Exploratory Data Analysis - Terrorism (#Task4)

Author : Snehal Prasad

Importing all the libraries required for Analysis

In [37]:

```
import numpy as np
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Importing and observing the data

In [38]:

```
import warnings
warnings.filterwarnings('ignore')

data = pd.read_csv (r"C:\Users\A\Downloads\Global Terrorism - START data\globalterrorismdb_
df=pd.DataFrame(data)
print("Data has been successfully imported")
df.head()
```

```
Data has been successfully imported
```

Out[38]:

	eventid	iyear	imonth	iday	approxdate	extended	resolution	country	country_txt
0	197000000001	1970	7	2	NaN	0	NaN	58	Dominican Republic
1	197000000002	1970	0	0	NaN	0	NaN	130	Mexico
2	197001000001	1970	1	0	NaN	0	NaN	160	Philippines
3	197001000002	1970	1	0	NaN	0	NaN	78	Greece
4	197001000003	1970	1	0	NaN	0	NaN	101	Japan

5 rows × 135 columns

In [53]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 181691 entries, 0 to 181690
Data columns (total 20 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   Region          181691 non-null  object 
 5   Province/State  181270 non-null  object 
 6   City            181257 non-null  object 
 7   Latitude        177135 non-null  float64
 8   Longitude       177134 non-null  float64
 9   Location        55495 non-null   object 
 10  Summary         115562 non-null  object 
 11  Attack Type     181691 non-null  object 
 12  Target Type     181691 non-null  object 
 13  Group Name      181691 non-null  object 
 14  Motive          50561 non-null   object 
 15  Weapon Type     181691 non-null  object 
 16  Killed          181691 non-null  float64
 17  Wounded         181691 non-null  float64
 18  Add Notes       28289 non-null   object 
 19  Casualty        181691 non-null  float64
dtypes: float64(5), int64(3), object(12)
memory usage: 27.7+ MB
```

In [40]:

```
df.shape
```

Out[40]:

```
(181691, 135)
```

In [41]:

```
df.columns
```

Out[41]:

```
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 [42]:

```
for i in df.columns:
    print(i,end=", ")
```

```
eventid, iyear, imonth, iday, approxdate, extended, resolution, country, cou
ntry_txt, region, region_txt, provstate, city, latitude, longitude, specific
ity, vicinity, location, summary, crit1, crit2, crit3, doubtterr, alternativ
e, alternative_txt, multiple, success, suicide, attacktype1, attacktype1_tx
t, attacktype2, attacktype2_txt, attacktype3, attacktype3_txt, targtype1, ta
rgtype1_txt, targsubtype1, targsubtype1_txt, corp1, target1, natlty1, natlty
1_txt, targtype2, targtype2_txt, targsubtype2, targsubtype2_txt, corp2, targ
et2, natlty2, natlty2_txt, targtype3, targtype3_txt, targsubtype3, targsubty
pe3_txt, corp3, target3, natlty3, natlty3_txt, gname, gsubname, gname2, gsub
name2, gname3, gsubname3, motive, guncertain1, guncertain2, guncertain3, ind
ividual, nperps, nperpcap, claimed, claimmode, claimmode_txt, claim2, claimm
ode2, claimmode2_txt, claim3, claimmode3, claimmode3_txt, compclaim, weaptyp
e1, weaptype1_txt, weapsubtype1, weapsubtype1_txt, weaptype2, weaptype2_txt,
weapsubtype2, weapsubtype2_txt, weaptype3, weaptype3_txt, weapsubtype3, weap
subtype3_txt, weaptype4, weaptype4_txt, weapsubtype4, weapsubtype4_txt, weap
detail, nkill, nkillus, nkillter, nwound, nwoundus, nwoundte, property, prop
extent, propextent_txt, propvalue, propcomment, ishostkid, nhostkid, nhostki
dus, nhours, ndays, divert, kidhijcountry, ransom, ransomamt, ransomamtus, r
ansompaid, ransompaidus, ransomnote, hostkidoutcome, hostkidoutcome_txt, nre
leased, addnotes, scite1, scite2, scite3, dbsource, INT_LOG, INT_IDEO, INT_M
ISC, INT_ANY, related,
```

Cleaning the data

In [43]:

```
df=df[["iyear","imonth","iday","country_txt","region_txt","provstate","city",
       "latitude","longitude","location","summary","attacktype1_txt","targtype1_txt",
       "gname","motive","weaptype1_txt","nkill","nwound","addnotes"]]
df.head()
```

Out[43]:

	iyear	imonth	iday	country_txt	region_txt	provstate	city	latitude	longitude	loc
0	1970	7	2	Dominican Republic	Central America & Caribbean	NaN	Santo Domingo	18.456792	-69.951164	
1	1970	0	0	Mexico	North America	Federal	Mexico city	19.371887	-99.086624	
2	1970	1	0	Philippines	Southeast Asia	Tarlac	Unknown	15.478598	120.599741	
3	1970	1	0	Greece	Western Europe	Attica	Athens	37.997490	23.762728	
4	1970	1	0	Japan	East Asia	Fukouka	Fukouka	33.580412	130.396361	

In [44]:

```
df.rename(columns={"iyear":"Year","imonth":"Month","iday":"Day","country_txt":"Country",
                   "region_txt":"Region","provstate":"Province/State","city":"City",
                   "latitude":"Latitude","longitude":"Longitude","location":"Location",
                   "summary":"Summary","attacktype1_txt":"Attack Type","targtype1_txt":"Tar
                   "gname":"Group Name","motive":"Motive","weaptype1_txt":"Weapon Type",
                   "nkill":"Killed","nwound":"Wounded","addnotes":"Add Notes"},inplace=True
```

In [48]:

```
df.isnull().sum()
```

Out[48]:

```
Year                   0
Month                  0
Day                    0
Country                0
Region                 0
Province/State       421
City                 434
Latitude            4556
Longitude           4557
Location          126196
Summary            66129
Attack Type            0
Target Type            0
Group Name             0
Motive            131130
Weapon Type            0
Killed             10313
Wounded            16311
Add Notes         153402
dtype: int64
```

In [49]:

```
df["Killed"]=df["Killed"].fillna(0)
df["Wounded"]=df["Wounded"].fillna(0)
df["Casualty"]=df["Killed"]+df["Wounded"]
```

In [50]:

```
df.describe()
```

Out[50]:

	Year	Month	Day	Latitude	Longitude	Kill
count	181691.000000	181691.000000	181691.000000	177135.000000	1.771340e+05	181691.000(
mean	2002.638997	6.467277	15.505644	23.498343	-4.586957e+02	2.2668
std	13.259430	3.388303	8.814045	18.569242	2.047790e+05	11.227(
min	1970.000000	0.000000	0.000000	-53.154613	-8.618590e+07	0.000(
25%	1991.000000	4.000000	8.000000	11.510046	4.545640e+00	0.000(
50%	2009.000000	6.000000	15.000000	31.467463	4.324651e+01	0.000(
75%	2014.000000	9.000000	23.000000	34.685087	6.871033e+01	2.000(
max	2017.000000	12.000000	31.000000	74.633553	1.793667e+02	1570.000(

Observation:
1) The data consists of terrorist activities ranging from the year: 1970 to 2017
2) Maximum number of people killed in an event were: 1570
3) Maximum number of people wounded in an event were: 8191
4) Maximum number of total casualties in an event were: 9574

Visualizing the data

Number of attacks in each year

In [51]:

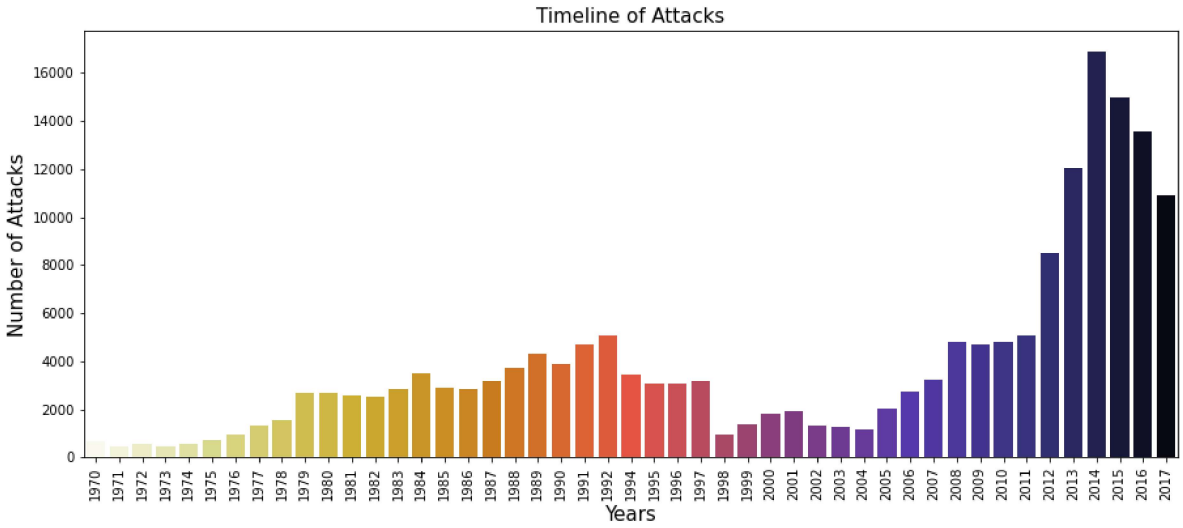
```
attacks=df["Year"].value_counts(dropna=False).sort_index().to_frame().reset_index().rename(
attacks.head()
```

Out[51]:

Year	Attacks
1970	651
1971	471
1972	568
1973	473
1974	581

In [52]:

```
plt.subplots(figsize=(15,6))
sns.countplot('Year',data=df,palette='CMRmap_r')
plt.xticks(rotation=90)
plt.title("Timeline of Attacks",fontsize=15)
plt.xlabel("Years",fontsize=15)
plt.ylabel("Number of Attacks",fontsize=15)
plt.show()
```



1) Most number of attacks (16903) in 2014
2) Least number of attacks (471) in 1971

In [18]:

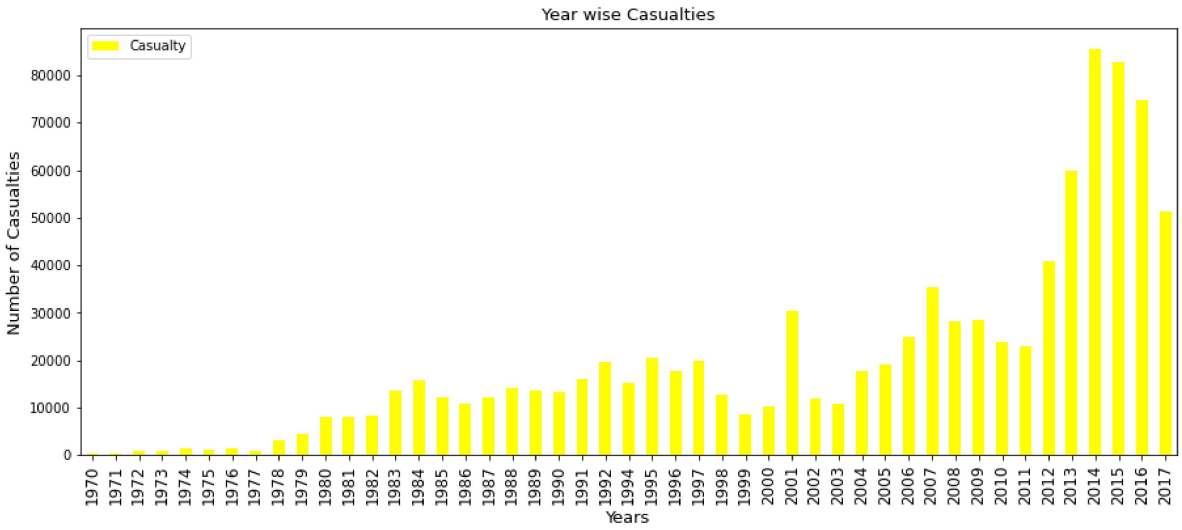
```
yc=df[["Year","Casualty"]].groupby("Year").sum()
yc.head()
```

Out[18]:

Year	Casualty
1970	386.0
1971	255.0
1972	975.0
1973	865.0
1974	1404.0

In [19]:

```
yc.plot(kind="bar",color="yellow",figsize=(15,6))
plt.title("Year wise Casualties",fontsize=13)
plt.xlabel("Years",fontsize=13)
plt.xticks(fontsize=12)
plt.ylabel("Number of Casualties",fontsize=13)
plt.show()
```



Region wise Attacks

1) Distribution of terrorist Attacks over Regions From 1970-2017

In [20]:

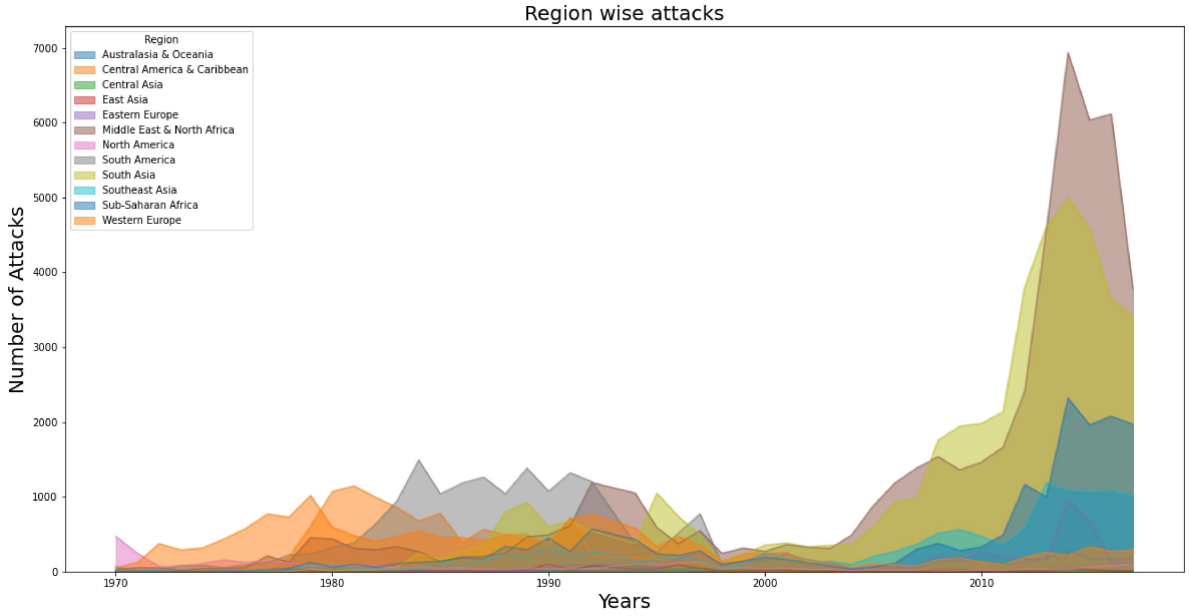
```
reg=pd.crosstab(df.Year,df.Region)
reg.head()
```

Out[20]:

Region	Australasia & Oceania	Central America & Caribbean	Central Asia	East Asia	Eastern Europe	Middle East & North Africa	North America	South America	South Asia	Sout
Year										
1970	1	7	0	2	12	28	472	65	1	
1971	1	5	0	1	5	55	247	24	0	
1972	8	3	0	0	1	53	73	33	1	
1973	1	6	0	2	1	19	64	83	1	
1974	1	11	0	4	2	42	111	81	2	

In [21]:

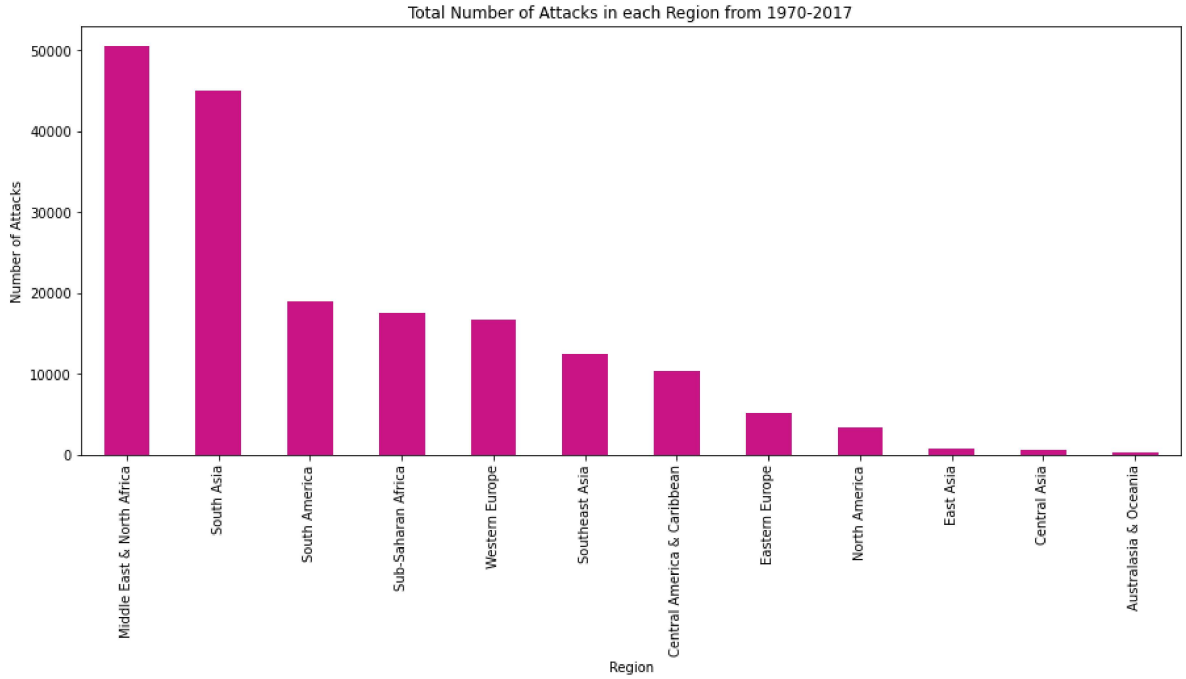
```
reg.plot(kind="area", stacked=False, alpha=0.5,figsize=(20,10))
plt.title("Region wise attacks",fontsize=20)
plt.xlabel("Years",fontsize=20)
plt.ylabel("Number of Attacks",fontsize=20)
plt.show()
```



1.Total Teerorist Attacks in each Region from 1970-2017

In [22]:

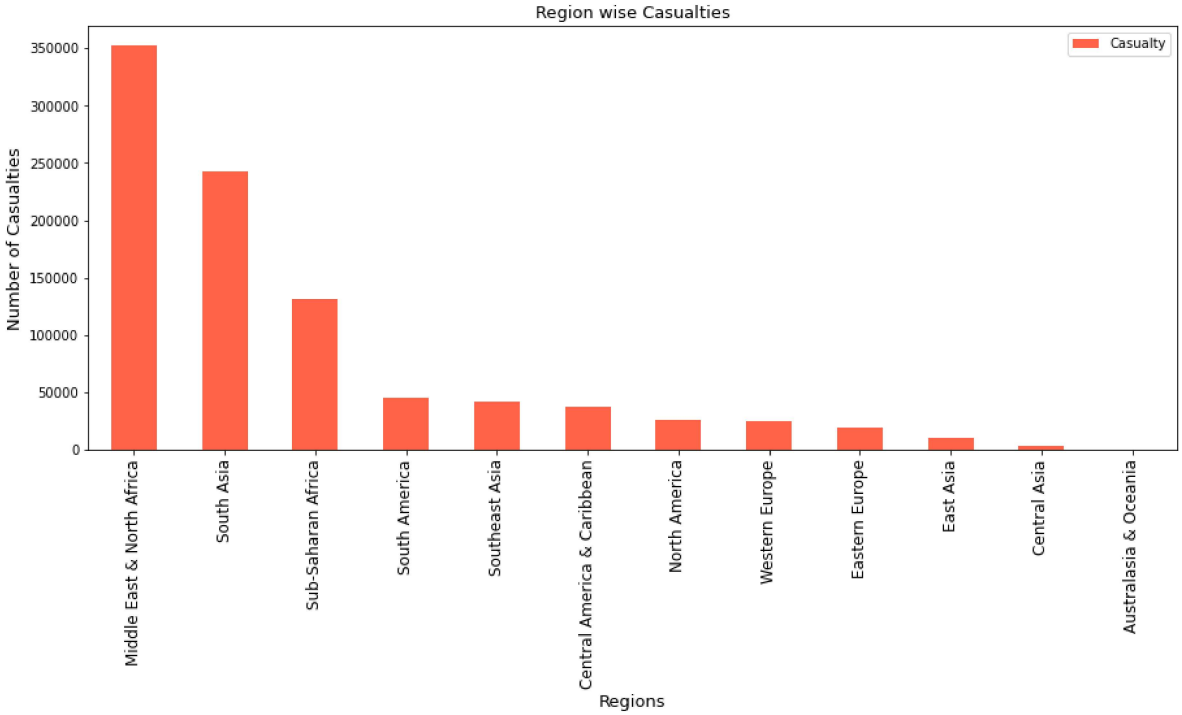
```
regt=reg.transpose()
regt["Total"]=regt.sum(axis=1)
ra=regt["Total"].sort_values(ascending=False)
ra

ra.plot(kind="bar",color='mediumvioletred',figsize=(15,6))
plt.title("Total Number of Attacks in each Region from 1970-2017")
plt.xlabel("Region")
plt.ylabel("Number of Attacks")
plt.show()
```



1) Total Casualties (Killed+Wounded) in each region

In [23]:

```
rc=df[["Region","Casualty"]].groupby("Region").sum().sort_values(by="Casualty",ascending=Fa
rc.plot(kind="bar",color="tomato",figsize=(15,6))
plt.title("Region wise Casualties",fontsize=13)
plt.xlabel("Regions",fontsize=13)
plt.xticks(fontsize=12)
plt.ylabel("Number of Casualties",fontsize=13)
plt.show()
```



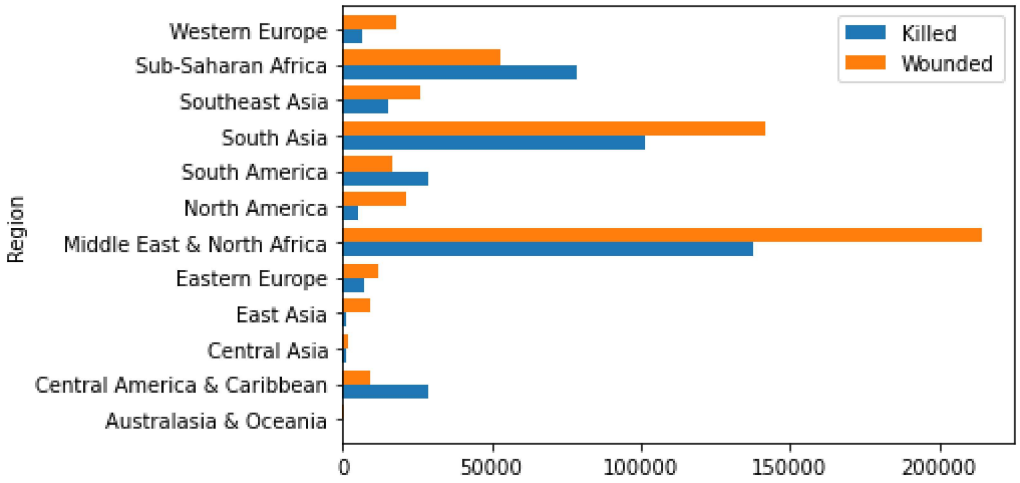
Total number of Killed and Wounded people in each region

In [24]:

```
killed_and_wounded_per_region = df.groupby('Region')[['Killed','Wounded']].sum()
killed_and_wounded_per_region.plot(kind='barh',width=0.8)
```

Out[24]:

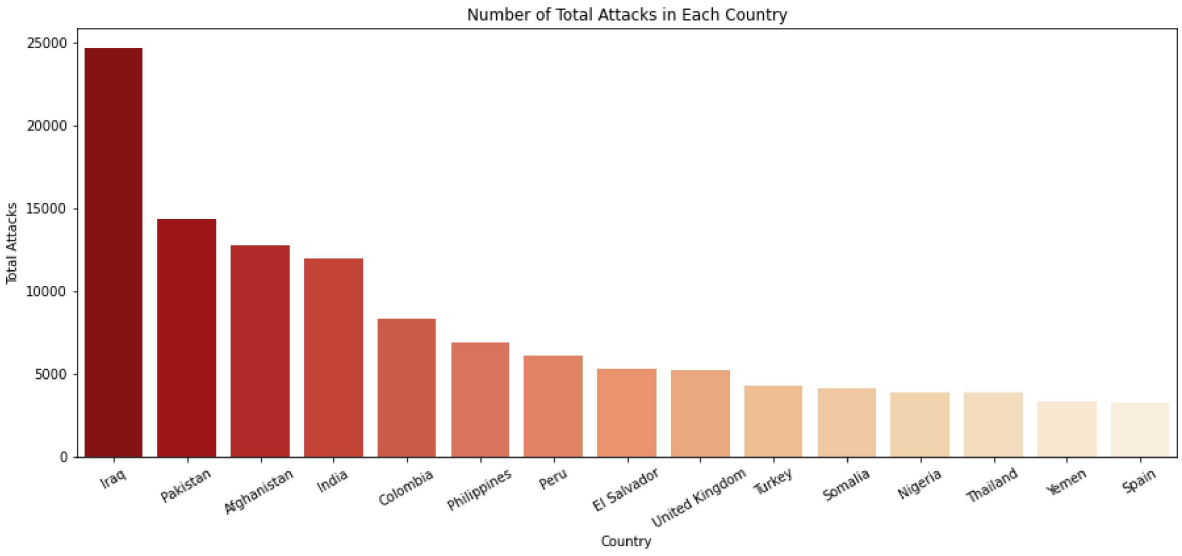
```
<AxesSubplot:ylabel='Region'>
```



Number of Total attacks in each country

In [25]:

```
plt.subplots(figsize=(15,6))
country_attacks=df.Country.value_counts()[:15].reset_index()
country_attacks.columns=['Country','Total Attacks']
sns.barplot(x=country_attacks.Country,y=country_attacks['Total Attacks'],palette='OrRd_r')
plt.xticks(rotation=30)
plt.title('Number of Total Attacks in Each Country')
plt.show()
```



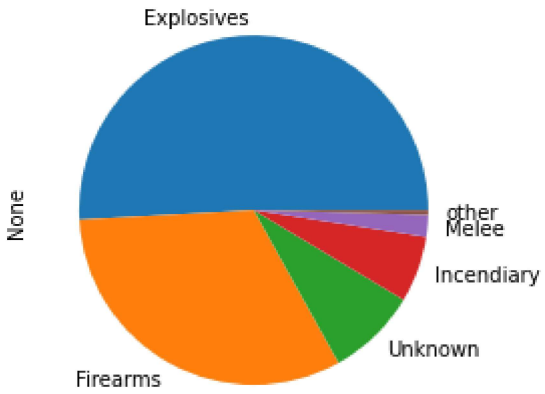
Weapons Used

In [26]:

```
weapon_series = df['Weapon Type'].value_counts()
values_to_show = 5
weapon_series = weapon_series.nlargest(values_to_show).append(pd.Series(weapon_series.nsmal
weapon_series.plot.pie()
```

Out[26]:

```
<AxesSubplot:ylabel='None'>
```



Thank You

In [ ]: