Despite it prevalence in the media, mass shootings are statistically rare events, representing an availability bias, and give the impression that these events are common. Coupled with the psychological trauma and nature of the incidents being the violent loss of life, person can feel as if the world is on fire, but incidents of homicide or accidental shootings are more prevalent. What is clear based on statistical science is that mass shootings are increasing in both frequency and the number of victims. This can be constituted as a public health crisis given the scale of increase and nature of the incident in the ability of certain type of firearms to cause a significant amount of damage with a short period of time.

Mass Killings in America

In [ ]:
```
import pandas as pd, numpy as np, seaborn as sns, matplotlib.pyplot as plt, matplotlib as mlt, geopandas as gp, geodatasets
mlt.rcParams['axes.facecolor'] = "lightgray"
```

In [ ]:
```
mk_incident_public = "https://query.data.world/s/f5axjxaynrhsg2qkkpgxjzoqbdvkat?dws=00000"
mkip_data = pd.read_table(mk_incident_public, delimiter=",", header=0, skipinitialspace=True, engine="python", encoding="utf-8", enco
mkip_data.rename(str.title, axis="columns", inplace=True)
mkip_data.rename({"Incident_Id":"Incident_ID","Firstcod":"FirstCOD", "Secondcod":"SecondCOD" }, axis="columns", inplace=True)
mkip_data['FirstCOD'] = [s.title() for s in mkip_data['FirstCOD']]
mkip_data['Location_Type'] = [s.title() for s in mkip_data['Location_Type']]
```

In [ ]:
```
mkip_data.head(3)
```

Out[ ]:

	Incident_ID	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD	SecondCOD	Type	Situation_Type	Location_T
0	577	2023-06-18	Kellogg	ID	1	4	0	Shooting	NaN	Other	Interpersonal conflict	Residence/O She
1	578	2023-06-15	Sequatchie	TN	1	5	1	Shooting	Smoke inhalation & burns	Family	Family issue	Residence/O She
2	576	2023-05-27	Mesa	AZ	1	4	1	Shooting	NaN	Public	NaN	Mult

Exploratory Data Analysis

In [ ]:
```
mkip_data.tail(3)
```

Out[ ]:

	Incident_ID	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD	SecondCOD	Type	Situation_Type	Locat
556	97	2006-02-21	Mesa	AZ	1	5	0	Shooting	NaN	Felony	Other	Reside
557	109	2006-01-30	Goleta	CA	1	7	0	Shooting	NaN	Public	Employment grievance	Governme
558	98	2006-01-01	Richmond	VA	2	7	0	Stabbing	Strangulation	Felony	Robbery	Reside

In [ ]:
```
mkip_data.index
```

Out[ ]:
```
RangeIndex(start=0, stop=559, step=1)
```

In [ ]:
```
mkip_data.shape
```

Out[ ]:
```
(559, 16)
```

In [ ]:
```
mkip_data.size
```

Out[ ]:
```
8944
```

In [ ]:
```
mkip_data.ndim
```

Out[ ]:
```
2
```

In [ ]:
```
mkip_data.dtypes
```

Out[ ]:
```
Incident_ID              int64
Date                    object
City                    object
State                   object
Num_Offenders            int64
Num_Victims_Killed       int64
Num_Victims_Injured      int64
FirstCOD                object
SecondCOD               object
Type                    object
Situation_Type          object
Location_Type           object
Location                object
Longitude              float64
Latitude               float64
Narrative               object
dtype: object
```

In [ ]:
```
mkip_data.columns
```

Out[ ]:
```
Index(['Incident_ID', 'Date', 'City', 'State', 'Num_Offenders',
       'Num_Victims_Killed', 'Num_Victims_Injured', 'FirstCOD', 'SecondCOD',
       'Type', 'Situation_Type', 'Location_Type', 'Location', 'Longitude',
       'Latitude', 'Narrative'],
      dtype='object')
```

In [ ]:
```
for ele in ['FirstCOD', 'SecondCOD','Type', 'Situation_Type', 'Location_Type', 'Location']:
    print(mkip_data[ele].unique())
```

```
['Shooting' 'Smoke Inhalation & Burns' 'Stabbing' 'Vehicle Crash'
 'Asphyxiation' 'Blunt Force' 'Unknown' 'Drowning' 'Strangulation'
 'Pushing/Jumping']
[nan 'Smoke inhalation & burns' 'Stabbing' 'Asphyxiation' 'Vehicle crash'
 'Other' 'Strangulation' 'Blunt force' 'Drowning']
['Other' 'Family' 'Public' 'Suspected felony' 'Felony' 'Undetermined'
 'Unsolved']
['Interpersonal conflict' 'Family issue' nan 'Other' 'Indiscriminate'
 'Employment grievance' 'Arson' 'Drug trade' 'Despondency' 'Hate'
 'Robbery' 'Profit' 'Undetermined' 'Gang conflict' 'Terrorism']
['Residence/Other Shelter' 'Multiple' 'Commercial/Retail/Entertainment'
 'School/College' 'Open Space' 'Vehicle' 'Medical Facility'
 'House Of Worship' 'Government/Transit']
['Residence' 'Multiple' 'Commercial/Retail' 'Bar/Club/Restaurant' 'School'
 'Open space' 'Vehicle' 'Medical facility' 'Hotel/Motel'
 'House of worship' 'Government/Transit' 'Shelter/Drug house' 'College']
```

In [ ]:

```
mkip_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 559 entries, 0 to 558
Data columns (total 16 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Incident_ID          559 non-null    int64  
 1   Date                 559 non-null    object 
 2   City                 559 non-null    object 
 3   State                559 non-null    object 
 4   Num_Offenders        559 non-null    int64  
 5   Num_Victims_Killed   559 non-null    int64  
 6   Num_Victims_Injured  559 non-null    int64  
 7   FirstCOD             559 non-null    object 
 8   SecondCOD            78 non-null     object 
 9   Type                 559 non-null    object 
 10  Situation_Type       536 non-null    object 
 11  Location_Type        559 non-null    object 
 12  Location             559 non-null    object 
 13  Longitude            559 non-null    float64
 14  Latitude             559 non-null    float64
 15  Narrative            559 non-null    object 
dtypes: float64(2), int64(4), object(10)
memory usage: 70.0+ KB
```

In [ ]:

```
cat = mkip_data['FirstCOD'].unique()
c = mkip_data.groupby(mkip_data['FirstCOD']).FirstCOD.count()
duplicates = set()
# print(sum(1 for item in cat if all(item in mkip_data['FirstCod'])))
sum(1 for cat in mkip_data['FirstCOD'])
```

Out[ ]: 559

In [ ]:

```
mkip_data.select_dtypes(include="object").nunique()
```

Out[ ]:

```
Date              533
City              429
State              48
FirstCOD           10
SecondCOD           8
Type                7
Situation_Type     14
Location_Type       9
Location           13
Narrative         559
dtype: int64
```

In [ ]:

```
if not(any(mkip_data.notna())) or not(any(mkip_data.notnull())):
    print("All Good")
else:
    print("Work to be Done")
```

```
Work to be Done
```

In [ ]:

```
100 * mkip_data.isnull().sum() / mkip_data.shape[0]
```

Out[ ]:

```
Incident_ID              0.000000
Date                     0.000000
City                     0.000000
State                    0.000000
Num_Offenders            0.000000
Num_Victims_Killed       0.000000
Num_Victims_Injured      0.000000
FirstCOD                 0.000000
SecondCOD               86.046512
Type                     0.000000
Situation_Type           4.114490
Location_Type            0.000000
Location                 0.000000
Longitude                0.000000
Latitude                 0.000000
Narrative                0.000000
dtype: float64
```

Data Analysis and Visualization

In [ ]:

```python
mkip_data[["Num_Offenders","Num_Victims_Killed","Num_Victims_Injured"]].describe().round()
```

Out[ ]:

	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured
count	559.0	559.0	559.0
mean	1.0	5.0	4.0
std	1.0	4.0	37.0
min	0.0	4.0	0.0
25%	1.0	4.0	0.0
50%	1.0	4.0	0.0
75%	1.0	5.0	1.0
max	7.0	60.0	867.0

In [ ]:

```python
print("Total Number of Mass Violence Incidents: ",  mkip_data.Incident_ID.sum())
print("Total Number of Victims: ",  mkip_data.Num_Victims_Killed.sum())
print("Total Number of Victims: ",  mkip_data.Num_Victims_Injured.sum())
```

```
Total Number of Mass Violence Incidents:  158612
Total Number of Victims:  2905
Total Number of Victims:  2002
```

In [ ]:

```python
mkip_data.groupby("Date", axis=0).filter(lambda x: (x.nunique() > 1).any()).iloc[:, 1:8]
# mkip_data["Same_Day"] = mkip_data.groupby("Date", axis=0).nunique()
```

Out[ ]:

	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD
6	2023-04-30	Mojave	CA	1	4	0	Shooting
7	2023-04-30	Henryetta	OK	1	6	0	Shooting
30	2022-11-20	Colorado Springs	CO	1	5	17	Shooting
31	2022-11-20	Hennessey	OK	1	4	1	Shooting
35	2022-11-04	Orlando	FL	1	4	1	Shooting
36	2022-11-04	La Plata	MD	1	4	0	Shooting
40	2022-10-17	Woodbridge	VA	1	4	0	Shooting
41	2022-10-17	South Fulton	GA	1	4	5	Smoke Inhalation & Burns
43	2022-10-09	Inman	SC	1	5	0	Shooting
44	2022-10-09	Henryetta	OK	1	4	0	Shooting
60	2022-05-14	Buffalo	NY	1	10	3	Shooting
61	2022-05-14	Worcester	MA	1	4	3	Smoke Inhalation & Burns
68	2022-01-23	Milwaukee	WI	1	6	0	Shooting
69	2022-01-23	Inglewood	CA	1	4	1	Shooting
81	2021-09-05	Lakeland	FL	1	4	2	Shooting
82	2021-09-05	Houston	TX	1	4	0	Shooting
119	2020-06-04	San Antonio	TX	1	5	0	Asphyxiation
120	2020-06-04	Valhermoso Springs	AL	2	7	0	Shooting
123	2020-03-15	Moncure	NC	1	6	0	Shooting
124	2020-03-15	Springfield	MO	1	4	2	Shooting
153	2019-07-06	Port Angeles	WA	1	4	0	Smoke Inhalation & Burns
154	2019-07-06	North St. Louis County	MO	2	5	0	Shooting
199	2018-01-28	Melcroft	PA	1	4	1	Shooting
200	2018-01-28	Reading	PA	4	4	0	Shooting
222	2017-04-07	Houston	TX	1	4	0	Shooting
223	2017-04-07	Columbia	SC	2	4	0	Blunt Force
225	2017-03-22	Rothschild	WI	1	4	0	Shooting
226	2017-03-22	Sacramento	CA	1	4	0	Blunt Force
256	2016-04-22	Appling	GA	1	5	0	Shooting
257	2016-04-22	Piketon	OH	4	8	0	Shooting
280	2015-07-22	Broken Arrow	OK	2	5	1	Stabbing
281	2015-07-22	Suwanee	GA	1	4	0	Shooting
282	2015-07-18	Modesto	CA	1	5	0	Stabbing
283	2015-07-18	Chicago	IL	1	4	0	Smoke Inhalation & Burns
320	2014-02-20	Alturas	CA	1	4	2	Shooting
321	2014-02-20	Indianapolis	IN	4	4	0	Shooting
325	2013-11-23	Parsons	KS	1	4	0	Strangulation
326	2013-11-23	Tulsa	OK	1	4	1	Shooting
338	2013-07-26	Clarksburg	WV	1	4	0	Shooting
339	2013-07-26	Hialeah	FL	1	6	0	Shooting
388	2011-07-07	Wheatland	WY	1	4	1	Shooting
389	2011-07-07	Grand Rapids	MI	1	7	2	Shooting
397	2011-02-11	Brooklyn	NY	1	4	5	Stabbing
398	2011-02-11	Willowbrook	CA	1	4	0	Shooting
446	2009-03-29	Carthage	NC	1	8	2	Shooting
447	2009-03-29	Santa Clara	CA	1	5	1	Shooting
487	2008-02-07	Kirkwood	MO	1	6	1	Shooting
488	2008-02-07	Los Angeles	CA	1	4	1	Shooting
489	2008-02-02	Tinley Park	IL	1	5	1	Shooting
490	2008-02-02	Cockeysville	MD	1	4	0	Shooting

	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD
529	2006-10-14	Kansas City	KS	2	4	0	Shooting
530	2006-10-14	Bonaparte	IA	1	5	0	Shooting

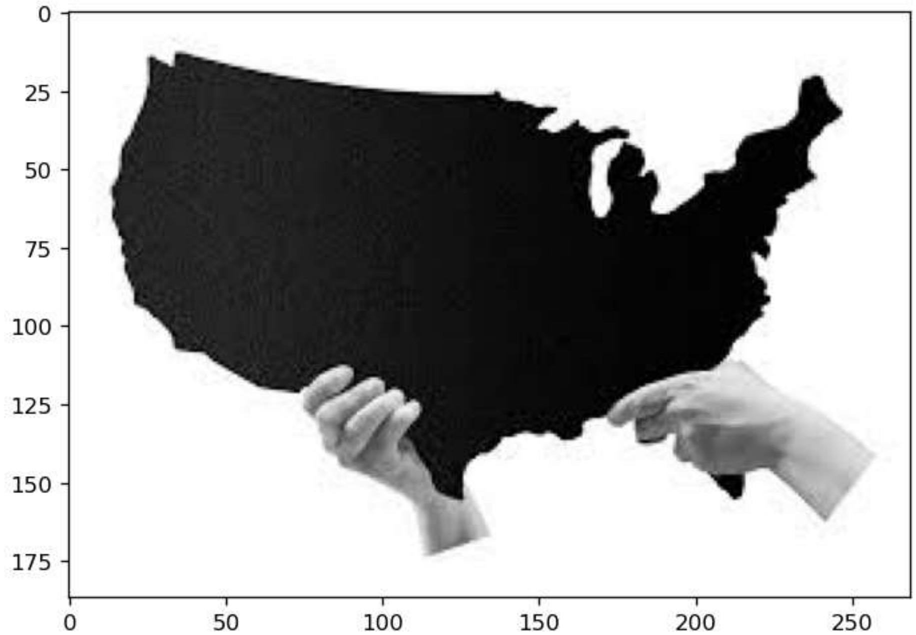
Mass Violence by State

In [ ]:
```
state_count = mkip_data.groupby(["State"]).Incident_ID.count().sort_values(ascending=False)
state_num_victims = mkip_data.groupby(["State"]).Num_Victims_Killed.sum().sort_values(ascending=False)
print(state_num_victims)
```

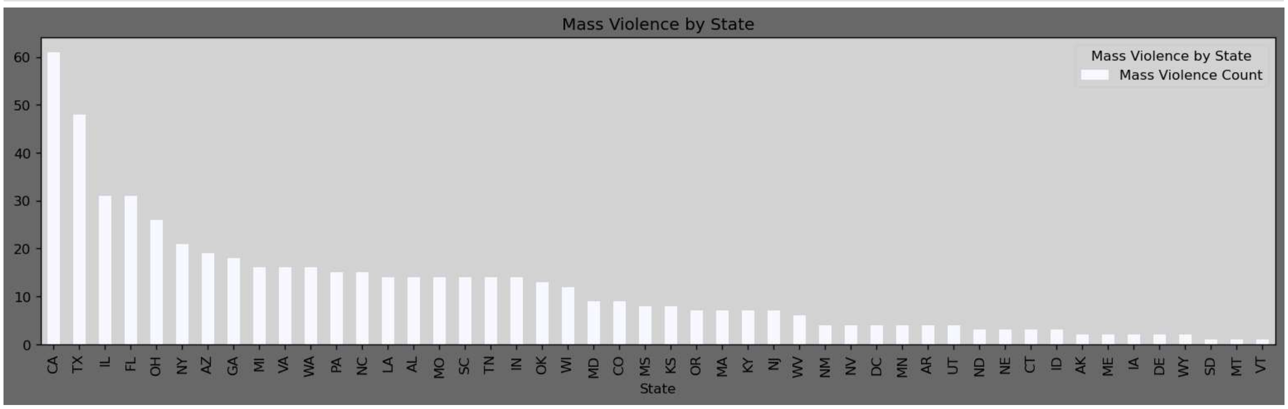
```
State
CA    320
TX    301
FL    196
IL    149
OH    123
VA    110
NY    107
GA     85
AZ     83
NV     80
PA     76
WA     75
MI     70
AL     69
TN     68
NC     68
SC     67
IN     65
MO     63
OK     61
WI     61
LA     60
CO     56
CT     40
MS     38
MD     38
OR     35
KS     35
KY     31
NJ     28
MA     28
WV     26
DC     24
UT     20
NM     19
MN     18
AR     17
NE     16
ID     12
ND     12
IA     10
AK      8
ME      8
DE      8
WY      8
SD      5
VT      4
MT      4
Name: Num_Victims_Killed, dtype: int64
```

In [ ]:
```
from PIL import Image
plt.figure(figsize=(8,4), constrained_layout=True, dpi=120)
plt.imshow(np.asarray(Image.open('images.png')))
```

Out[ ]: `<matplotlib.image.AxesImage at 0x24ee57bf4c0>`



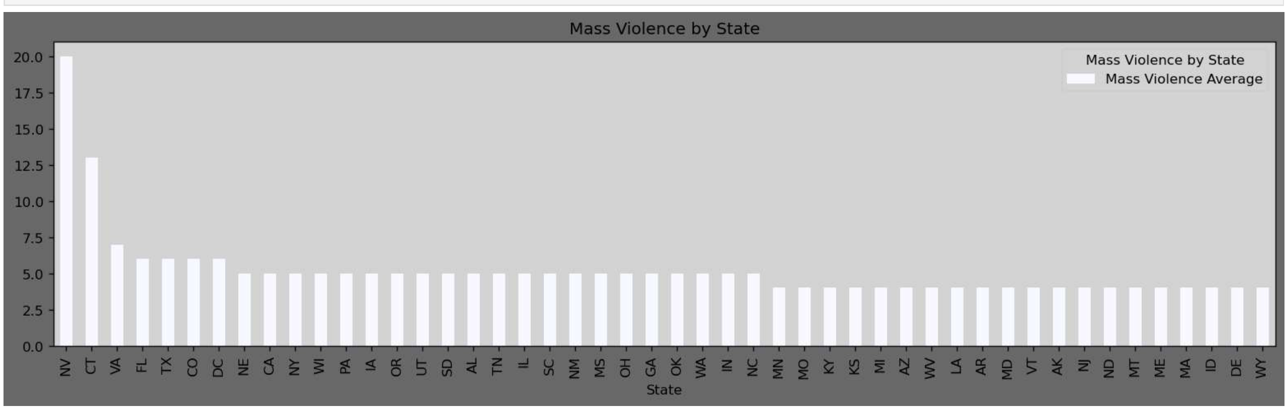
In [ ]:
```
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by State")
state_count.plot.bar(color="ghostwhite")
axe.legend(labels=["Mass Violence Count"], loc="upper right", mode="none", title="Mass Violence by State")
axe.set_xlabel("State")
axe.set_title("Mass Violence by State")
axe.plot();
```



In [ ]:
```
state_avg_victims = mkip_data.groupby(["State"]).Num_Victims_Killed.mean().sort_values(ascending=False).round()
state_avg_victims
```

Out[ ]:
```
State
NV    20.0
CT    13.0
VA     7.0
FL     6.0
TX     6.0
CO     6.0
DC     6.0
NE     5.0
CA     5.0
NY     5.0
WI     5.0
PA     5.0
IA     5.0
OR     5.0
UT     5.0
SD     5.0
AL     5.0
TN     5.0
IL     5.0
SC     5.0
NM     5.0
MS     5.0
OH     5.0
GA     5.0
OK     5.0
WA     5.0
IN     5.0
NC     5.0
MN     4.0
MO     4.0
KY     4.0
KS     4.0
MI     4.0
AZ     4.0
WV     4.0
LA     4.0
AR     4.0
MD     4.0
VT     4.0
AK     4.0
NJ     4.0
ND     4.0
MT     4.0
ME     4.0
MA     4.0
ID     4.0
DE     4.0
WY     4.0
Name: Num_Victims_Killed, dtype: float64
```

In [ ]:
```
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by State")
# state_avg_victims.plot.line(color="black")
state_avg_victims.plot.bar(color="ghostwhite")
axe.legend(labels=["Mass Violence Average"], loc="upper right", mode="none", title="Mass Violence by State")
axe.set_xlabel("State")
axe.set_title("Mass Violence by State")
axe.plot();
```

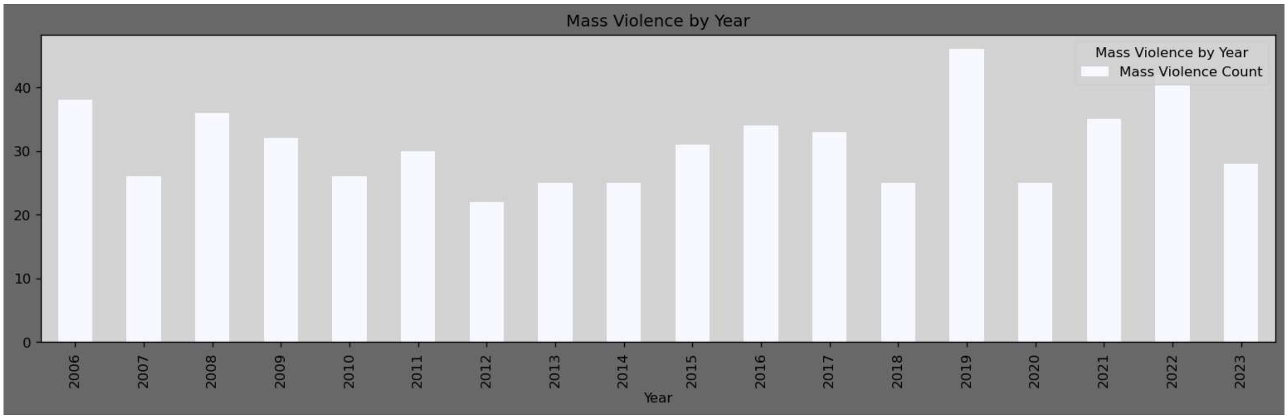


Mass Violence by Year

In [ ]:
```
mkip_data["Date_Year"] = pd.to_datetime(mkip_data["Date"], errors="coerce", yearfirst=True).dt.year
date_count = mkip_data.groupby(["Date_Year"]).Incident_ID.count()
print(date_count)
```

```
Date_Year
2006    38
2007    26
2008    36
2009    32
2010    26
2011    30
2012    22
2013    25
2014    25
2015    31
2016    34
2017    33
2018    25
2019    46
2020    25
2021    35
2022    42
2023    28
Name: Incident_ID, dtype: int64
```

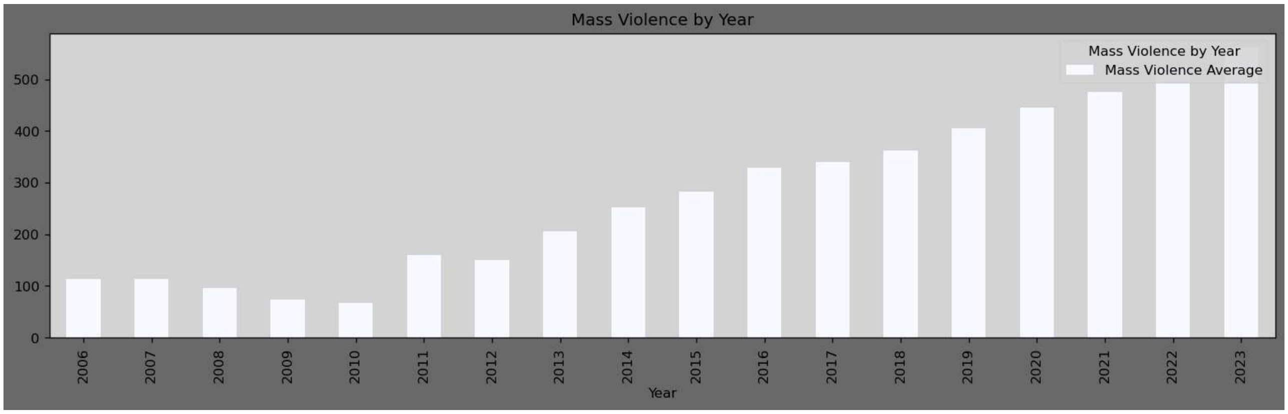
In [ ]:
```
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by Year")
date_count.plot.bar(color="ghostwhite")
axe.legend(labels=["Mass Violence Count"], loc="upper right", mode="none", title="Mass Violence by Year")
axe.set_xlabel("Year")
axe.set_title("Mass Violence by Year")
axe.plot();
```



In [ ]:
```
mkip_data["Date_Year"] = pd.to_datetime(mkip_data["Date"], errors="coerce", yearfirst=True).dt.year
date_mean = mkip_data.groupby(["Date_Year"]).Incident_ID.mean().round(1)
print(date_mean)
```

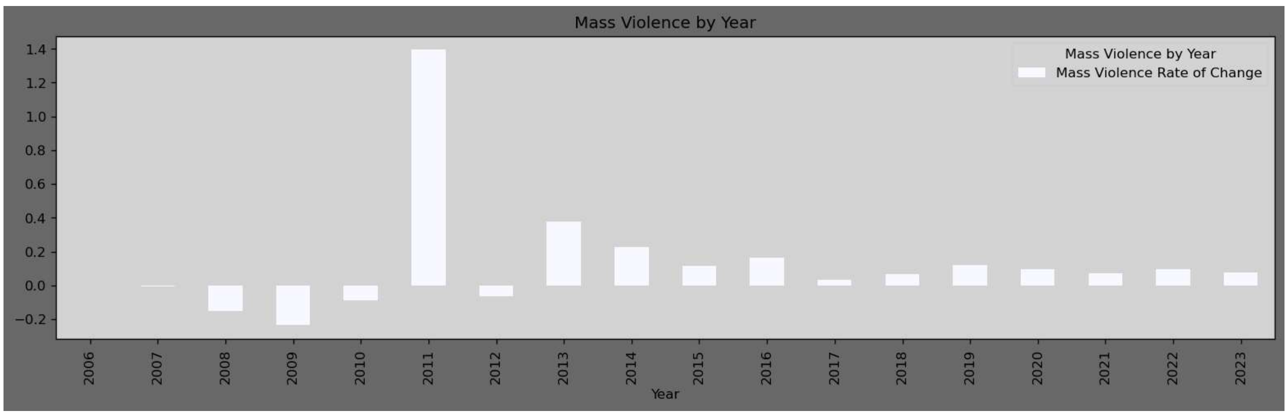
```
Date_Year
2006    113.3
2007    112.9
2008     96.0
2009     73.4
2010     66.8
2011    159.9
2012    149.8
2013    206.2
2014    253.0
2015    282.0
2016    328.9
2017    339.5
2018    362.0
2019    405.8
2020    444.4
2021    476.1
2022    522.0
2023    561.3
Name: Incident_ID, dtype: float64
```

In [ ]:
```
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by Year")
date_mean.plot.bar(color="ghostwhite")
axe.legend(labels=["Mass Violence Average"], loc="upper right", mode="none", title="Mass Violence by Year")
axe.set_xlabel("Year")
axe.set_title("Mass Violence by Year")
axe.plot();
```



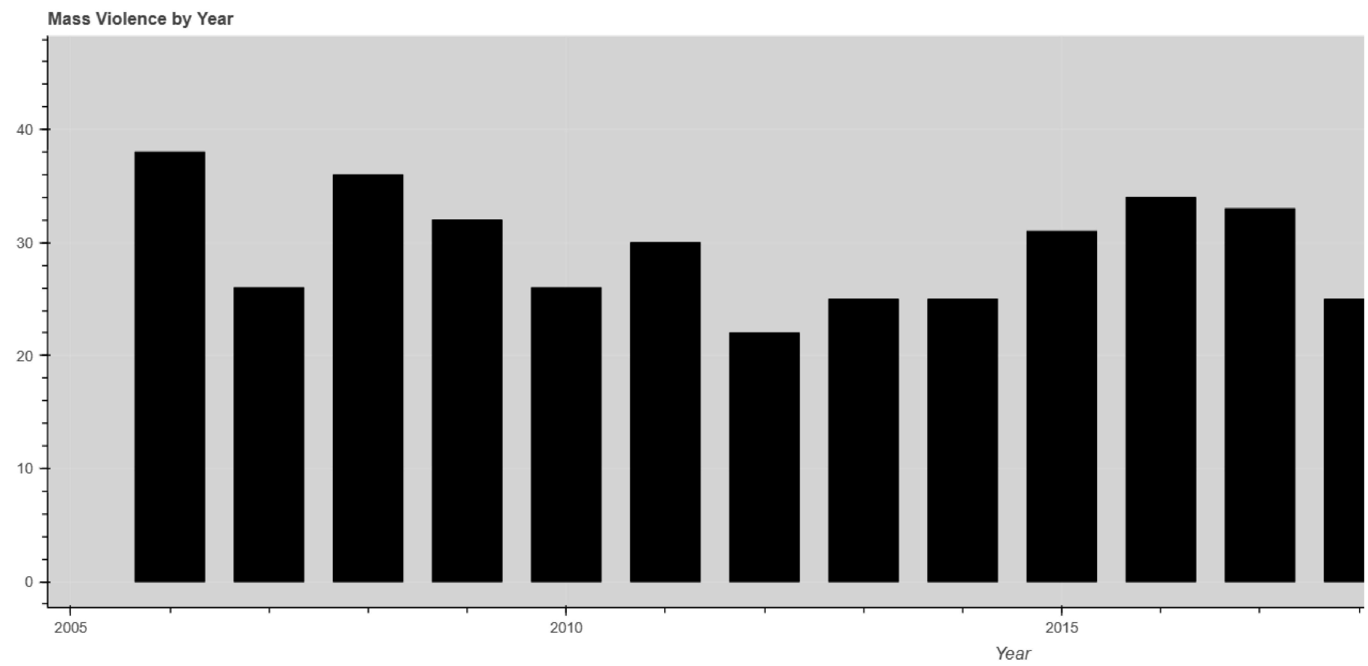
In [ ]:
```python
roc_mean = date_mean.pct_change()
```

In [ ]:
```python
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by Year")
roc_mean.plot.bar(color="ghostwhite")
axe.legend(labels=["Mass Violence Rate of Change"], loc="upper right", mode="none", title="Mass Violence by Year")
axe.set_xlabel("Year")
axe.set_title("Mass Violence by Year")
axe.plot();
```



In [ ]:
```python
from bokeh.plotting import figure
from bokeh.io import output_notebook, show
output_notebook()

fig = figure(width=1500, height=500,  title="Mass Violence by Year")
fig.grid.grid_line_alpha=0.3
fig.xaxis.axis_label="Year"
fig.background_fill_color = "lightgray"
fig.vbar(x=date_count.index, top=date_count.values, color="black", width=0.7)

show(fig)
```

BokehJS 2.4.3 successfully loaded.

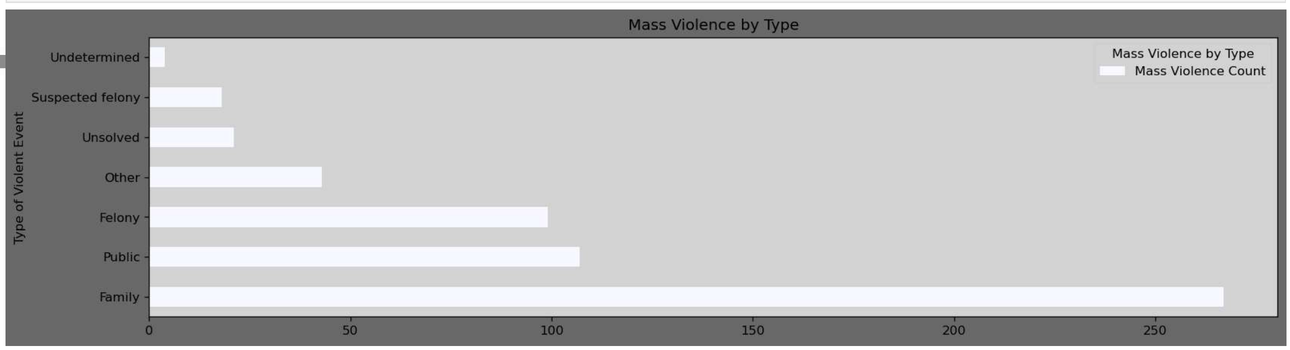


Mass Shoots by Type

In [ ]:
```
type_count= mkip_data.groupby(["Type"]).Incident_ID.count().sort_values(ascending=False)
print(type_count)
```

```
Type
Family              267
Public              107
Felony               99
Other                43
Unsolved             21
Suspected felony     18
Undetermined          4
Name: Incident_ID, dtype: int64
```

In [ ]:
```
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by Type")
type_count.plot.barh(color="ghostwhite")
axe.legend(labels=["Mass Violence Count"], loc="upper right", mode="none", title="Mass Violence by Type")
axe.set_ylabel("Type of Violent Event")
axe.set_title("Mass Violence by Type")
axe.plot();
```

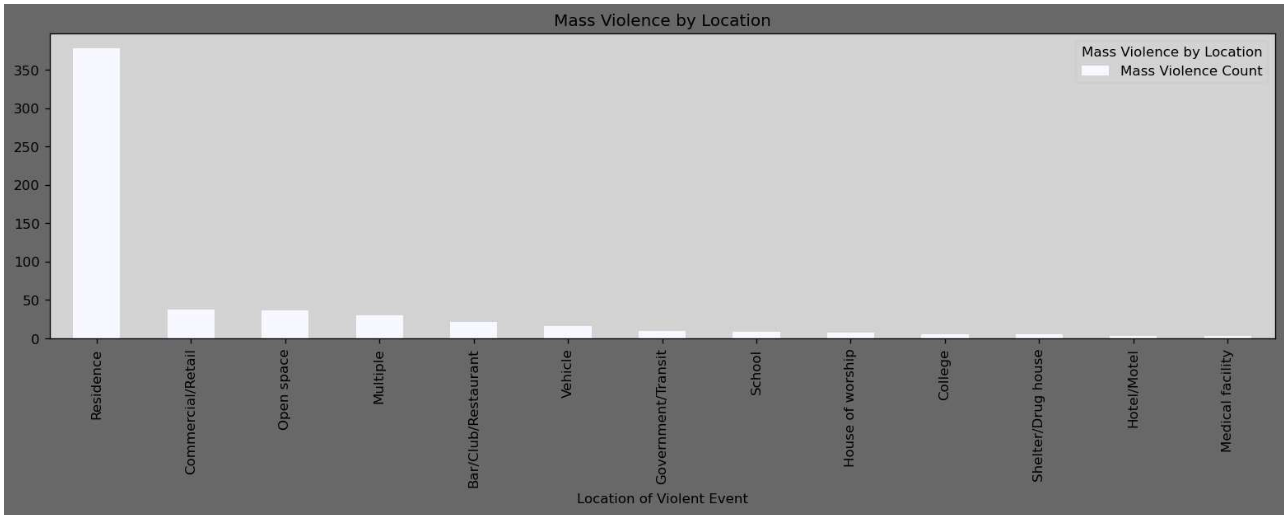


Mass Violence by Location

In [ ]:
```
location_count = mkip_data.groupby(["Location"]).Location.count().sort_values(ascending=False)
print(location_count)
```

```
Location
Residence              378
Commercial/Retail       37
Open space              36
Multiple                30
Bar/Club/Restaurant     21
Vehicle                 16
Government/Transit      10
School                   8
House of worship         7
College                  5
Shelter/Drug house       5
Hotel/Motel              3
Medical facility         3
Name: Location, dtype: int64
```

In [ ]:
```
fig, axe = plt.subplots(figsize=(16,4), constrained_layout=False, dpi=120, facecolor="dimgray", label="Mass Violence by Location")
location_count.plot.bar (color="ghostwhite")
axe.legend(labels=["Mass Violence Count"], loc="upper right", mode="none", title="Mass Violence by Location")
axe.set_xlabel("Location of Violent Event")
axe.set_title("Mass Violence by Location")
axe.plot();
```



Mass Violence by Location Type

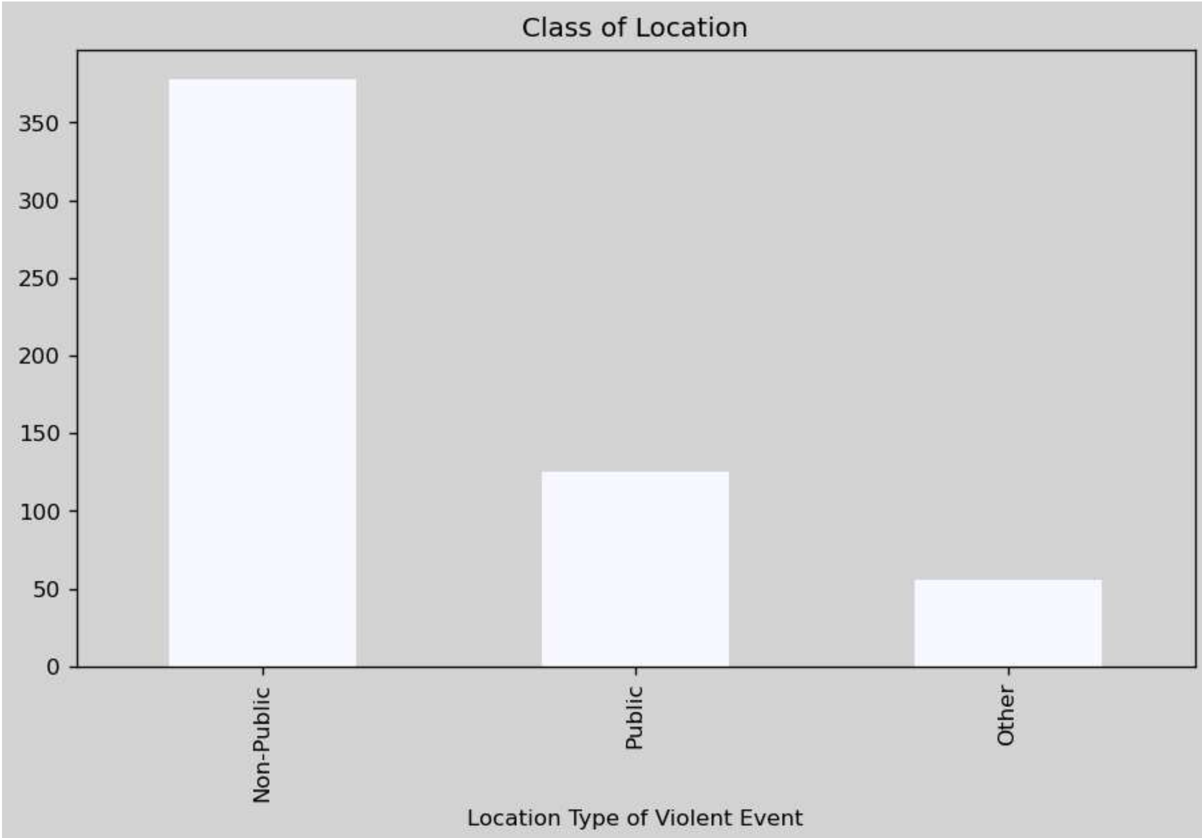
In [ ]:
```
print(mkip_data["Location"].unique())
hashLoc = {
"Public" : ['Commercial/Retail', 'Bar/Club/Restaurant', 'School', 'Open space', 'Medical facility', 'Hotel/Motel',
'House of worship','Shelter/Drug house', 'College'],
"Non-Public" : ['Residence'],
"Other" : ['Multiple', 'Vehicle', 'Government/Transit']}
def replace_loc(mkip_data_loc):
    return "".join(key for key, values in hashLoc.items() for value in values if value in mkip_data_loc)

mkip_data["Type_Location"] = mkip_data["Location"].apply(replace_loc)
```

```
['Residence' 'Multiple' 'Commercial/Retail' 'Bar/Club/Restaurant' 'School'
 'Open space' 'Vehicle' 'Medical facility' 'Hotel/Motel'
 'House of worship' 'Government/Transit' 'Shelter/Drug house' 'College']
```

In [ ]:
```
typeLoc_count = mkip_data.groupby(["Type_Location"]).Type_Location.count().sort_values(ascending=False)
print(typeLoc_count)
```

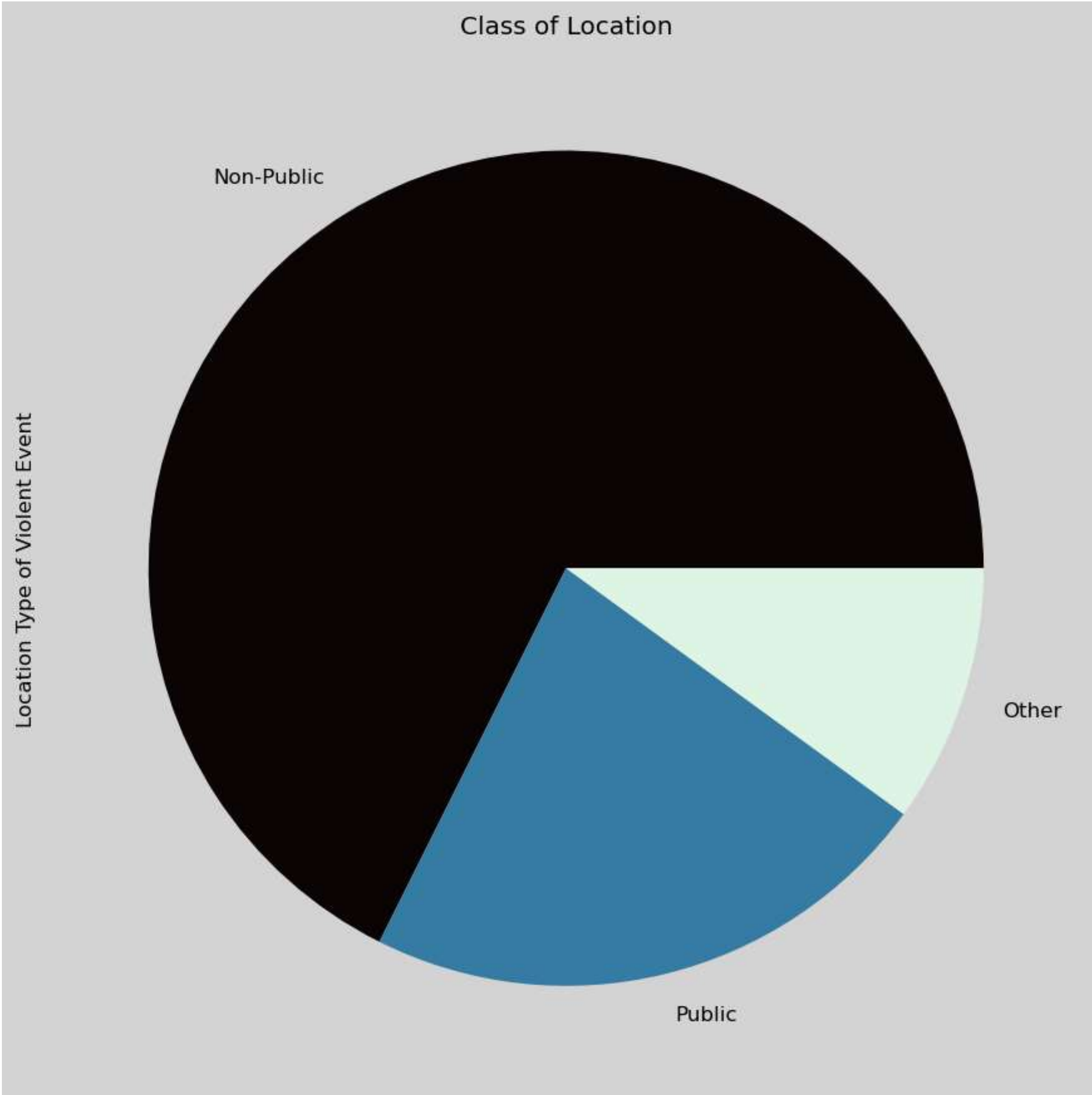
```
Type_Location
Non-Public    378
Public        125
Other          56
Name: Type_Location, dtype: int64
```

In [ ]:
```
fig, axe = plt.subplots(figsize=(9,5), constrained_layout=False, dpi=120, facecolor="lightgray", label="Mass Violence by Type")
typeLoc_count.plot.bar(stacked=False, color="ghostwhite")
axe.set_xlabel("Location Type of Violent Event")
axe.set_title("Class of Location")
axe.plot();
```

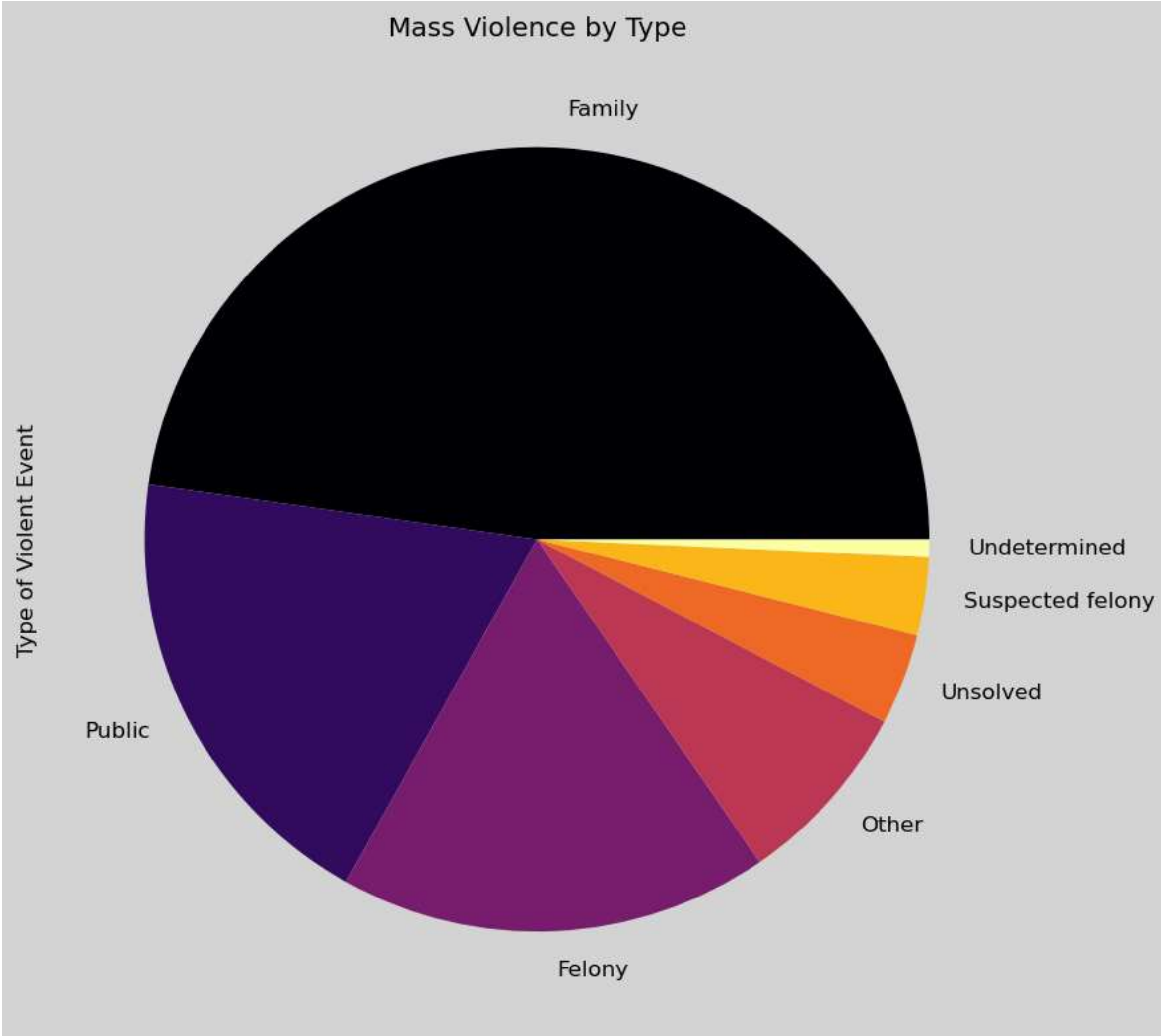


In [ ]:

```
fig, axe = plt.subplots(figsize=(10,9), constrained_layout=False, dpi=120, facecolor="lightgray", label="Mass Violence by Type")
typeLoc_count.plot.pie(stacked=False, cmap="mako")
axe.set_ylabel("Location Type of Violent Event")
axe.set_title("Class of Location")
axe.plot();
```



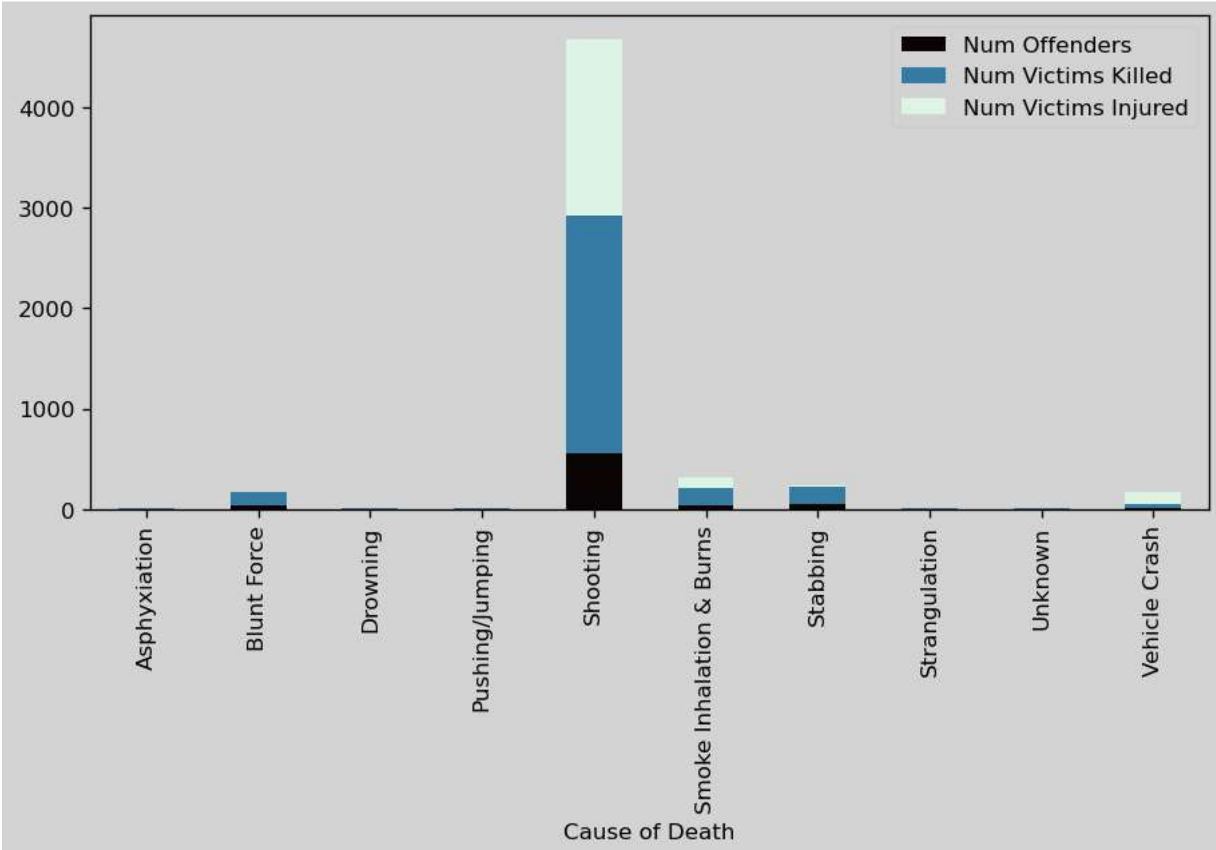
In [ ]:
```
fig, axe = plt.subplots(figsize=(10,8), constrained_layout=False, dpi=120, facecolor="lightgray", label="Mass Violence by Type")
type_count.plot.pie(cmap="inferno")
axe.set_ylabel("Type of Violent Event")
axe.set_title("Mass Violence by Type")
axe.plot();
```



In [ ]:

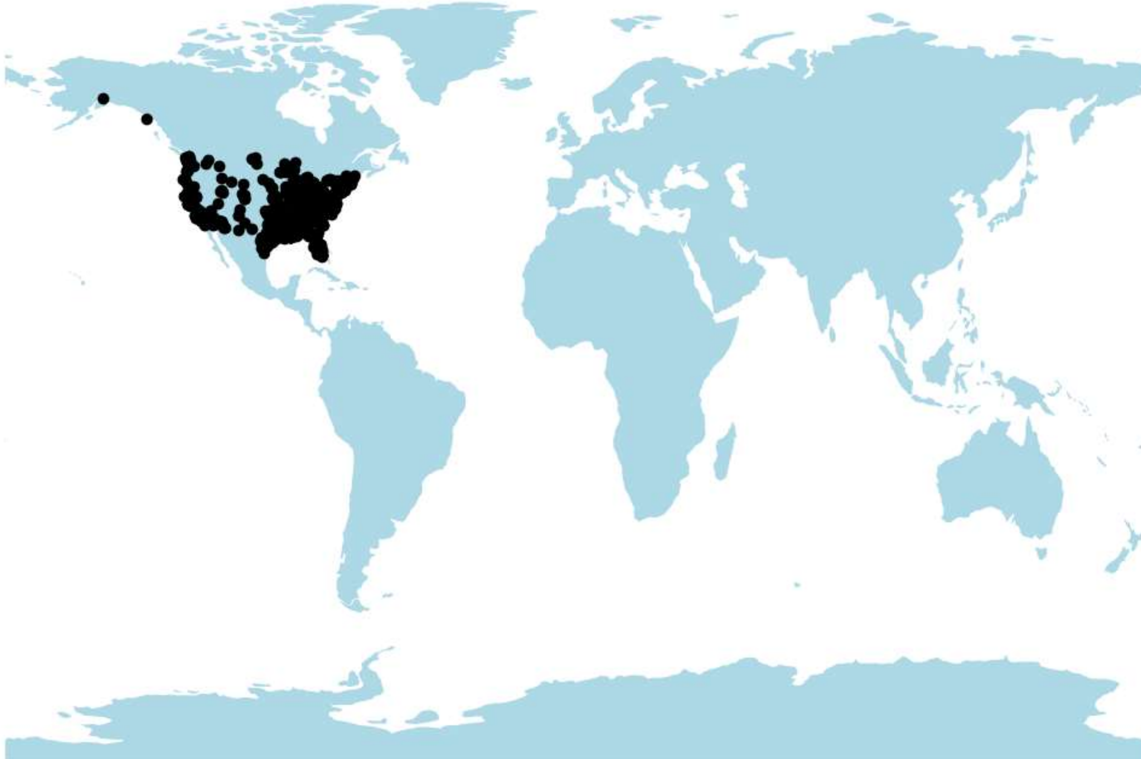
```
fig, axe = plt.subplots(figsize=(9,4), constrained_layout=False, dpi=120, facecolor="lightgray", label="Mass Violence by Type")
stack = mkip_data.groupby("FirstCOD")["FirstCOD",'Num_Offenders', 'Num_Victims_Killed', 'Num_Victims_Injured'].sum()
stack.rename({"Num_Offenders":"Num Offenders","Num_Victims_Killed":"Num Victims Killed", "Num_Victims_Injured":"Num Victims Injured"
stack.plot.bar(ax=axe, stacked=True, cmap="mako")
axe.set_xlabel("Cause of Death")
# axe.set_title("Class of Location")
axe.plot();
```

```
C:\Users\Owner\AppData\Local\Temp\ipykernel_25660\3346088684.py:2: FutureWarning: Indexing with multiple keys (implicitly converted t
o a tuple of keys) will be deprecated, use a list instead.
  stack = mkip_data.groupby("FirstCOD")["FirstCOD",'Num_Offenders', 'Num_Victims_Killed', 'Num_Victims_Injured'].sum()
C:\Users\Owner\AppData\Local\Temp\ipykernel_25660\3346088684.py:2: FutureWarning: The default value of numeric_only in DataFrameGroup
By.sum is deprecated. In a future version, numeric_only will default to False. Either specify numeric_only or select only columns whi
ch should be valid for the function.
  stack = mkip_data.groupby("FirstCOD")["FirstCOD",'Num_Offenders', 'Num_Victims_Killed', 'Num_Victims_Injured'].sum()
```



In [ ]:
```
gdf = gp.GeoDataFrame(mkip_data, geometry=gp.points_from_xy(mkip_data.Longitude, mkip_data.Latitude), crs="EPSG:4326")
usa = gp.read_file(geodatasets.get_path('naturalearth.land'))
ax = usa.plot(figsize=(20,9), color="lightblue")
ax.set_axis_off()
gdf.plot(ax=ax, color="black")
```

Out[ ]: <Axes: >



In [ ]:
```
mk_offenders_public = "https://query.data.world/s/di7bmak4rf6qkkmu2v64xfsei64y2j?dws=00000"
mkop_data = pd.read_table(mk_incident_public, delimiter=",", header=0, skipinitialspace=True, engine="python", encoding="utf-8", enco
mkop_data.rename(str.title, axis="columns", inplace=True)
mkop_data.rename({"Incident_Id":"Incident_ID","Firstcod":"FirstCOD", "Secondcod":"SecondCOD" }, axis="columns", inplace=True)
mkop_data['FirstCOD'] = [s.title() for s in mkip_data['FirstCOD']]
mkop_data['Location_Type'] = [s.title() for s in mkip_data['Location_Type']]
mkop_data.head(1)
```

Out[ ]:

	Incident_ID	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD	SecondCOD	Type	Situation_Type	Location_Type
0	577	2023-06-18	Kellogg	ID	1	4	0	Shooting	NaN	Other	Interpersonal conflict	Residence/Other Shelter

In [ ]:
```
mk_victims_public = "https://query.data.world/s/zrk6mdov7k3wkg3sve4e6wylul746d?dws=00000"
mkvp_data = pd.read_table(mk_incident_public, delimiter=",", header=0, skipinitialspace=True, engine="python", encoding="utf-8", enco
mkvp_data.rename(str.title, axis="columns", inplace=True)
mkvp_data.rename({"Incident_Id":"Incident_ID","Firstcod":"FirstCOD", "Secondcod":"SecondCOD" }, axis="columns", inplace=True)
mkvp_data['FirstCOD'] = [s.title() for s in mkip_data['FirstCOD']]
mkvp_data['Location_Type'] = [s.title() for s in mkip_data['Location_Type']]
mkvp_data.head(1)
```

Out[ ]:

	Incident_ID	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD	SecondCOD	Type	Situation_Type	Location_Type
0	577	2023-06-18	Kellogg	ID	1	4	0	Shooting	NaN	Other	Interpersonal conflict	Residence/Other Shelter

In [ ]:
```
mk_weapons_public = "https://query.data.world/s/tsuydmpezwzbjormbolhpi6eo34opd?dws=00000"
mkwp_data = pd.read_table(mk_incident_public, delimiter=",", header=0, skipinitialspace=True, engine="python", encoding="utf-8", enco
mkwp_data.rename(str.title, axis="columns", inplace=True)
mkwp_data.rename({"Incident_Id":"Incident_ID","Firstcod":"FirstCOD", "Secondcod":"SecondCOD" }, axis="columns", inplace=True)
mkwp_data['FirstCOD'] = [s.title() for s in mkip_data['FirstCOD']]
```

```python
mkwp_data['Location_Type'] = [s.title() for s in mkip_data['Location_Type']]
mkwp_data.head(1)
```

Out[ ]:

	Incident_ID	Date	City	State	Num_Offenders	Num_Victims_Killed	Num_Victims_Injured	FirstCOD	SecondCOD	Type	Situation_Type	Location_Type
0	577	2023-06-18	Kellogg	ID	1	4	0	Shooting	NaN	Other	Interpersonal conflict	Residence/Other Shelter

@Credit: USA TODAY/AP/Northeastern University

About the data

The USA TODAY/AP/Northeastern University mass killing database contains information on incidents, offenders, victims and weapons for all multiple homicides with four or more victims killed in the United States from 2006 to the present.

Definition

A mass killing is defined as the intentional killing of four or more victims – excluding the deaths of unborn children and the offender(s) – by any means within a 24-hour period.

This definition includes cases involving all weapons (shooting, blunt force, stabbing, explosives), types (public, felony-related, and familicides), motivations (domestic dispute, profit, revenge, terrorism, hate), victim-offender relationships (stranger, family, acquaintance, co-worker), and number of locations. The time frame of 24 hours was chosen to eliminate conflation with spree killers who kill multiple victims over several days in different locations and to satisfy the traditional requirement of occurring in a “single incident,” even if that incident involves an offender targeting multiple locations in an extended assault but within a relatively short time span. However, offenders who kill four or more victims during any 24-hour period of time as part of a multi-day spree are included, as are all their victims within seven days of the mass killing. Negligent homicides related to driving under the influence or accidental fires are excluded because of the lack of intent. Finally, only incidents occurring within the 50 U.S. states and the District of Columbia are included in the database.

Consistent with the traditional definition, fatal mass shootings are mass killings (four or more victim fatalities) in which most or all the victims are killed by gunfire. This differs from an alternative definition used by the Gun Violence Archive that includes incidents in which at least four victims are shot regardless of whether the injury is fatal. Less than 5% of the mass shootings listed in the Gun Violence Archive are defined as mass killings in our database. Our definition of a fatal mass shooting also differs from an active shooter event which, as characterized by the FBI, involves an individual actively engaged in killing or attempting to kill people in a populated area. Less than 25% of active shooter events result in four or more victim fatalities, constituting a mass killing.

Methods

Researchers at USA TODAY first identified potential incidents using the FBI’s Supplementary Homicide Reports (SHR). Homicide incidents in the SHR were flagged as potential mass killing cases if four or more victims were reported on the same record, and the type of homicide was coded as “murder or non-negligent manslaughter.” Cases were subsequently verified utilizing media accounts, court documents, academic journal articles, books and local law enforcement records obtained through Freedom of Information Act (FOIA) requests. Each data point was corroborated by multiple sources, which were compiled into a single document to assess the quality of information. When sources were contradictory, official law enforcement or court records were used, when available, followed by the most recent media or academic source. Case information was subsequently compared with other available mass killing or mass shooting databases to ensure validity. Incidents listed in the SHR that could not be independently verified were excluded from the database.

In 2016, primary data collection and verification efforts shifted from USA TODAY to Northeastern University. Northeastern researchers conducted extensive searches for incidents not reported in the SHR during the time period, utilizing internet search engines including Lexis-Nexis, Google News, and Newspapers.com. Search terms included: [number] dead, [number] killed, [number] slain, [number] murdered, [number] homicide, mass murder, mass shooting, massacre, rampage, family killing, familicide and arson murder. Offender, victim and location names were also directly searched when available. Northeastern University researchers also independently verified data collected by USA TODAY staff and filled in missing information, sometimes involving updated reports on older cases.

In December 2018, a Memo of Understanding (MOU) was signed by The Associated Press, USA TODAY and Northeastern University to formalize a joint initiative to maintain and expand the mass killing database previously housed at USA TODAY. The Associated Press hosts the database and maintains the data entry tool, USA TODAY has developed and maintains the public website for and visualizations of the database, and Northeastern University manages data collection and updates.

The full database currently consists of four linked data tables with a total of 59 data fields (not counting indicators for the availability of offender/victim identity) –18 fields for each incident, 20 fields for each offender, 13 fields for each victim killed and eight fields for each weapon used. Most variables, with the notable exception of victim names, are available for public download. The remaining data is reserved for individuals affiliated with The Associated Press, USA TODAY/Gannett, and Northeastern University's School of Criminology and Criminal Justice, and others by permission of all three organizations. Moving forward, additional variables may be added to the full database as well as the public subset. While USA TODAY respects diversity of gender, this database instead uses sex as a datapoint as is common in crime statistics.

Any questions or corrections concerning the data should be directed to James Alan Fox at j.fox@northeastern.edu.

Credits

Research and reporting: Karina Zaiets and George Petras

Design and development: Veronica Bravo and Mitchell Thorson

Editing: Shawn J. Sullivan

Paul Overberg, Meghan Hoyer, Mark Hannan, Jodi Upton, Barbie Hansen, and Erin Durkin contributed to the original 2012 data reporting effort at USA TODAY.