Gun Violence Data Analysis and Visualization in USA

Gun violence in the US results in tens of thousands of deaths and injuries annually. In 2013, there were 73,505 nonfatal firearm injuries which included 11,208 homicides, 21,175 suicides, 505 deaths due to accidental or negligent discharge of a firearm, and 281 deaths due to firearms use with "undetermined intent".

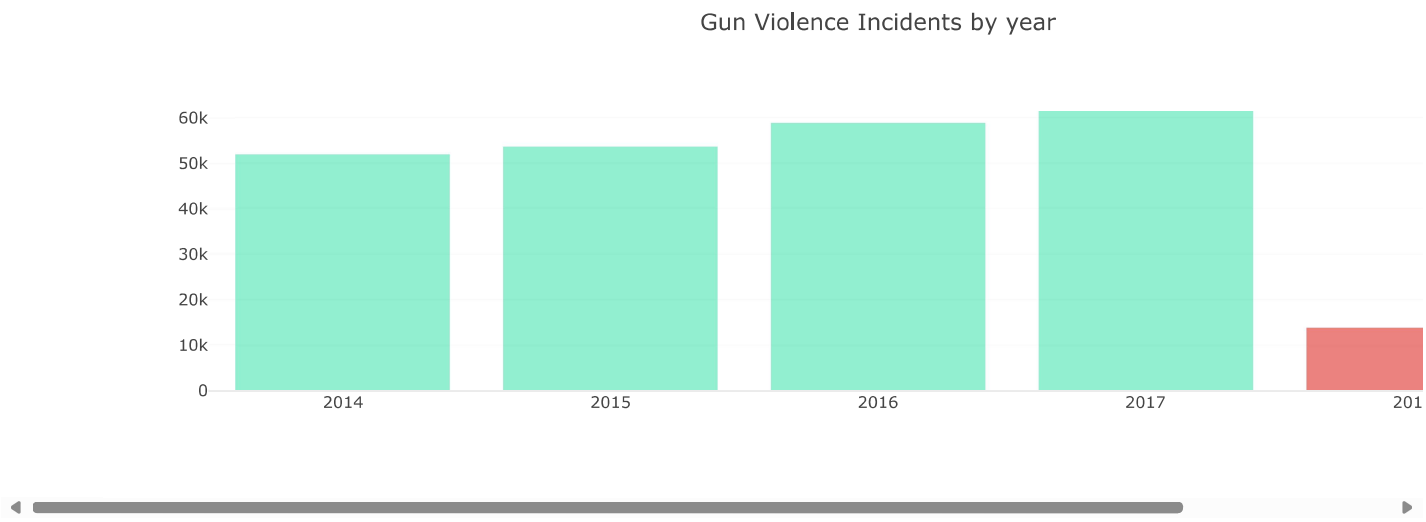
In [1]:
```
# Import the required libraries

from plotly.offline import init_notebook_mode, iplot
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
import plotly.graph_objs as go
from wordcloud import WordCloud
from textblob import TextBlob
import plotly.plotly as py
from plotly import tools
import seaborn as sns
import pandas as pd
import string, os, random
import calendar
from PIL import Image
import numpy as np


init_notebook_mode(connected=True)
punc = string.punctuation
from datetime import datetime

# Load the dataset
path = "/gun-violence-data/gun-violence-data.csv"
df = pd.read_csv(path)

# Create some additional features
df['year'] = df['date'].dt.year
df['monthday'] = df['date'].dt.day
df['weekday'] = df['date'].dt.weekday
df['loss'] = df['n_killed'] + df['n_injured']
```

Exploring Time Related Trends of Gun Violence

Number of Gun Violence Incidents by Years

Plot the number of gun violence incidents by years.

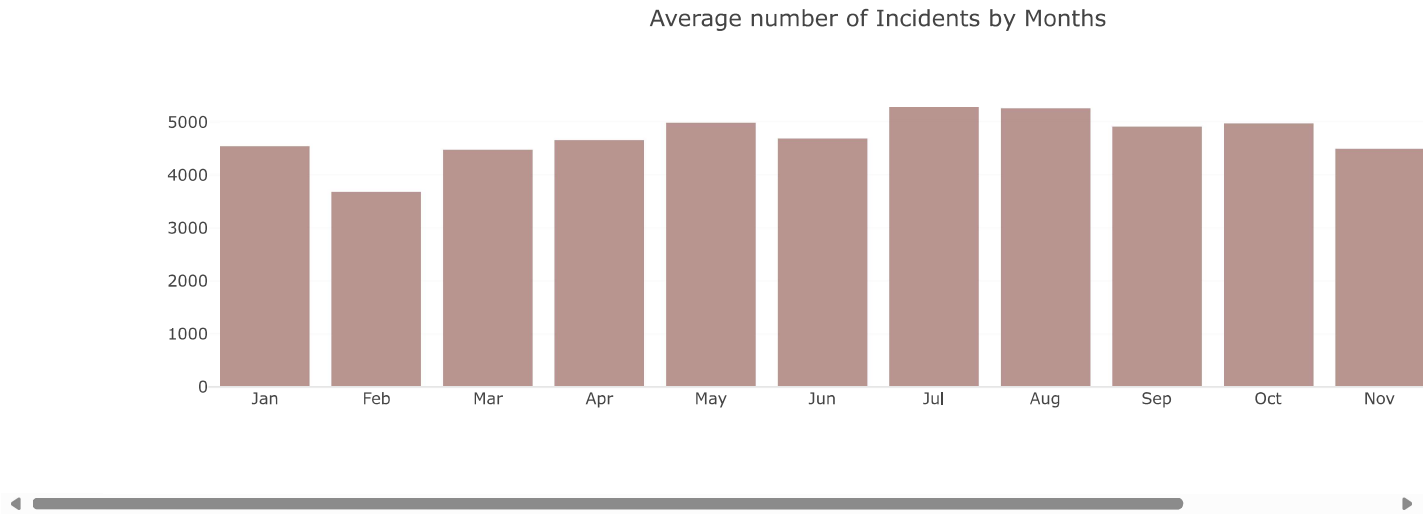
In [2]:
```
def create_stack_bar_data(col):
    aggregated = df[col].value_counts()
    x_values = aggregated.index.tolist()
    y_values = aggregated.values.tolist()
    return x_values, y_values

x1, y1 = create_stack_bar_data('year')
x1 = x1[:-1]
y1 = y1[:-1]
trace1 = go.Bar(x=x1, y=y1, opacity=0.75, name="year count", marker=dict(color=['rgba(10, 220, 150, 0.6)', 'rgba(10, 220, 150, 0.6)', 'rgb
layout = dict(height=400, title='Gun Violence Incidents by year', legend=dict(orientation="h"));
fig = go.Figure(data=[trace1], layout=layout);
iplot(fig);
```



From the above graph, we can observe that the number of gun violence incidents is increasing every year. In the year 2014, there were about 51 thousand incidents reported. The number increased to 53 thousand in the next year, and 2016 saw a bigger jump with close to 58 thousand incidents reported. In 2017, the number of gun violence incidents further increased to 61 thousand. A significant increase of 10,000 incidents has been observed from 2014 to 2017. We are still halfway through 2018, and this number is expected to continue growing.

Average number of incidents per month

In [3]:
```
tempdf = df[df['year'].isin(['2014','2015','2016','2017'])]
tempdf1 = tempdf.groupby(['year','month']).agg({'month' : 'count'}).rename(columns={'month': 'month_count'}).reset_index()
aggregated = tempdf1.groupby(['month']).agg({'month_count' : 'mean'})
x2 = aggregated.index.tolist()
y2 = aggregated.month_count.tolist()
mapp = {}
for m,v in zip(x2, y2):
    mapp[m] = v
xn = [calendar.month_abbr[int(x)] for x in sorted(x2)]
vn = [mapp[x] for x in sorted(x2)]

trace1 = go.Bar(x=xn, y=vn, opacity=0.75, name="month", marker=dict(color='rgba(100, 20, 10, 0.6)'))
layout = dict(height=400, title='Average number of Incidents by Months', legend=dict(orientation="h"));
fig = go.Figure(data=[trace1], layout=layout)
iplot(fig, filename='stacked-bar')
```

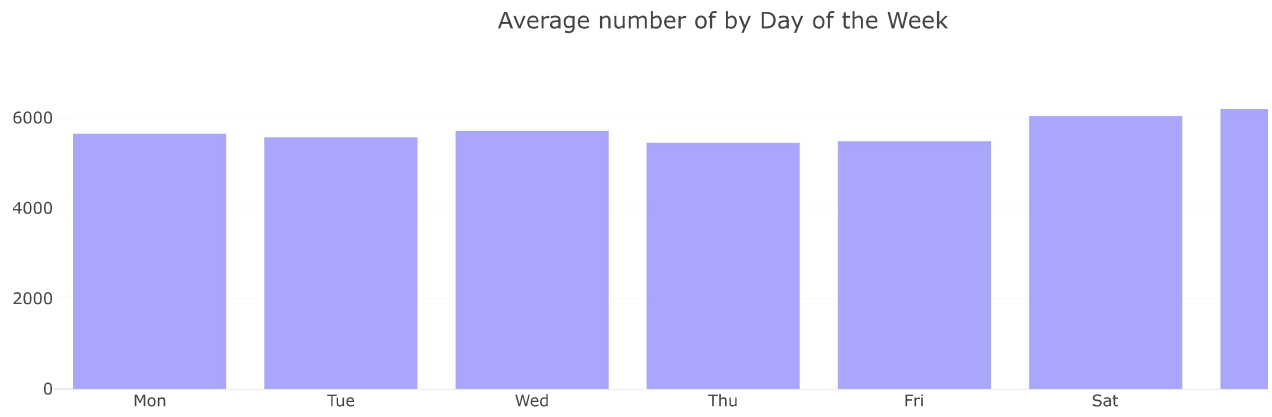


The months of July and August have the highest average number of gun violence incidents compared to any other month. In these two months, there are about 5,200 incidents reported every year. February has the lowest average number of incidents, about 3,700 every year. In the next sections, I will try to explore whether there are specific dates on which the number of gun violence incidents is consistently higher.

Average Number of Gun Violence Incidents by Day of the Week

In [4]:
```
tempdf1 = df.groupby(['year', 'weekday']).agg({'weekday' : 'count'}).rename(columns={'weekday' : 'weekday_count'}).reset_index()
aggregated = tempdf1.groupby(['weekday']).agg({'weekday_count' : 'mean'})

x2 = aggregated.index.tolist()
y2 = aggregated.weekday_count.tolist()

weekmap = {0:'Mon', 1:'Tue', 2:'Wed', 3:'Thu', 4:'Fri', 5:'Sat', 6:'Sun'}
x2 = [weekmap[x] for x in x2]
wkmp = {}
```

```
for j,x in enumerate(x2):
    wkmp[x] = y2[j]
order = list(weekmap.values())
ordervals = [wkmp[val] for val in order]

trace1 = go.Bar(x=order, y=ordervals, opacity=0.75, name="weekday", marker=dict(color='rgba(61, 60, 250, 0.6)'))
layout = dict(height=400, title='Average number of by Day of the Week', legend=dict(orientation="h"));

fig = go.Figure(data=[trace1], layout=layout)
iplot(fig, filename='stacked-bar')
```

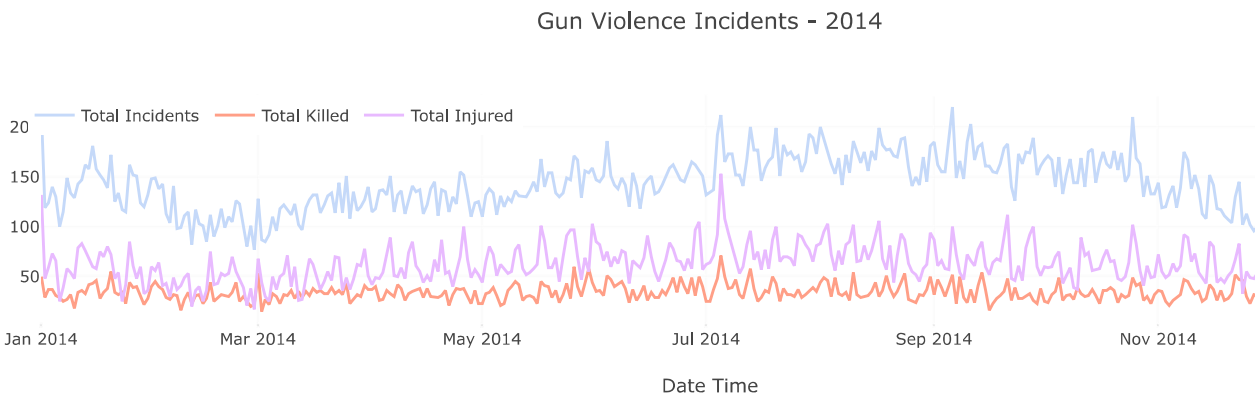


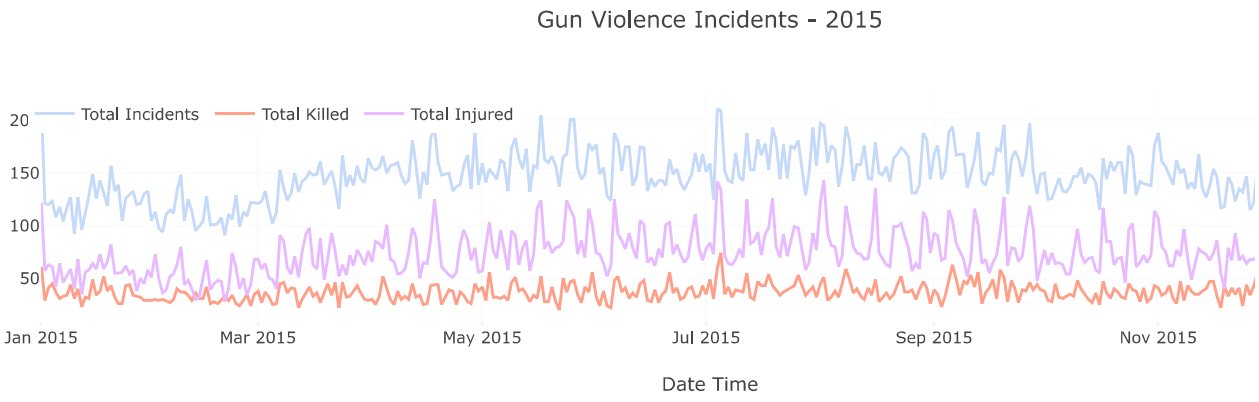
The number of gun violence incidents is higher on weekends, with approximately 6,000 incidents occurring on Saturdays and Sundays across the entire US.

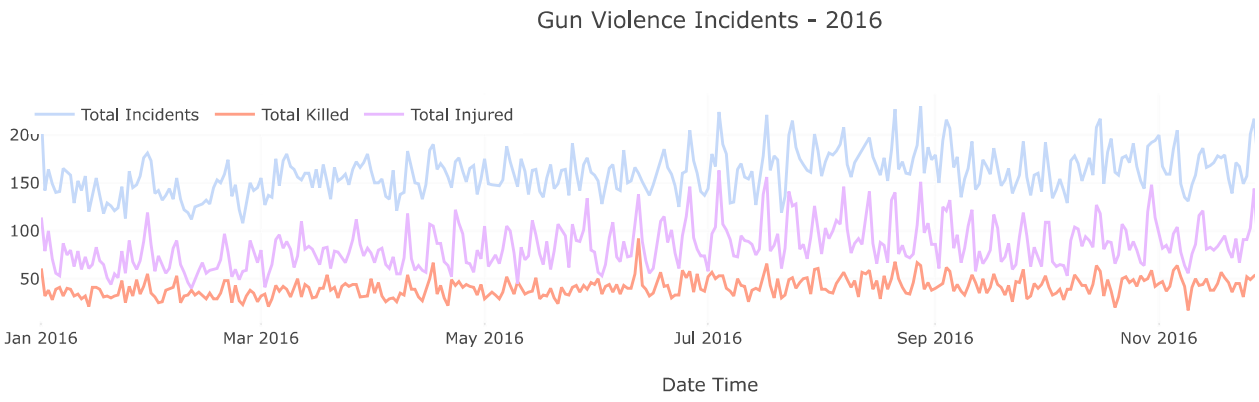
Time Series plot of Total Incidents, People Killed, People Injured

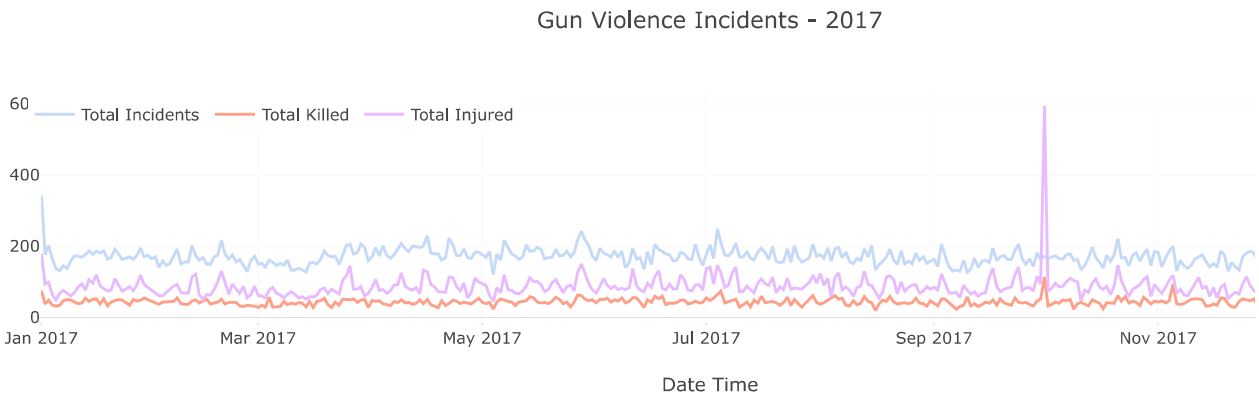
In [5]:
```
# 2014 time series
temp = df[df['year'] == 2014].groupby('date').agg({'state' : 'count', 'n_killed' : 'sum', 'n_injured' : 'sum'}).reset_index().rename(colu


trace1 = go.Scatter(x = temp.date, y = temp.incidents, name='Total Incidents', mode = "lines", marker = dict(color = '#c5d9f9'))
trace2 = go.Scatter(x = temp.date, y = temp.n_killed, name="Total Killed", mode = "lines", marker = dict(color = '#ff9f87'))
trace3 = go.Scatter(x = temp.date, y = temp.n_injured, name="Total Injured", mode = "lines", marker = dict(color = '#e8baff'))

data = [trace1, trace2, trace3]
layout = dict(height=350, title = 'Gun Violence Incidents - 2014', legend=dict(orientation="h", x=-.01, y=1), xaxis= dict(title='Date Tim
fig = dict(data = data, layout = layout)
iplot(fig)


# 2015 time series
temp = df[df['year'] == 2015].groupby('date').agg({'state' : 'count', 'n_killed' : 'sum', 'n_injured' : 'sum'}).reset_index().rename(colu


trace1 = go.Scatter(x = temp.date, y = temp.incidents, name='Total Incidents', mode = "lines", marker = dict(color = '#c5d9f9'))
trace2 = go.Scatter(x = temp.date, y = temp.n_killed, name="Total Killed", mode = "lines", marker = dict(color = '#ff9f87'))
trace3 = go.Scatter(x = temp.date, y = temp.n_injured, name="Total Injured", mode = "lines", marker = dict(color = '#e8baff'))

data = [trace1, trace2, trace3]
layout = dict(height=350, title = 'Gun Violence Incidents - 2015', legend=dict(orientation="h", x=-.01, y=1), xaxis= dict(title='Date Tim
fig = dict(data = data, layout = layout)
iplot(fig)


# 2016 time series
temp = df[df['year'] == 2016].groupby('date').agg({'state' : 'count', 'n_killed' : 'sum', 'n_injured' : 'sum'}).reset_index().rename(colu


trace1 = go.Scatter(x = temp.date, y = temp.incidents, name='Total Incidents', mode = "lines", marker = dict(color = '#c5d9f9'))
trace2 = go.Scatter(x = temp.date, y = temp.n_killed, name="Total Killed", mode = "lines", marker = dict(color = '#ff9f87'))
trace3 = go.Scatter(x = temp.date, y = temp.n_injured, name="Total Injured", mode = "lines", marker = dict(color = '#e8baff'))

data = [trace1, trace2, trace3]
layout = dict(height=350,title = 'Gun Violence Incidents - 2016', legend=dict(orientation="h", x=-.01, y=1), xaxis= dict(title='Date Time
fig = dict(data = data, layout = layout)
iplot(fig)


# 2017 time series
temp = df[df['year'] == 2017].groupby('date').agg({'state' : 'count', 'n_killed' : 'sum', 'n_injured' : 'sum'}).reset_index().rename(colu


trace1 = go.Scatter(x = temp.date, y = temp.incidents, name='Total Incidents', mode = "lines", marker = dict(color = '#c5d9f9'))
trace2 = go.Scatter(x = temp.date, y = temp.n_killed, name="Total Killed", mode = "lines", marker = dict(color = '#ff9f87'))
trace3 = go.Scatter(x = temp.date, y = temp.n_injured, name="Total Injured", mode = "lines", marker = dict(color = '#e8baff'))
```

```
data = [trace1, trace2, trace3]
layout = dict(height=350,title = 'Gun Violence Incidents - 2017', legend=dict(orientation="h", x=-.01, y=1), xaxis= dict(title='Date Time
fig = dict(data = data, layout = layout)
iplot(fig)
```









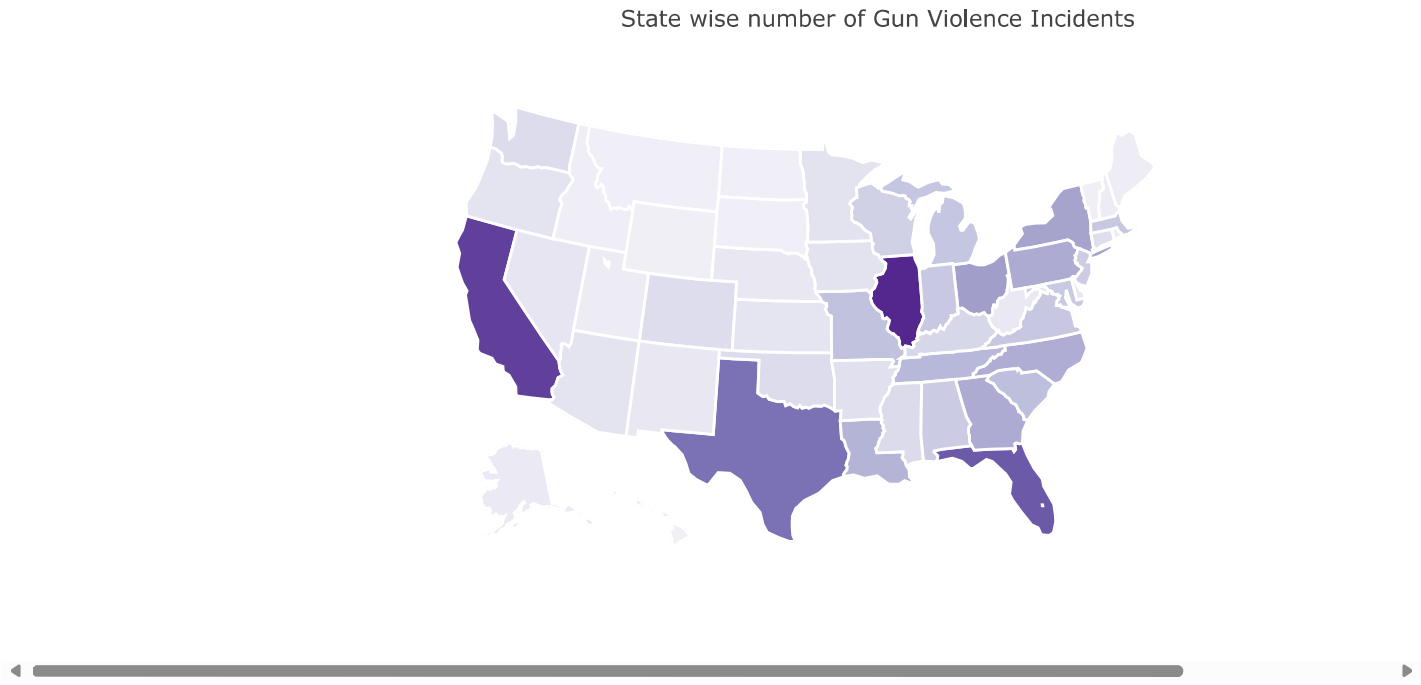
Exploring Location related trends

Number of Incidents by every state

In [6]:

```
states_df = df['state'].value_counts()

statesdf = pd.DataFrame()
statesdf['state'] = states_df.index
statesdf['counts'] = states_df.values

scl = [[0.0, 'rgb(242,240,247)'],[0.2, 'rgb(218,218,235)'],[0.4, 'rgb(188,189,220)'],\
            [0.6, 'rgb(158,154,200)'],[0.8, 'rgb(117,107,177)'],[1.0, 'rgb(84,39,143)']]

state_to_code = {'District of Columbia' : 'dc','Mississippi': 'MS', 'Oklahoma': 'OK', 'Delaware': 'DE', 'Minnesota': 'MN', 'Illinois': 'I
statesdf['state_code'] = statesdf['state'].apply(lambda x : state_to_code[x])

data = [ dict(
        type='choropleth',
        colorscale = scl,
        autocolorscale = False,
        locations = statesdf['state_code'],
        z = statesdf['counts'],
        locationmode = 'USA-states',
        text = statesdf['state'],
        marker = dict(
            line = dict (
                color = 'rgb(255,255,255)',
                width = 2
            ) ),
        colorbar = dict(
            title = "Gun Violence Incidents")
        ) ]

layout = dict(
        title = 'State wise number of Gun Violence Incidents',
        geo = dict(
            scope='usa',
            projection=dict( type='albers usa' ),
            showlakes = True,
            lakecolor = 'rgb(255, 255, 255)'),
             )

# data = [ dict(
#         type='choropleth',
#         colorscale = scl,
#         autocolorscale = False,
#         locations = statesdf['state_code'],
#         z = statesdf['counts'],
#         locationmode = 'USA-states',
#         text = statesdf['state'],
#         marker = dict(
#             line = dict (
#                 color = 'rgb(255,255,255)',
#                 width = 2
#             ) ),
#         colorbar = dict(
#             title = "Gun Violence Incidents")
#         ) ]

# layout = dict(
#         title = 'State wise number of Gun Violence Incidents',
#         geo = dict(
#             scope='usa',
#             projection=dict( type='albers usa' ),
#             showlakes = True,
#             lakecolor = 'rgb(255, 255, 255)'),
```

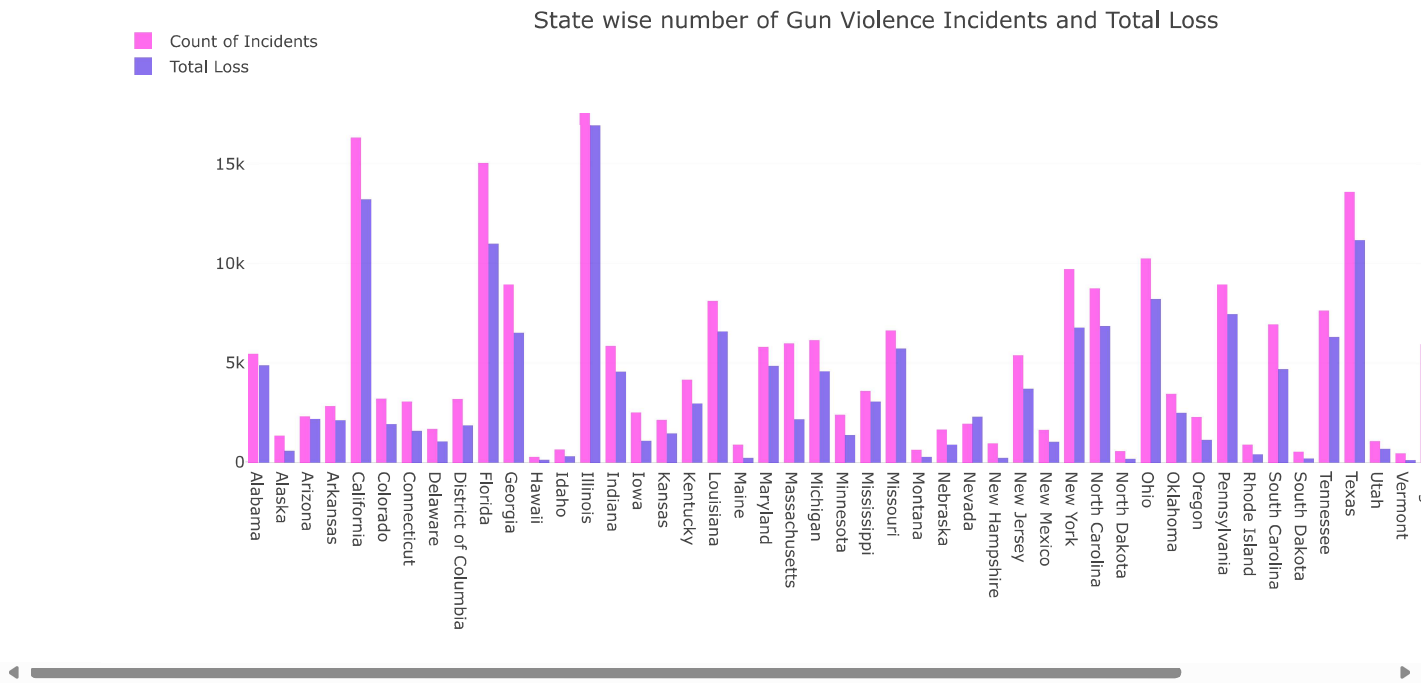
```
#             )

fig = dict( data=data, layout=layout )
iplot( fig, filename='d3-cloropleth-map' )
```



From the above map, Illinois State has the highest number of gun violence incidents reported in the last 5 years, with a total of 17,000 incidents. It is followed by California, with approximately 16,000 incidents reported, Florida with 15,000 incidents, and Texas with 13,000 gun violence incidents. In the next section, I will explore the states with the highest gun violence incidents per 100,000 people.

Total Loss (Injured + Killed) by every state

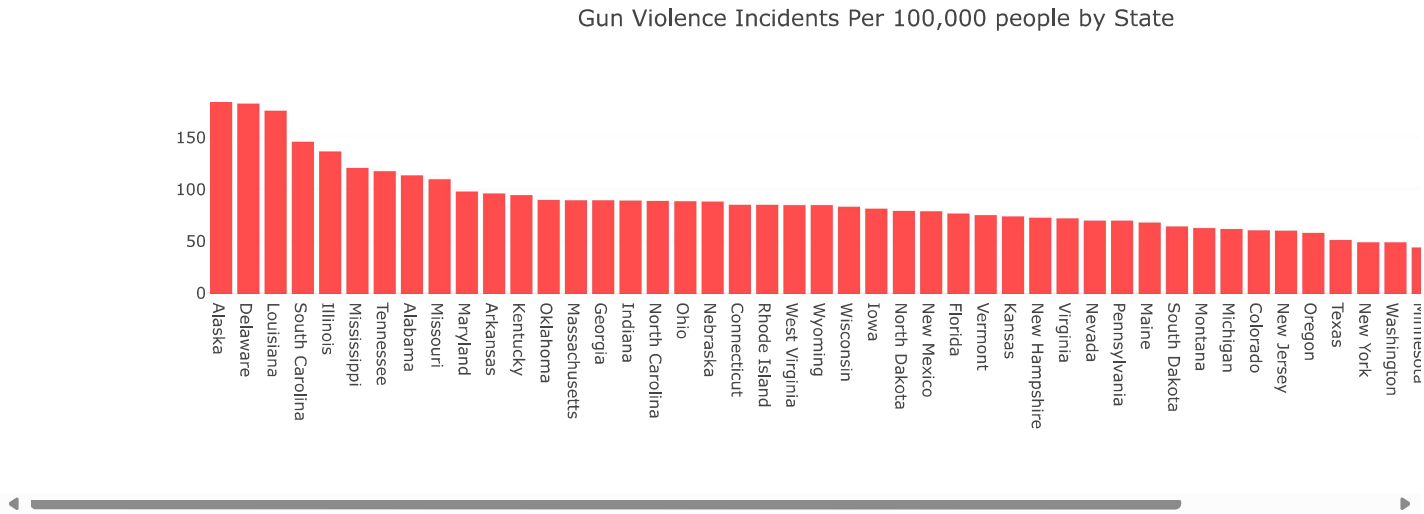
In [7]:
```
statdf = df.reset_index().groupby(by=['state']).agg({'loss':'sum', 'year':'count'}).rename(columns={'year':'count'})
statdf['state'] = statdf.index

trace1 = go.Bar(
    x=statdf['state'],
    y=statdf['count'],
    name='Count of Incidents',
    marker=dict(color='rgb(255,10,225)'),
    opacity=0.6
)
trace2 = go.Bar(
    x=statdf['state'],
    y=statdf['loss'],
    name='Total Loss',
    marker=dict(color='rgb(58,22,225)'),
    opacity=0.6
)

data = [trace1, trace2]
layout = go.Layout(
    barmode='group',
    margin=dict(b=150),
    legend=dict(dict(x=-.1, y=1.2)),
    title = 'State wise number of Gun Violence Incidents and Total Loss',
)

fig = go.Figure(data=data, layout=layout)
iplot(fig, filename='grouped-bar')
```



Incidents Per 100,000 People by State

Let's explore how many gun violence incidents occur in different states per 100,000 people. Additionally, let's identify the states with a high number of incidents relative to their population.

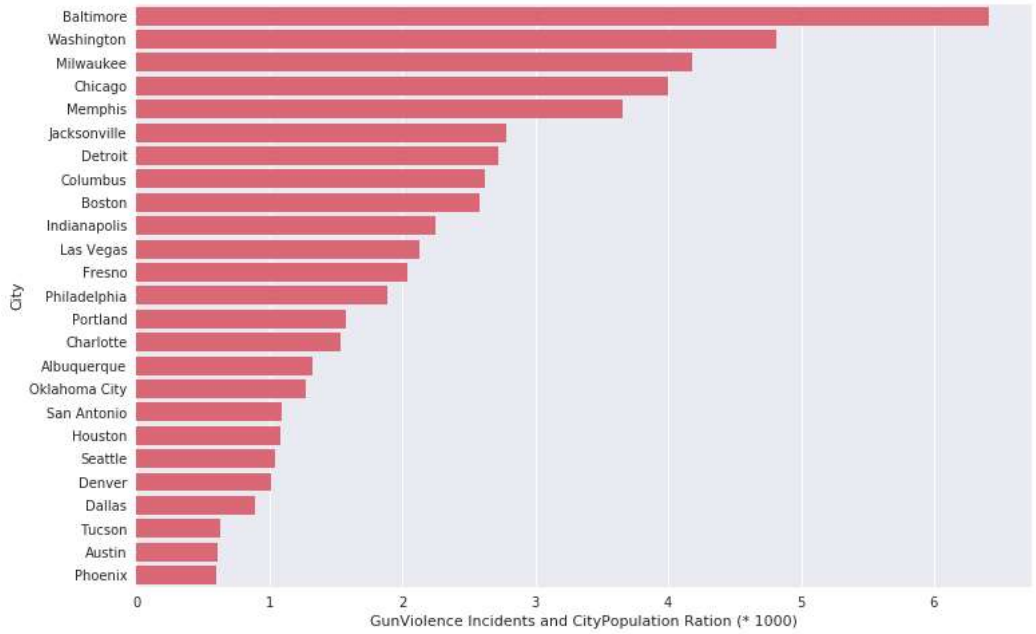
In [8]:
```
census_2013 = {'Mississippi': 2991207, 'Iowa': 3090416, 'Oklahoma': 3850568, 'Delaware': 925749, 'Minnesota': 5420380, 'Alaska': 735132,
statesdf['population'] = statesdf['state'].apply(lambda x : census_2013[x])
statesdf['incidents_per_k'] = statesdf.apply(lambda x : float(x['counts'])*100000 / x['population'], axis=1)
tempdf = statesdf.sort_values('incidents_per_k', ascending = False)[1:50]
trace1 = go.Bar(
    x=tempdf.state,
    y=tempdf.incidents_per_k,
    name='Location Types',
    orientation = 'v',
    marker=dict(color='red'),
    opacity=0.7
)

data = [trace1]
layout = go.Layout(
    height=400,
    margin=dict(b=150),
    barmode='group',
    legend=dict(dict(x=-.1, y=1.2)),
    title = 'Gun Violence Incidents Per 100,000 people by State',
)

fig = go.Figure(data=data, layout=layout)
iplot(fig, filename='grouped-bar')
```



Alaska has the highest number of gun violence incidents per 100,000 people, approximately equal to 183, while Arizona, Utah, and Idaho are the states with the fewest number of incidents in the population-adjusted dataset. It's interesting to note that California, which is one of the top states with a high number of gun violence incidents, has a very low rate per 100,000 incidents. Illinois still ranks among the top 5 states with the most gun violence incidents.

Population Adjusted Dataset - Top Cities (Incidents Per 100,000 People)

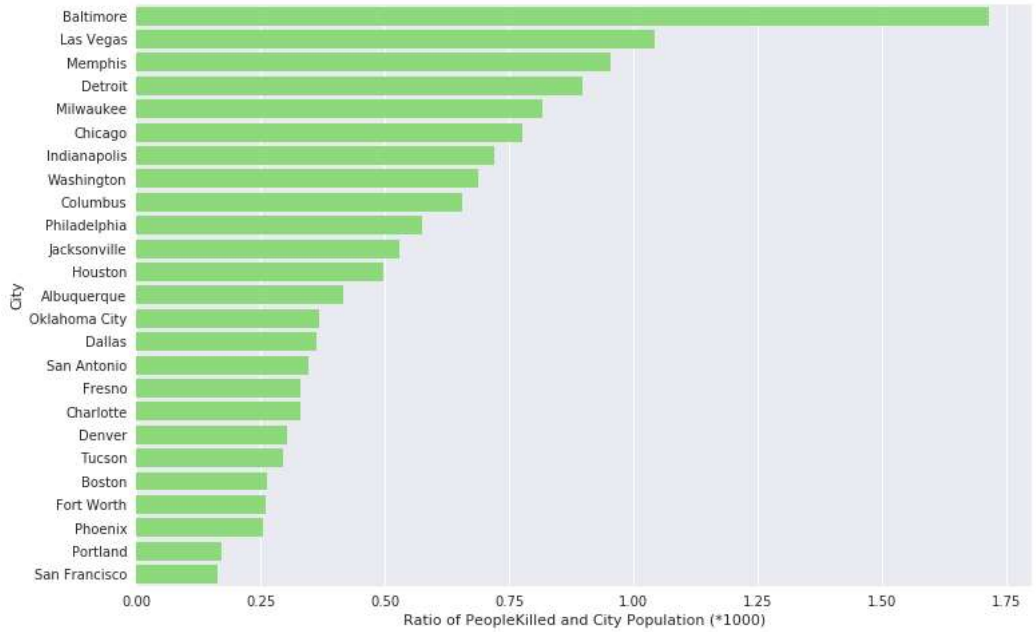
In [9]:
```
population = {'Carol Stream': '40069', 'La Quinta': '40956', 'Leander': '42761', 'Edinburg': '87650', 'Chino Hills': '78822', 'Pinellas Pa
df['city_population'] = df['city_or_county'].apply(lambda x : int(population[x]) if x in population else 0)

tempdf = df.groupby(by=['city_or_county']).agg({'n_killed': 'sum', 'n_injured' : 'sum', 'city_population' : 'mean', 'state' : 'count'}).re
tempdf['incidents_population_ratio'] = 1000*tempdf['total_incidents'] / (tempdf['city_population']+1)
tempdf['killed_population_ratio'] = 1000*tempdf['total_killed'] / (tempdf['city_population']+1)
tempdf['injured_population_ratio'] = 1000*tempdf['total_injured'] / (tempdf['city_population']+1)
tempdf['loss_population_ratio'] = 1000*(tempdf['total_killed'] + tempdf['total_injured']) / (tempdf['city_population']+1)

i_p = tempdf.sort_values(['incidents_population_ratio'], ascending=[False])
i_p = i_p[i_p['city_population'] > 500000][:25]
sns.set(rc={'figure.figsize':(12,8)})
ax = sns.barplot(y='city_or_county', x='incidents_population_ratio', data=i_p, color='#ed5569')
ax.set(xlabel='GunViolence Incidents and CityPopulation Ration (* 1000)', ylabel='City');
```



Baltimore (a city in Maryland) has the highest ratio of gun violence incidents to its city population, in contrast to Chicago, which had the highest absolute number of gun violence incidents. Baltimore had a total of 3,943 gun violence incidents, with a population of 614,664 in 2017. Chicago, on the other hand, experienced more than 10,000 gun violence incidents in recent years. However, it is one of the most populated cities in the US, with a population of 2,704,958 in 2017, ranking fourth in terms of the ratio of gun violence incidents to population, following Baltimore, Washington, and Milwaukee.

Population Adjusted Dataset - Top Cities (People Killed Per 100,000 People)

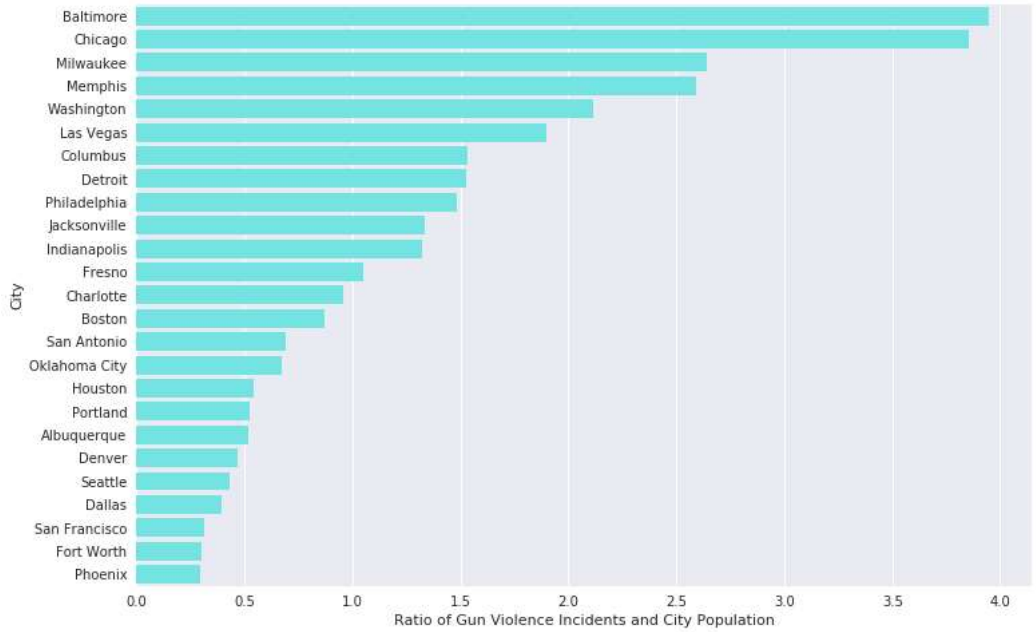
In [10]:
```
i_p = tempdf.sort_values(['killed_population_ratio'], ascending=[False])
i_p = i_p[i_p['city_population'] > 500000][:25]
sns.set(rc={'figure.figsize':(12,8)})
ax = sns.barplot(y='city_or_county', x='killed_population_ratio', data=i_p, color='#85e86d')
ax.set(xlabel='Ratio of PeopleKilled and City Population (*1000)', ylabel='City');
```



Baltimore again tops the list with approximately 1,000 people killed in gun violence incidents, resulting in a ratio of 1.716382. Las Vegas, with a population of 632,912, and Memphis, with a population size of 652,717, come in 2nd and 3rd place with 660 and 623 people killed, respectively. Their respective ratios of people killed to population are 1.04 and 0.95. Chicago, where the maximum number of incidents has occurred, has a killed-to-population ratio of 0.77, placing it in the 6th spot.

Population Adjusted Dataset - Top Cities (People Injured Per 100,000 People)

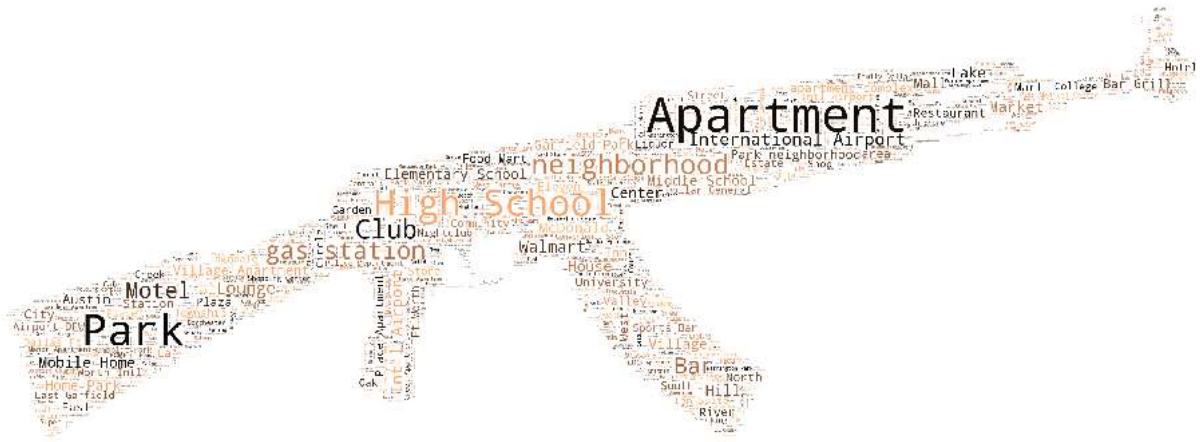
In [11]:
```
i_p = tempdf.sort_values(['injured_population_ratio'], ascending=[False])
i_p = i_p[i_p['city_population'] > 500000][:25]
sns.set(rc={'figure.figsize':(12,8)})
ax = sns.barplot(y='city_or_county', x='injured_population_ratio', data=i_p, color='#60f7f4')
ax.set(xlabel='Ratio of Gun Violence Incidents and City Population', ylabel='City');
```



Baltimore and Chicago are the cities with the highest ratio of people injured in gun violence incidents to their respective city populations. Additionally, Milwaukee, Memphis, and Washington are among the top cities where the ratio of people getting injured in gun violence incidents is significantly higher.

Common Areas of Gun Violence Incidents

In [12]:
```
mask = np.array(Image.open('../input/word-cloud-masks/gun.png'))
txt = " ".join(df['location_description'].dropna())
wc = WordCloud(mask=mask, max_words=1200, stopwords=STOPWORDS, colormap='copper', background_color='White').generate(txt)
plt.figure(figsize=(16,18))
plt.imshow(wc)
plt.axis('off')
plt.title('');
```

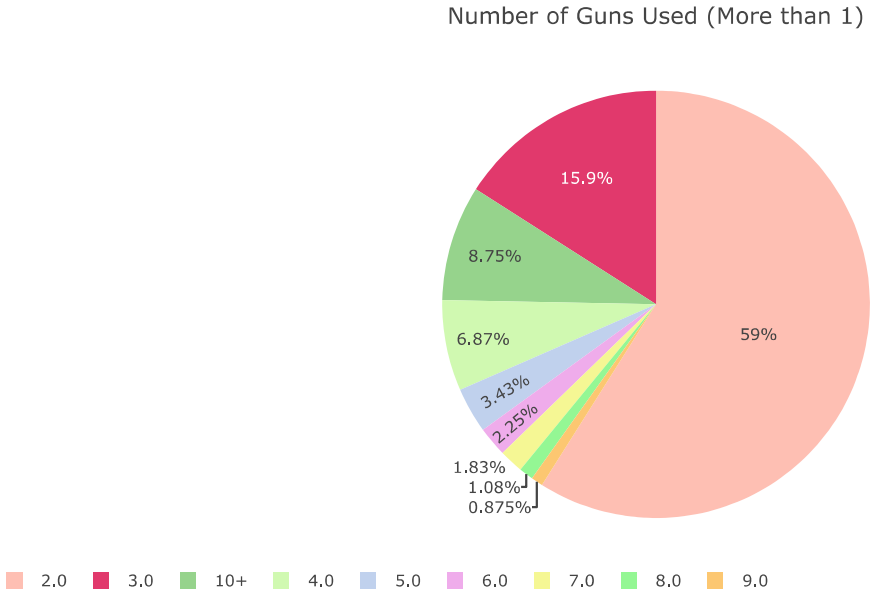


The above word cloud depicts that the most significant number of gun violence incidents are reported in high schools, apartments, parks, gas stations, Walmart, motels, bars, clubs, etc.

Other Characteristics of Gun Violence Incidents

Number of Guns Used (greater than 1)

In [13]:
```
df['n_guns'] = df['n_guns_involved'].apply(lambda x : "10+" if x>=10 else str(x))

tempdf = df['n_guns'].value_counts().reset_index()
tempdf = tempdf[tempdf['index'] != 'nan']
tempdf = tempdf[tempdf['index'] != '1.0']

labels = list(tempdf['index'])
values = list(tempdf['n_guns'])

trace1 = go.Pie(labels=labels, values=values, marker=dict(colors = ['#FEBFB3', '#E1396C', '#96D38C', '#D0F9B1', '#c0d1ed', '#efaceb', '#f
layout = dict(height=500, title='Number of Guns Used (More than 1)', legend=dict(orientation="h"));
fig = go.Figure(data=[trace1], layout=layout)
iplot(fig)
```



Most Serious Gun Violence Incidents

Let's identify the most serious gun violence incidents that have occurred in the recent past. The seriousness of these incidents is defined by the total number of fatalities and injuries.

In [14]:
```
df1 = df.sort_values(['loss'], ascending=[False])
df1[['date', 'state', 'city_or_county', 'address', 'n_killed', 'n_injured']].head(10)
```

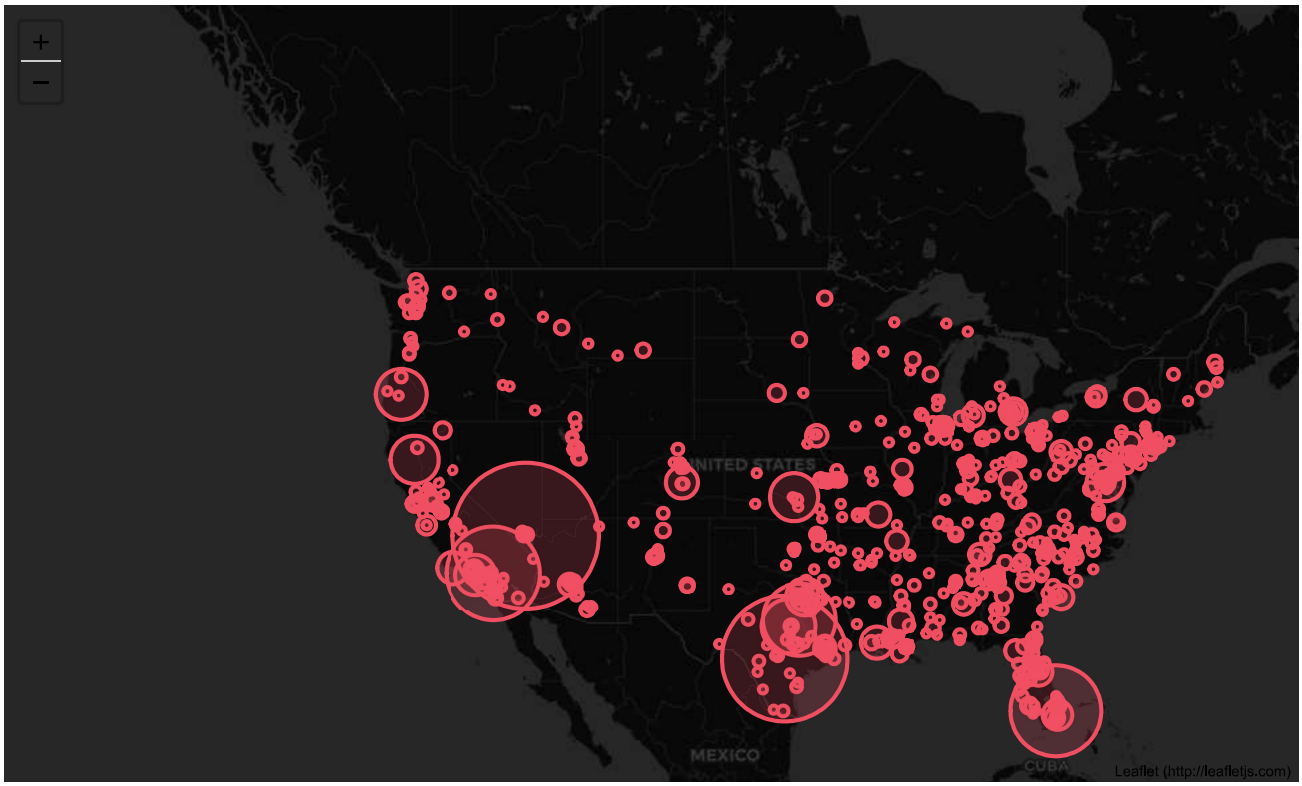
Out[14]:

	date	state	city_or_county	address	n_killed	n_injured
239677	2017-10-01	Nevada	Las Vegas	Mandalay Bay 3950 Blvd S	59	489
130448	2016-06-12	Florida	Orlando	1912 S Orange Avenue	50	53
217151	2017-11-05	Texas	Sutherland Springs	216 4th St	27	20
101531	2015-12-02	California	San Bernardino	1365 South Waterman Avenue	16	19
232745	2018-02-14	Florida	Pompano Beach (Parkland)	5901 Pine Island Rd	17	17
70511	2015-05-17	Texas	Waco	4671 S Jack Kultgen Fwy	9	18
195845	2017-07-01	Arkansas	Little Rock	220 W 6th St	0	25
137328	2016-07-25	Florida	Fort Myers	3580 Evans Ave	2	19
11566	2014-04-02	Texas	Fort Hood	Motor Pool Road and Tank Destroyer Boulevard	4	16
92624	2015-10-01	Oregon	Roseburg	1140 Umpqua College Rd	10	9

Let's plot the locations of these incidents on the map.

In [15]:
```
smalldf = df[df['n_killed'] >= 3][['latitude', 'longitude', 'loss', 'n_killed']].dropna()
map1 = folium.Map(location=[39.50, -98.35], tiles='CartoDB dark_matter', zoom_start=3.5)
map2 = folium.Map([39.50, -98.35], zoom_start=3.5, tiles='cartodbdark_matter')
markers = []
for i, row in smalldf.iterrows():
    loss = row['loss']
    if row['loss'] > 100:
        loss = row['loss']*0.1
    folium.CircleMarker([float(row['latitude']), float(row['longitude'])], radius=float(loss), color='#ef4f61', fill=True).add_to(map1)
# markers.append([float(row['Latitude']), float(row['Longitude']), row['n_killed']])
# map2.add_child(plugins.HeatMap(markers, radius=12))
map1
```

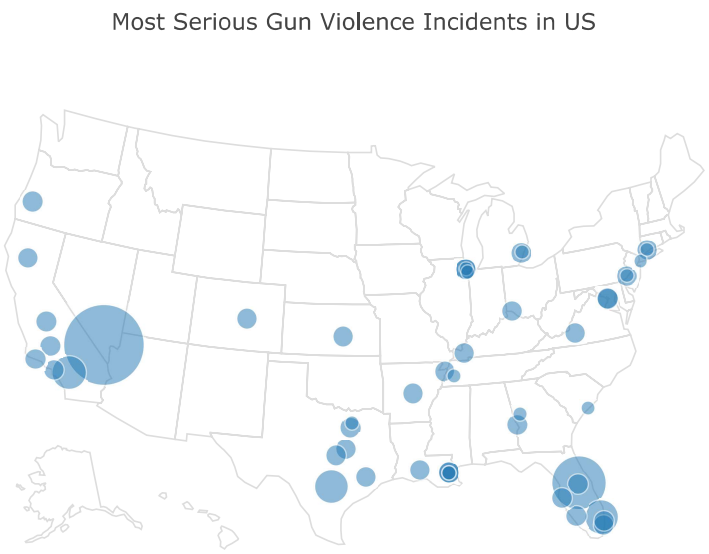
Out[15]:



Let's plot the top 10 incidents by the loss

In [16]:
```
limits = [(0,2),(3,10),(11,20),(21,50),(50,3000)]
colors = ["rgb(0,116,217)","rgb(255,65,54)","rgb(133,20,75)","rgb(255,133,27)","lightgrey"]


def returnsize(x):
    if x > 200:
        return 60
    elif x > 80:
        return 40
    elif x > 30:
        return 25
    elif x > 10:
        return 15
    elif x > 5:
        return 10
```

```
    else:
        return 5

df1['size'] = df1['loss'].apply(lambda x: returnsize(x))
dfs = df1.head(50)

data = [ dict(
        type = 'scattergeo',
        locationmode = 'USA-states',
        lon = dfs['longitude'],
        lat = dfs['latitude'],
        text = dfs['city_or_county'],
        mode = 'markers',
        marker = dict(
            size = dfs['size'],
            opacity = 0.5,
            cmin = 0,
        ))]

layout = dict(
        title = 'Most Serious Gun Violence Incidents in US',
        colorbar = True,
        geo = dict(
            projection=dict( type='albers usa' ),
            subunitcolor = "rgb(221, 221, 221)",
            subunitwidth = 1.0
        ),
    )

fig = dict( data=data, layout=layout )
iplot( fig, validate=False)
```

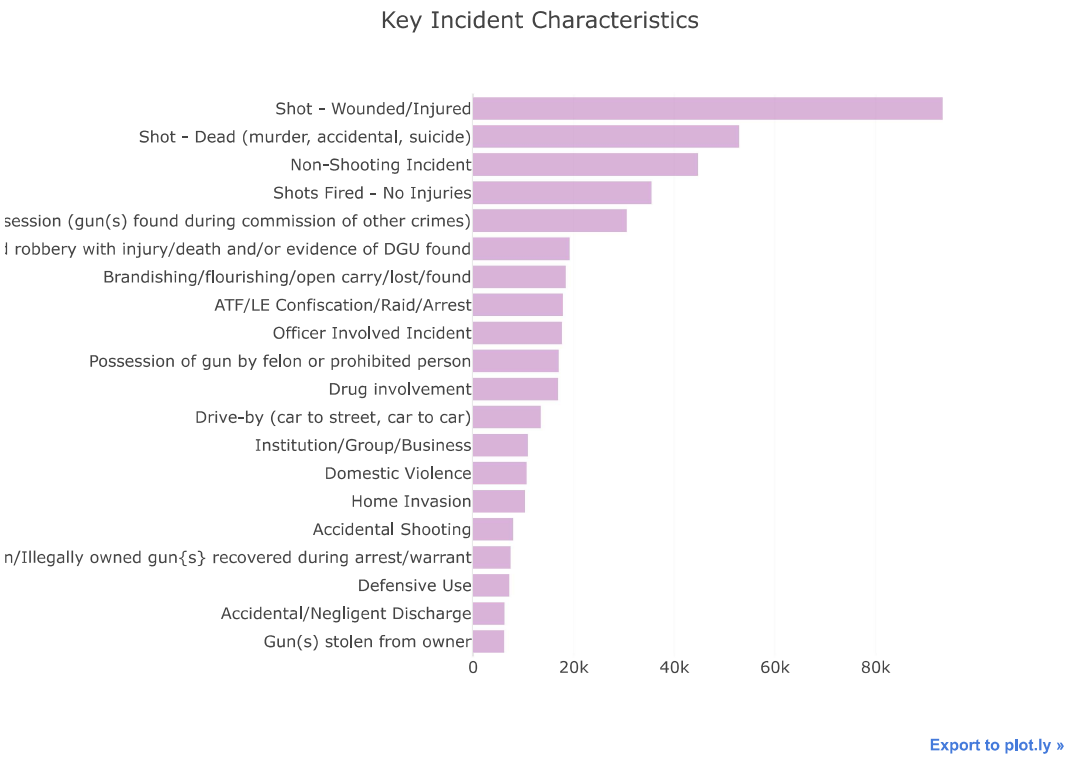


The Las Vegas shootings were one of the most serious gun violence incidents in which more than 600 people were harmed, including 60 fatalities. The Orlando shootings in 2016, in which about 50 people were killed and another 50 were injured, represent another significant gun violence event in the past few years. Another recent incident was the Texas shootings in November 2017, which resulted in the deaths of more than 25 people.

Key characteristics of Gun Violence Incidents

In [17]:
```
from collections import Counter
big_text = "||".join(df['incident_characteristics'].dropna()).split("||")
incidents = Counter(big_text).most_common(20)
xx = [x[0] for x in incidents]
yy = [x[1] for x in incidents]

trace1 = go.Bar(
    x=yy[::-1],
    y=xx[::-1],
    name='Incident Characterisitcs',
    marker=dict(color='purple'),
    opacity=0.3,
    orientation="h"
)
data = [trace1]
layout = go.Layout(
    barmode='group',
    margin=dict(l=350),
    width=800,
```

```
    height=600,
    legend=dict(dict(x=-.1, y=1.2)),
    title = 'Key Incident Characteristics',
)

fig = go.Figure(data=data, layout=layout)
iplot(fig, filename='grouped-bar')
```

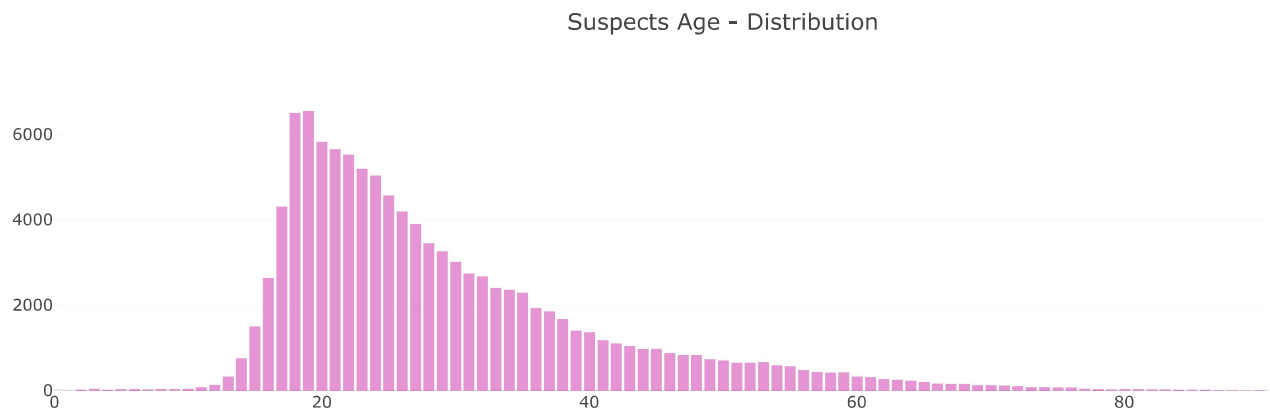


Prominent Age of Gun Violence Suspects

Let's determine the main age groups of the gun violence suspects from the data. I've written code to extract this information by combining multiple columns.

In [18]:
```
def get_user_mapping(txt):
    if txt == "NA":
        return {}
    mapping = {}
    for d in txt.split("||"):
        try:
            key = d.split("::")[0]
            val = d.split("::")[1]
            if key not in mapping:
                mapping[key] = val
        except:
            pass
    return mapping

df['participant_type'] = df['participant_type'].fillna("NA")
df['participant_type_map'] = df['participant_type'].apply(lambda x : get_user_mapping(x))
df['participant_age'] = df['participant_age'].fillna("NA")
df['participant_age_map'] = df['participant_age'].apply(lambda x : get_user_mapping(x))
df['participant_gender'] = df['participant_gender'].fillna("NA")
df['participant_gender_map'] = df['participant_gender'].apply(lambda x : get_user_mapping(x))

## Finding the Suspect Age Groups
suspect_age_groups = {}
for i, row in df.iterrows():
    suspects = []
    for k,v in row['participant_type_map'].items():
        if "suspect" in v.lower():
            suspects.append(k)
    for suspect in suspects:
        if suspect in row['participant_age_map']:
            ag = row['participant_age_map'][suspect]
            if ag not in suspect_age_groups:
                suspect_age_groups[ag] = 0
            else:
                suspect_age_groups[ag] += 1

trace1 = go.Bar(x=list(suspect_age_groups.keys()), y=list(suspect_age_groups.values()), opacity=0.75, name="month", marker=dict(color='rgb
layout = dict(height=400, title='Suspects Age - Distribution', xaxis=dict(range=[0, 100]), legend=dict(orientation="h"));
```

```
fig = go.Figure(data=[trace1], layout=layout)
iplot(fig)
```



We can observe that teenagers aged 18 to 19 are primarily involved in gun violence incidents, which is a disheartening insight.

Prominent Age of Gun Violence Victims

Similarly, let's plot the ages of people affected in gun violence incidents.

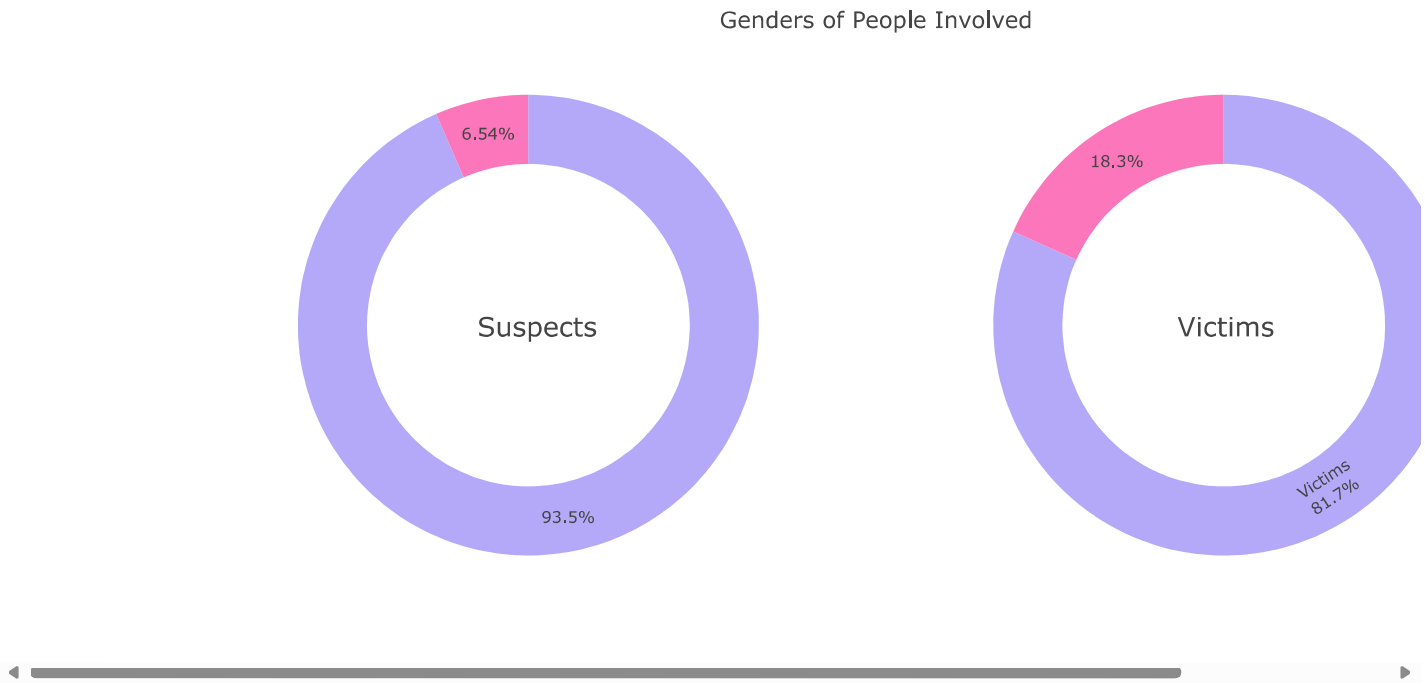
In [19]:
```
victim_age_groups = {}
for i, row in df.iterrows():
    victims = []
    for k,v in row['participant_type_map'].items():
        if "victim" in v.lower():
            victims.append(k)
    for victim in victims:
        if victim in row['participant_age_map']:
            ag = row['participant_age_map'][victim]
            if ag not in victim_age_groups:
                victim_age_groups[ag] = 0
            else:
                victim_age_groups[ag] += 1

trace1 = go.Bar(x=list(victim_age_groups.keys()), y=list(victim_age_groups.values()), opacity=0.75, name="month", marker=dict(color='brow
layout = dict(height=400, title='Victims Age - Distribution', xaxis=dict(range=[0, 100]), legend=dict(orientation="h"));
fig = go.Figure(data=[trace1], layout=layout)
iplot(fig)
```

Gender Groups - Victims and Suspects

Let's examine the distribution of gender among people involved in the incidents, both suspects and victims.

In [20]:
```
## find gender groups
suspect_age_groups = {}
victim_age_groups = {}
for i, row in df.iterrows():
    suspects = []
```

```
    for k,v in row['participant_type_map'].items():
        if "suspect" in v.lower():
            suspects.append(k)
    for suspect in suspects:
        if suspect in row['participant_gender_map']:
            ag = row['participant_gender_map'][suspect]
            if ag not in suspect_age_groups:
                suspect_age_groups[ag] = 0
            else:
                suspect_age_groups[ag] += 1

    victims = []
    for k,v in row['participant_type_map'].items():
        if "victim" in v.lower():
            victims.append(k)
    for victim in victims:
        if victim in row['participant_gender_map']:
            ag = row['participant_gender_map'][victim]
            if ag not in victim_age_groups:
                victim_age_groups[ag] = 0
            else:
                victim_age_groups[ag] += 1

del victim_age_groups['Male, female']
fig = {
  "data": [
    {
      "values": list(suspect_age_groups.values()),
      "labels": list(suspect_age_groups.keys()),
      "marker" : dict(colors=['#fc76bb','#b4a9f9']),
      "name": "Suspects",
      "domain": {"x": [0, .48]},
      "hoverinfo":"label+percent+name",
      "hole": .7,
      "type": "pie"
    },
    {
      "values": list(victim_age_groups.values()),
      "labels": list(victim_age_groups.keys()),
      "text": ["Victims"],
      "marker" : dict(colors=['#b4a9f9','#fc76bb']),
      "textposition":"inside",
      "domain": {"x": [.52, 1]},
      "name": "Victims",
      "hoverinfo":"label+percent+name",
      "hole": .7,
      "type": "pie"
    }],
  "layout": {
          "showlegend" : False,
        "title":"Genders of People Involved",
        "annotations": [
            {
                "font": {
                    "size": 20
                },
                "showarrow": False,
                "text": "Suspects",
                "x": 0.20,
                "y": 0.5
            },
            {
                "font": {
                    "size": 20
                },
                "showarrow": False,
                "text": "Victims",
                "x": 0.8,
                "y": 0.5
            }
        ]
    }
}
iplot(fig, filename='donut')
```



Analysis of the Number of Guns Registered by State

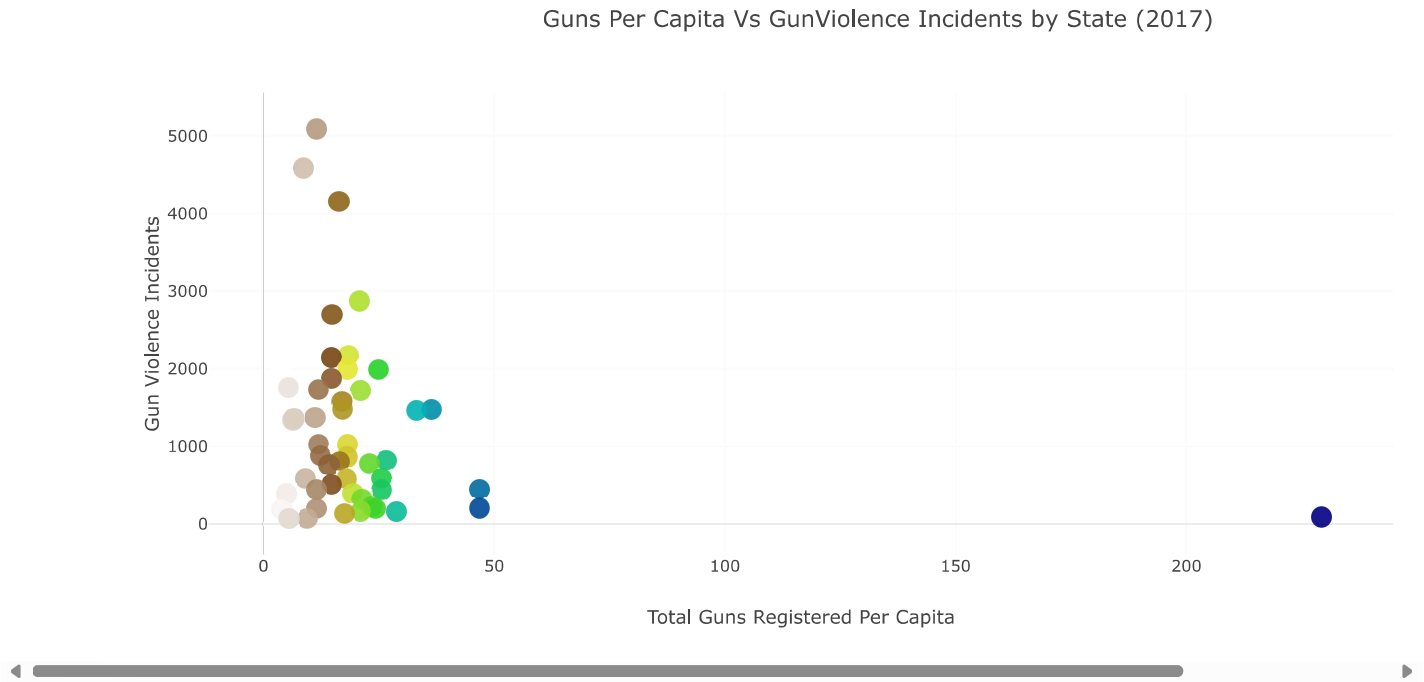
Does having more guns lead to more gun violence incidents?

According to The Washington Post, there are more than 350 million guns in the U.S. It will be interesting to determine whether states with a higher number of guns per capita also experience a higher number of gun violence incidents, and vice versa. I obtained data for the year 2017 from the following source, which includes information on guns per capita and total guns registered by state.

Source of Data - https://www.thoughtco.com/gun-owners-percentage-of-state-populations-3325153

In [21]:
```
from io import StringIO

ownership = StringIO("""Rank,state,guns_per_capita,guns_registered
1,Wyoming,229.24,132806
2,Washington D.C.,68.05,47228
3,New Hampshire,46.76,64135
4,New Mexico,46.73,97580
5,Virginia,36.34,307822
6,Alabama,33.15,161641
7,Idaho,28.86,49566
8,Arkansas,26.57,79841
9,Nevada,25.64,76888
10,Arizona,25.61,179738
11,Louisiana,24.94,116831
12,South Dakota,24.29,21130
13,Utah,23.48,72856
14,Connecticut,22.96,82400
15,Alaska,21.38,15824
16,Montana,21.06,22133
17,South Carolina,21.01,105601
18,Texas,20.79,588696
19,West Virginia,19.42,35264
20,Pennsylvania,18.45,236377
21,Georgia,18.22,190050
22,Kentucky,18.2,81068
23,Oklahoma,18.13,71269
24,Kansas,18.06,52634
25,North Dakota,17.56,13272
26,Indiana,17.1,114019
27,Maryland,17.03,103109
28,Colorado,16.48,92435
29,Florida,16.35,343288
30,Ohio,14.87,173405
31,North Carolina,14.818,152238
32,Oregon,14.816,61383
33,Tennessee,14.76,99159
34,Minnesota,14.22,79307
35,Washington,12.4,91835
36,Missouri,11.94,72996
37,Mississippi,11.89,35494
38,Nebraska,11.57,22234
39,Maine,11.5,15371
40,Illinois,11.44,146487
41,Wisconsin,11.19,64878
```

```
42,Vermont,9.41,5872
43,Iowa,9.05,28494
44,California,8.71,344622
45,Michigan,6.59,65742
46,New Jersey,6.38,57507
47,Hawaii,5.5,7859
48,Massachusetts,5.41,37152
49,Delaware,5.04,4852
50,Rhode Island,3.98,37152
51,New York,3.83,76207""")

ownership = pd.read_csv(ownership)

states_df = df[df['year'] == 2017]['state'].value_counts()
statesdf = pd.DataFrame()
statesdf['state'] = states_df.index
statesdf['counts'] = states_df.values
statesdf = statesdf.merge(ownership, on='state')

data = [
    {
        'x': statesdf['guns_per_capita'],
        'y': statesdf['counts'],
        'mode': 'markers',
        'text' : statesdf['state'],
        'marker': {
            'color': statesdf['Rank'],
            'size': 15,
            'showscale' : True,
            'colorscale' : 'Earth',
            "colorbar" : dict(
                title='Rank (guns_per_capita)'
            ),
            'opacity': 0.9
        }
    }
]

layout = go.Layout(title="Guns Per Capita Vs GunViolence Incidents by State (2017)",
                   xaxis=dict(title='Total Guns Registered Per Capita'),
                   yaxis=dict(title='Gun Violence Incidents')

                  )
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```

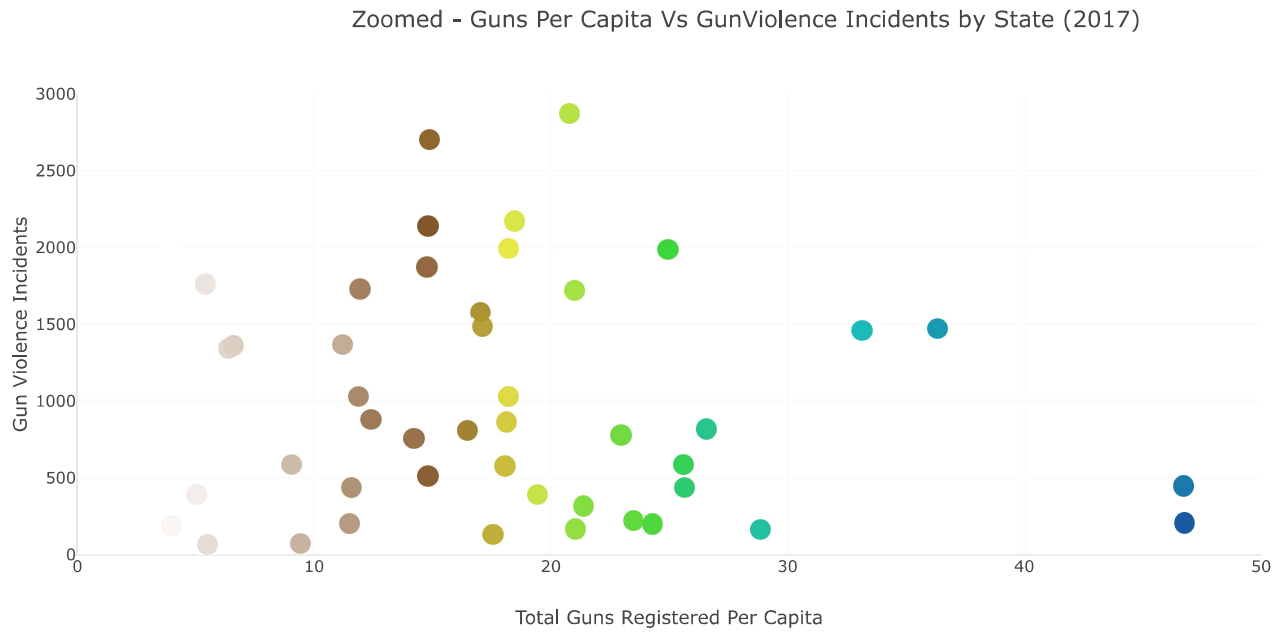


According to this data, Wyoming had the highest number of guns registered per capita, equal to 229. However, there were only 91 incidents of gun violence in 2017. In contrast, states such as California, Illinois, and Florida had lower numbers of guns registered per capita but significantly higher numbers of incidents, as evident in the top left corner of the above graph.

Let's zoom in and examine the states after removing some outliers, such as Wyoming, California, Illinois, and Florida.

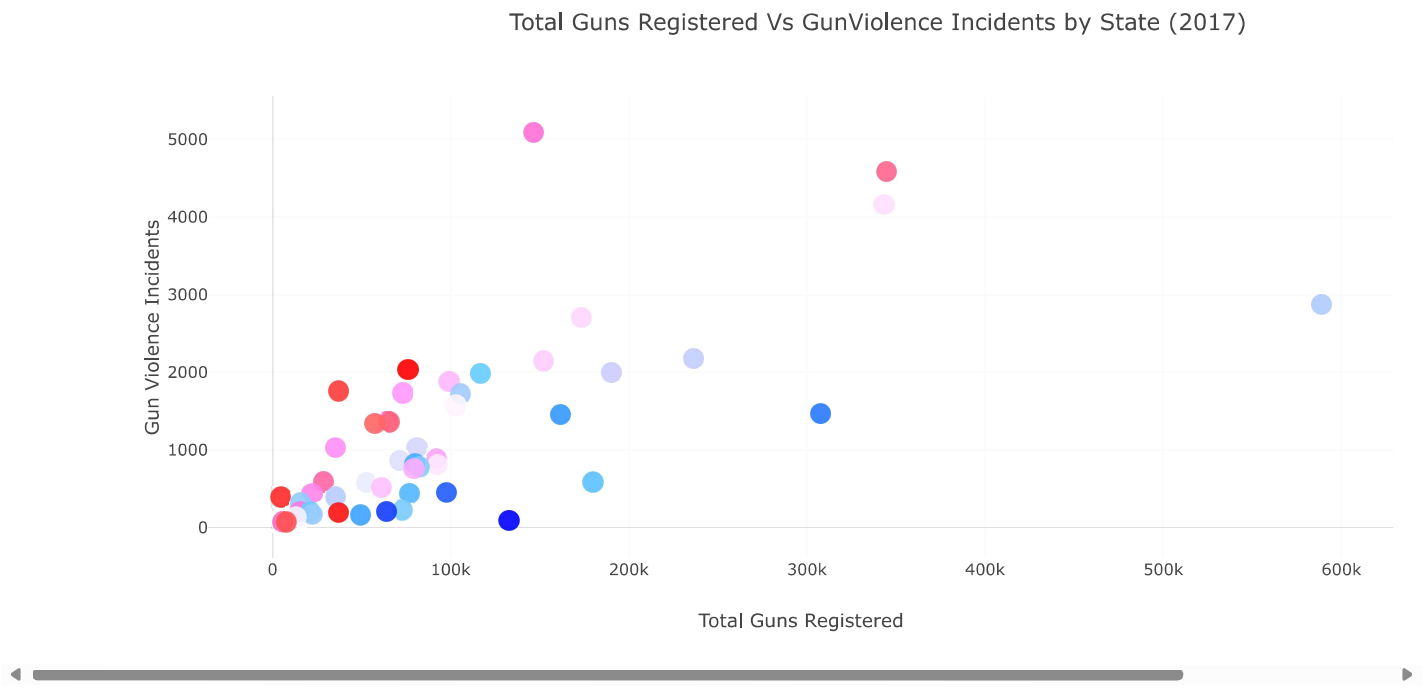
In [22]:
```
#zooomed in

data = [
    {
        'x': statesdf['guns_per_capita'],
```

```
        'y': statesdf['counts'],
        'mode': 'markers',
        'text' : statesdf['state'],
        'marker': {
            'color': statesdf['Rank'],
            'size': 15,
            'showscale' : True,
            'colorscale' : 'Earth',
            "colorbar" : dict(
                title='Rank (guns_per_capita)'
            ),
            'opacity': 0.9
        }
    }
]

layout = go.Layout(title="Zoomed - Guns Per Capita Vs GunViolence Incidents by State (2017)",
                   xaxis=dict(title='Total Guns Registered Per Capita', range =[0, 50]),
                   yaxis=dict(title='Gun Violence Incidents', range=[0, 3000])

                  )
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```



Let's plot the total number of registered guns instead of guns per capita.

In [23]:
```
data = [
    {
        'x': statesdf['guns_registered'],
        'y': statesdf['counts'],
        'mode': 'markers',
        'text' : statesdf['state'],
        'marker': {
            'color': statesdf['Rank'],
            'size': 15,
            'showscale' : True,
            'colorscale' : 'Picnic',
            "colorbar" : dict(
                title='Rank (guns_per_capita)'
            ),
            'opacity': 0.9
        }
    }
]


layout = go.Layout(title="Total Guns Registered Vs GunViolence Incidents by State (2017)",
                   xaxis=dict(title='Total Guns Registered'),
                   yaxis=dict(title='Gun Violence Incidents'))
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```



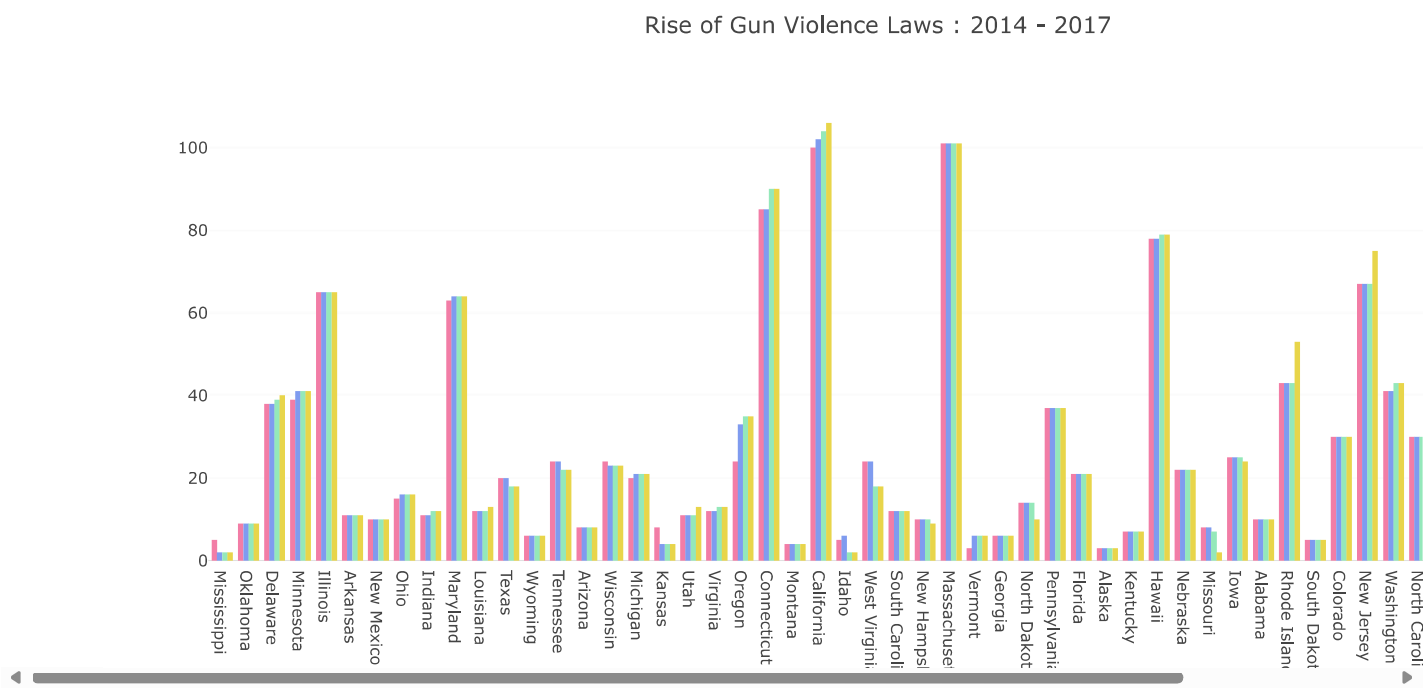
So, we can see that people in Texas own the highest number of guns across America, but the number of incidents is much higher in a few other states. However, the numbers for California and Florida suggest that owning more guns is associated with a higher number of gun violence incidents.

Gun Laws Impact on Gun Violence Incidents

Another important metric to analyze is the stringency of gun laws in different states and how they correlate with the number of gun violence incidents. I have obtained gun laws data for the years 2014 to 2018 from the following source:

Let's also examine the rise of gun violence laws in different states from 2014 to 2018.

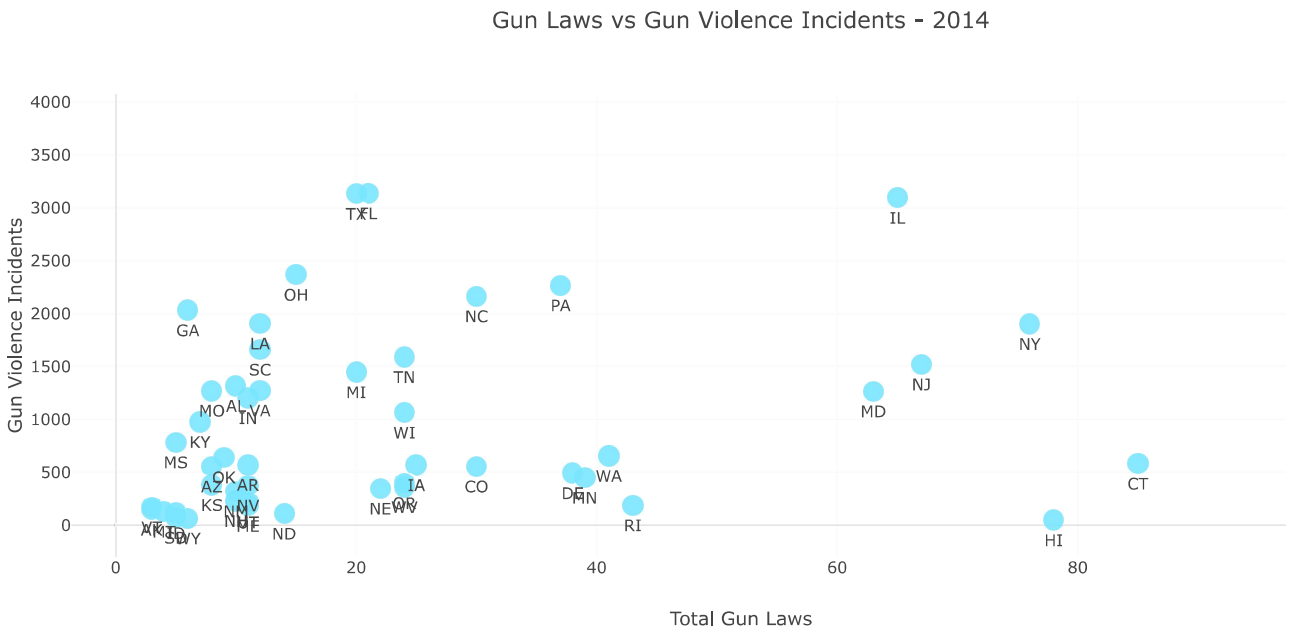
In [24]:
```
state_to_code = {'District of Columbia' : 'dc','Mississippi': 'MS', 'Oklahoma': 'OK', 'Delaware': 'DE', 'Minnesota': 'MN', 'Illinois': 'I
laws_2014 = {'Mississippi': 5, 'Oklahoma': 9, 'Delaware': 38, 'Minnesota': 39, 'Illinois': 65, 'Arkansas': 11, 'New Mexico': 10, 'Ohio': 
laws_2015 = {'Mississippi': 2, 'Oklahoma': 9, 'Delaware': 38, 'Minnesota': 41, 'Illinois': 65, 'Arkansas': 11, 'New Mexico': 10, 'Ohio': 
laws_2016 = {'Mississippi': 2, 'Oklahoma': 9, 'Delaware': 39, 'Minnesota': 41, 'Illinois': 65, 'Arkansas': 11, 'New Mexico': 10, 'Ohio': 
laws_2017 = {'Mississippi': 2, 'Oklahoma': 9, 'Delaware': 40, 'Minnesota': 41, 'Illinois': 65, 'Arkansas': 11, 'New Mexico': 10, 'Ohio': 

y1, y2, y3, y4 = [], [], [], []
x1 = []
for x, y in laws_2014.items():
    y1.append(y)
    y2.append(laws_2015[x])
    y3.append(laws_2016[x])
    y4.append(laws_2017[x])
    x1.append(x)

trace1 = go.Bar(x=x1,y=y1,name='2014', marker=dict(color="#f27da6"))
trace2 = go.Bar(x=x1,y=y2,name='2015', marker=dict(color="#7f9bef"))
trace3 = go.Bar(x=x1,y=y3,name='2016', marker=dict(color="#94e8ba"))
trace4 = go.Bar(x=x1,y=y4,name='2017', marker=dict(color="#e8d54a"))

data = [trace1, trace2, trace3, trace4]
layout = go.Layout(barmode='group', title="Rise of Gun Violence Laws : 2014 - 2017")
fig = go.Figure(data=data, layout=layout)
iplot(fig, filename='grouped-bar')
```



Impact of Number of Gun Laws vs. Gun Violence Incidents: 2014 - 2017

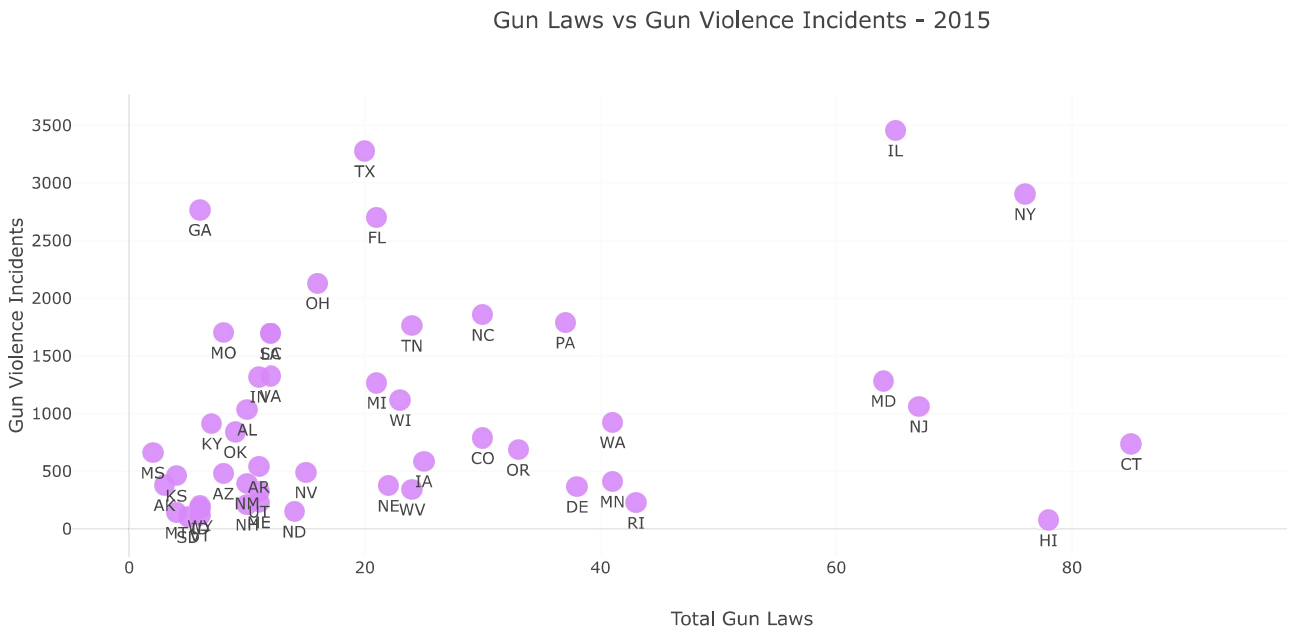
Let's analyze the relationship between the number of gun violence laws and the incidents of gun violence for different states over these years.

In [25]:
```
states_df = df[df['year'] == 2014]['state'].value_counts()
statesdf = pd.DataFrame()
statesdf['state'] = states_df.index
statesdf['counts'] = states_df.values
statesdf['laws'] = statesdf['state'].apply(lambda x : laws_2014[x] if x in laws_2014 else "")

statesdf['state'] = statesdf['state'].apply(lambda x : state_to_code[x])

data = [
    {
        'x': statesdf['laws'],
        'y': statesdf['counts'],
        'mode': 'markers+text',
        'text' : statesdf['state'],
        'textposition' : 'bottom center',
        'marker': {
            'color': "#7ae6ff",
            'size': 15,
            'opacity': 0.9
        }
    }
]

layout = go.Layout(title="Gun Laws vs Gun Violence Incidents - 2014",
                   xaxis=dict(title='Total Gun Laws'),
                   yaxis=dict(title='Gun Violence Incidents')
                  )
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```



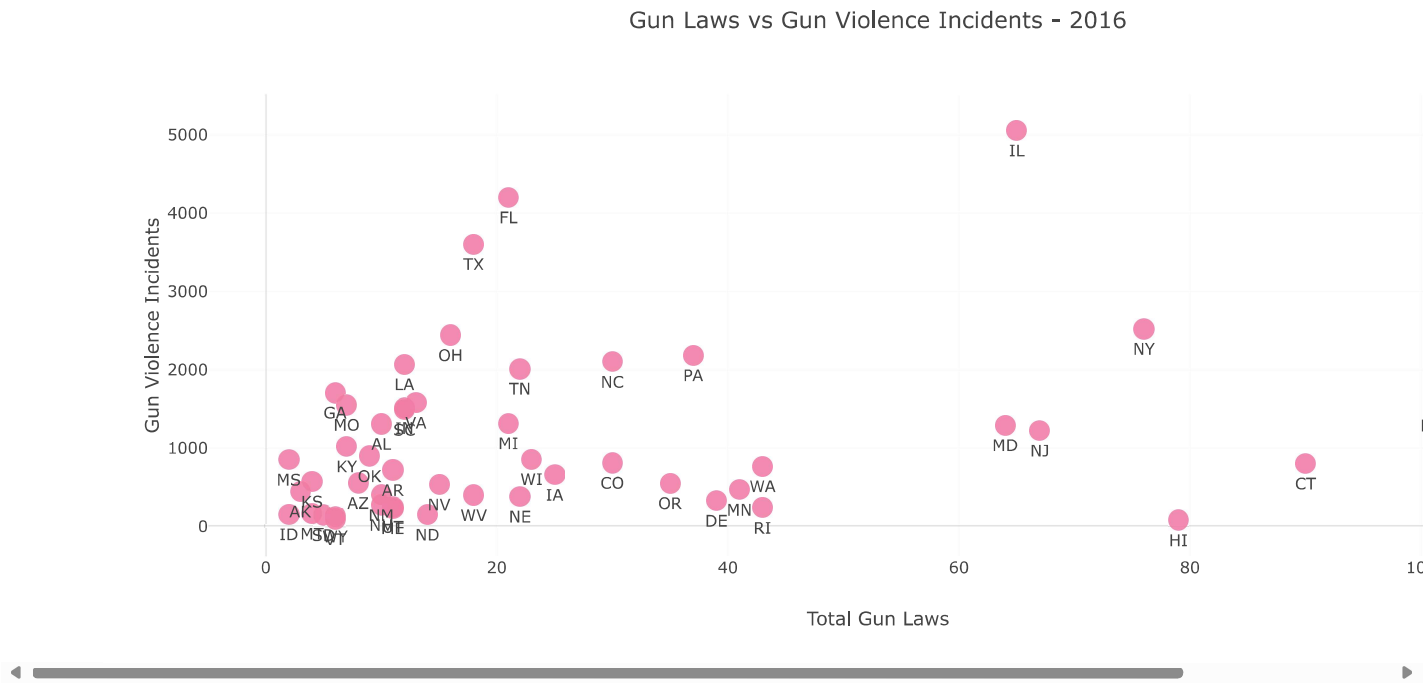
In [26]:

```
states_df = df[df['year'] == 2015]['state'].value_counts()
statesdf = pd.DataFrame()
statesdf['state'] = states_df.index
statesdf['counts'] = states_df.values
statesdf['laws'] = statesdf['state'].apply(lambda x : laws_2015[x] if x in laws_2015 else "")

statesdf['state'] = statesdf['state'].apply(lambda x : state_to_code[x])

data = [
    {
        'x': statesdf['laws'],
        'y': statesdf['counts'],
        'mode': 'markers+text',
        'text' : statesdf['state'],
        'textposition' : 'bottom center',
        'marker': {
            'color': "#d889f9",
            'size': 15,
            'opacity': 0.9
        }
    }
]

layout = go.Layout(title="Gun Laws vs Gun Violence Incidents - 2015",
                   xaxis=dict(title='Total Gun Laws'),
                   yaxis=dict(title='Gun Violence Incidents')
                  )
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```



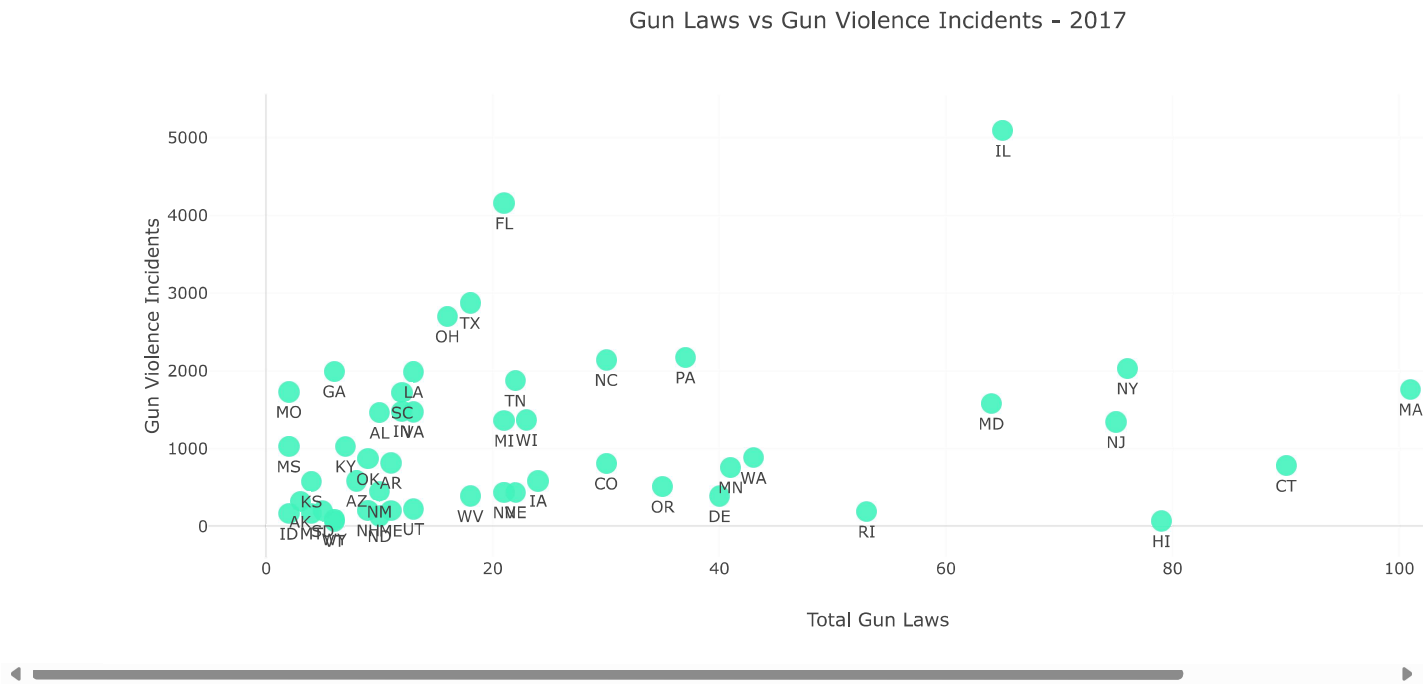
In [27]:

```
states_df = df[df['year'] == 2016]['state'].value_counts()
statesdf = pd.DataFrame()
statesdf['state'] = states_df.index
statesdf['counts'] = states_df.values
statesdf['laws'] = statesdf['state'].apply(lambda x : laws_2016[x] if x in laws_2016 else "")

statesdf['state'] = statesdf['state'].apply(lambda x : state_to_code[x])

data = [
    {
        'x': statesdf['laws'],
        'y': statesdf['counts'],
        'mode': 'markers+text',
        'text' : statesdf['state'],
        'textposition' : 'bottom center',
        'marker': {
            'color': "#f27da6",
            'size': 15,
            'opacity': 0.9
        }
    }
]

layout = go.Layout(title="Gun Laws vs Gun Violence Incidents - 2016",
                   xaxis=dict(title='Total Gun Laws'),
                   yaxis=dict(title='Gun Violence Incidents')
                  )
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```



In [28]:

```
states_df = df[df['year'] == 2017]['state'].value_counts()
statesdf = pd.DataFrame()
statesdf['state'] = states_df.index
statesdf['counts'] = states_df.values
statesdf['laws'] = statesdf['state'].apply(lambda x : laws_2017[x] if x in laws_2017 else "")

statesdf['state'] = statesdf['state'].apply(lambda x : state_to_code[x])

data = [
    {
        'x': statesdf['laws'],
        'y': statesdf['counts'],
        'mode': 'markers+text',
        'text' : statesdf['state'],
        'textposition' : 'bottom center',
        'marker': {
            'color': "#42f4bc",
            'size': 15,
            'opacity': 0.9
        }
    }
]

layout = go.Layout(title="Gun Laws vs Gun Violence Incidents - 2017",
                   xaxis=dict(title='Total Gun Laws'),
                   yaxis=dict(title='Gun Violence Incidents')
                  )
fig = go.Figure(data = data, layout = layout)
iplot(fig, filename='scatter-colorscale')
```



From the above plots, we can observe that states like California, Massachusetts, Hawaii, New York, New Jersey, Illinois, Maryland, and Rhode Island have a considerably higher number of gun violence laws. Among them, only California and Illinois have the highest number of gun violence incidents. In contrast, states such as Texas and Florida have considerably fewer gun violence laws but a high number of incidents.