

# Exploratory Data Analysis on Dataset – Terrorism

## Exploratory Data Analysis

Exploratory Data Analysis (EDA) refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.

Exploratory Data Analysis (EDA) is an approach to analyze the data using visual techniques. It is used to discover trends, patterns, or to check assumptions with the help of statistical summary and graphical representations.

```
[ ] # Importing all the important Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import seaborn as sns
%matplotlib inline
import plotly.express as px
import plotly.graph_objects as go
from collections import Counter

import warnings
warnings.filterwarnings('ignore')

[ ] #loading dataset
# Read the CSV file
dataset=pd.read_csv('globalterrorismdb_0718dist.csv',encoding='ISO-8859-1',low_memory=False)
dataset.head()
```

The screenshot shows a Jupyter Notebook interface with the following content:

**Code Cell 1:**

```
[ ] # Read the CSV file
dataset=pd.read_csv('globalterrorismdb_0718dist.csv',encoding='ISO-8859-1',low_memory=False)
dataset.head()
```

**Output:**

	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
0	197000000001	1970	7	2	NaN	0	NaN	58	Dominican Republic	2	...	NaN	NaN	NaN	NaN	PGIS	0.0	0.0	0.0	0.0	NaN
1	197000000002	1970	0	0	NaN	0	NaN	130	Mexico	1	...	NaN	NaN	NaN	NaN	PGIS	0.0	1.0	1.0	1.0	NaN
2	197001000001	1970	1	0	NaN	0	NaN	160	Philippines	5	...	NaN	NaN	NaN	NaN	PGIS	-9.0	-9.0	1.0	1.0	NaN
3	197001000002	1970	1	0	NaN	0	NaN	78	Greece	8	...	NaN	NaN	NaN	NaN	PGIS	-9.0	-9.0	1.0	1.0	NaN
4	197001000003	1970	1	0	NaN	0	NaN	101	Japan	4	...	NaN	NaN	NaN	NaN	PGIS	-9.0	-9.0	1.0	1.0	NaN

5 rows x 135 columns

**Code Cell 2:**

```
[ ] # Shape of Dataset
dataset.shape
```

**Output:**

```
(2909, 135)
```

**Code Cell 3:**

```
[ ] # Dataset Columns
dataset.columns
```

**Output:**

```
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)
```

**Code Cell 4:**

```
[ ] #Rename the necessary columns
```

```
Exploratory Data Analysis on Dataset - Terrorism.ipynb
File Edit View Insert Runtime Tools Help Last saved at 5:24 PM

+ Code + Text

[ ] #Checking for Missing data:
dataset.isnull().sum()

Year      0
Month     0
Day        0
Country   0
Region    0
City       0
latitude  48
longitude  48
AttackType 0
Killed    305
Wounded   1062
Casualties 1071
Target     47
Group      0
Target_type 0
Weapon_type 0
dtype: int64

[ ] #Removing the Missing data:
dataset.dropna(axis=0, inplace=True)
dataset.shape

(1759, 16)

[ ] #Re-Checking for Missing Data:
dataset.isnull().sum()

Year      0
Month     0
Day        0
Country   0
Region    0
City       0
latitude  48
longitude  48
AttackType 0
Killed    305
Wounded   1062
Casualties 1071
Target     47
Group      0
Target_type 0
Weapon_type 0
dtype: int64
```

```
Exploratory Data Analysis on Dataset - Terrorism.ipynb
File Edit View Insert Runtime Tools Help Last saved at 5:24 PM

+ Code + Text

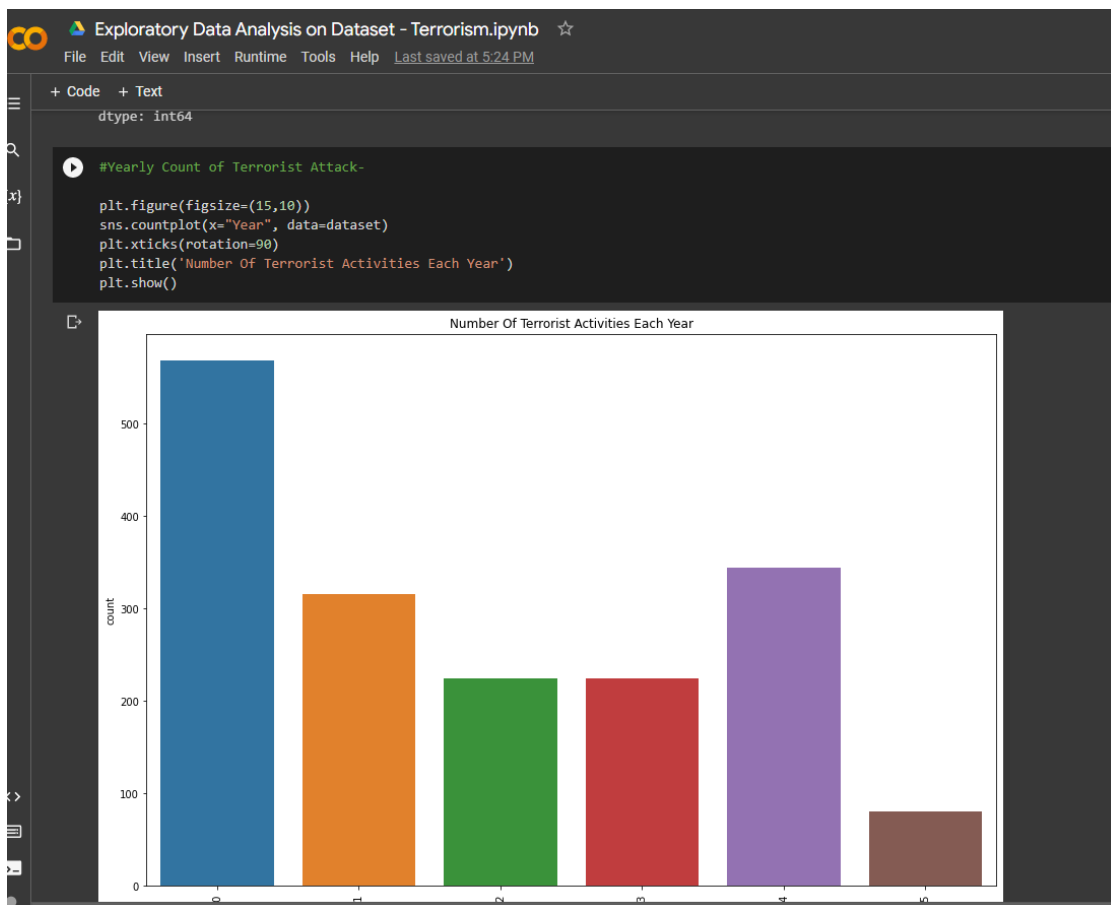
[ ] Country      0
Region          0
City            0
latitude        48
longitude        48
AttackType      0
Killed          305
Wounded         1062
Casualties      1071
Target          47
Group           0
Target_type     0
Weapon_type     0
dtype: int64

[ ] #Removing the Missing data:
dataset.dropna(axis=0, inplace=True)
dataset.shape

(1759, 16)

[ ] #Re-Checking for Missing Data:
dataset.isnull().sum()

Year      0
Month     0
Day        0
Country   0
Region    0
City       0
latitude  0
longitude  0
AttackType 0
Killed     0
Wounded    0
Casualties 0
Target     0
Group      0
Target_type 0
Weapon_type 0
dtype: int64
```



## Observation

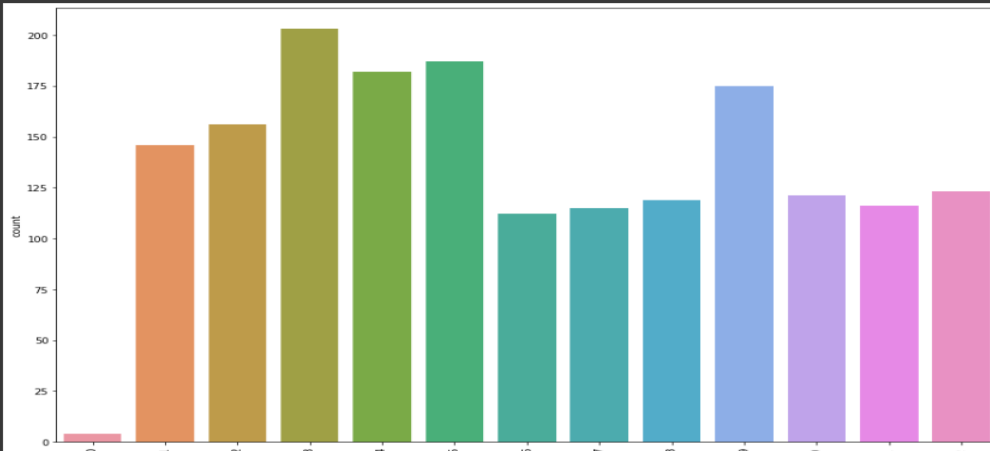
From the graph we can see 2013-17 marks the highest attacks with 2014 having the highest.

There has been a gradual increase in Terror Activities since 2004.

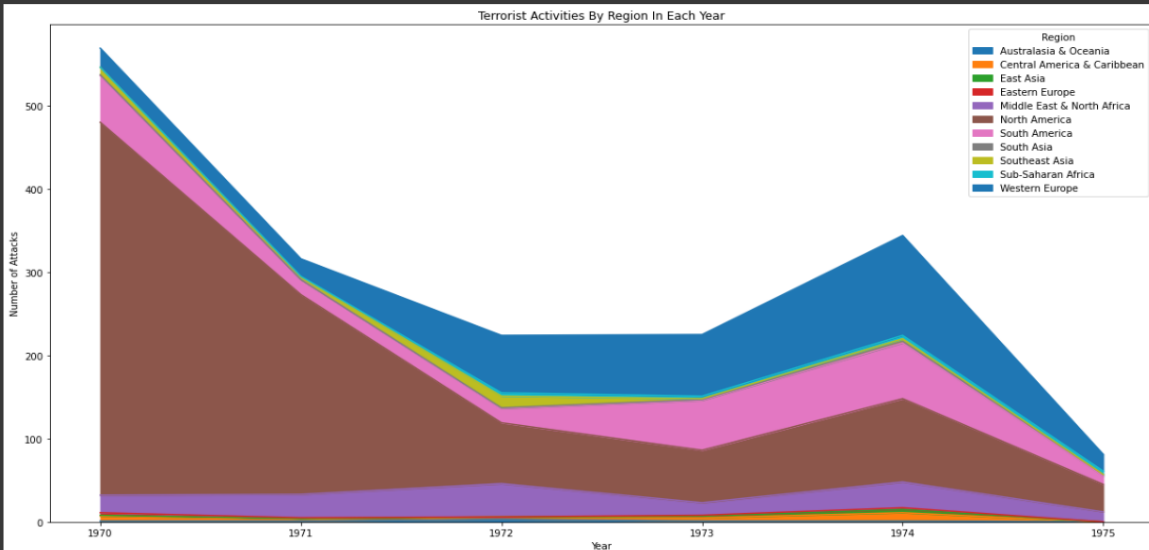
Highest number of terror activities occurred in the year 2014.

After 2014 the terror activities started to decrease.

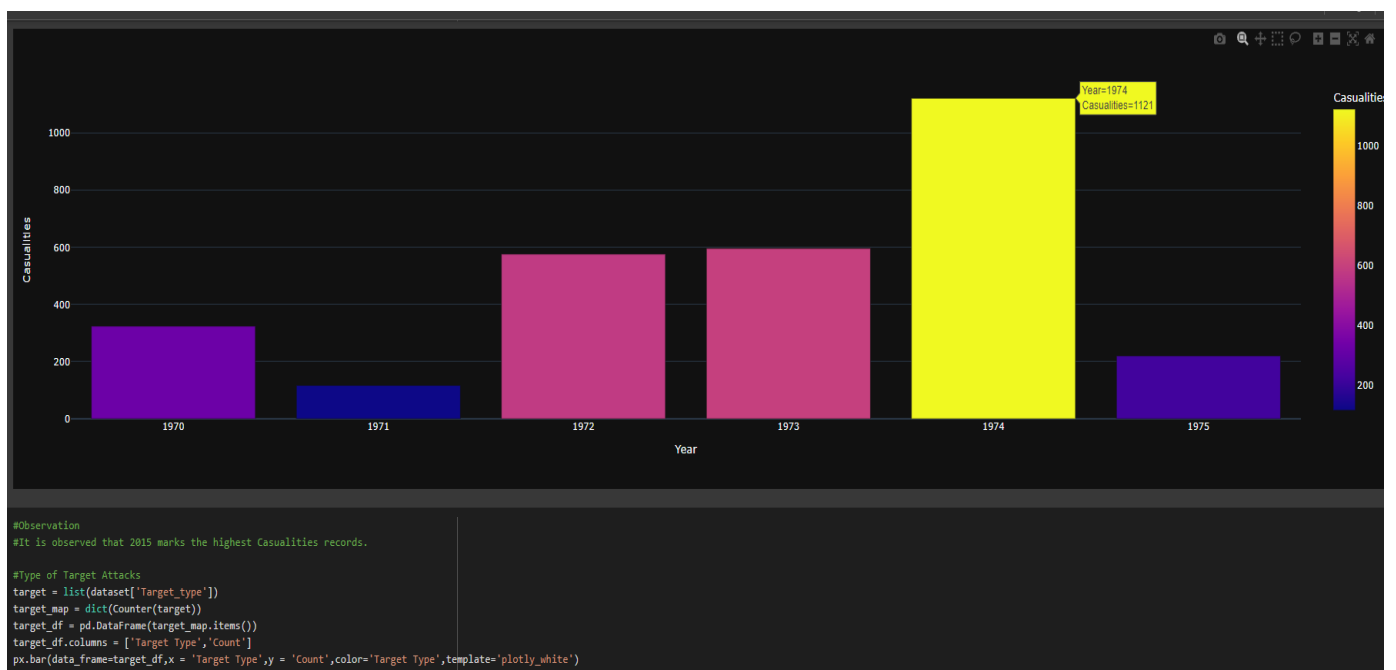
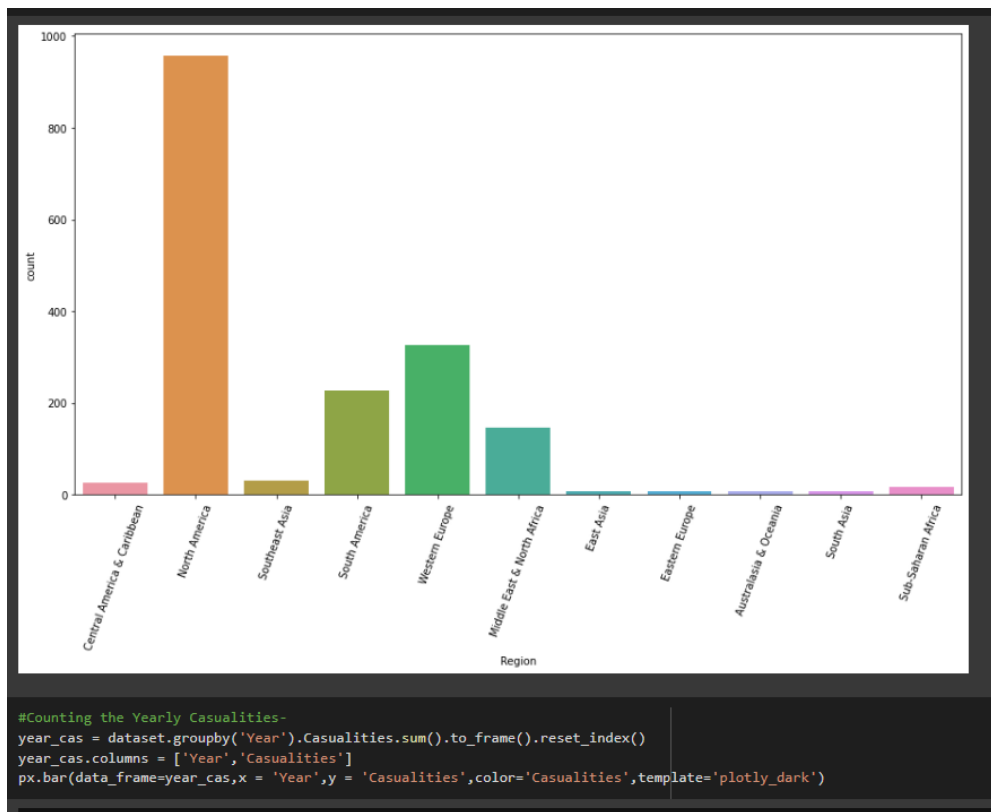
```
#month analysis
dataset['Month'].value_counts()
plt.figure(figsize = (15, 10))
sns.countplot(x='Month', data = dataset)
plt.xticks(rotation=90)
plt.show()
```

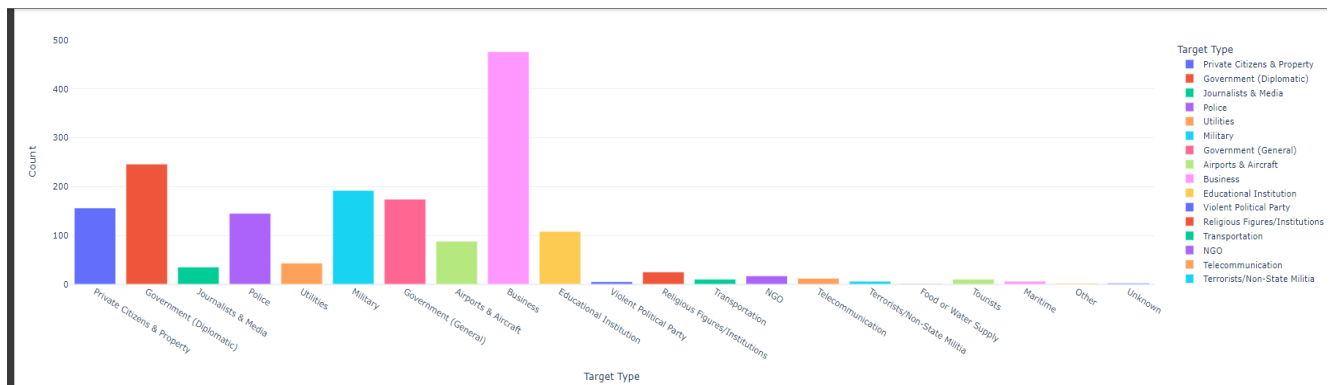


```
#Terrorist Activities By Region In Each Year
pd.crosstab(dataset.Year, dataset.Region).plot(kind='area',figsize=(20,10))
plt.title('Terrorist Activities By Region In Each Year')
plt.ylabel('Number of Attacks')
plt.show()
```



```
#REGION AFFECTED BY TERRORIST ATTACK
dataset['Region'].value_counts()
plt.figure(figsize=(15,8))
sns.countplot(x='Region',data=dataset)
plt.xticks(rotation=70)
plt.show()
```





```
# Observation
# Private Citizens and Property Counts the highest amongst all.

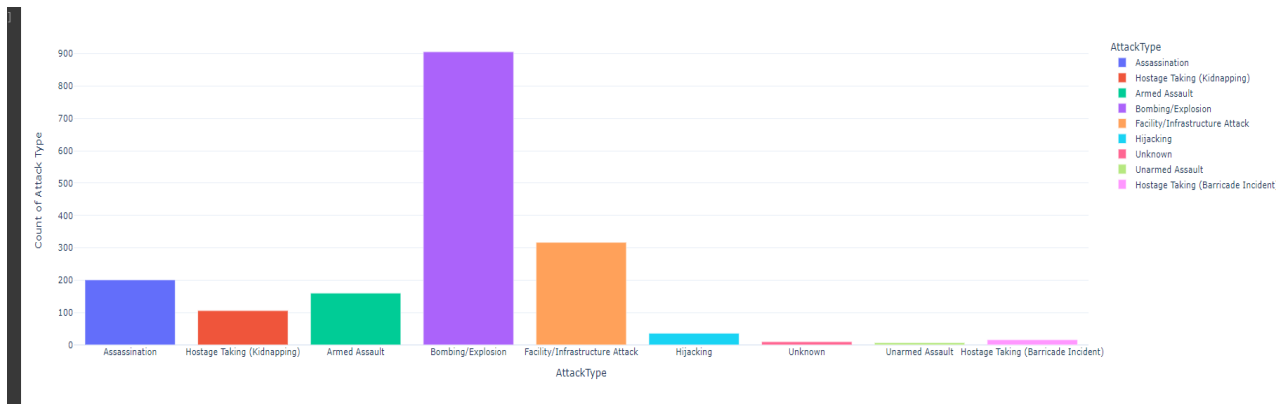
# Analysing the Type of Attacks:-
# Counting the Casualties according to the Attack Type
AttackType=dataset.pivot_table(columns='AttackType',values='Casualties',aggfunc='sum')
AttackType = AttackType.T
AttackType['Type'] = AttackType.index
```

```
#plotting the Attack Type
labels = AttackType.columns.tolist()
attack=AttackType.T
values=attack.values.tolist()
values = sum(values,[])
attack_type = list(dataset['AttackType'].unique())
fig = go.Figure(data=[go.Pie(labels = attack_type,values=values,hole=.3)])
fig.update_layout(template = 'gridon')
fig.show()
```



```
# Observation
# Bombing and Explosion method shows the highest chosen type.

# Count of Weapon Chosen for Attack.
from collections import Counter
values = list(dataset['AttackType'])
value_map = dict(Counter(values))
value_dataset = pd.DataFrame(value_map.items())
value_dataset.columns = ["AttackType","Count of Attack Type"]
px.bar(data_frame=value_dataset,x = 'AttackType',y = 'Count of Attack Type',color = 'AttackType',template="plotly_white")
```



```

] # Observation
# Again, Bombing and Explosion shows the highest.

# Plotting the HOT-ZONE of Terrorism on the highest year of Terrorist Attack i.e. 2014.
import folium
from folium.plugins import MarkerCluster
year=dataset[dataset['Year']==2014]
mapData=year.loc[:,['city','longitude']]
mapData=mapData.dropna().values.tolist()

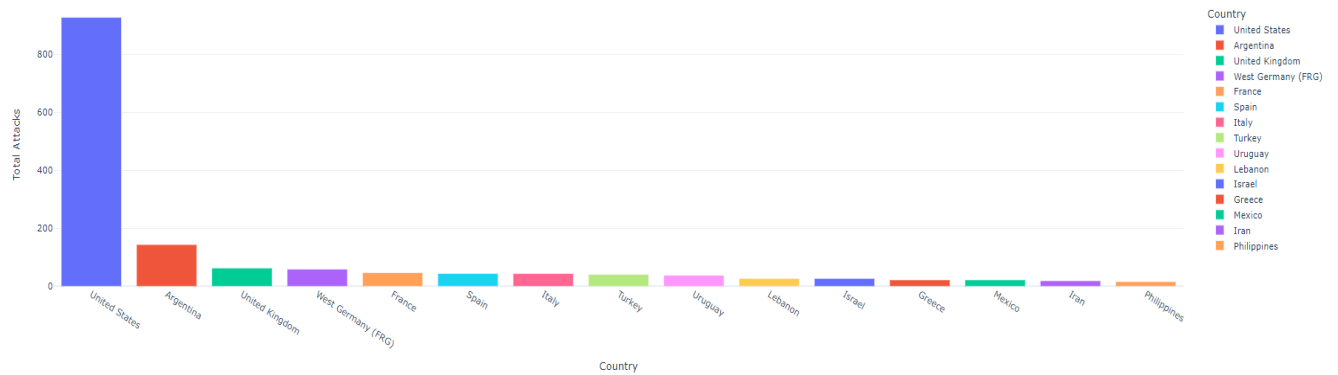
map = folium.Map(location = [0, 50], tiles='CartoDB positron', zoom_start=2)
markerCluster = folium.plugins.MarkerCluster().add_to(map)
for point in range(0, len(mapData)):
    folium.Marker(location=[mapData[point][1],mapData[point][2]],
                  popup = mapData[point][0]).add_to(markerCluster)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

```



```
[ ] # Iraq shows the highest Terror Attacks followed by other Middle-east region.
```

```
# Top 15 Countries showing the Highest Terror Attack.
plt.figure(figsize=(15,6))
country_attack=dataset.Country.value_counts()[1:15].reset_index()
country_attack.columns= ["Country", "Total Attacks"]
px.bar(data_frame= country_attack,x= 'Country',y= 'Total Attacks',color= 'Country',template='plotly_white')
```



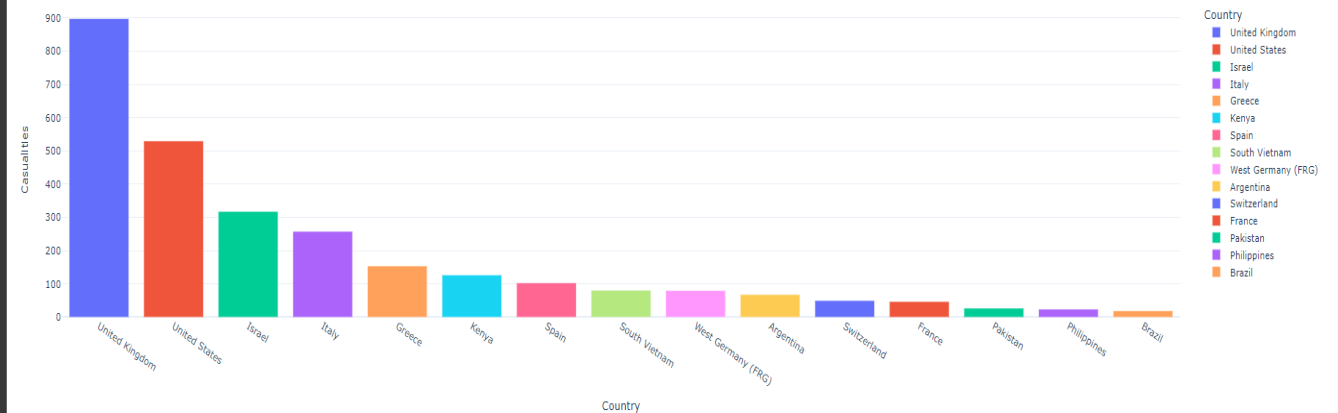
<Figure size 1080x432 with 0 Axes>

```
[ ] # Observation
# Iraq, again the highest followed by Pakistan, Afghanistan and India.
```

```
# Counting the Total Number of Casualties in each Country.
plt.figure(figsize=(15, 8))
cas_count= dataset.groupby("Country").Casualties.sum().to_frame().reset_index().sort_values("Casualties", ascending=False)[1:15]
px.bar(data_frame=cas_count,x= 'Country',y= 'Casualties',color='Country',template='plotly_white')
```

```
[ ] # Observation
# Iraq, again the highest followed by Pakistan, Afghanistan and India.
```

```
# Counting the Total Number of Casualties in each Country.
plt.figure(figsize=(15, 8))
cas_count= dataset.groupby("Country").Casualties.sum().to_frame().reset_index().sort_values("Casualties", ascending=False)[1:15]
px.bar(data_frame=cas_count,x= 'Country',y= 'Casualties',color= 'Country',template='plotly_white')
```

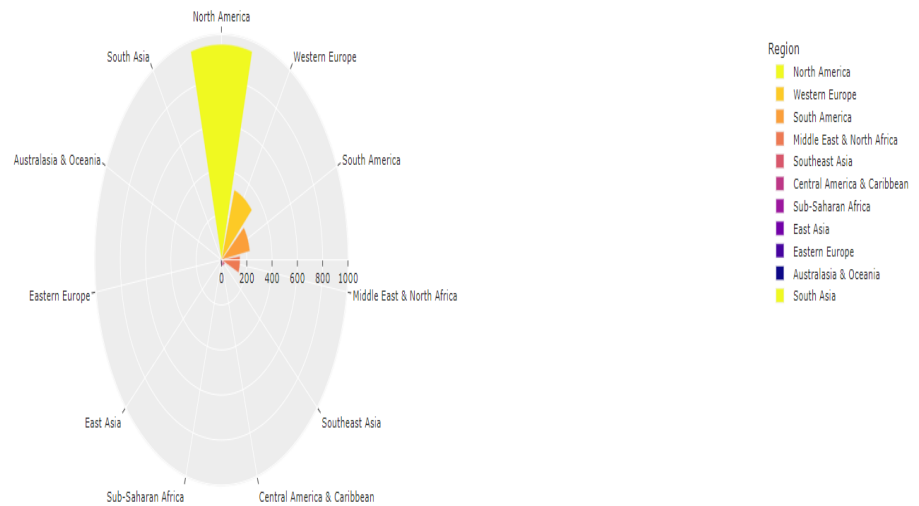


<Figure size 1080x576 with 0 Axes>

```
[ ] # Count of Terror Attack Region-Wise.
```

```
region_attacks = dataset.Region.value_counts().to_frame().reset_index()
region_attacks.columns= ["Region", "Total Attacks"]
fig = px.bar_polar(data_frame=region_attacks,r= 'Total Attacks',theta='Region',color= 'Region',
template="ggplot2", color_discrete_sequence= px.colors.sequential.Plasma_r)
fig.show()
```

```
[ ] # Count of Terror Attack Region-Wise.
region_attacks = dataset.Region.value_counts().to_frame().reset_index()
region_attacks.columns = ['Region', 'Total Attacks']
fig = px.bar_polar(data_frame=region_attacks, r = 'Total Attacks', theta='Region', color = 'Region',
                  template="ggplot2", color_discrete_sequence= px.colors.sequential.Plasma_r)
fig.show()
```



## Observation

Middle East and North Africa shows the highest followed by South Asia.

## Conclusion

- Hot zones of terrorism is Middle east and North Africa so, we should focus in these region.
- Iraq, Afghanistan and Pakistan most suffered country, Government should be aware from the citizens of these countries.
- Terrorist like to target Private citizens, Army and Police mostly, Security should be tighten in all these areas.
- BOMBING and EXPLOSIVE are most used weapon and attack type b=used by terrorist Government should tighten borders and should strict arms law.
- Most number of attacks were done by unknown group or not an group terrorist
- All country should have to make pact to tackle terrorism because after 2005 there is rapid increase in Terrorist Activities.