

The Sparks Foundation - GRIP JULY2021

Task4- Explotatory Data Analysis - Terrorism

Perform Explotatory Data analysis on data set 'Global terrorism'

Aurthor- Tanushree Gaur

Importing libraries

```
In [5]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

```
In [12]: df=pd.read_csv("C:\Users\Tanu.ueeschand\Downloads\Globalterrorismdb_0718dt1.csv",encoding='latin1')
df.head()
```

```
Out[12]:
```

	eventid	year	month	day	approxdate	extended	resolution	country	city	region	...	addressee	scile1	scile2	scile3	dbsource	INT_LOG	INT_IDEO	INT_MISC	INT_ANY	related	
0	197000000001	1970	7	2	0	NaN	0	NaN	58	Dominican Republic	2	...	NaN	NaN	NaN	NaN	PGIS	0	0	0	0	NaN
1	197000000002	1970	0	0	NaN	0	NaN	130	Mexico	1	...	NaN	NaN	NaN	NaN	NaN	PGIS	0	1	1	1	NaN
2	197000000003	1970	1	0	NaN	0	NaN	160	Philippines	5	...	NaN	NaN	NaN	NaN	NaN	PGIS	-9	-9	-9	1	NaN
3	197000000002	1970	1	0	NaN	0	NaN	78	Greece	8	...	NaN	NaN	NaN	NaN	NaN	PGIS	-9	-9	1	1	NaN
4	197000000003	1970	1	0	NaN	0	NaN	101	Japan	4	...	NaN	NaN	NaN	NaN	NaN	PGIS	-9	-9	1	1	NaN

5 rows × 23 columns

Checking columns

```
In [14]: df.columns
Index(['eventid', 'year', 'month', 'day', 'approxdate', 'extended', 'resolution', 'country', 'city', 'region', 'addressee', 'scile1', 'scile2', 'scile3', 'dbsource', 'INT_LOG', 'INT_IDEO', 'INT_MISC', 'INT_ANY', 'related'], dtype='object', length=23)
```

```
In [15]: df.rename(columns={'year':'Year','month':'Month','day':'Day','country_txt':'Country','provstate':'State','region_txt':'Region','attacktype_txt':'AttackType','target':'Target','rnk1':'Killed','reason':'Wounded','summary':'Summary','ignme':'Group','targettype_txt':'Target_Type','weapntype_txt':'Weapon_Type',' motive':'Motive'}, inplace=True)
```

for the better preprocessing of data we take only relevant columns

```
In [16]: df=df[['Year', 'Month', 'Day', 'Country', 'State', 'Region', 'City', 'Latitude', 'Longitude', 'AttackType', 'Killed', 'Wounded', 'Target', 'Summary', 'Group', 'Target_Type', 'Weapon_Type', 'Motive']]
```

checking for null values in data

```
In [18]: df.isnull().sum()
Year      0
Month     0
Day        0
Country    0
State      0
City       421
Region     424
Latitude   4566
Longitude  4557
AttackType 0
Killed     19313
Wounded    18311
Target     8266
Summary    66129
Group      0
Target_Type 0
Weapon_Type 0
Motive     131159
dtype: int64
```

Checking the information of the data

```
In [18]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 181691 entries, 0 to 181690
Data columns (total 23 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    177129 non-null  float64
 8   Longitude  177134 non-null  float64
 9   AttackType  177137 non-null  float64
10   Killed      171378 non-null  float64
11   Wounded     163380 non-null  float64
12   Target      181850 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      56561 non-null  object
dtypes: float64(4), int64(5), object(13)
memory usage: 25.0+ MB
```

```
In [19]: df.describe()
Out[19]:
```

	Year	Month	Day	latitude	longitude	Killed	Wounded
count	181691.000000	181691.000000	181691.000000	177135.000000	1771340e+05	173178.000000	165380.000000
mean	2002.458997	6.457277	15.505644	23.498343	-4.586957e+02	2.403272	3.167668
std	13.259430	3.368303	8.814045	18.569242	2.047790e+05	11.545741	35.846292
min	1970.000000	0.000000	0.000000	53.154613	-8.618950e+07	0.000000	0.000000
25%	1991.000000	4.000000	8.000000	11.010406	4.540504e+00	0.000000	0.000000
50%	2009.000000	6.000000	15.000000	31.467463	4.324651e+01	0.000000	0.000000
75%	2014.000000	9.000000	23.000000	34.685087	6.871033e+01	2.000000	2.000000
max	2017.000000	12.000000	31.000000	74.633953	1.793679e+02	1570.000000	8191.000000

Looking at the details of the GLOBAL TERRORISM

```
In [21]: print("Country with the most attacks:",df['Country'].value_counts().idxmax())
print("City with the most attacks:",df['City'].value_counts().idxmax())
#as first entry is unknown so we are indexing it from 1
print("Region with the most attacks:",df['Region'].value_counts().idxmax())
print("Year with the most attacks:",df['Year'].value_counts().idxmax())
print("Month with the most attacks:",df['Month'].value_counts().idxmax())
print("Group with the most attacks:",df['Group'].value_counts().idxmax())
print("Most Attack Types:",df['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

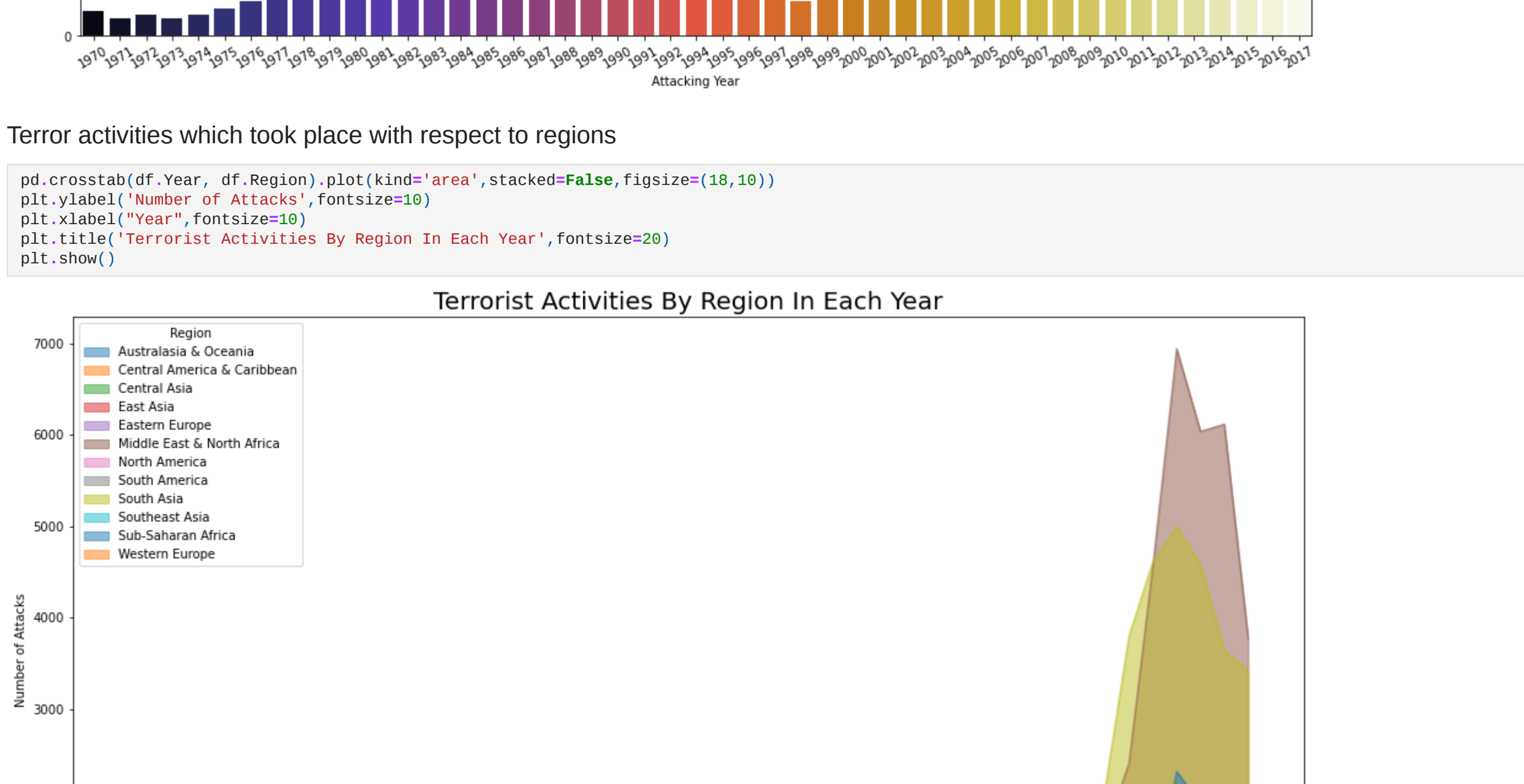
Attacks taken place with respect to Years

```
In [22]: year = df['Year'].unique()
years_count = df['Year'].value_counts(dropna = False).sort_index()
plt.figure(figsize = (18,10))
sns.barplot(x = year, y = years_count, palette = "CMRmap")
plt.xticks(rotation = 38)
plt.xlabel('Attacking Year', fontsize=10)
plt.ylabel('Number of Attacks Each Year', fontsize=18)
plt.title('Attacks In Years', fontsize=20)
plt.show()
```



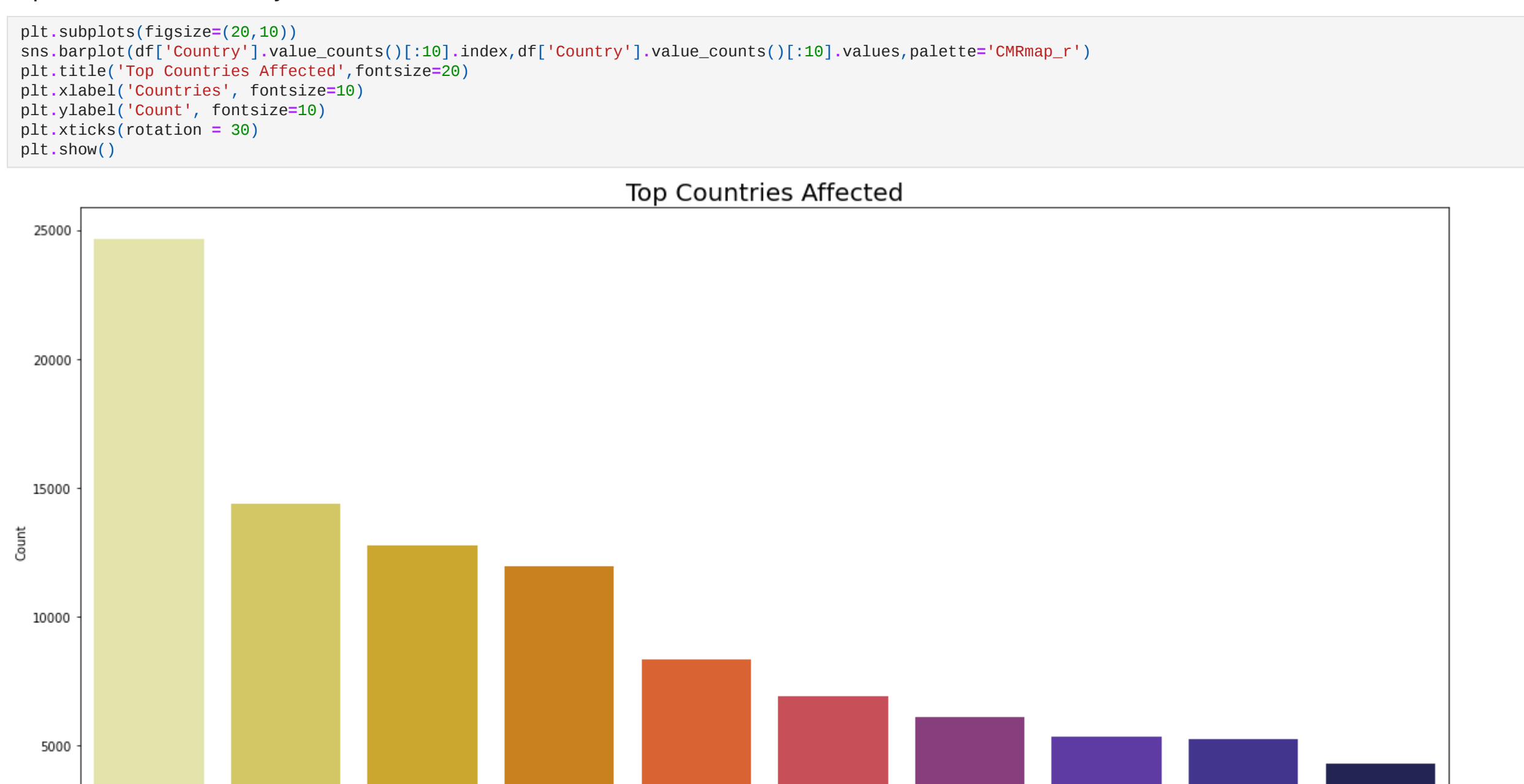
Terror activities which took place with respect to regions

```
In [23]: pd.crosstab(df.Year, df.Region, plot(kind='area', stacked=False, figsize=(18,10))
plt.xlabel('Number of Attacks', fontsize=20)
plt.ylabel('Year', fontsize=10)
plt.title('Terrorist Activities By Region In Each Year', fontsize=20)
plt.show()
```



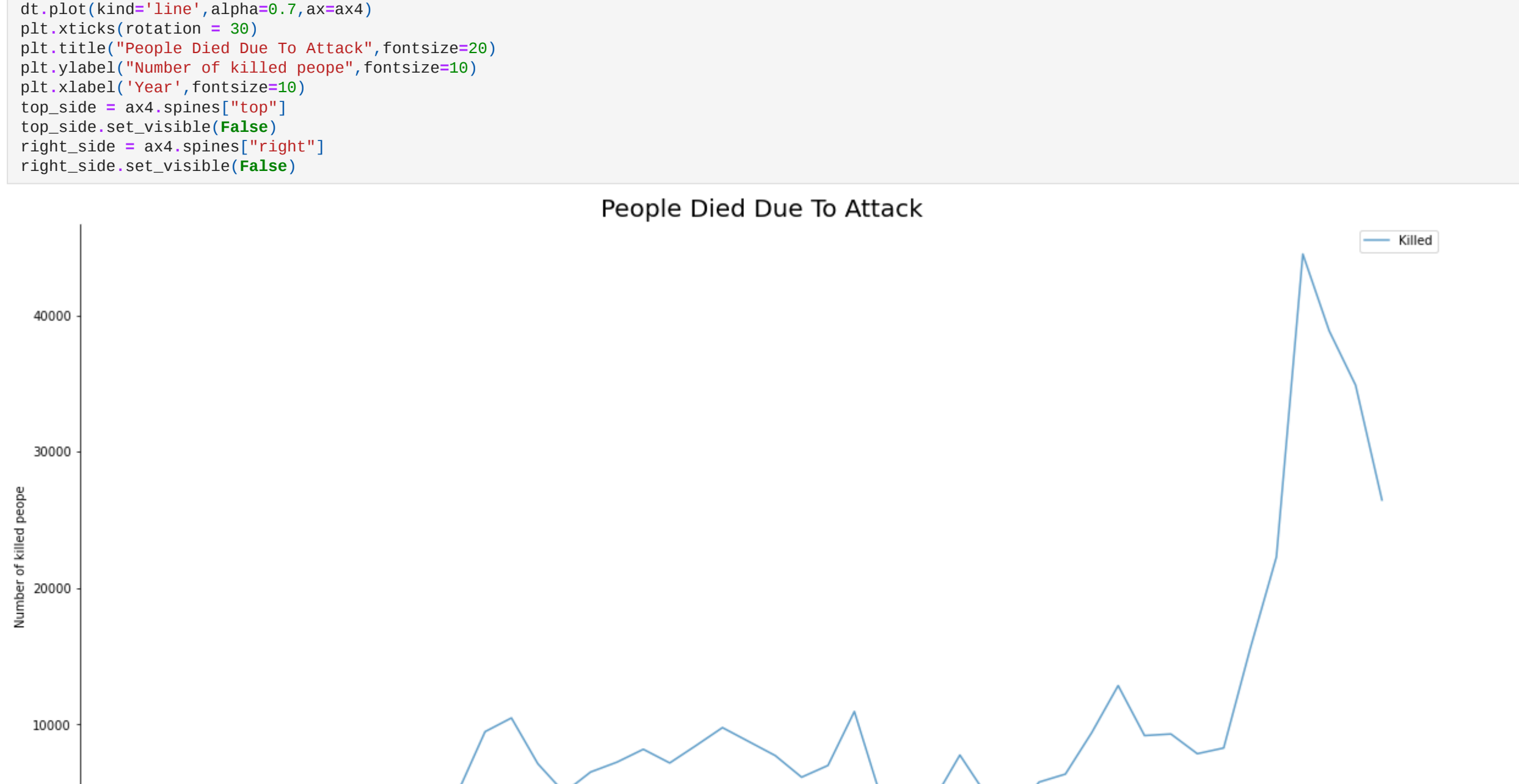
Top Countries affected by the terror attack

```
In [25]: df.subplots(figsize=(20,10))
plt.subplot(df['Country'].value_counts().idxmax().head(10).plot(kind='bar', figsize=(20,10)))
plt.title('Top Countries Affected', fontsize=20)
plt.xlabel('Countries', fontsize=10)
plt.ylabel('Count', fontsize=10)
plt.title('Number of Attacks Each Year', fontsize=18)
plt.xticks(rotation = 38)
plt.show()
```



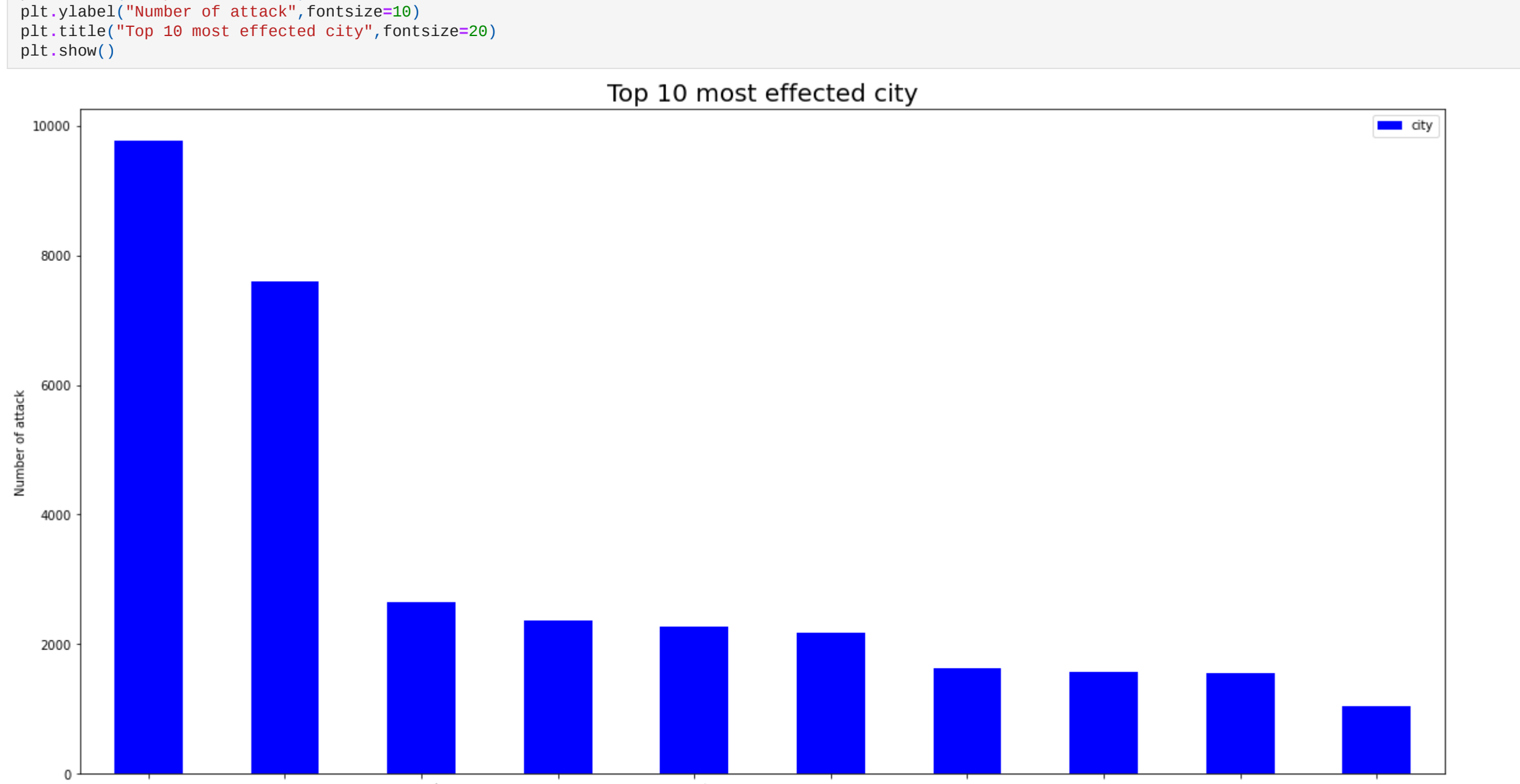
Casualties took place

```
In [26]: dc = df[['Year', 'Killed']].groupby(['Year']).sum()
fig, ax1 = plt.subplots(figsize=(20,10))
plt.plot(kind='line', linewidth=7, ax=ax1)
plt.xticks(rotation = 38)
plt.xlabel('People Died Due To Attack', fontsize=20)
plt.ylabel('Number of killed people', fontsize=18)
plt.title('Year', fontsize=10)
top_side = ax1.spines['top']
top_side.set_visible(False)
right_side = ax1.spines['right']
right_side.set_visible(False)
```



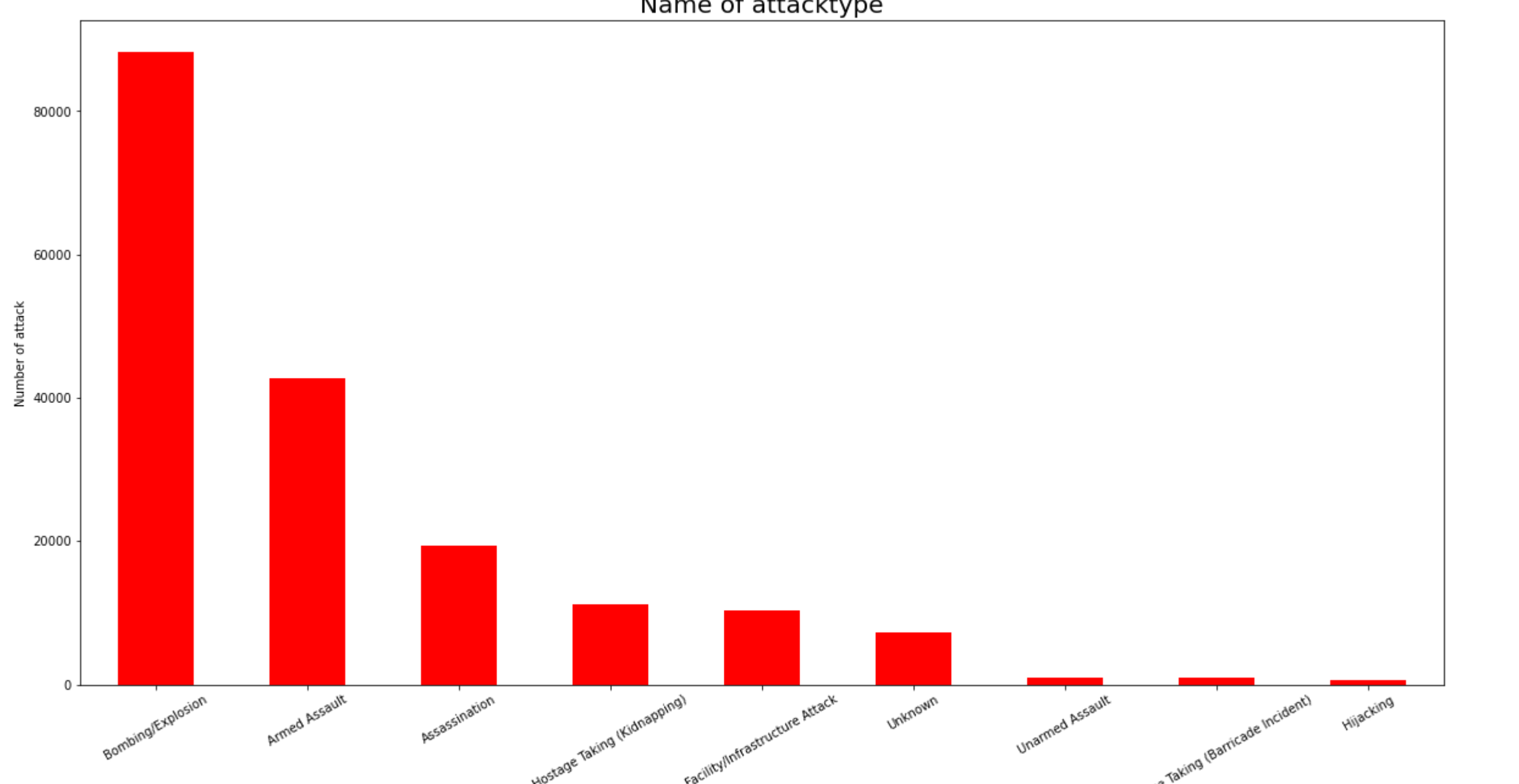
Plot of top 10 cities which got affected due to Attack

```
In [28]: df['City'].value_counts().to_frame().sort_values('City', axis=0, ascending=False).head(10).plot(kind='bar', figsize=(20,10), color='blue')
plt.xticks(rotation=90)
plt.xlabel('City', fontsize=10)
plt.ylabel('Number of attacks', fontsize=10)
plt.title('Top 10 most effected city', fontsize=20)
plt.show()
```



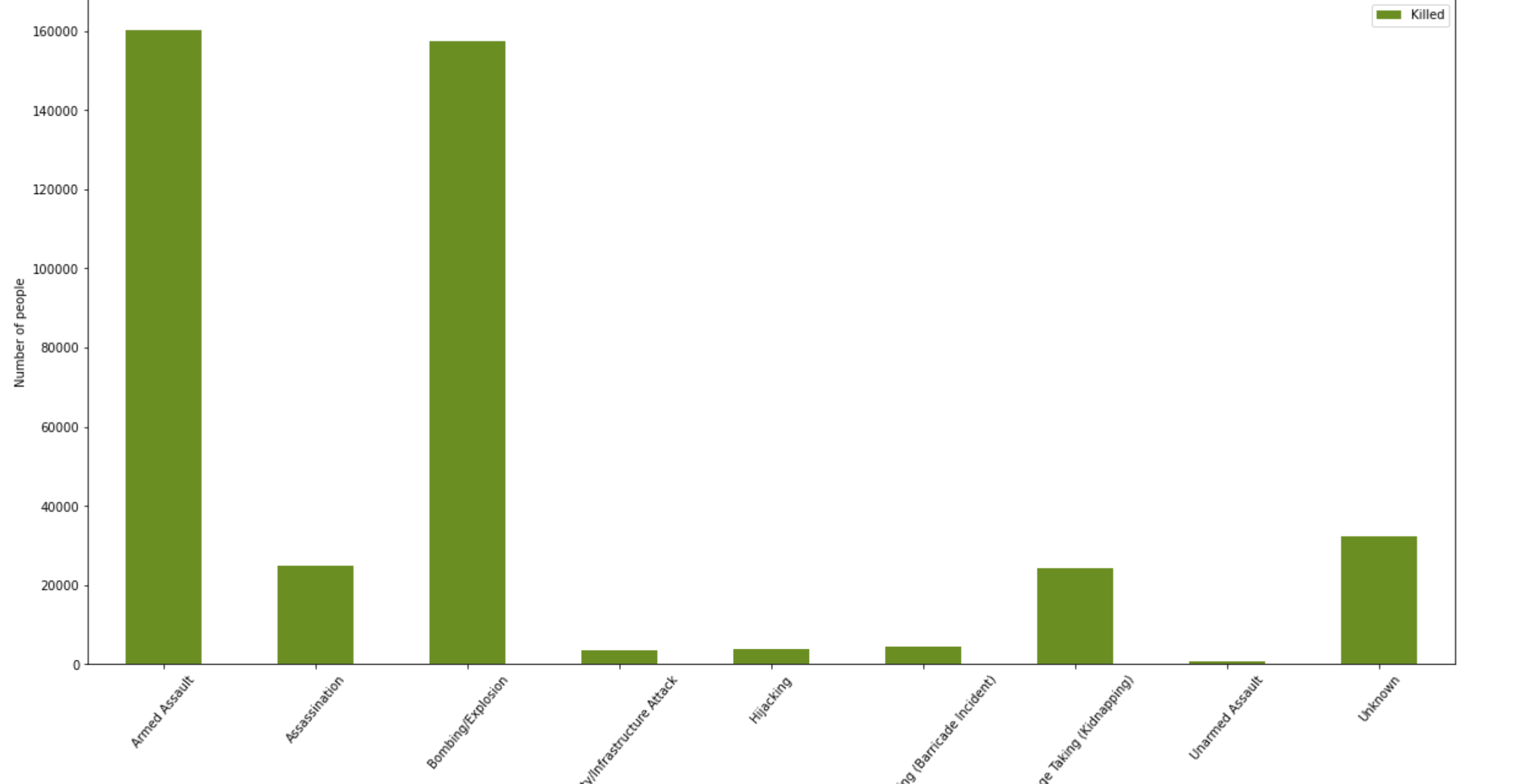
Type of Attack happened in Global terrorism

```
In [30]: df['AttackType'].value_counts().plot(kind='bar', figsize=(20,10), color='red')
plt.xticks(rotation = 38)
plt.xlabel('AttackType', fontsize=10)
plt.ylabel('Number of attack', fontsize=10)
plt.title('Name of attacktype', fontsize=20)
plt.show()
```



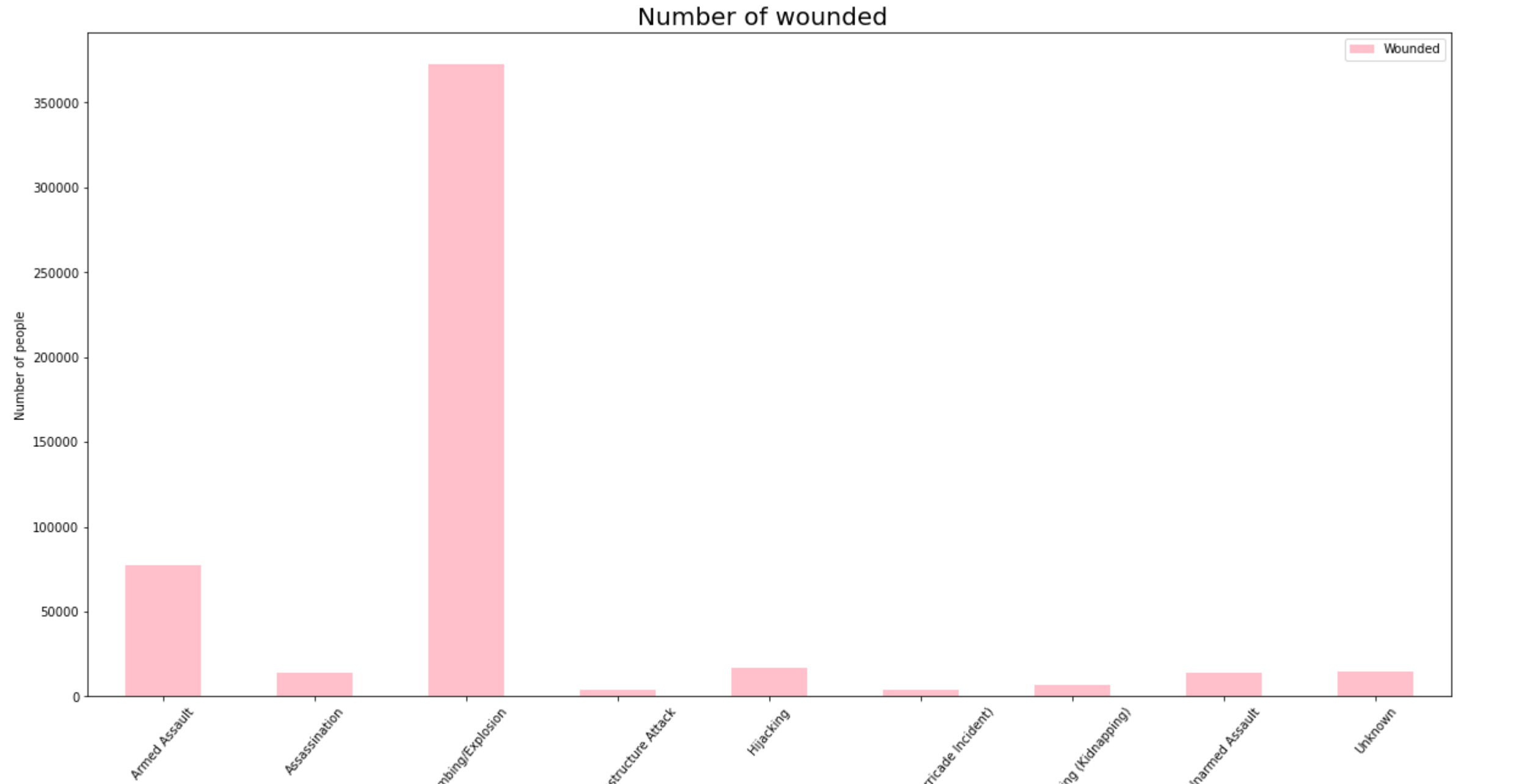
People killed in the attack

```
In [31]: df[['AttackType', 'Killed']].groupby(['AttackType'], axis=0).sum().plot(kind='bar', figsize=(20,10), color='olive')
plt.xticks(rotation=38)
plt.xlabel('Number of killed people', fontsize=20)
plt.ylabel('Attack type', fontsize=10)
plt.title('Number of killed people', fontsize=20)
plt.show()
```



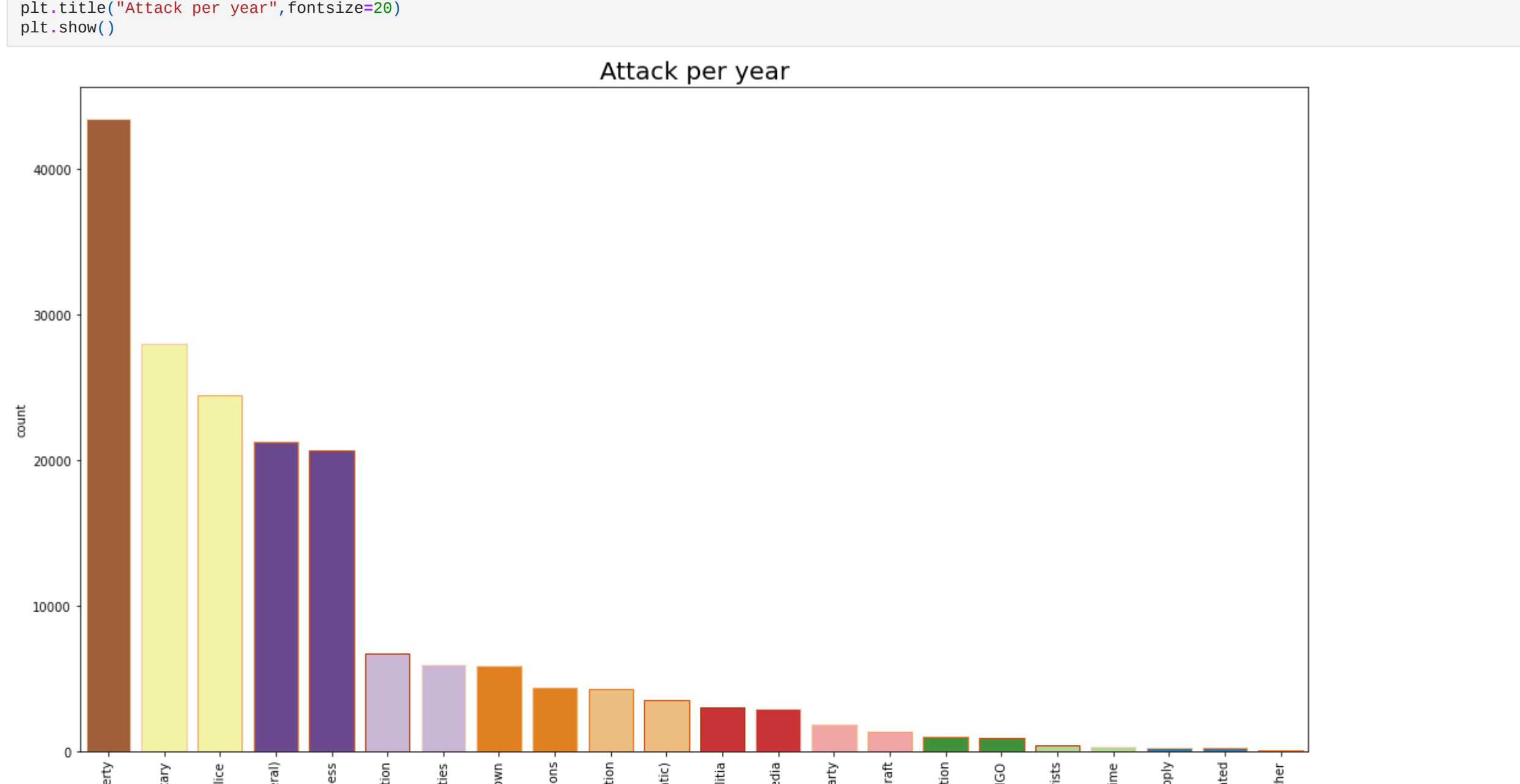
Number of people wounded due to attack with respect to attack type

```
In [33]: df[['AttackType', 'Wounded']].groupby(['AttackType'], axis=0).sum().plot(kind='bar', figsize=(20,10), color='pink')
plt.xticks(rotation=38)
plt.xlabel('Number of wounded people', fontsize=20)
plt.ylabel('Attack type', fontsize=10)
plt.title('Number of wounded people', fontsize=20)
plt.show()
```



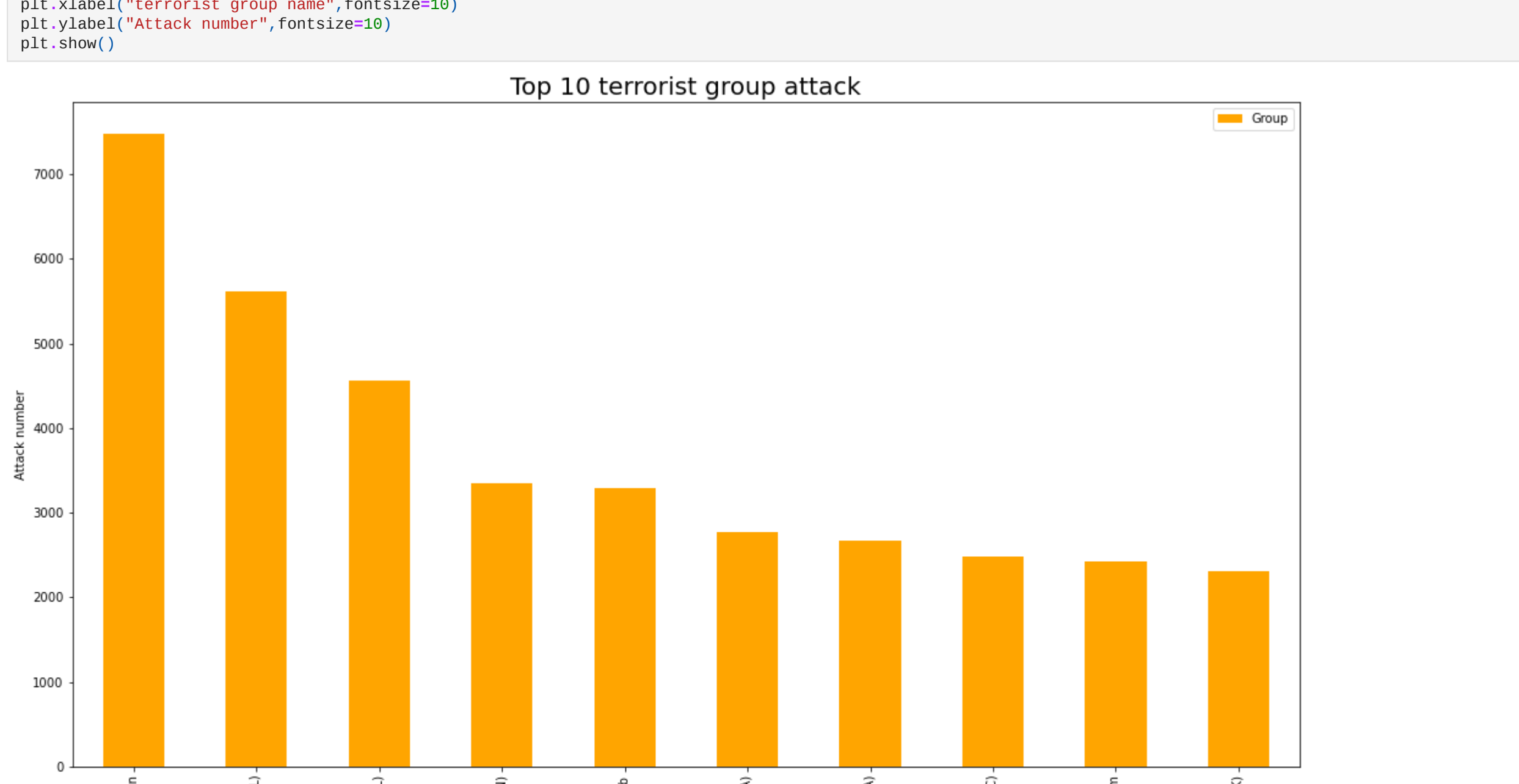
Types of Attack happened in a year

```
In [34]: sns.countplot(df['Target_Type'], order=df['Target_Type'].value_counts().index, palette='Paired_r', edgecolor=sns.color_palette('oranges'));
plt.xticks(rotation=90)
plt.xlabel('Number of people', fontsize=20)
plt.ylabel('AttackType', fontsize=10)
plt.title('Attack per year', fontsize=20)
plt.show()
```



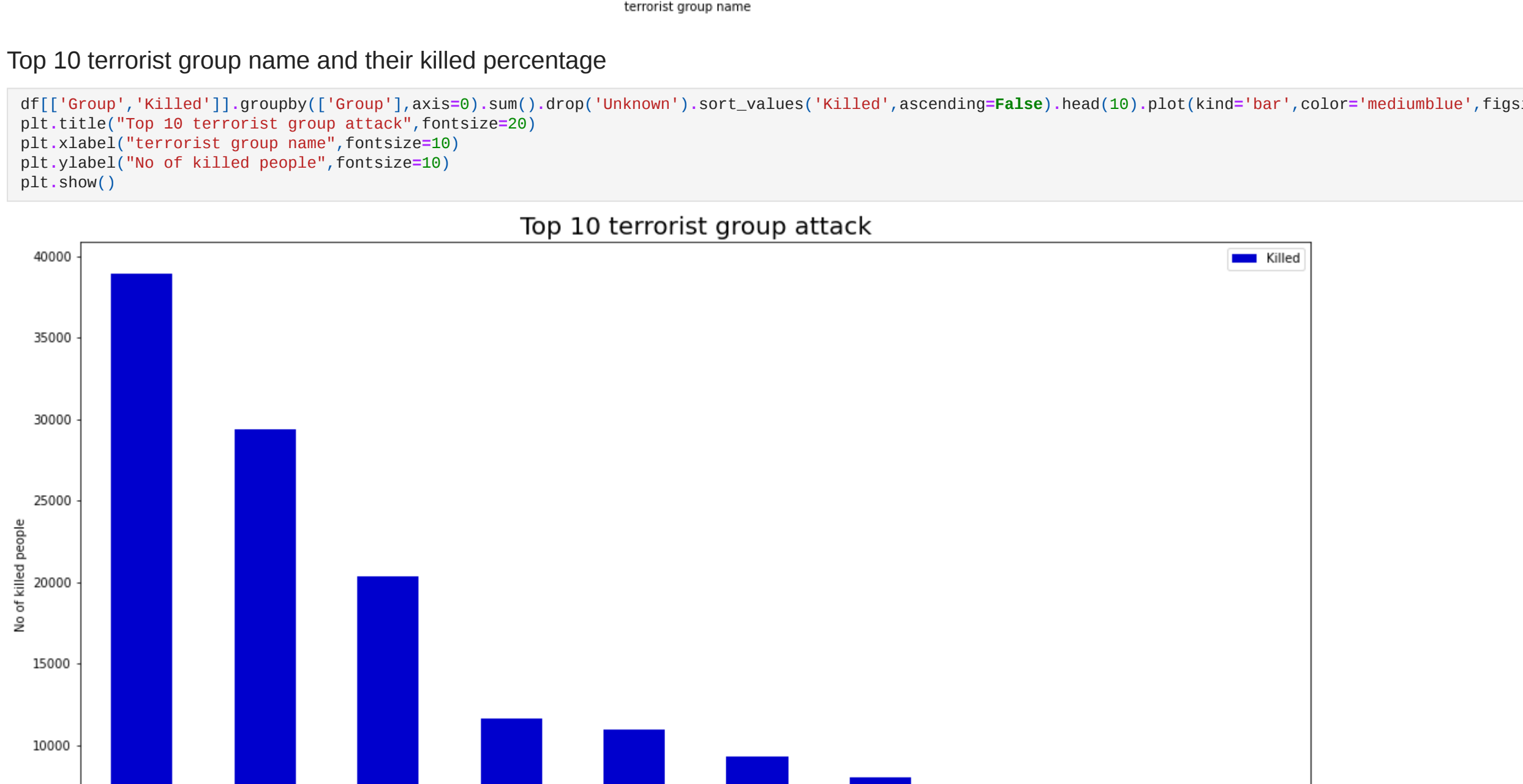
Top 10 terrorist group which supports the terrorism

```
In [36]: df['Group'].value_counts().to_frame().drop('Unknown').head(10).plot(kind='bar', colors='orange', figsize=(18,10))
plt.xticks(rotation=90)
plt.xlabel('terrorist group name', fontsize=20)
plt.ylabel('Attack number', fontsize=10)
plt.title('Attack number', fontsize=20)
plt.show()
```



Top 10 terrorist group name and their killed percentage

```
In [37]: df[['Group', 'Country', 'Killed']].groupby(['Group'], axis=0).sum().sort_values('Killed', ascending=False).head(10).plot(kind='bar', color='mediumpurple', figsize=(18,10))
plt.xticks(rotation=90)
plt.xlabel('terrorist group name', fontsize=20)
plt.ylabel('No of killed people', fontsize=10)
plt.title('No of killed people', fontsize=20)
plt.show()
```



List of the Terrorist group lead by the country and the casualty count

```
In [38]: df[['Group', 'Country', 'Killed']].groupby(['Group'], axis=0).sum().sort_values('Killed', ascending=False).drop('Unknown').reset_index().head(10)
```

	Group	Country	Killed
0	Islamic State of Iraq and the Levant (ISIL)	Iraq	31550.0
1	Boko Haram	Nigeria	29250.0
2	Shining Path (SL)	Peru	11950.0
3	Liberation Tigers of Tamil Eelam (LTTE)	Sri Lanka	10920.0
4	Al-Shabaab	Somalia	8170.0
5	Farabundo Marti National Liberation Front (FMLN)	El Salvador	8170.0
6	Islamic State of Iraq and the Levant (ISIL)	Syria	6883.0
7	Neoprogressive Democratic Force (FDN)	Nicaragua	6630.0
8	Tehrik-i-Taliban Pakistan (TTP)	Pakistan	6514.0

Total Number of people killed in terror attack

```
In [40]: kill = df.loc[:, 'Killed']
print("Number of people killed by terror attack", int(sum(kill.dropna())))
Number of people killed by terror attack: 411868
```

Total Number of people killed in terror attack with respect to Attacktype

```
In [41]: typekill = df.pivot_table(columns='AttackType', values='Killed', aggfunc='sum')
AttackType Killed Assault Assassination Bombing/Explosion Facility/Infrastructure Attack Hijacking Hostage Taking (Barricade Incident) Hostage Taking (Kidnapping) Unarmed Assault Unknown
Killed 162297.0 24920.0 157321.0 3642.0 3718.0 4478.0 2423.0 880.0 32381.0
```

Total Number of people killed in terror attack with respect to Country

```
In [42]: countrykill = df.pivot_table(columns='Country', values='Killed', aggfunc='sum')
Country Afghanistan Albania Algeria Andorra Angola Antigua and Barbuda Argentina Armenia Australia Austria ... Vietnam Wallis and Futuna West Bank and Gaza Strip Western Sahara Western Sahara Yemen Yugoslavia Zaire Zambia Zimbabwe
Killed 35894.0 42.0 11066.0 0.0 3043.0 0.0 490.0 37.0 23.0 30.0 ... 1.0 0.0 1350.0 97.0 8776.0 119.0 324.0 70.0
1 rows × 205 columns
```

Conclusion from the Evaluated data

- 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 Thank You

Tanushree Gaur

```
In [ ]:
```