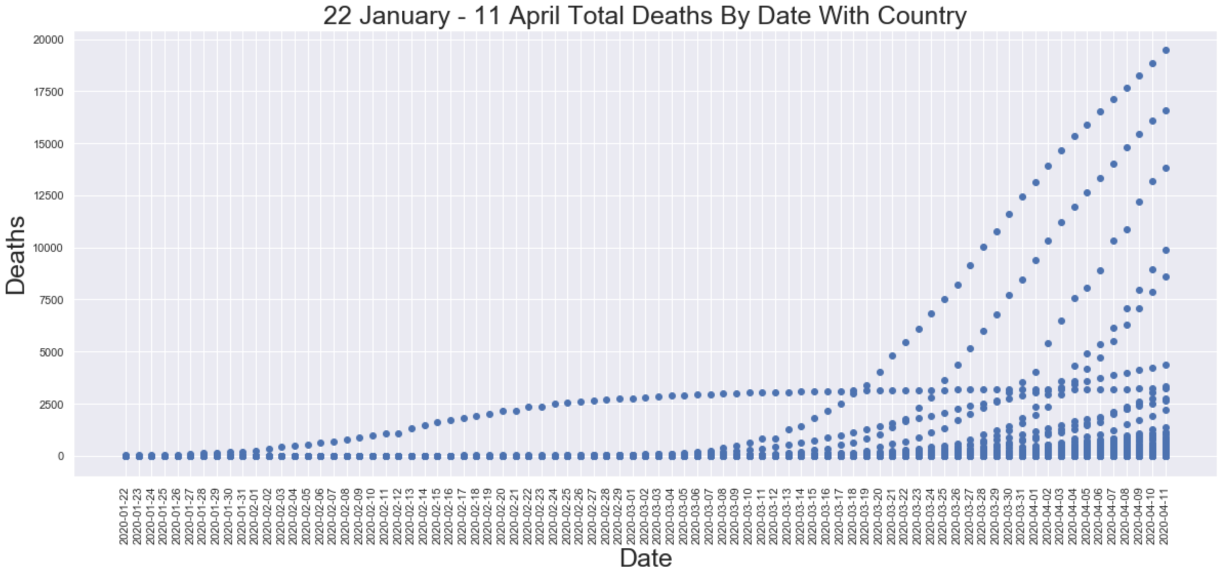
```
plt.figure(figsize=(20,8))
plt.scatter(data['Date'],data['ConfirmedCases'])
plt.title('22 January - 11 April Total ConfirmedCases By Date With Country', font
plt.xlabel('Date',fontsize=25)
plt.ylabel('ConfirmedCases',fontsize=25)
plt.xticks(rotation=90)
plt.show()
```



- Day By Day Increase The ConfirmedCases

Scatter Plot Date VS Deaths With Country

```
In [9]:  plt.figure(figsize=(20,8))
         plt.scatter(data['Date'],data['Fatalities'])
         plt.title('22 January - 11 April Total Deaths By Date With Country', fontsize=25
         plt.xlabel('Date', fontsize=25)
         plt.ylabel('Deaths',fontsize=25)
         plt.xticks(rotation=90)
         plt.show()
```



- Day By Day Increase The Deaths

Country Records Count

```
In [10]: data['Country_Region'].value_counts()
```

```
Out[10]: US                         4374
         China                      2673
         Canada                      972
         United Kingdom              891
         France                      891
                                    ...
         Angola                       81
         Zimbabwe                     81
         Kyrgyzstan                   81
         Saint Kitts and Nevis        81
         Chile                        81
         Name: Country_Region, Length: 184, dtype: int64
```

Maximum & Minimum Date

In [11]:

```
print(data['Date'].min())
print(data['Date'].max())
```

```
2020-01-22
2020-04-11
```

- Start date 22 january 2020
- End date 11 april 2020

Group Country & Select Largest 15 Country by ConfirmedCases

In [12]:

```
data_15 = data.groupby('Country_Region', as_index=False)['ConfirmedCases','Fatali
data_15 = data_15.nlargest(15,'ConfirmedCases')
data_15
```

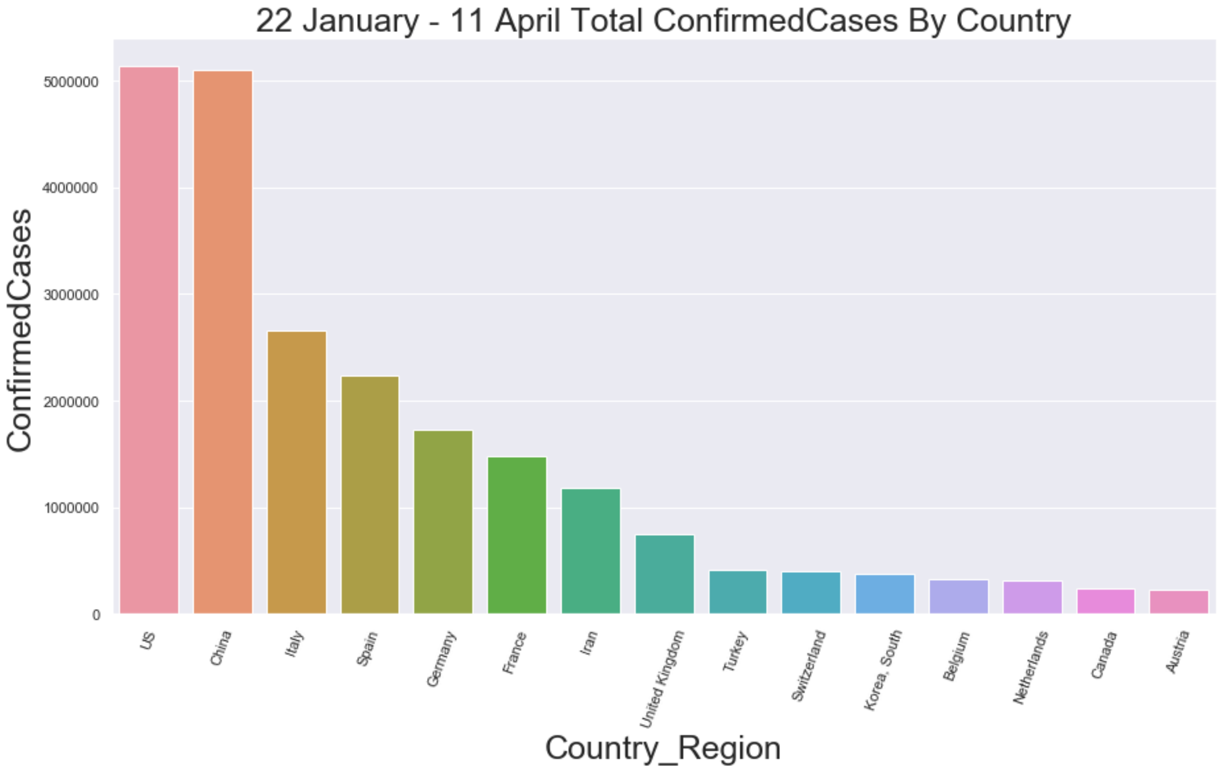
Out[12]:

	Country_Region	ConfirmedCases	Fatalities
171	US	5135445.0	147545.0
36	China	5096274.0	182450.0
84	Italy	2661341.0	297444.0
156	Spain	2237252.0	200412.0
65	Germany	1728391.0	24491.0
61	France	1476739.0	122427.0
80	Iran	1184893.0	75219.0
175	United Kingdom	749434.0	72805.0
170	Turkey	408544.0	8310.0
161	Switzerland	393842.0	11029.0
90	Korea, South	369752.0	4758.0
16	Belgium	319318.0	23652.0
120	Netherlands	310043.0	27083.0
32	Canada	233554.0	4282.0
9	Austria	220500.0	3175.0

Bar Plot ConfirmedCases For New Data

In [13]:

```
plt.figure(figsize=(15,8))    # For Figure Resize
sns.barplot(x='Country_Region',y='ConfirmedCases', data=data_15) # For Bar Plot
plt.title("22 January - 11 April Total ConfirmedCases By Country",fontsize = 25)
plt.xlabel('Country_Region',fontsize = 25)   # For X-axis Name
plt.ylabel('ConfirmedCases',fontsize = 25)   # For Y-axis Name
plt.xticks(rotation=70)   # For X label Value_Name rotation
plt.show()    # Show The Plot
```

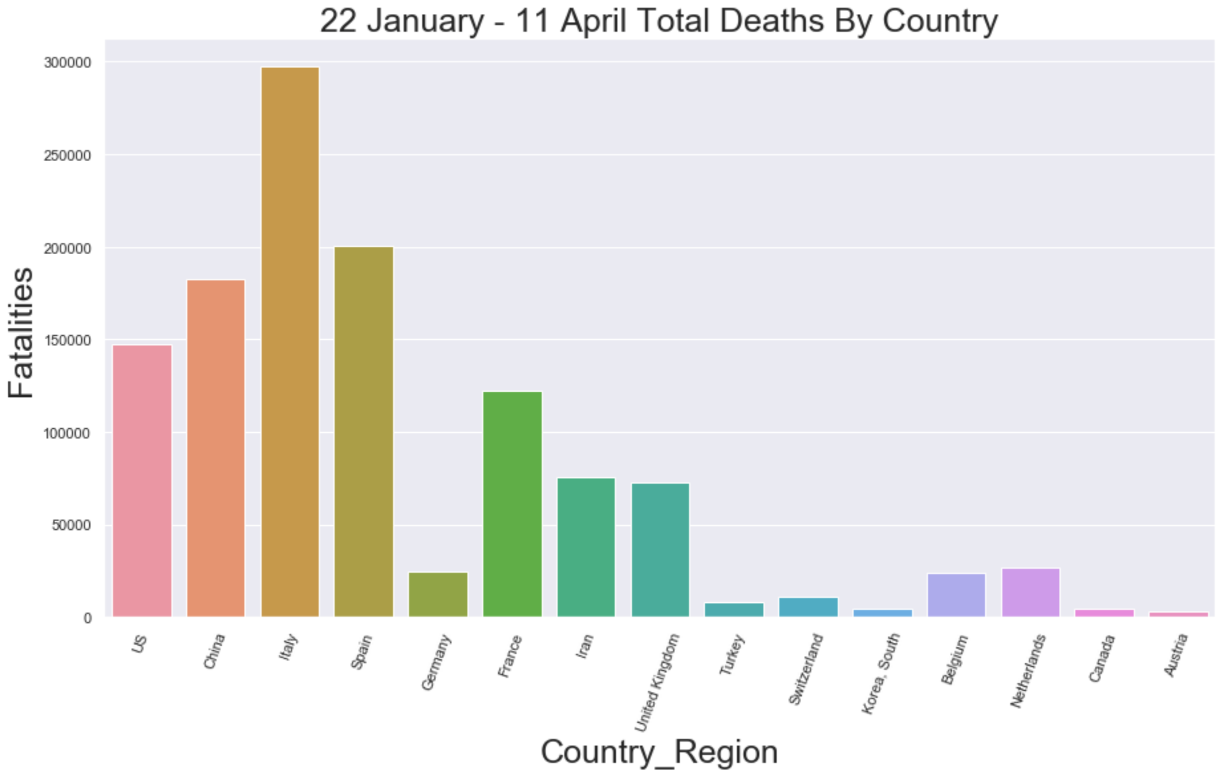


- The Highest Number of ConfirmedCases in US
- The lowest Number of ConfirmedCases in Austria

Bar Plot For Fatalities For New Data

In [14]:

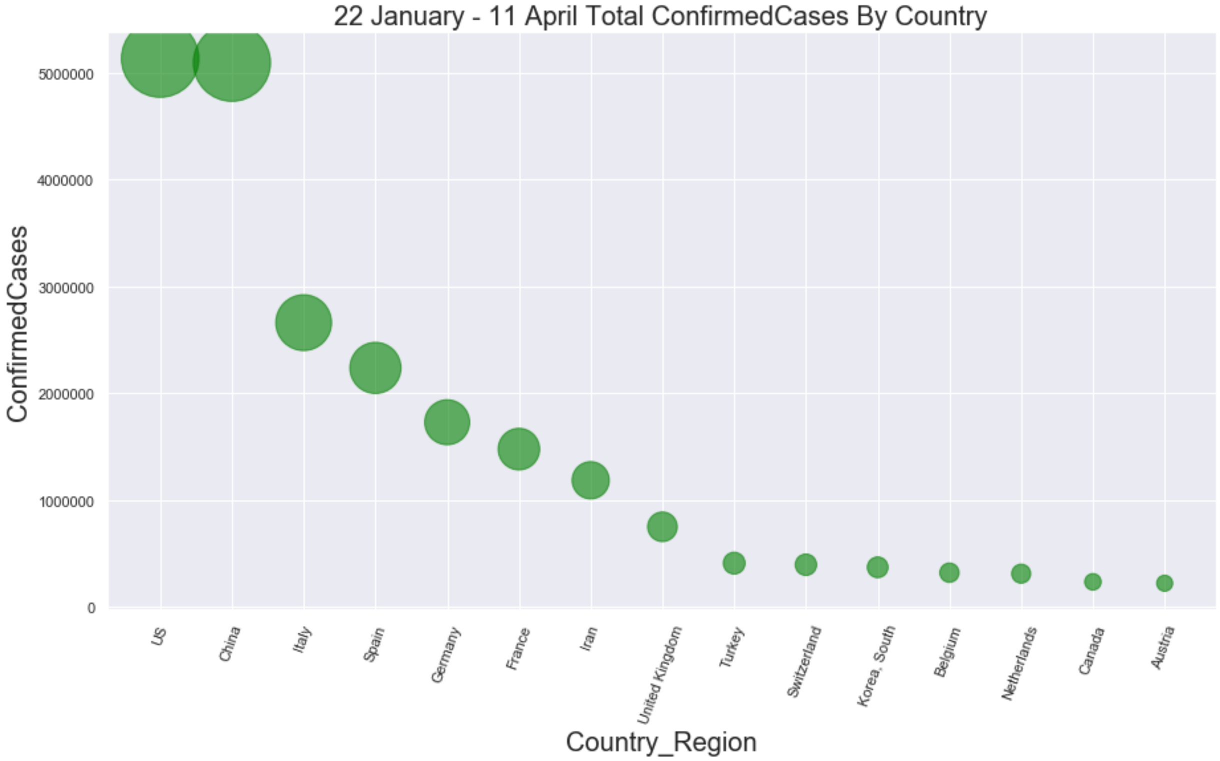
```
plt.figure(figsize=(15,8))  # For Figure Resize
sns.barplot(x='Country_Region',y='Fatalities', data=data_15)  # Show The Plot
plt.title("22 January - 11 April Total Deaths By Country",fontsize = 25)  # For
plt.xlabel('Country_Region',fontsize = 25)  # For X-axis Name
plt.ylabel('Fatalities',fontsize = 25)   # For Y-axis Name
plt.xticks(rotation=70)   # For X label Value_Name rotation
plt.show()   # Show The Plot
```



- The Highest Number of Deaths in Italy
- The lowest Number of Deaths in Austria

Scatter Plot For Country VS ConfirmedCases For New Data

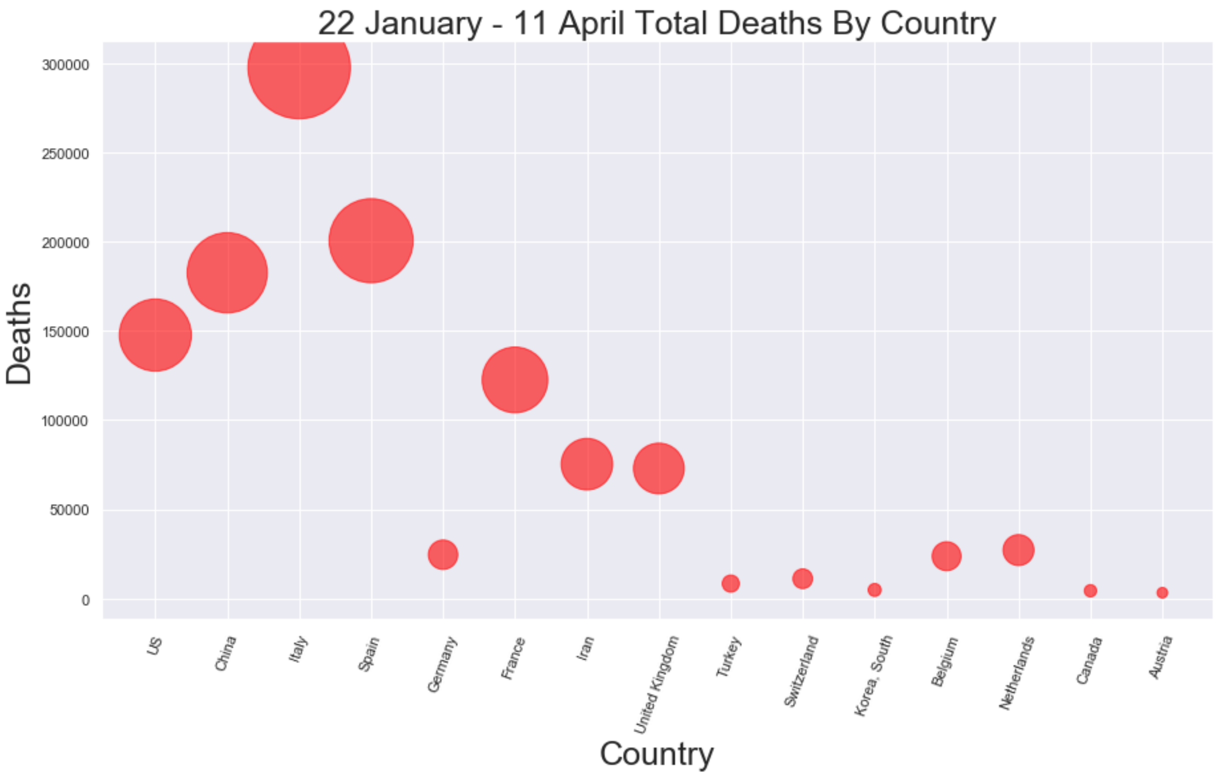
In [15]:
```
bubbol = np.array(data_15['ConfirmedCases']/1500) # For Bubbol Size
plt.figure(figsize=(15,8))
plt.scatter(data_15['Country_Region'],data_15['ConfirmedCases'],c='green',s=bubbo
plt.title("22 January - 11 April Total ConfirmedCases By Country",fontsize = 20)
plt.xlabel('Country_Region',fontsize = 20)
plt.ylabel('ConfirmedCases',fontsize = 20)
plt.xticks(rotation=70)
plt.show()
```



- The Highest Number of ConfirmedCases in US.
- The lowest Number of ConfirmedCases in Austria.

Scatter Plot For Country VS Deaths For New Data

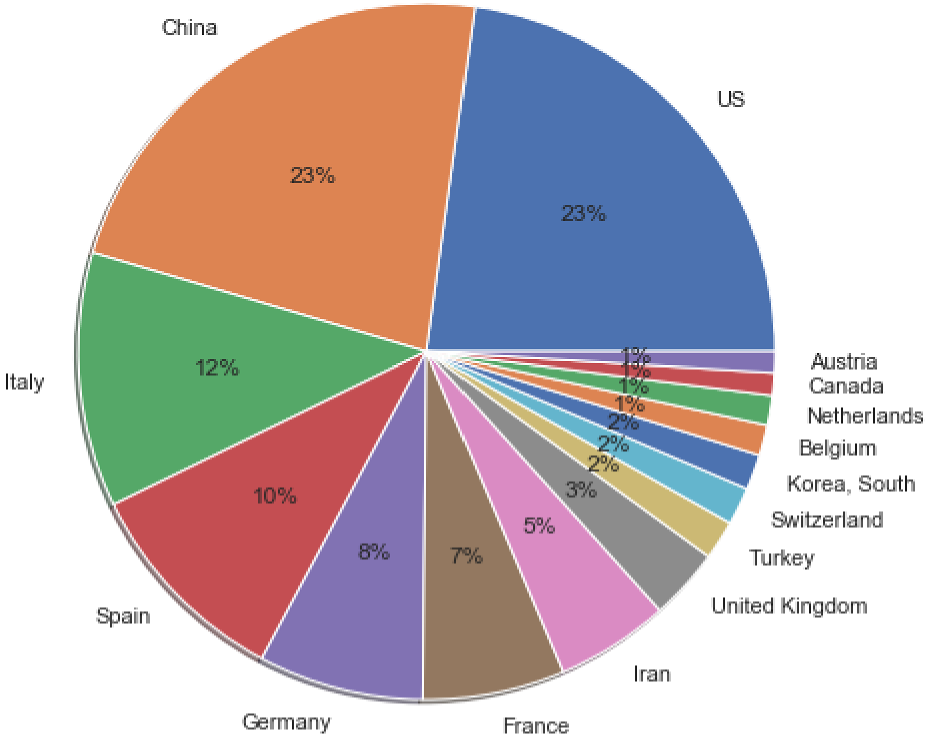
In [16]:
```
bubbol = np.array(data_15['Fatalities']/50) # For Bubbol Size
plt.figure(figsize=(15,8))
plt.scatter(data_15['Country_Region'],data_15['Fatalities'],c='red',s=bubbol, alp
plt.title("22 January - 11 April Total Deaths By Country",fontsize = 25)
plt.xlabel('Country',fontsize = 25)
plt.ylabel('Deaths',fontsize = 25)
plt.xticks(rotation=70)
plt.show()
```



- The Highest Number of Deaths in Italy.
- The lowest Number of Deaths in Austria.

Pie Chart For ConfirmedCases For New Data

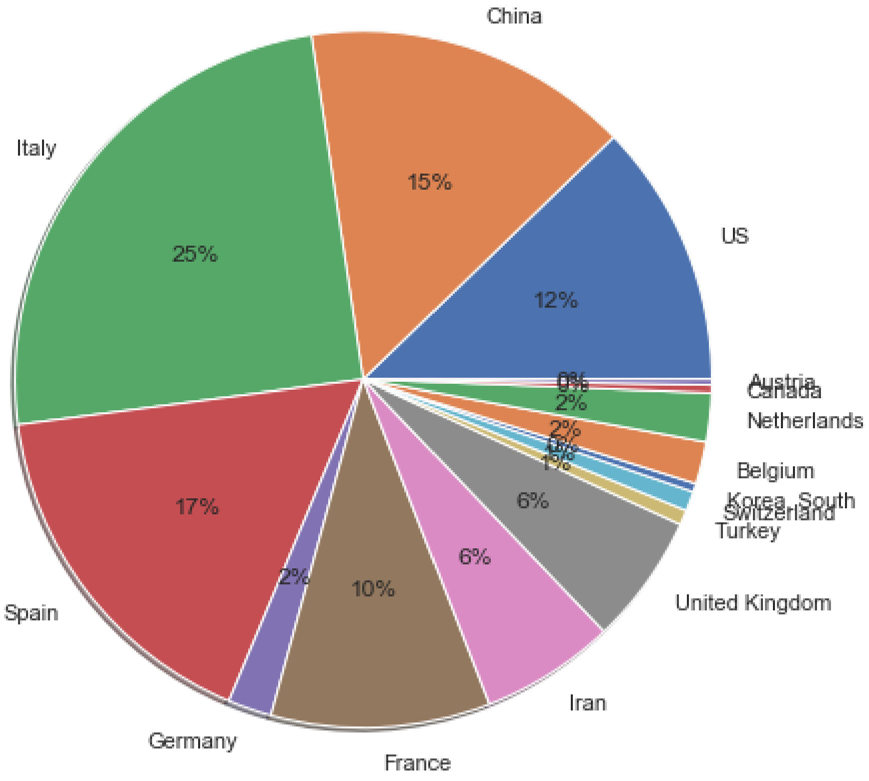
```
In [17]: plt.axis('equal')
         plt.pie(data_15['ConfirmedCases'],labels=data_15['Country_Region'], radius=2, aut
                 shadow=True)
         plt.show()
```



- The Highest Number of ConfirmedCases 23% in US & China .
- The lowest Number of ConfirmedCases 1% in Belgium, Netherlands, Canada & Austria.

Pie Chart For Deaths For New Data

In [18]:
```
plt.axis('equal')
plt.pie(data_15['Fatalities'],labels=data_15['Country_Region'], radius=2, autopct
        shadow=True)
plt.show()
```



- The Highest Number of Deaths 25% in Italy.
- The lowest Number of Deaths 0.2% in Austria.

The Date Convert Into YYYY-MM-DD Format

In [19]:
```
data['Date'] = pd.to_datetime(data['Date'])
```

Create New Dataset For Individual Date

In [20]:
```
data_81 = data.groupby('Date', as_index=False)['ConfirmedCases','Fatalities'].su
data_81 = data_81.nlargest(81,'ConfirmedCases')
data_81
```

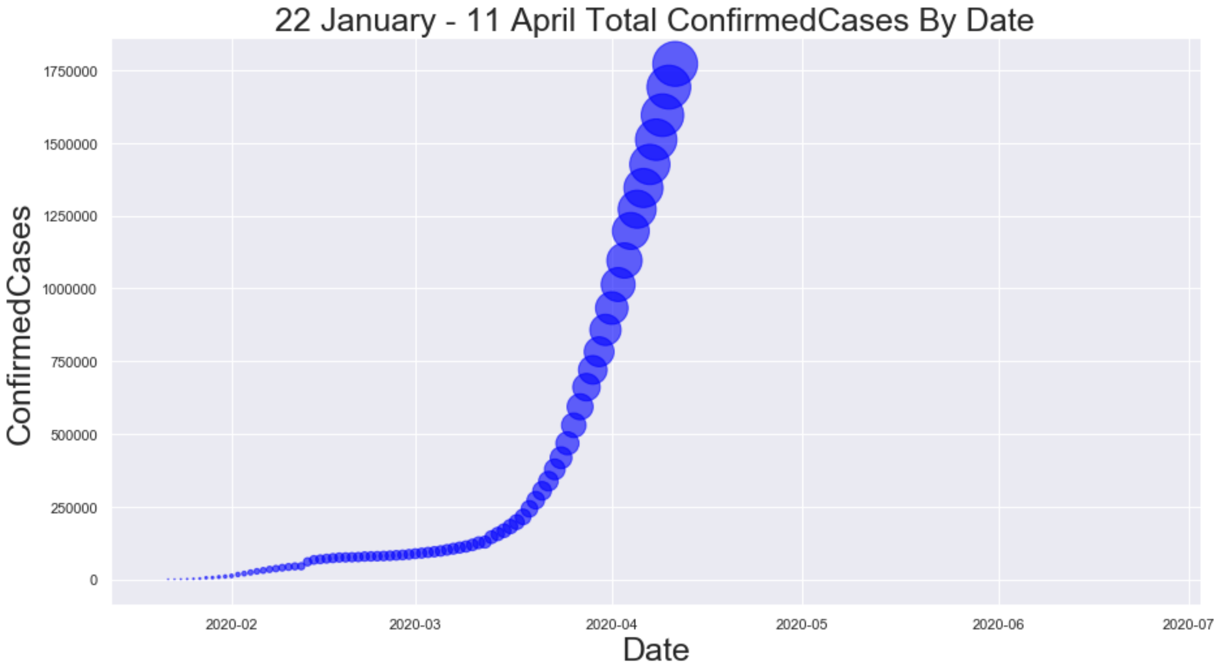
Out[20]:

	Date	ConfirmedCases	Fatalities
80	2020-04-11	1771337.0	108497.0
79	2020-04-10	1691542.0	102519.0
78	2020-04-09	1595174.0	95449.0
77	2020-04-08	1510928.0	88332.0
76	2020-04-07	1425923.0	81859.0
...	...	...	...
4	2020-01-26	2113.0	56.0
3	2020-01-25	1432.0	42.0
2	2020-01-24	939.0	26.0
1	2020-01-23	653.0	18.0
0	2020-01-22	554.0	17.0

81 rows × 3 columns

Scatter Plot Date VS ConfirmedCases For New Dataset

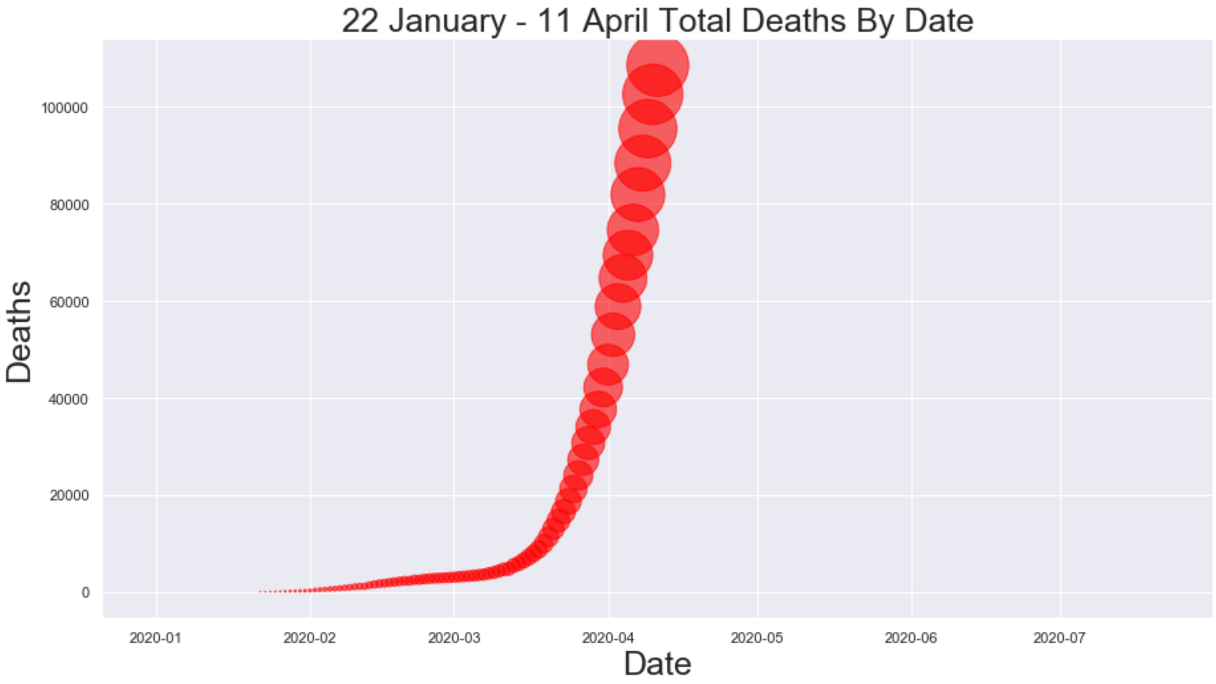
```
In [21]: bubbol = np.array(data_81['ConfirmedCases']/1500) # For Bubbol Size
         plt.figure(figsize=(15,8))
         plt.scatter(data_81['Date'],data_81['ConfirmedCases'],c='blue',s=bubbol, alpha=0.
         plt.title("22 January - 11 April Total ConfirmedCases By Date",fontsize=25)
         plt.xlabel('Date',fontsize=25)
         plt.ylabel('ConfirmedCases',fontsize=25)
         plt.show()
```



- Day By Day Increase The ConfirmedCases

Scatter Plot Date VS Deaths For New Dataset

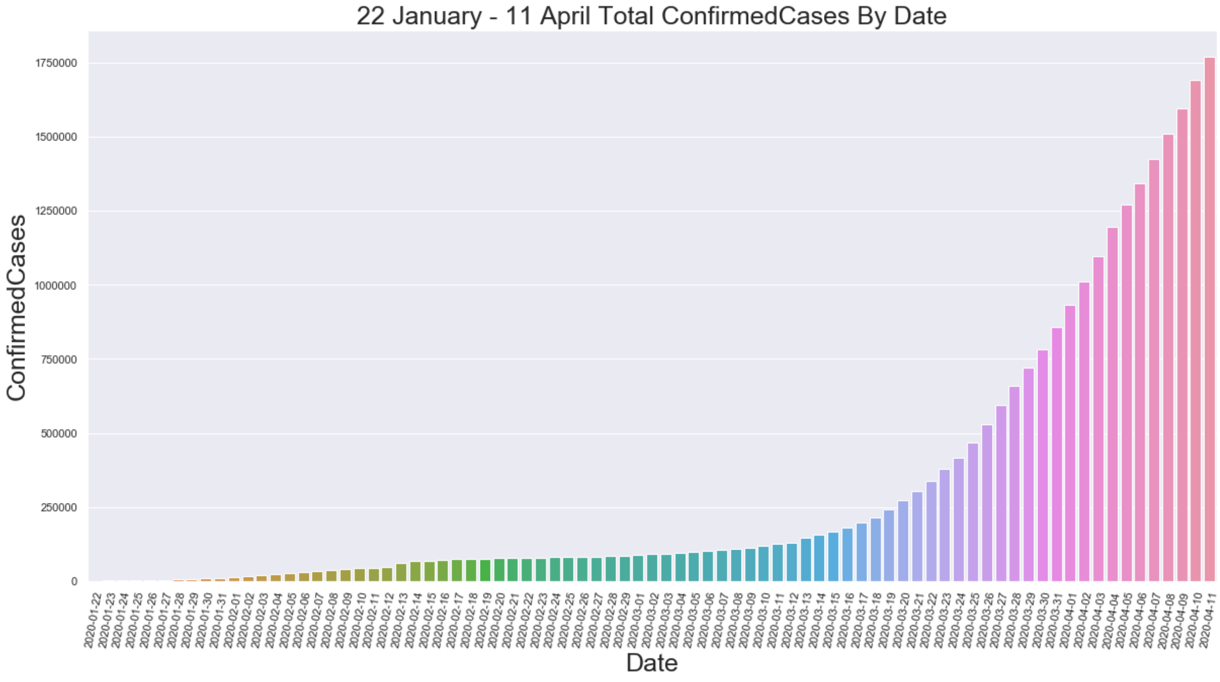
In [22]:
```
bubbol = np.array(data_81['Fatalities']/50) # For Bubbol Size
plt.figure(figsize=(15,8))
plt.scatter(data_81['Date'],data_81['Fatalities'],c='red',s=bubbol, alpha=0.6)
plt.title("22 January - 11 April Total Deaths By Date",fontsize=25)
plt.xlabel('Date',fontsize=25)
plt.ylabel('Deaths',fontsize=25)
plt.show()
```



- Day By Day Increase The Deaths

Bar Plot Date VS ConfirmedCases For New Dataset

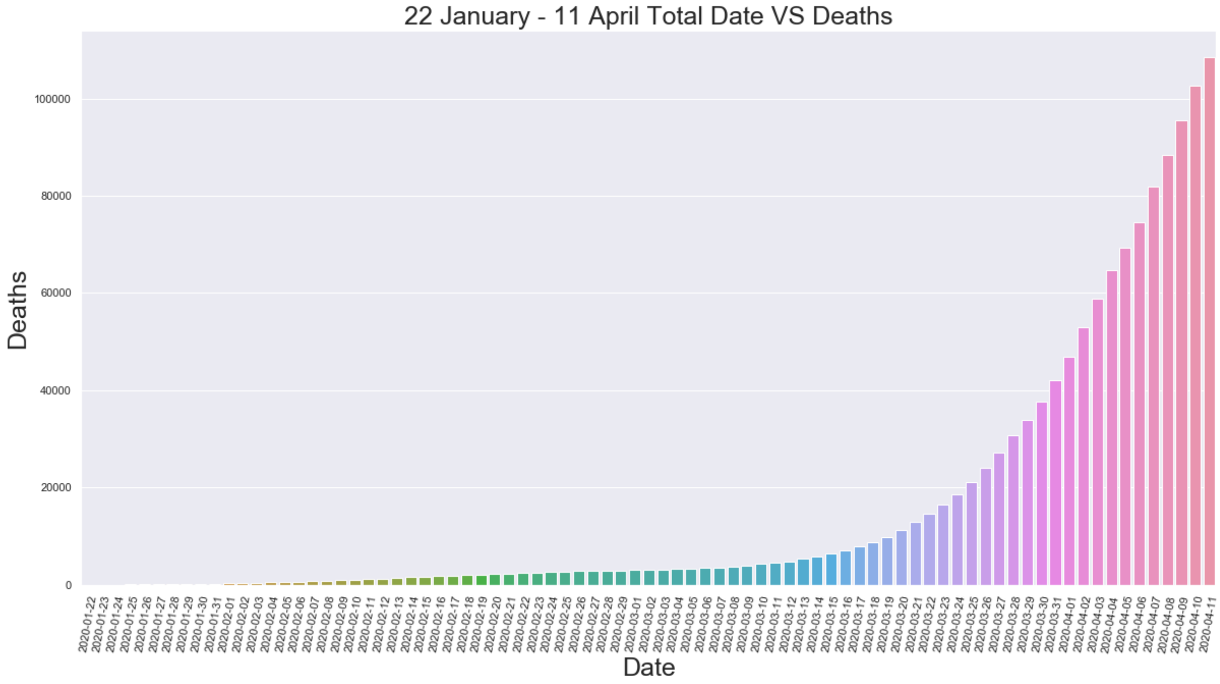
In [23]:
```
fig, ax = plt.subplots(figsize = (20,10))
fig = sns.barplot(x ="Date", y ="ConfirmedCases", data = data_81)

x_dates = data_81['Date'].dt.strftime('%Y-%m-%d').sort_values()
ax.set_xticklabels(labels=x_dates, rotation=80)
plt.title("22 January - 11 April Total ConfirmedCases By Date",fontsize=25)
plt.xlabel('Date',fontsize=25)
plt.ylabel('ConfirmedCases',fontsize=25)
plt.show()
```



- Day By Day Increase The ConfirmedCases.

Bar Plot Date VS Deaths For World

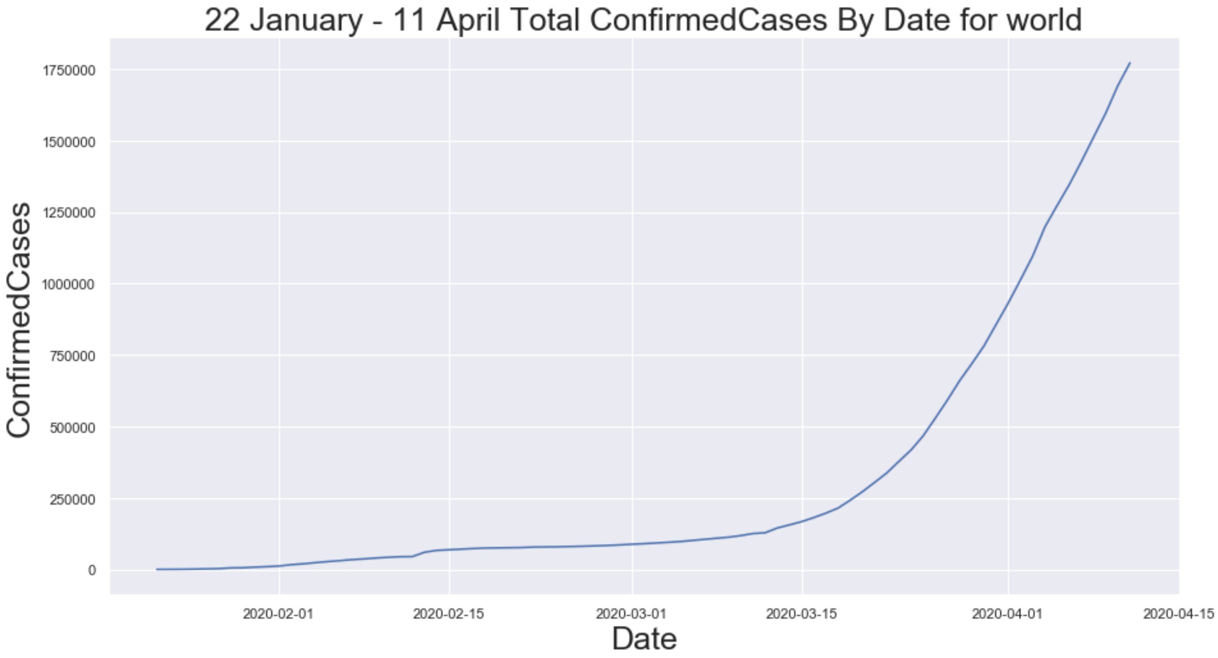
In [24]:
```
fig, ax = plt.subplots(figsize = (20,10))
fig = sns.barplot(x ="Date", y ="Fatalities", data = data_81)

x_dates = data_81['Date'].dt.strftime('%Y-%m-%d').sort_values()
ax.set_xticklabels(labels=x_dates, rotation=80)
plt.title("22 January - 11 April Total Date VS Deaths",fontsize=25)
plt.xlabel('Date',fontsize=25)
plt.ylabel('Deaths',fontsize=25)
plt.show()
```



- Day By Day Increase The Deaths.

Line Plot Date VS ConfirmedCases For World

In [25]:
```
plt.figure(figsize=(15,8))
sns.lineplot(x='Date',y='ConfirmedCases', data=data_81)
plt.title("22 January - 11 April Total ConfirmedCases By Date for world",fontsize
plt.xlabel('Date',fontsize=25)
plt.ylabel('ConfirmedCases',fontsize=25)
plt.show()
```

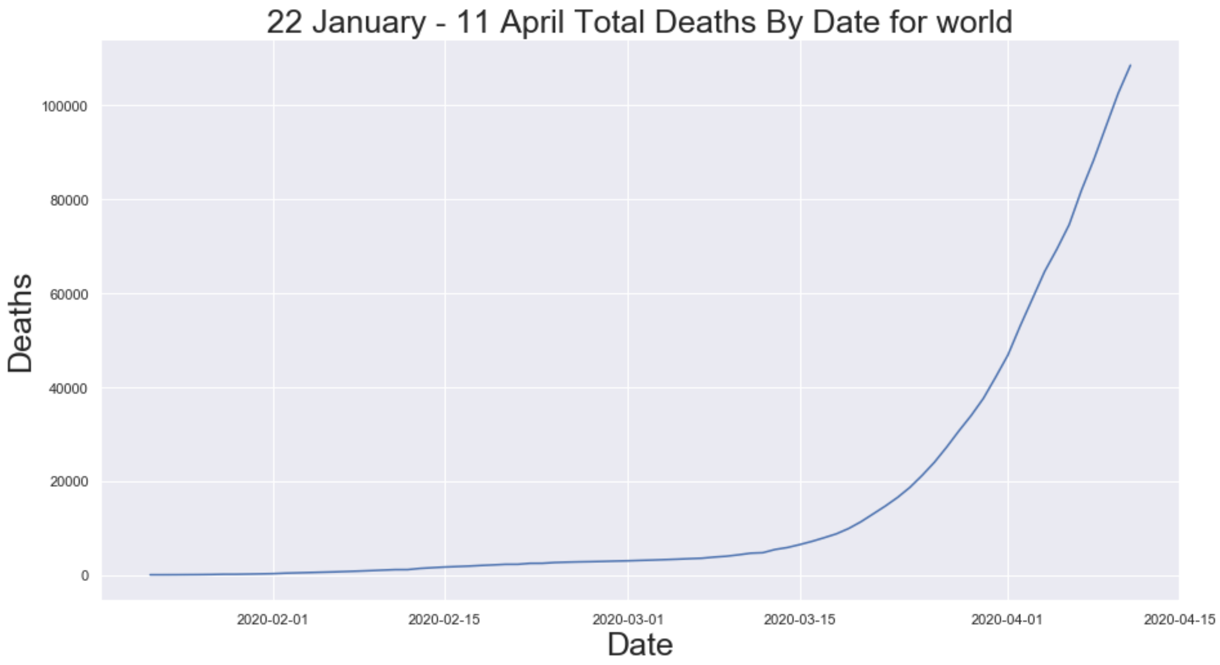


- Day By Day Increase The ConfirmedCases.

Line Plot Date VS Deaths For World

In [26]:

```
plt.figure(figsize=(15,8))
sns.lineplot(x='Date',y='Fatalities', data=data_81)
plt.title("22 January - 11 April Total Deaths By Date for world",fontsize=25)
plt.xlabel('Date',fontsize=25)
plt.ylabel('Deaths',fontsize=25)
plt.show()
```



- Day By Day Increase The Number of deaths.

Create New Dataset For Individual Date & Individual Country

In [27]:

```
data_all = data.groupby(['Date','Country_Region'], as_index=False)['ConfirmedCase
data_all
```

Out[27]:

	Date	Country_Region	ConfirmedCases	Fatalities
0	2020-01-22	Afghanistan	0.0	0.0
1	2020-01-22	Albania	0.0	0.0
2	2020-01-22	Algeria	0.0	0.0
3	2020-01-22	Andorra	0.0	0.0
4	2020-01-22	Angola	0.0	0.0
...	...	...	...	...
14899	2020-04-11	Vietnam	258.0	0.0
14900	2020-04-11	West Bank and Gaza	268.0	2.0
14901	2020-04-11	Western Sahara	4.0	0.0
14902	2020-04-11	Zambia	40.0	2.0
14903	2020-04-11	Zimbabwe	14.0	3.0

14904 rows × 4 columns

Select All United State Data From New Dataset

In [28]:

```
data_usa = data_all.query("Country_Region=='US'")
data_usa
```

Out[28]:

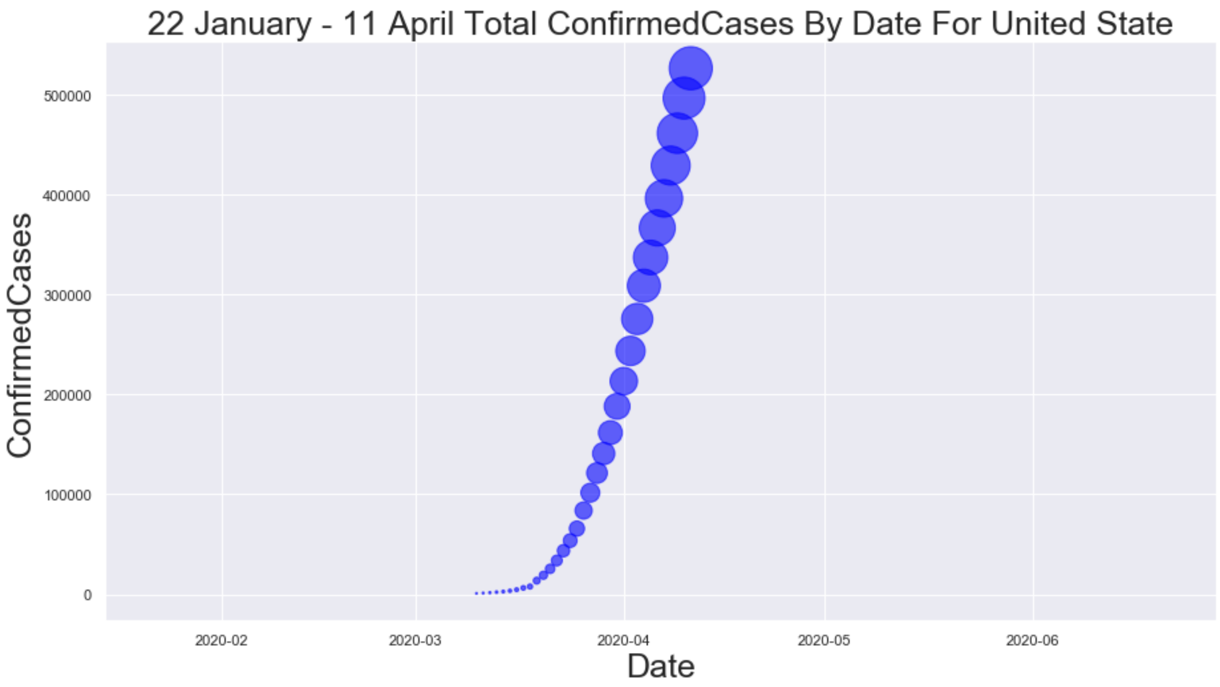
	Date	Country_Region	ConfirmedCases	Fatalities
171	2020-01-22	US	0.0	0.0
355	2020-01-23	US	0.0	0.0
539	2020-01-24	US	0.0	0.0
723	2020-01-25	US	0.0	0.0
907	2020-01-26	US	0.0	0.0
...	...	...	...	...
14155	2020-04-07	US	396063.0	12717.0
14339	2020-04-08	US	428889.0	14690.0
14523	2020-04-09	US	461274.0	16473.0
14707	2020-04-10	US	496372.0	18581.0
14891	2020-04-11	US	526233.0	20458.0

81 rows × 4 columns

Scatter Plot Date VS ConfirmedCases For United

Scatter Plot Date VS ConfirmedCases For United State

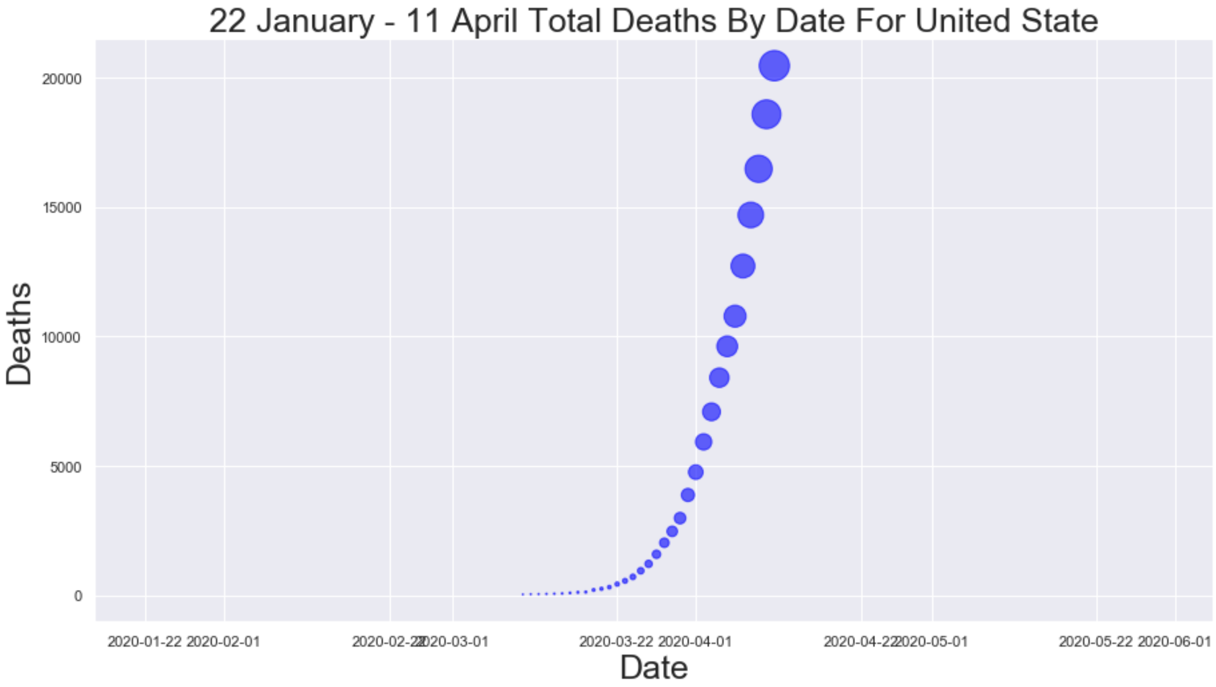
In [29]:
```
bubbol = np.array(data_usa['ConfirmedCases']/500) # For Bubbol Size
plt.figure(figsize=(15,8))
plt.scatter(data_usa['Date'],data_usa['ConfirmedCases'],c='blue',s=bubbol, alpha=
plt.title("22 January - 11 April Total ConfirmedCases By Date For United State",f
plt.xlabel('Date',fontsize=25)
plt.ylabel('ConfirmedCases',fontsize=25)
plt.show()
```



- Day By Day Increase The Number of ConfirmedCases In US.

Scatter Plot Date VS Deaths For United State

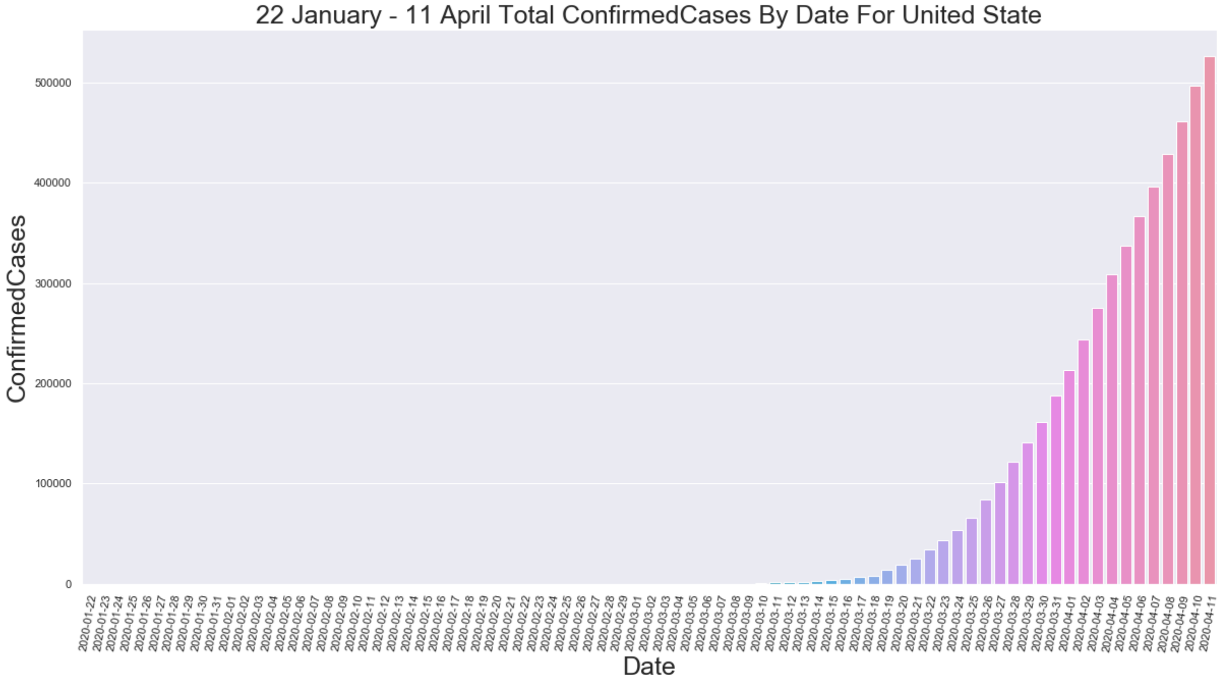
In [30]:
```
bubbol = np.array(data_usa['Fatalities']/40) # For Bubbol Size
plt.figure(figsize=(15,8))
plt.scatter(data_usa['Date'],data_usa['Fatalities'],c='blue',s=bubbol, alpha=0.6
plt.title("22 January - 11 April Total Deaths By Date For United State",fontsize=
plt.xlabel('Date',fontsize=25)
plt.ylabel('Deaths',fontsize=25)
plt.show()
```



- Day By Day Increase The Number of deaths In US.

Bar Plot Date VS ConfirmedCases For United State

```
In [31]: fig, ax = plt.subplots(figsize = (20,10))
         fig = sns.barplot(x ="Date", y ="ConfirmedCases", data = data_usa)

         x_dates = data_usa['Date'].dt.strftime('%Y-%m-%d').sort_values()
         ax.set_xticklabels(labels=x_dates, rotation=80)
         plt.title("22 January - 11 April Total ConfirmedCases By Date For United State",f
         plt.xlabel('Date',fontsize=25)
         plt.ylabel('ConfirmedCases',fontsize=25)
         plt.show()
```

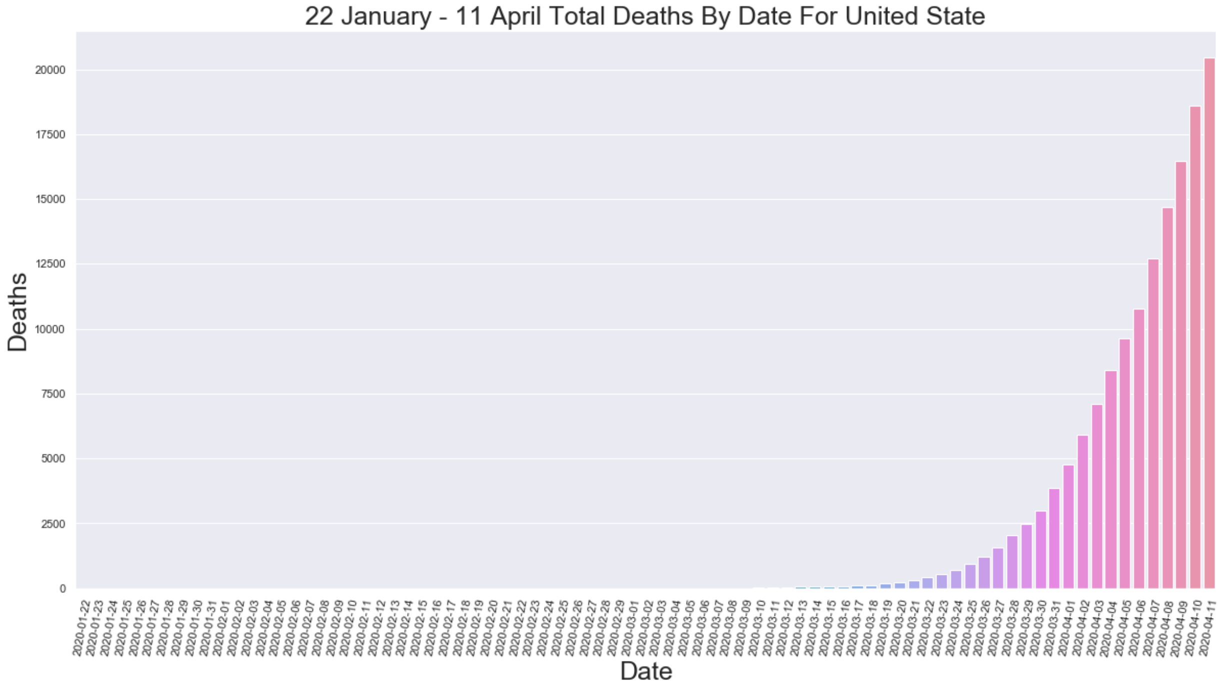


- Day By Day Increase The Number of ConfirmedCases In US.

Bar Plot Date VS Deaths For United State

In [32]:

```
fig, ax = plt.subplots(figsize = (20,10))
fig = sns.barplot(x ="Date", y ="Fatalities", data = data_usa)

x_dates = data_usa['Date'].dt.strftime('%Y-%m-%d').sort_values()
ax.set_xticklabels(labels=x_dates, rotation=80)
plt.title("22 January - 11 April Total Deaths By Date For United State",fontsize
plt.xlabel('Date',fontsize = 25)
plt.ylabel('Deaths',fontsize = 25)
plt.show()
```



- Day By Day Increase The Number of Deaths In US.