

# Exploratory\_Data\_Analysis\_Terrorism

In this task, we will be performing exploratory data analysis on the dataset "GlobalTerrorism" and try to find out the hot zone of terrorism. Also, we will derive the security issues and various insights.

```
In [1]: # Importing all the libraries needed in this notebook
import math
import warnings
import numpy as np
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import seaborn as sns
import plotly.offline as py
import plotly.graph_objs as go
import matplotlib.pyplot as plt
warnings.filterwarnings('ignore')
```

## Loading and Reading The Data Set

```
In [2]: global_terror = pd.read_csv('globalterrorismdb_0718dist.csv',encoding='ISO-8859-1')
```

```
In [3]: global_terror.head() # first five values of the dataset
```

```
Out[3]:
```

approxdate	extended	resolution	country	country_txt	region	...	addnotes	scite1	scite2	scite3	dbsource	INT_LOG
NaN	0	NaN	58	Dominican Republic	2	...	NaN	NaN	NaN	NaN	PGIS	0
NaN	0	NaN	130	Mexico	1	...	NaN	NaN	NaN	NaN	PGIS	0
NaN	0	NaN	160	Philippines	5	...	NaN	NaN	NaN	NaN	PGIS	-9
NaN	0	NaN	78	Greece	8	...	NaN	NaN	NaN	NaN	PGIS	-9
NaN	0	NaN	101	Japan	4	...	NaN	NaN	NaN	NaN	PGIS	-9

```
In [4]: global_terror.columns
```

```
Out[4]: Index(['eventid', 'iyear', 'imonth', 'iday', 'approxdate', 'extended',
              'resolution', 'country', 'country_txt', 'region',
              ...,
              'addnotes', 'scite1', 'scite2', 'scite3', 'dbsource', 'INT_LOG',
              'INT_IDEO', 'INT_MISC', 'INT_ANY', 'related'],
              dtype='object', length=135)
```

```
In [5]: global_terror.rename(columns={'iyear':'Year','imonth':'Month','iday':'Day','country_txt':'Country',
                                     'region_txt':'Region','attacktype1_txt':'AttackType','target1':'Target','n
                                     'nwound':'Wounded','summary':'Summary','gname':'Group','targettype1_txt':'Ta
                                     'weaptype1_txt':'Weapon_type','motive':'Motive'},inplace=True)
```

## Dropping out irrelevant columns

```
In [6]: # Important data for further processing
global_terror=global_terror[['Year','Month','Day','Country','state','Region','city','latitude','lon
                             'Wounded','Target','Summary','Group','Target_type','Weapon_type','Motive']]
```

```
In [7]: global_terror.head()
```

```
Out[7]:
```

	Year	Month	Day	Country	state	Region	city	latitude	longitude	AttackType	Killed	Wounded
0	1970	7	2	Dominican Republic	NaN	Central America & Caribbean	Santo Domingo	18.456792	-69.951164	Assassination	1.0	0.0
1	1970	0	0	Mexico	Federal	North America	Mexico city	19.371887	-99.086624	Hostage Taking (Kidnapping)	0.0	0.0
2	1970	1	0	Philippines	Tarlac	Southeast Asia	Unknown	15.478598	120.599741	Assassination	1.0	0.0
3	1970	1	0	Greece	Attica	Western Europe	Athens	37.997490	23.762728	Bombing/Explosion	NaN	0.0
4	1970	1	0	Japan	Fukouka	East Asia	Fukouka	33.580412	130.396361	Facility/Infrastructure Attack	NaN	0.0

```
In [8]: # Checking for the null values
global_terror.isnull().sum()
```

```
Out[8]: Year                0
Month                0
Day                0
Country                0
state              421
Region              0
city              434
latitude          4556
longitude          4557
AttackType         0
Killed            10313
Wounded           16311
Target             636
Summary           66129
Group              0
Target_type        0
Weapon_type        0
Motive            131130
dtype: int64
```

## Checking the dataset's information

```
In [9]: global_terror.info() # Returns the concise summary
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 181691 entries, 0 to 181690
Data columns (total 18 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Year                  181691 non-null  int64
1   Month                 181691 non-null  int64
2   Day                   181691 non-null  int64
3   Country               181691 non-null  object
4   state                 181270 non-null  object
5   Region                181691 non-null  object
6   city                  181257 non-null  object
7   latitude              177135 non-null  float64
8   longitude             177134 non-null  float64
9   AttackType            181691 non-null  object
10  Killed                171378 non-null  float64
11  Wounded               165380 non-null  float64
12  Target                181055 non-null  object
13  Summary               115562 non-null  object
14  Group                 181691 non-null  object
15  Target_type           181691 non-null  object
16  Weapon_type           181691 non-null  object
17  Motive                50561 non-null  object
dtypes: float64(4), int64(3), object(11)
memory usage: 25.0+ MB
```

# Data Visualization

## Destructive Features

```
In [10]: print("Country with the most attacks:",global_terror['Country'].value_counts().idxmax())
print("City with the most attacks:",global_terror['city'].value_counts().index[1]) #as first entry
print("Region with the most attacks:",global_terror['Region'].value_counts().idxmax())
print("Year with the most attacks:",global_terror['Year'].value_counts().idxmax())
print("Month with the most attacks:",global_terror['Month'].value_counts().idxmax())
print("Group with the most attacks:",global_terror['Group'].value_counts().index[1])
print("Most Attack Types:",global_terror['AttackType'].value_counts().idxmax())
```

Country with the most attacks: Iraq  
City with the most attacks: Baghdad  
Region with the most attacks: Middle East & North Africa  
Year with the most attacks: 2014  
Month with the most attacks: 5  
Group with the most attacks: Taliban  
Most Attack Types: Bombing/Explosion

```
In [13]: from wordcloud import WordCloud
from scipy import signal
cities = global_terror.state.dropna(False)
plt.subplots(figsize=(20,10))
wordcloud = WordCloud(background_color = 'black',
                      width = 500,
                      height = 400).generate(' '.join(cities))

plt.axis('off')
plt.imshow(wordcloud)
plt.show()
```

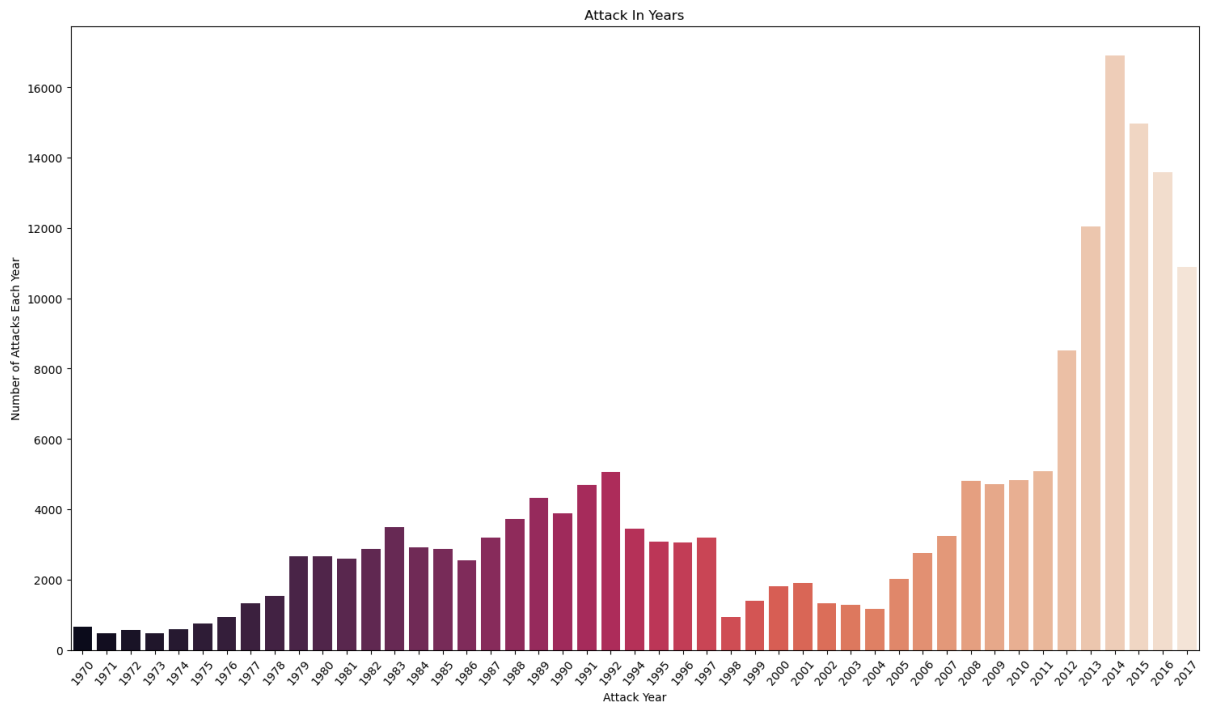


```
In [14]: global_terror['Year'].value_counts(dropna = False).sort_index()
```

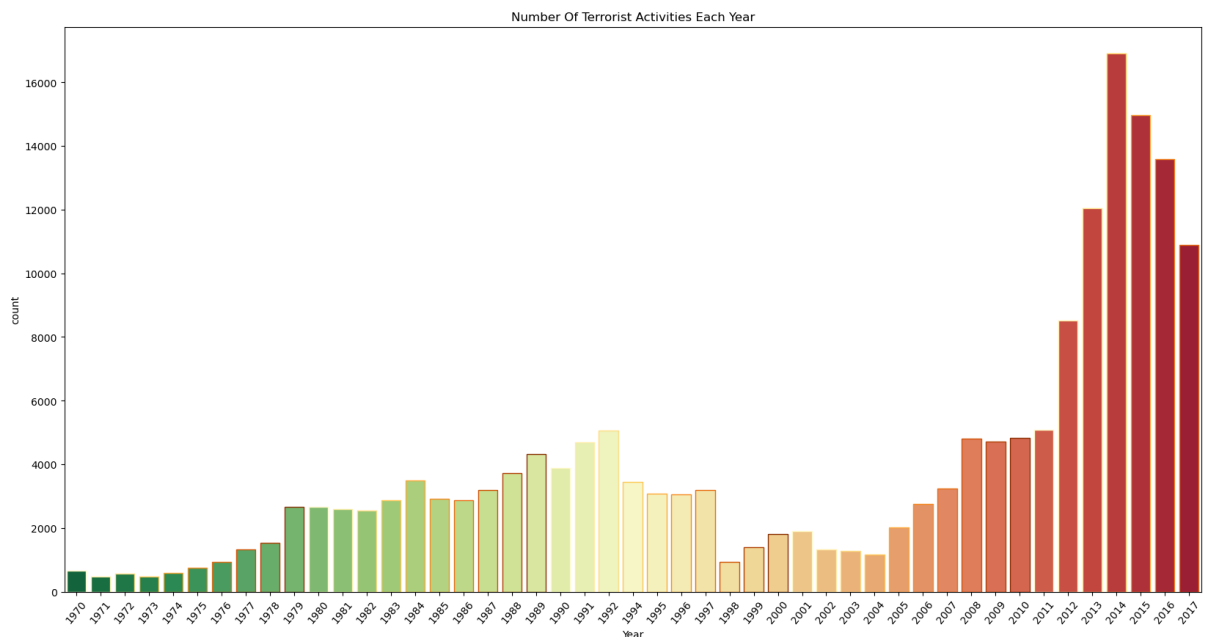
```
Out[14]: 1970      651
          1971      471
          1972      568
          1973      473
          1974      581
          1975      740
          1976      923
          1977     1319
          1978     1526
          1979     2662
          1980     2662
          1981     2586
          1982     2544
          1983     2870
          1984     3495
          1985     2915
          1986     2860
          1987     3183
          1988     3721
          1989     4324
          1990     3887
          1991     4683
          1992     5071
          1994     3456
          1995     3081
          1996     3058
          1997     3197
          1998      934
          1999     1395
          2000     1814
          2001     1906
          2002     1333
          2003     1278
          2004     1166
          2005     2017
          2006     2758
          2007     3242
          2008     4805
          2009     4721
          2010     4826
          2011     5076
          2012     8522
          2013    12036
          2014    16903
          2015    14965
          2016    13587
          2017    10900
          Name: Year, dtype: int64
```

## Terrorist Activities Each Year

```
In [15]: x_year = global_terror['Year'].unique()
y_count_years = global_terror['Year'].value_counts(dropna = False).sort_index()
plt.figure(figsize = (18,10))
sns.barplot(x = x_year,
            y = y_count_years,
            palette = 'rocket')
plt.xticks(rotation = 50)
plt.xlabel('Attack Year')
plt.ylabel('Number of Attacks Each Year')
plt.title('Attack In Years')
plt.show()
```

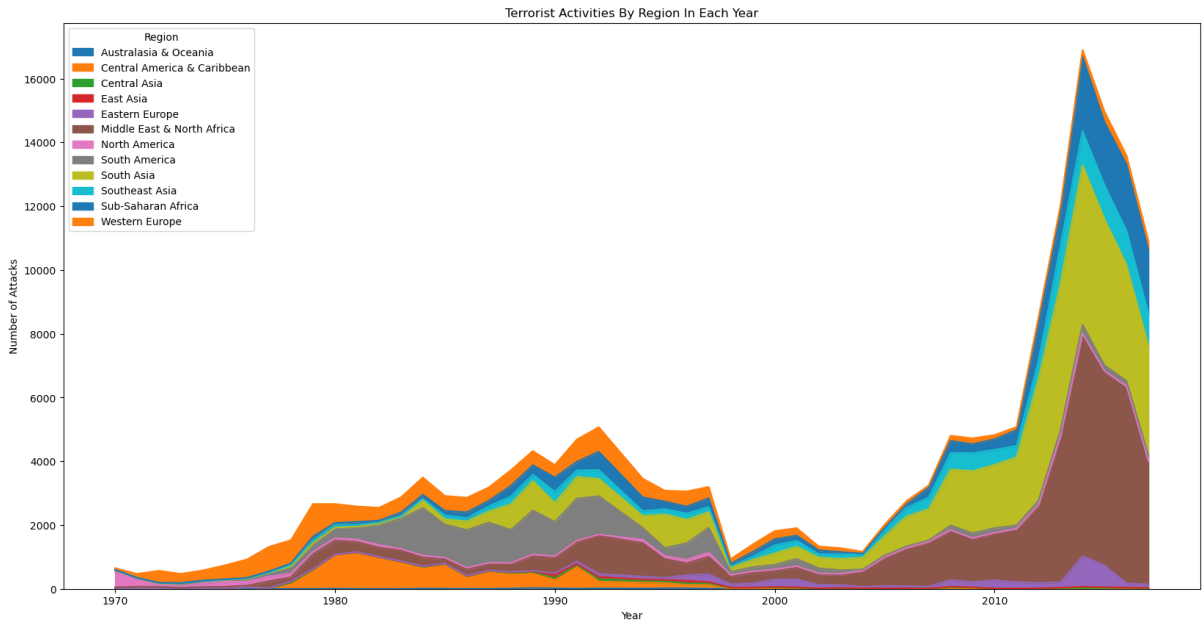


```
In [18]: plt.subplots(figsize=(20,10))
sns.countplot(x='Year', data=global_terror, palette='RdYlGn_r', edgecolor=sns.color_palette("YlOrRd", 10))
plt.xticks(rotation=50)
plt.title('Number Of Terrorist Activities Each Year')
plt.show()
```



# Terrorist Activities By Region In Each Year

```
In [19]: pd.crosstab(global_terror.Year, global_terror.Region).plot(kind='area',figsize=(20,10))
plt.title('Terrorist Activities By Region In Each Year')
plt.ylabel('Number of Attacks')
plt.show()
```



```
In [20]: global_terror['Wounded'] = global_terror['Wounded'].fillna(0).astype(int)
global_terror['Killed'] = global_terror['Killed'].fillna(0).astype(int)
global_terror['Casualties'] = global_terror['Killed'] + global_terror['Wounded']
```

```
In [21]: # Top 50 worst terrorist attacks
global_terror1 = global_terror.sort_values(by='Casualties',ascending=False)[:50]
```

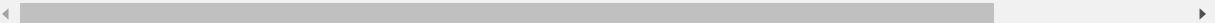
```
In [22]: heat=global_terror1.pivot_table(index='Country',columns='Year',values='Casualties')
heat.fillna(0,inplace=True)
```

```
In [23]: heat.head()
```

```
Out[23]:
```

	Year	1982	1984	1987	1992	1994	1995	1996	1997	1998	2001	...	2005	2006	2007	2008	2009	2013	2
Country																			
Afghanistan	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
Algeria	0.0	0.0	0.0	0.0	0.0	0.0	0.0	450.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
Chad	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	1161.0	0.0	0.0	
Egypt	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
Ethiopia	0.0	0.0	0.0	500.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	

5 rows × 21 columns



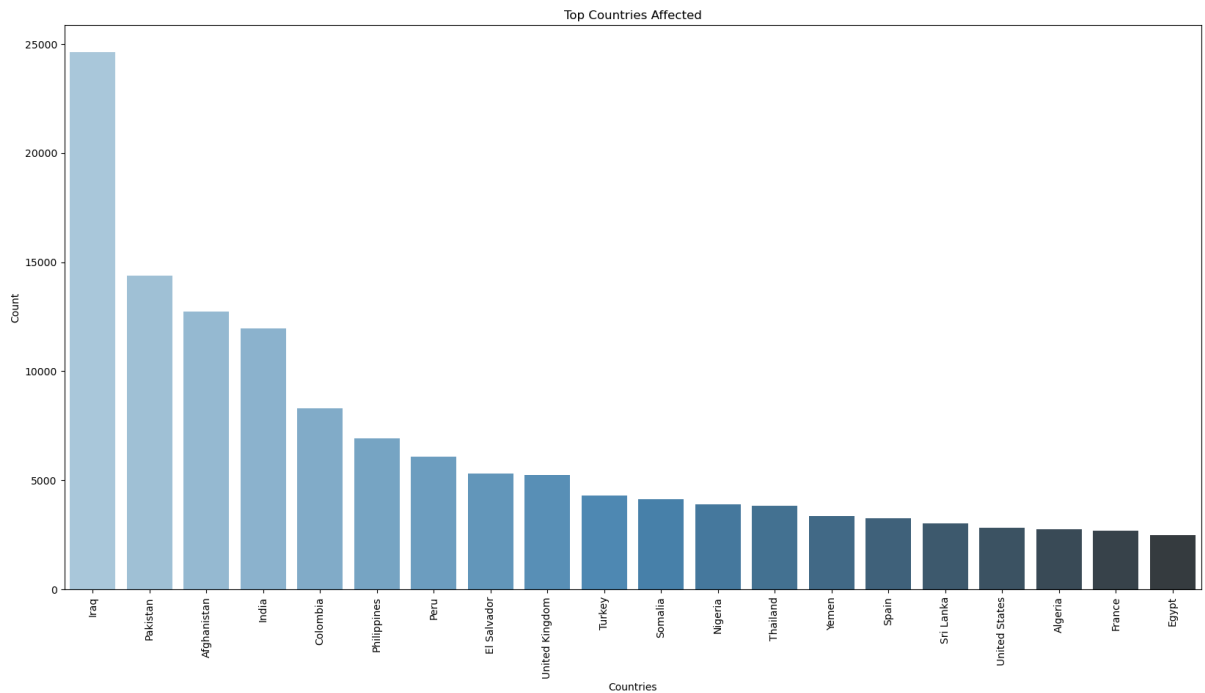
```
In [24]: import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
colorscale = [[0, '#edf8fb'], [.3, '#00BFFF'], [.6, '#8856a7'], [1, '#810f7c']]
heatmap = go.Heatmap(z=heat.values, x=heat.columns, y=heat.index, colorscale=colorscale)
data = [heatmap]
layout = go.Layout(
    title='Top 50 Worst Terror Attacks in History from 1982 to 2017',
    xaxis = dict(ticks='', nticks=20),
    yaxis = dict(ticks='')
)
fig = go.Figure(data=data, layout=layout)
py.iplot(fig, filename='heatmap', show_link=False)
```

```
In [25]: global_terror.Country.value_counts()[21]
```

```
Out[25]: Iraq                24636
Pakistan            14368
Afghanistan         12731
India               11960
Colombia            8306
Philippines         6908
Peru               6096
El Salvador         5320
United Kingdom      5235
Turkey             4292
Somalia            4142
Nigeria            3907
Thailand            3849
Yemen              3347
Spain              3249
Sri Lanka          3022
United States       2836
Algeria            2743
France             2693
Egypt              2479
Lebanon            2478
Name: Country, dtype: int64
```

## Top Countries Affected By Terrorist Attacks

```
In [27]: plt.subplots(figsize=(20,10))
sns.barplot(x=global_terror['Country'].value_counts()[:20].index, y=global_terror['Country'].value_counts()[:20].values)
plt.title('Top Countries Affected')
plt.xlabel('Countries')
plt.ylabel('Count')
plt.xticks(rotation=90)
plt.show()
```



## ANALYSIS ON CUSTOMIZED DATA

### Terrorist Attacks of a Particular year and their Locations

Let's look at the terrorist acts in the world over a certain year.

```
In [30]: import folium
from folium.plugins import MarkerCluster
```

```
In [31]: filterYear = global_terror['Year'] == 2001
```

```
In [32]: filterData = global_terror[filterYear] # filter data
# filterData.info()
reqFilterData = filterData.loc[:, 'city': 'longitude'] # get the required fields
reqFilterData = reqFilterData.dropna() # drop NaN values in latitude and longitude
reqFilterDataList = reqFilterData.values.tolist()
# reqFilterDataList
```



```
In [33]: map = folium.Map(location = [0, 50], tiles='CartoDB positron', zoom_start=2)
markerCluster = folium.plugins.MarkerCluster().add_to(map)
for point in range(0, len(reqFilterDataList)):
    folium.Marker(location=reqFilterDataList[point][1],reqFilterDataList[point][2],
                  popup = reqFilterDataList[point][0]).add_to(markerCluster)
map
```

Out[33]: Make this Notebook Trusted to load map: File -> Trust Notebook

From the above map, we can depict that the maximum attacks carried out in the year 2001 was on the African Continent, almost 1325 attacks. Then, the continent South America faced the highest number of attacks, i.e. 258.

## Terrorist's Organizations Operations In Each Country

```
In [35]: global_terror.Group.value_counts()[1:20]
```

```
Out[35]: Taliban 7478
Islamic State of Iraq and the Levant (ISIL) 5613
Shining Path (SL) 4555
Farabundo Marti National Liberation Front (FMLN) 3351
Al-Shabaab 3288
New People's Army (NPA) 2772
Irish Republican Army (IRA) 2671
Revolutionary Armed Forces of Colombia (FARC) 2487
Boko Haram 2418
Kurdistan Workers' Party (PKK) 2310
Basque Fatherland and Freedom (ETA) 2024
Communist Party of India - Maoist (CPI-Maoist) 1878
Maoists 1630
Liberation Tigers of Tamil Eelam (LTTE) 1606
National Liberation Army of Colombia (ELN) 1561
Tehrik-i-Taliban Pakistan (TTP) 1351
Palestinians 1125
Houthi extremists (Ansar Allah) 1062
Al-Qaida in the Arabian Peninsula (AQAP) 1020
Name: Group, dtype: int64
```

```
In [36]: test = global_terror[global_terror.Group.isin(['Shining Path (SL)', 'Taliban', 'Islamic State of Iraq and the Levant (ISIL)'])]
```

```
In [37]: test.Country.unique()
```

```
Out[37]: array(['Peru', 'Bolivia', 'Colombia', 'Argentina', 'Brazil', 'Mexico',  
              'Afghanistan', 'Pakistan', 'Syria', 'Iraq', 'Turkey', 'Tunisia',  
              'Lebanon', 'Turkmenistan', 'Israel', 'Belgium', 'Egypt', 'Libya',  
              'Saudi Arabia', 'West Bank and Gaza Strip', 'France', 'Bahrain',  
              'Jordan', 'Somalia', 'Germany', 'Yemen', 'Philippines', 'Malaysia',  
              'Indonesia', 'Russia', 'Georgia', 'United Kingdom', 'Iran',  
              'Australia'], dtype=object)
```

```
In [38]: global_terror_df_group = global_terror.dropna(subset=['latitude', 'longitude'])
```

```
In [39]: global_terror_df_group = global_terror_df_group.drop_duplicates(subset=['Country', 'Group'])
```

```
In [40]: terrorist_groups = global_terror.Group.value_counts()[1:8].index.tolist()  
global_terror_df_group = global_terror_df_group.loc[global_terror_df_group.Group.isin(terrorist_g  
print(global_terror_df_group.Group.unique())
```

```
["New People's Army (NPA)" 'Irish Republican Army (IRA)'  
'Shining Path (SL)' 'Farabundo Marti National Liberation Front (FMLN)'  
'Taliban' 'Al-Shabaab' 'Islamic State of Iraq and the Levant (ISIL)']
```

```
In [41]: map = folium.Map(location=[50, 0], tiles="CartoDB positron", zoom_start=2)  
markerCluster = folium.plugins.MarkerCluster().add_to(map)  
for i in range(0, len(global_terror_df_group)):  
    folium.Marker([global_terror_df_group.iloc[i]['latitude'], global_terror_df_group.iloc[i]['lon  
                    popup='Group: {}<br>Country: {}'.format(global_terror_df_group.iloc[i]['Group'],  
                    global_terror_df_group.iloc[i]['Country'])).add_to(map)  
map
```

```
Out[41]: Make this Notebook Trusted to load map: File -> Trust Notebook
```

```
In [42]: m1 = folium.Map(location=[50, 0], tiles="CartoDB positron", zoom_start=2)
marker_cluster = MarkerCluster(
    name='clustered icons',
    overlay=True,
    control=False,
    icon_create_function=None
)
for i in range(0, len(global_terror_df_group)):
    marker=folium.Marker([global_terror_df_group.iloc[i]['latitude'], global_terror_df_group.iloc[i]['longitude']],
        popup='Group:{}<br>Country:{}'.format(global_terror_df_group.iloc[i]['Group'],
            global_terror_df_group.iloc[i]['Country'])
    folium.Popup(popup).add_to(marker)
    marker_cluster.add_child(marker)
marker_cluster.add_to(m1)
folium.TileLayer('openstreetmap').add_to(m1)
#folium.TileLayer('Mapbox Bright').add_to(m1)
folium.TileLayer('cartodbdark_matter').add_to(m1)
folium.TileLayer('stamentoner').add_to(m1)
folium.LayerControl().add_to(m1)

m1
```

Out[42]: Make this Notebook Trusted to load map: File -> Trust Notebook

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

```
Out[43]:
```

	Year	Month	Day	Country	state	Region	city	latitude	longitude	AttackType	Killed	Wounded
0	1970	7	2	Dominican Republic	NaN	Central America & Caribbean	Santo Domingo	18.456792	-69.951164	Assassination	1	
1	1970	0	0	Mexico	Federal	North America	Mexico city	19.371887	-99.086624	Hostage Taking (Kidnapping)	0	
2	1970	1	0	Philippines	Tarlac	Southeast Asia	Unknown	15.478598	120.599741	Assassination	1	
3	1970	1	0	Greece	Attica	Western Europe	Athens	37.997490	23.762728	Bombing/Explosion	0	
4	1970	1	0	Japan	Fukouka	East Asia	Fukouka	33.580412	130.396361	Facility/Infrastructure Attack	0	

```
In [44]: # Total Number of people killed in terror attack
killData = global_terror.loc[:, 'Killed']
print('Number of people killed by terror attack:', int(sum(killData.dropna())))# drop the NaN value
```

Number of people killed by terror attack: 411868

```
In [45]: # Let's Look at what types of attacks these deaths were made of.
attackData = global_terror.loc[:, 'AttackType']
# attackData
typeKillData = pd.concat([attackData, killData], axis=1)
```

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

```
Out[46]:
```

	AttackType	Killed
0	Assassination	1
1	Hostage Taking (Kidnapping)	0
2	Assassination	1
3	Bombing/Explosion	0
4	Facility/Infrastructure Attack	0

```
In [47]: typeKillFormatData = typeKillData.pivot_table(columns='AttackType', values='Killed', aggfunc='sum')
typeKillFormatData
```

```
Out[47]:
```

AttackType	Armed Assault	Assassination	Bombing/Explosion	Facility/Infrastructure Attack	Hijacking	Hostage Taking (Barricade Incident)	Hostage Taking (Kidnapping)	Unarmed Assault
Killed	160297	24920	157321	3642	3718	4478	24231	8

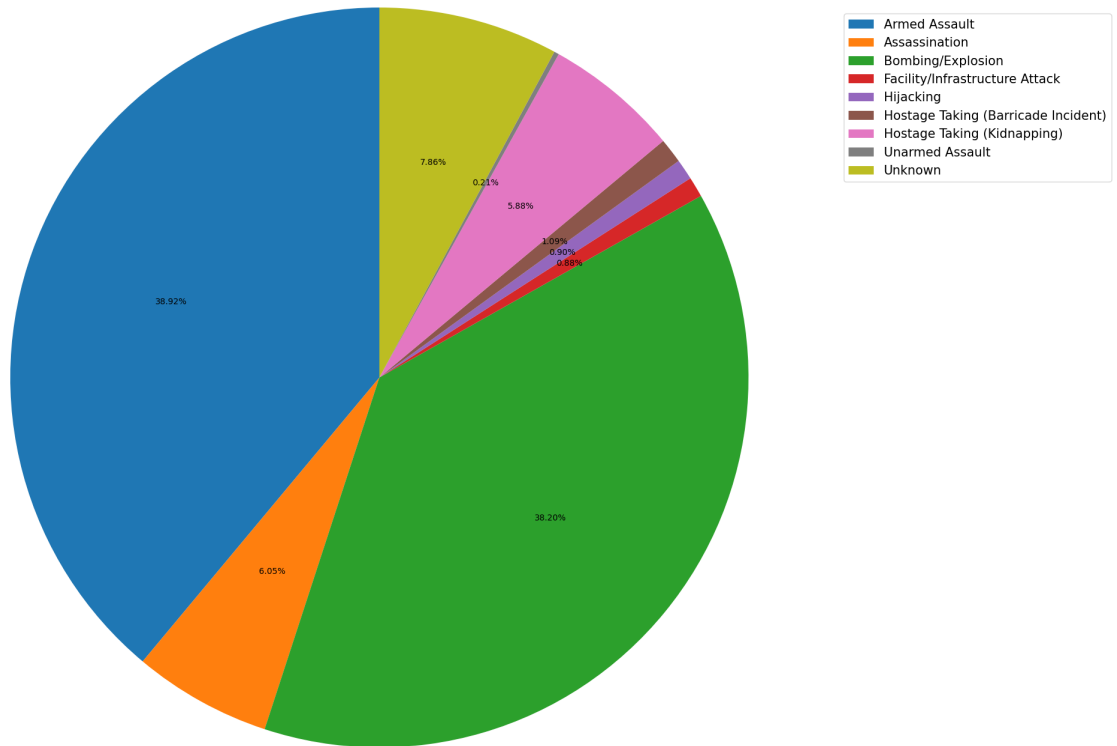
```
In [48]: typeKillFormatData.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1 entries, Killed to Killed
Data columns (total 9 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Armed Assault                        1 non-null     int32
1   Assassination                       1 non-null     int32
2   Bombing/Explosion                   1 non-null     int32
3   Facility/Infrastructure Attack      1 non-null     int32
4   Hijacking                          1 non-null     int32
5   Hostage Taking (Barricade Incident) 1 non-null     int32
6   Hostage Taking (Kidnapping)         1 non-null     int32
7   Unarmed Assault                    1 non-null     int32
8   Unknown                            1 non-null     int32
dtypes: int32(9)
memory usage: 152.0+ bytes
```

```
In [50]: labels = typeKillFormatData.columns.tolist()
transposed = typeKillFormatData.T
values = transposed.values.flatten().tolist() # Flatten the 2D array

fig, ax = plt.subplots(figsize=(20, 20), subplot_kw=dict(aspect="equal"))
plt.pie(values, startangle=90, autopct='%0.2f%%')
plt.title('Types Of Terrorist Attacks That Cause Deaths', fontsize=30)
plt.legend(labels, loc='upper right', bbox_to_anchor=(1.3, 0.9), fontsize=15) # Location of the legend
plt.show()
```

Types Of Terrorist Attacks That Cause Deaths



```
In [51]: global_terror.head(2)
```

```
Out[51]:
```

	Year	Month	Day	Country	state	Region	city	latitude	longitude	AttackType	Killed	Wounded	T
0	1970	7	2	Dominican Republic	NaN	Central America & Caribbean	Santo Domingo	18.456792	-69.951164	Assassination	1	0	Gu
1	1970	0	0	Mexico	Federal	North America	Mexico city	19.371887	-99.086624	Hostage Taking (Kidnapping)	0	0	N Cl dau

```
In [52]: # Number of Killed in Terrorist Attacks by Countries
countryData = global_terror.loc[:, 'Country']
# countyData
countryKillData = pd.concat([countryData, killData], axis=1)
```

```
In [53]: countryKillFormatData = countryKillData.pivot_table(columns='Country', values='Killed', aggfunc='sum', index='Country')
countryKillFormatData
```

Out[53]:

Country	Afghanistan	Albania	Algeria	Andorra	Angola	Antigua and Barbuda	Argentina	Armenia	Australia	Austria	...	Vietnam
Killed	39384	42	11066	0	3043	0	490	37	23	30	...	1

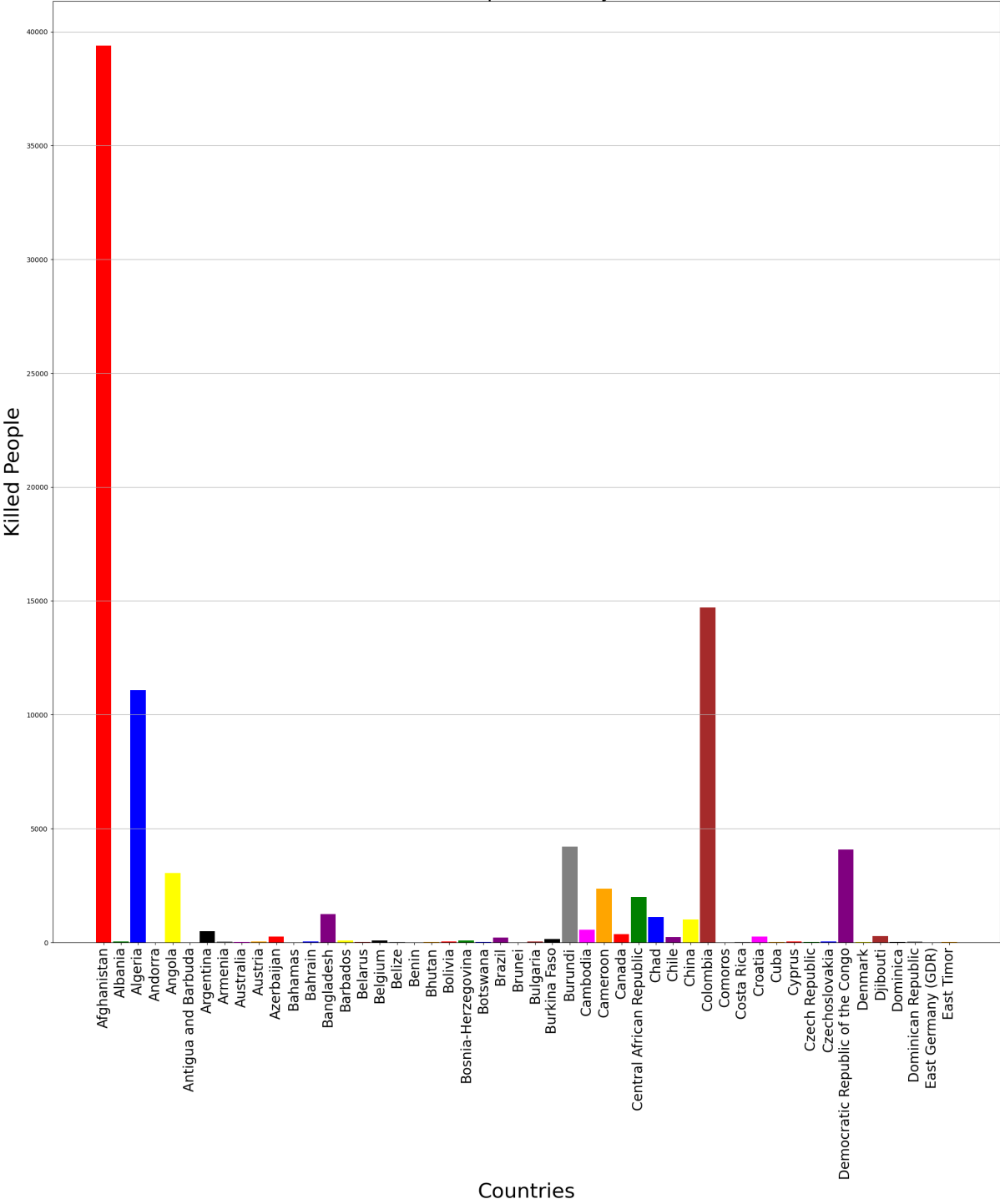
1 rows × 205 columns



```
In [54]: fig_size = plt.rcParams["figure.figsize"]
fig_size[0]=25
fig_size[1]=25
plt.rcParams["figure.figsize"] = fig_size
```

```
In [55]: labels = countryKillFormatData.columns.tolist()
labels = labels[:50] #50 bar provides nice view
index = np.arange(len(labels))
transpose = countryKillFormatData.T
values = transpose.values.tolist()
values = values[:50]
values = [int(i[0]) for i in values] # convert float to int
colors = ['red', 'green', 'blue', 'purple', 'yellow', 'brown', 'black', 'gray', 'magenta', 'orange']
fig, ax = plt.subplots(1, 1)
ax.yaxis.grid(True)
fig_size = plt.rcParams["figure.figsize"]
fig_size[0]=25
fig_size[1]=25
plt.rcParams["figure.figsize"] = fig_size
plt.bar(index, values, color = colors, width = 0.9)
plt.ylabel('Killed People', fontsize=30)
plt.xlabel('Countries', fontsize = 30)
plt.xticks(index, labels, fontsize=20, rotation=90)
plt.title('Number of People Killed By Countries', fontsize = 30)
# print(fig_size)
plt.show()
```

Number of People Killed By Countries

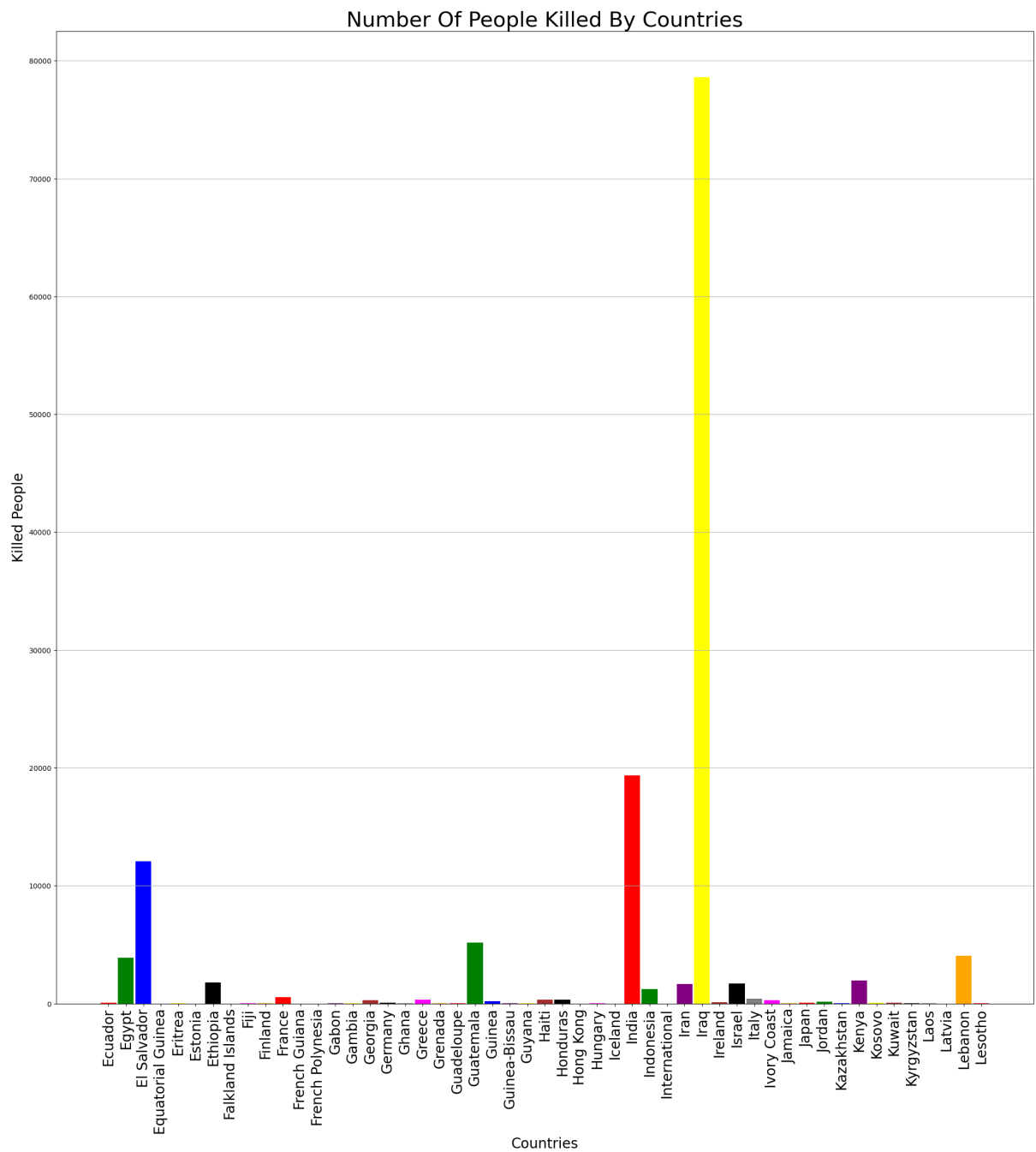




```

In [56]: labels = countryKillFormatData.columns.tolist()
labels = labels[50:101]
index = np.arange(len(labels))
transpose = countryKillFormatData.T
values = transpose.values.tolist()
values = values[50:101]
values = [int(i[0]) for i in values]
colors = ['red', 'green', 'blue', 'purple', 'yellow', 'brown', 'black', 'gray', 'magenta', 'orange']
fig, ax = plt.subplots(1, 1)
ax.yaxis.grid(True)
fig_size = plt.rcParams["figure.figsize"]
fig_size[0]=20
fig_size[1]=20
plt.rcParams["figure.figsize"] = fig_size
plt.bar(index, values, color = colors, width = 0.9)
plt.ylabel('Killed People', fontsize=20)
plt.xlabel('Countries', fontsize = 20)
plt.xticks(index, labels, fontsize=20, rotation=90)
plt.title('Number Of People Killed By Countries', fontsize = 30)
plt.show()

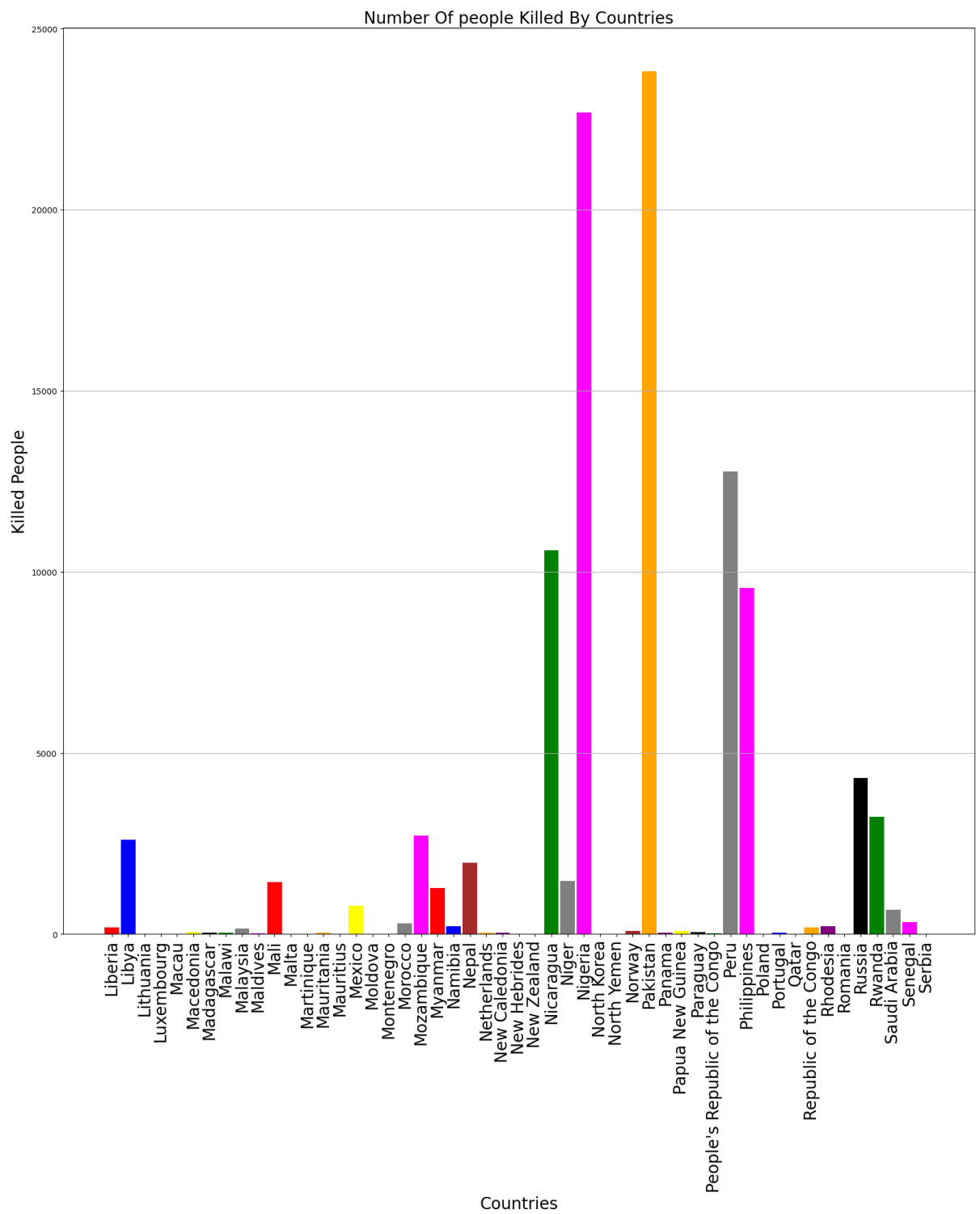
```



```

In [57]: labels = countryKillFormatData.columns.tolist()
labels = labels[101:152]
index = np.arange(len(labels))
transpose = countryKillFormatData.T
values = transpose.values.tolist()
values = values[101:152]
values = [int(i[0]) for i in values]
colors = ['red', 'blue', 'brown', 'orange', 'purple', 'yellow', 'black', 'green', 'gray', 'magenta']
fig, ax = plt.subplots(1, 1)
ax.yaxis.grid(True)
fig_size = plt.rcParams["figure.figsize"]
fig_size[0]=25
fig_size[1]=25
plt.rcParams["figure.figsize"] = fig_size
plt.bar(index, values, color = colors, width = 0.9)
plt.ylabel('Killed People', fontsize=20)
plt.xlabel('Countries', fontsize = 20)
plt.xticks(index, labels, fontsize=20, rotation=90)
plt.title('Number Of people Killed By Countries', fontsize = 20)
plt.show()

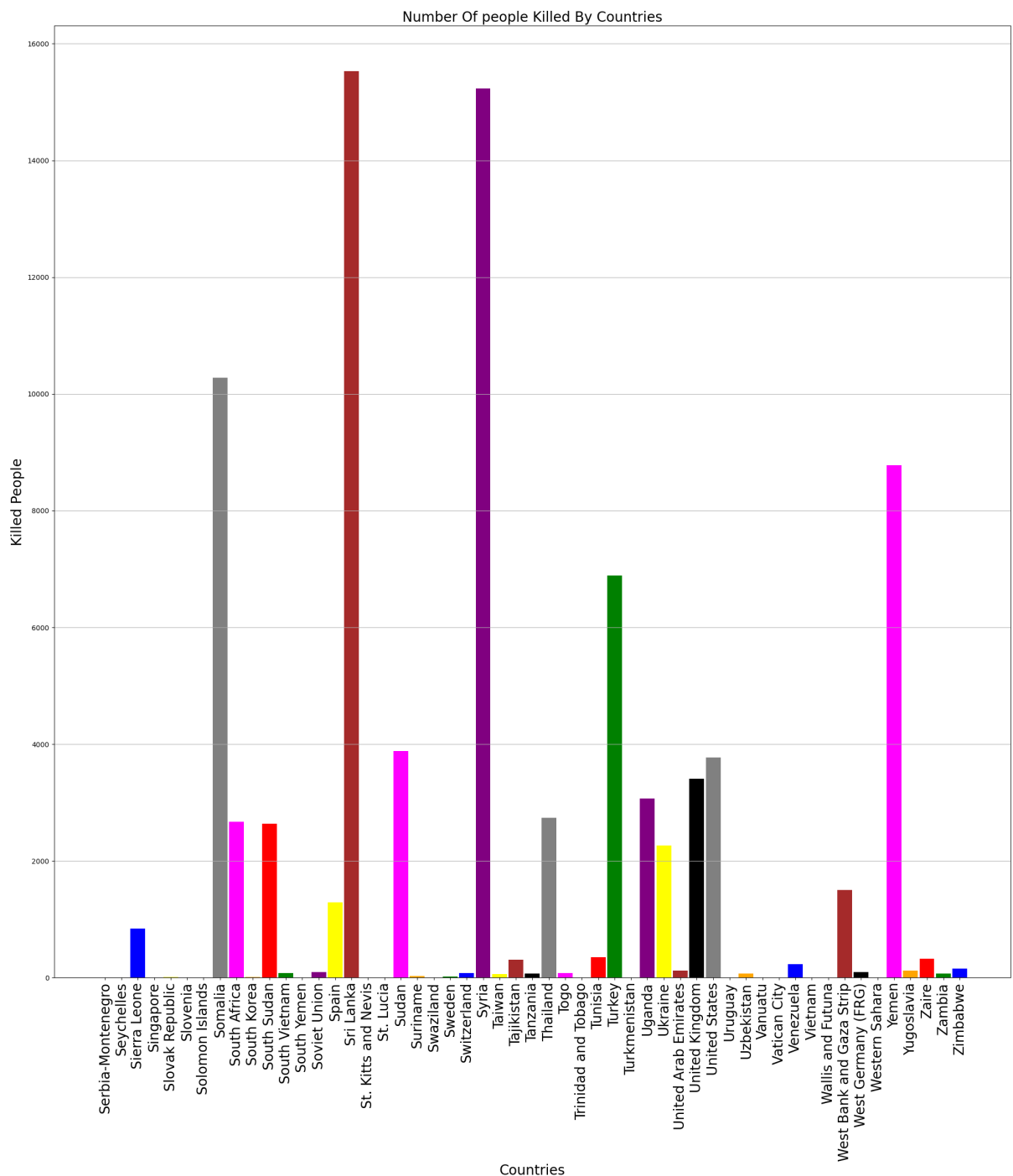
```



```

In [58]: labels = countryKillFormatData.columns.tolist()
labels = labels[152:206]
index = np.arange(len(labels))
transpose = countryKillFormatData.T
values = transpose.values.tolist()
values = values[152:206]
values = [int(i[0]) for i in values]
colors = ['red', 'green', 'blue', 'purple', 'yellow', 'brown', 'black', 'gray', 'magenta', 'orange']
fig, ax = plt.subplots(1, 1)
ax.yaxis.grid(True)
fig_size = plt.rcParams["figure.figsize"]
fig_size[0]=25
fig_size[1]=25
plt.rcParams["figure.figsize"] = fig_size
plt.bar(index, values, color = colors, width = 0.9)
plt.ylabel('Killed People', fontsize=20)
plt.xlabel('Countries', fontsize = 20)
plt.xticks(index, labels, fontsize=20, rotation=90)
plt.title('Number Of people Killed By Countries', fontsize = 20)
plt.show()

```



## CONCLUSIONS :

*From the above graphs, we can see that the countries where most people are killed are : Afghanistan, Columbia, Iran, Sri lanka, Syria, Somalia, Yemen naming a few. Even though there is a perception that Muslims are supporters of terrorism, but Muslims are the people who are most damaged by terrorist attacks.*

So after different type of analyzation ,Overall terrorism is suddenly increased from 2010 and I have ranked the Hot zone in terms of terrorist activities in all regions -

### #Middle East & North Africa

#### Iraq

Overall having highest Number of terrorist rate and sudden increase in terrorist activity after year 2010

#### Libya

Overall having low terrorism activity, Second Highest in terms of successful terrorist activities which takes place in this country and an increased in terrorism rate after 2010.

#### Yemen

Overall having low terrorism activity, Third Highest in terms of successful terrorist activities which takes place in this country and an increased in terrorism rate after 2010.

### #South Asia

#### Afghanistan

Overall having 1st Highest terrorist rates and 1st highest in terms of extended terrorist rates and a very high increase in terrorism rate after 2010.

#### Pakistan

Overall having 2nd highest terrorist rates but there is a good decrease in terrorist rates if we consider the 20s century and an increase in terrorism rate after 2010 but not as that of increase like Afghanistan and India.

#### India

Overall having 3rd highest terrorist rates but if we consider extended terrorist rate so this country is 2nd highest and a very high increase in terrorism rate after 2010.

### #Sub-Saharan Africa

#### Nigeria

Overall having low terrorism activity but 1st Highest in terms of extended terrorist activities which takes place in this country and 2nd in terms of increased of terrorism rate after 2010.

#### Somalia

Overall having low terrorism activity but 2nd Highest in terms of extended terrorist activities which takes place in this country and 1st in terms of increased of terrorism rate after 2010.

#### Sudan

Overall having low terrorism activity but 3rd Highest in terms of extended terrorist activities which takes place in this country and a bit increase in terrorism rate after 2010.

### #Solution

- More security surveillance required at Iraq.

- Noticing the trends of terrorism activities , hugely populated regions suffer major kill ratios. This must be controlled with strict border rules

**Thank You!**

**GitHub:** <https://github.com/anujtiwari21?tab=repositories>  
(<https://github.com/anujtiwari21?tab=repositories>).