> **Tip**: Welcome to the Investigate a Dataset project! You will find tips in quoted sections like this to help organize your approach to your investigation. Once you complete this project, remove these **Tip** sections from your report before submission. First things first, you might want to double-click this Markdown cell and change the title so that it reflects your dataset and investigation.

Project: US Census and FBI Gun Data Analysis

Table of Contents

- Introduction
- Data Wrangling
- Exploratory Data Analysis
- Conclusions

Introduction

Dataset Description

Two datasets have been selected for this analysis. The FBI's National Instant Criminal Background Check System (NCIS) data which contains the number of firearm checks by month, state, and the typecomes; and the U.S. census data which contains information about the population and demographic as of July 1, 2016 and April 1,2010. The NICS is used by to determine whether a prospective buyer is eligible to buy firearms or explosives. Gun shops call into this system to ensure that each customer does not have a criminal record or isn’t otherwise ineligible to make a purchase. Both dataset are connected by states, therefore, we can compare the changes in gun registration and population by states.

Question(s) for Analysis

Which states have the highest and the lowest gun registrations? Which states have had the highest growth in gun registrations? What is the overall trend of gun purchases? Is the change in population correlated with the change in gun purchases?

In [1]:
```
# Use this cell to set up import statements for all of the packages that you p
lan to use

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
% matplotlib inline

np.set_printoptions(suppress=True)


# Remember to include a 'magic word' so that your visualizations are plotted i
nline with the notebook. See this page for more:
# http://ipython.readthedocs.io/en/stable/interactive/magics.html
```

In [2]:
```
# Upgrade pandas to use dataframe.explode() function.
!pip install --upgrade pandas==0.25.0
```

```
Requirement already up-to-date: pandas==0.25.0 in /opt/conda/lib/python3.6/si
te-packages (0.25.0)
Requirement already satisfied, skipping upgrade: pytz>=2017.2 in /opt/conda/l
ib/python3.6/site-packages (from pandas==0.25.0) (2017.3)
Requirement already satisfied, skipping upgrade: numpy>=1.13.3 in /opt/conda/
lib/python3.6/site-packages (from pandas==0.25.0) (1.19.5)
Requirement already satisfied, skipping upgrade: python-dateutil>=2.6.1 in /o
pt/conda/lib/python3.6/site-packages (from pandas==0.25.0) (2.6.1)
Requirement already satisfied, skipping upgrade: six>=1.5 in /opt/conda/lib/p
ython3.6/site-packages (from python-dateutil>=2.6.1->pandas==0.25.0) (1.11.0)
```

Data Wrangling

In [3]:
```
# Load your data and print out a few lines. Perform operations to inspect data
types and look for instances of missing or possibly errant data.

df_gun_purchase = pd.read_excel("gun_data[1].xlsx")
df_census_data = pd.read_csv("U.S._Census_Data[1].csv")
```

In [4]:
```
#check for the data structure of the df_gun_purchase

df_gun_purchase.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12485 entries, 0 to 12484
Data columns (total 27 columns):
month                        12485 non-null object
state                        12485 non-null object
permit                       12461 non-null float64
permit_recheck               1100 non-null float64
handgun                      12465 non-null float64
long_gun                     12466 non-null float64
other                        5500 non-null float64
multiple                     12485 non-null int64
admin                        12462 non-null float64
prepawn_handgun              10542 non-null float64
prepawn_long_gun             10540 non-null float64
prepawn_other                5115 non-null float64
redemption_handgun           10545 non-null float64
redemption_long_gun          10544 non-null float64
redemption_other             5115 non-null float64
returned_handgun             2200 non-null float64
returned_long_gun            2145 non-null float64
returned_other               1815 non-null float64
rentals_handgun              990 non-null float64
rentals_long_gun             825 non-null float64
private_sale_handgun         2750 non-null float64
private_sale_long_gun        2750 non-null float64
private_sale_other           2750 non-null float64
return_to_seller_handgun     2475 non-null float64
return_to_seller_long_gun    2750 non-null float64
return_to_seller_other       2255 non-null float64
totals                       12485 non-null int64
dtypes: float64(23), int64(2), object(2)
memory usage: 2.6+ MB
```

In [5]:
```
#check for the composition of the first five rows in the df_gun_purchase

df_gun_purchase.head()
```

Out[5]:

	month	state	permit	permit_recheck	handgun	long_gun	other	multiple	admin	prep
0	2017-09	Alabama	16717.0	0.0	5734.0	6320.0	221.0	317	0.0	
1	2017-09	Alaska	209.0	2.0	2320.0	2930.0	219.0	160	0.0	
2	2017-09	Arizona	5069.0	382.0	11063.0	7946.0	920.0	631	0.0	
3	2017-09	Arkansas	2935.0	632.0	4347.0	6063.0	165.0	366	51.0	
4	2017-09	California	57839.0	0.0	37165.0	24581.0	2984.0	0	0.0	

5 rows × 27 columns

In [6]:
```
#check for the composition of the last five rows in the df_gun_purchase

df_gun_purchase.tail()
```

Out[6]:

	month	state	permit	permit_recheck	handgun	long_gun	other	multiple	admin
12480	1998-11	Virginia	0.0	NaN	14.0	2.0	NaN	8	0.0
12481	1998-11	Washington	1.0	NaN	65.0	286.0	NaN	8	1.0
12482	1998-11	West Virginia	3.0	NaN	149.0	251.0	NaN	5	0.0
12483	1998-11	Wisconsin	0.0	NaN	25.0	214.0	NaN	2	0.0
12484	1998-11	Wyoming	8.0	NaN	45.0	49.0	NaN	5	0.0

5 rows × 27 columns

In [7]:
```
#check for duplication in the df_gun_purchase dataframe

df_gun_purchase.duplicated().sum()
```

Out[7]: 0

In [8]:
```
# check sum of all null values in df_gun_purchase dataframe
df_gun_purchase.isnull().sum()
```

Out[8]:
```
month                            0
state                            0
permit                          24
permit_recheck               11385
handgun                         20
long_gun                        19
other                         6985
multiple                         0
admin                           23
prepawn_handgun               1943
prepawn_long_gun              1945
prepawn_other                 7370
redemption_handgun            1940
redemption_long_gun           1941
redemption_other              7370
returned_handgun             10285
returned_long_gun            10340
returned_other               10670
rentals_handgun              11495
rentals_long_gun             11660
private_sale_handgun          9735
private_sale_long_gun         9735
private_sale_other            9735
return_to_seller_handgun     10010
return_to_seller_long_gun     9735
return_to_seller_other       10230
totals                           0
dtype: int64
```

In [9]:
```
#check for number of rows and columns in df_gun_purchase
df_gun_purchase.shape
```

Out[9]:
```
(12485, 27)
```

In [10]:
```
# check for the datatype of each columns of the df_gun_purchase dataset
df_gun_purchase.dtypes
```

Out[10]:
```
month                         object
state                         object
permit                       float64
permit_recheck               float64
handgun                      float64
long_gun                     float64
other                        float64
multiple                       int64
admin                        float64
prepawn_handgun              float64
prepawn_long_gun             float64
prepawn_other                float64
redemption_handgun           float64
redemption_long_gun          float64
redemption_other             float64
returned_handgun             float64
returned_long_gun            float64
returned_other               float64
rentals_handgun              float64
rentals_long_gun             float64
private_sale_handgun         float64
private_sale_long_gun        float64
private_sale_other           float64
return_to_seller_handgun     float64
return_to_seller_long_gun    float64
return_to_seller_other       float64
totals                         int64
dtype: object
```

In [11]:
```
# check the number of rows and columns in the census dataframe
df_census_data.shape
```

Out[11]:
```
(85, 52)
```

In [12]:
```
# Review the first 5 rows of the dataset

df_census_data.head()
```

Out[12]:

	Fact	Fact Note	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Connecticut
0	Population estimates, July 1, 2016, (V2016)	NaN	4,863,300	741,894	6,931,071	2,988,248	39,250,017	5,540,545	3,576,452
1	Population estimates base, April 1, 2010, (V2...	NaN	4,780,131	710,249	6,392,301	2,916,025	37,254,522	5,029,324	3,574,114
2	Population, percent change - April 1, 2010 (es...	NaN	1.70%	4.50%	8.40%	2.50%	5.40%	10.20%	0.10%
3	Population, Census, April 1, 2010	NaN	4,779,736	710,231	6,392,017	2,915,918	37,253,956	5,029,196	3,574,097
4	Persons under 5 years, percent, July 1, 2016, ...	NaN	6.00%	7.30%	6.30%	6.40%	6.30%	6.10%	5.20%

5 rows × 52 columns

In [13]:
```
# Review the last 25 rows of the dataset

df_census_data.tail(25)
```

Out[13]:

	Fact	Fact Note	Alabama	Alaska	Arizona	Arkansas	California	Colorado
60	Veteran-owned firms, 2012	NaN	41,943	7,953	46,780	25,915	252,377	51,722
61	Nonveteran-owned firms, 2012	NaN	316,984	56,091	427,582	192,988	3,176,341	469,524
62	Population per square mile, 2010	NaN	94.4	1.2	56.3	56	239.1	48.5
63	Land area in square miles, 2010	NaN	50,645.33	570,640.95	113,594.08	52,035.48	155,779.22	103,641.89
64	FIPS Code	NaN	"01"	"02"	"04"	"05"	"06"	"08"
65	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
66	NOTE: FIPS Code values are enclosed in quotes ...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
67	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
68	Value Notes	NaN	NaN	NaN	NaN	NaN	NaN	NaN
69	1	Includes data not distributed by county.	NaN	NaN	NaN	NaN	NaN	NaN
70	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
71	Fact Notes	NaN	NaN	NaN	NaN	NaN	NaN	NaN
72	(a)	Includes persons reporting only one race	NaN	NaN	NaN	NaN	NaN	NaN
73	(b)	Hispanics may be of any race, so also are incl...	NaN	NaN	NaN	NaN	NaN	NaN
74	(c)	Economic Census - Puerto Rico data are not com...	NaN	NaN	NaN	NaN	NaN	NaN
75	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
76	Value Flags	NaN	NaN	NaN	NaN	NaN	NaN	NaN
77	-	Either no or too few sample observations were ...	NaN	NaN	NaN	NaN	NaN	NaN

	Fact	Fact Note	Alabama	Alaska	Arizona	Arkansas	California	Colorado
78	D	Suppressed to avoid disclosure of confidential...	NaN	NaN	NaN	NaN	NaN	NaN
79	F	Fewer than 25 firms	NaN	NaN	NaN	NaN	NaN	NaN
80	FN	Footnote on this item in place of data	NaN	NaN	NaN	NaN	NaN	NaN
81	NaN	Not available	NaN	NaN	NaN	NaN	NaN	NaN
82	S	Suppressed; does not meet publication standards	NaN	NaN	NaN	NaN	NaN	NaN
83	X	Not applicable	NaN	NaN	NaN	NaN	NaN	NaN
84	Z	Value greater than zero but less than half uni...	NaN	NaN	NaN	NaN	NaN	NaN

25 rows × 52 columns

In [14]:
```
# check the data structure of the census data
df_census_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 85 entries, 0 to 84
Data columns (total 52 columns):
Fact                80 non-null object
Fact Note           28 non-null object
Alabama             65 non-null object
Alaska              65 non-null object
Arizona             65 non-null object
Arkansas            65 non-null object
California          65 non-null object
Colorado            65 non-null object
Connecticut         65 non-null object
Delaware            65 non-null object
Florida             65 non-null object
Georgia             65 non-null object
Hawaii              65 non-null object
Idaho               65 non-null object
Illinois            65 non-null object
Indiana             65 non-null object
Iowa                65 non-null object
Kansas              65 non-null object
Kentucky            65 non-null object
Louisiana           65 non-null object
Maine               65 non-null object
Maryland            65 non-null object
Massachusetts       65 non-null object
Michigan            65 non-null object
Minnesota           65 non-null object
Mississippi         65 non-null object
Missouri            65 non-null object
Montana             65 non-null object
Nebraska            65 non-null object
Nevada              65 non-null object
New Hampshire       65 non-null object
New Jersey          65 non-null object
New Mexico          65 non-null object
New York            65 non-null object
North Carolina      65 non-null object
North Dakota        65 non-null object
Ohio                65 non-null object
Oklahoma            65 non-null object
Oregon              65 non-null object
Pennsylvania        65 non-null object
Rhode Island        65 non-null object
South Carolina      65 non-null object
South Dakota        65 non-null object
Tennessee           65 non-null object
Texas               65 non-null object
Utah                65 non-null object
Vermont             65 non-null object
Virginia            65 non-null object
Washington          65 non-null object
West Virginia       65 non-null object
Wisconsin           65 non-null object
Wyoming             65 non-null object
dtypes: object(52)
memory usage: 34.7+ KB
```

In [15]:
```
#check for duplication in the census dataset
print(df_census_data.duplicated().sum())
df_census_data[df_census_data.duplicated()]
```

3

Out[15]:

	Fact	Fact Note	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Connecticut	Delawa
67	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
70	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
75	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na

3 rows × 52 columns

In [16]:
```
# check sum of all null values in census dataset
null_data = df_census_data.isnull().sum()
null_data
```

Out[16]:

```
Fact                5
Fact Note          57
Alabama            20
Alaska             20
Arizona            20
Arkansas           20
California         20
Colorado           20
Connecticut        20
Delaware           20
Florida            20
Georgia            20
Hawaii             20
Idaho              20
Illinois           20
Indiana            20
Iowa               20
Kansas             20
Kentucky           20
Louisiana          20
Maine              20
Maryland           20
Massachusetts      20
Michigan           20
Minnesota          20
Mississippi        20
Missouri           20
Montana            20
Nebraska           20
Nevada             20
New Hampshire      20
New Jersey         20
New Mexico         20
New York           20
North Carolina     20
North Dakota       20
Ohio               20
Oklahoma           20
Oregon             20
Pennsylvania       20
Rhode Island       20
South Carolina     20
South Dakota       20
Tennessee          20
Texas              20
Utah               20
Vermont            20
Virginia           20
Washington         20
West Virginia      20
Wisconsin          20
Wyoming            20
dtype: int64
```

In [17]:
```
# check if the last 20 data constitute the null values in census dataset
null_tail20 = df_census_data.tail(20).isnull().sum()
null_data == null_tail20
```

Out[17]:
```
Fact               True
Fact Note         False
Alabama            True
Alaska             True
Arizona            True
Arkansas           True
California         True
Colorado           True
Connecticut        True
Delaware           True
Florida            True
Georgia            True
Hawaii             True
Idaho              True
Illinois           True
Indiana            True
Iowa               True
Kansas             True
Kentucky           True
Louisiana          True
Maine              True
Maryland           True
Massachusetts      True
Michigan           True
Minnesota          True
Mississippi        True
Missouri           True
Montana            True
Nebraska           True
Nevada             True
New Hampshire      True
New Jersey         True
New Mexico         True
New York           True
North Carolina     True
North Dakota       True
Ohio               True
Oklahoma           True
Oregon             True
Pennsylvania       True
Rhode Island       True
South Carolina     True
South Dakota       True
Tennessee          True
Texas              True
Utah               True
Vermont            True
Virginia           True
Washington         True
West Virginia      True
Wisconsin          True
Wyoming            True
dtype: bool
```

Data Cleaning

Gun Purchase Dataset:

First, the gun purchase dataset contains information for 50 US states and 5 US territories. The census dataset contains the information for only 50 US states. The first step will be to drop the columns for the 5 US territories from the gun purchase dataset to make it comparable with the census dataset.

Second, the gun purchase dataset contains null values. These values were replaced with zero. The rows were added together and the summed values were compared with original 'totals' column to ensure that the values are the same. Replacing the null values with a different number such as the mean will cause the summed value to be different from the original 'totals' column.

Third, the 'month' column were splitted into year and month column to make it easier to group data by month and year based on data exploration need.

Fourth, switch the values in the 'state' column from uppercase to lower case and convert the spaces to underscore.

Fifth, transpose all the rows in the 'state' column to column and the use the 'total' column as new values for the transposed cells.

Census Dataset

First, drop the 'Fact Note' column and the last 21 rows becuase they have Nan values.

Second, switch the column values from uppercase to lower case and convert the spaces to underscore.

Third, remove "%" from the third row in the dataset. The reason is because the current data type is object and it must be converted to a quantitative data.

In [18]:
```
#Retrieve the list of all states in the census dataset

state_list = df_census_data.columns[2:].tolist()
state_list
```

Out[18]:
```
['Alabama',
 'Alaska',
 'Arizona',
 'Arkansas',
 'California',
 'Colorado',
 'Connecticut',
 'Delaware',
 'Florida',
 'Georgia',
 'Hawaii',
 'Idaho',
 'Illinois',
 'Indiana',
 'Iowa',
 'Kansas',
 'Kentucky',
 'Louisiana',
 'Maine',
 'Maryland',
 'Massachusetts',
 'Michigan',
 'Minnesota',
 'Mississippi',
 'Missouri',
 'Montana',
 'Nebraska',
 'Nevada',
 'New Hampshire',
 'New Jersey',
 'New Mexico',
 'New York',
 'North Carolina',
 'North Dakota',
 'Ohio',
 'Oklahoma',
 'Oregon',
 'Pennsylvania',
 'Rhode Island',
 'South Carolina',
 'South Dakota',
 'Tennessee',
 'Texas',
 'Utah',
 'Vermont',
 'Virginia',
 'Washington',
 'West Virginia',
 'Wisconsin',
 'Wyoming']
```

In [19]:
```
# Get the list of states in df_gun_purchase that do not exist in the census da
taset

states_gun_data = df_gun_purchase['state'].unique().tolist()

states = []

def update_state(data):
    for state in data:
        if state not in state_list:
            states.append(state)

update_state(states_gun_data)
states
```

Out[19]:
```
['District of Columbia',
 'Guam',
 'Mariana Islands',
 'Puerto Rico',
 'Virgin Islands']
```

In [20]:
```
#drop the states that do not exist in the census dataset

df_gun_purchase.set_index('state', inplace=True)
df_gun_purchase = df_gun_purchase.drop(states)
df_gun_purchase
```

Out[20]:

state	month	permit	permit_recheck	handgun	long_gun	other	multiple	admin
Alabama	2017-09	16717.0	0.0	5734.0	6320.0	221.0	317	0.0
Alaska	2017-09	209.0	2.0	2320.0	2930.0	219.0	160	0.0
Arizona	2017-09	5069.0	382.0	11063.0	7946.0	920.0	631	0.0
Arkansas	2017-09	2935.0	632.0	4347.0	6063.0	165.0	366	51.0
California	2017-09	57839.0	0.0	37165.0	24581.0	2984.0	0	0.0
Colorado	2017-09	4356.0	0.0	15751.0	13448.0	1007.0	1062	0.0
Connecticut	2017-09	4343.0	673.0	4834.0	1993.0	274.0	0	0.0
Delaware	2017-09	275.0	0.0	1414.0	1538.0	66.0	68	0.0
Florida	2017-09	10784.0	0.0	39199.0	17949.0	2319.0	1721	1.0
Georgia	2017-09	12074.0	0.0	10933.0	7982.0	315.0	494	0.0
Hawaii	2017-09	946.0	0.0	0.0	0.0	0.0	0	0.0
Idaho	2017-09	5162.0	0.0	3058.0	5241.0	187.0	205	0.0
Illinois	2017-09	15712.0	71432.0	18290.0	10201.0	0.0	814	0.0
Indiana	2017-09	18241.0	0.0	16093.0	11332.0	1123.0	597	79.0
Iowa	2017-09	5847.0	1217.0	151.0	2640.0	27.0	1	1.0
Kansas	2017-09	1567.0	3.0	4518.0	5025.0	306.0	297	0.0
Kentucky	2017-09	378384.0	0.0	8112.0	7543.0	253.0	543	1.0
Louisiana	2017-09	1827.0	0.0	10495.0	11573.0	635.0	776	0.0
Maine	2017-09	783.0	0.0	3026.0	4220.0	179.0	186	1.0
Maryland	2017-09	2424.0	0.0	3389.0	4897.0	168.0	34	0.0
Massachusetts	2017-09	7160.0	0.0	4749.0	2808.0	449.0	158	0.0
Michigan	2017-09	16571.0	19.0	8654.0	10676.0	379.0	163	0.0

state	month	permit	permit_recheck	handgun	long_gun	other	multiple	admin
Minnesota	2017-09	25645.0	0.0	4862.0	12677.0	346.0	273	0.0
Mississippi	2017-09	1362.0	0.0	6260.0	6035.0	206.0	405	0.0
Missouri	2017-09	791.0	0.0	16993.0	14395.0	1050.0	991	0.0
Montana	2017-09	1076.0	0.0	2395.0	4878.0	140.0	216	3.0
Nebraska	2017-09	3036.0	113.0	110.0	1989.0	11.0	4	0.0
Nevada	2017-09	1952.0	0.0	3992.0	2509.0	251.0	237	0.0
New Hampshire	2017-09	2795.0	0.0	4410.0	3248.0	132.0	3	2.0
New Jersey	2017-09	0.0	0.0	3985.0	3040.0	140.0	0	0.0
...	...	...	...	...	...	...	...	...
Massachusetts	1998-11	0.0	NaN	4.0	39.0	NaN	0	0.0
Michigan	1998-11	579.0	NaN	3.0	443.0	NaN	0	0.0
Minnesota	1998-11	9.0	NaN	27.0	280.0	NaN	3	0.0
Mississippi	1998-11	0.0	NaN	286.0	491.0	NaN	7	0.0
Missouri	1998-11	0.0	NaN	116.0	458.0	NaN	4	0.0
Montana	1998-11	0.0	NaN	101.0	98.0	NaN	2	0.0
Nebraska	1998-11	88.0	NaN	1.0	96.0	NaN	1	0.0
Nevada	1998-11	0.0	NaN	75.0	76.0	NaN	8	0.0
New Hampshire	1998-11	0.0	NaN	8.0	46.0	NaN	1	0.0
New Jersey	1998-11	0.0	NaN	20.0	53.0	NaN	2	2.0
New Mexico	1998-11	0.0	NaN	86.0	121.0	NaN	4	0.0
New York	1998-11	0.0	NaN	40.0	279.0	NaN	0	0.0
North Carolina	1998-11	524.0	NaN	87.0	695.0	NaN	4	0.0
North Dakota	1998-11	0.0	NaN	20.0	38.0	NaN	0	0.0

state	month	permit	permit_recheck	handgun	long_gun	other	multiple	admin
Ohio	1998-11	0.0	NaN	502.0	434.0	NaN	16	0.0
Oklahoma	1998-11	0.0	NaN	259.0	361.0	NaN	13	0.0
Oregon	1998-11	0.0	NaN	153.0	186.0	NaN	5	0.0
Pennsylvania	1998-11	0.0	NaN	5.0	8.0	NaN	4	0.0
Rhode Island	1998-11	0.0	NaN	13.0	23.0	NaN	2	0.0
South Carolina	1998-11	0.0	NaN	0.0	6.0	NaN	0	0.0
South Dakota	1998-11	0.0	NaN	8.0	66.0	NaN	0	0.0
Tennessee	1998-11	0.0	NaN	19.0	85.0	NaN	3	0.0
Texas	1998-11	0.0	NaN	1384.0	1349.0	NaN	60	1.0
Utah	1998-11	0.0	NaN	98.0	169.0	NaN	0	0.0
Vermont	1998-11	0.0	NaN	23.0	35.0	NaN	0	1.0
Virginia	1998-11	0.0	NaN	14.0	2.0	NaN	8	0.0
Washington	1998-11	1.0	NaN	65.0	286.0	NaN	8	1.0
West Virginia	1998-11	3.0	NaN	149.0	251.0	NaN	5	0.0
Wisconsin	1998-11	0.0	NaN	25.0	214.0	NaN	2	0.0
Wyoming	1998-11	8.0	NaN	45.0	49.0	NaN	5	0.0

11350 rows × 26 columns

In [21]:

```
#Reset the ondex of the df_gun_purchase

df_gun_purchase = df_gun_purchase.reset_index()
```

In [22]:
```
# replace all Nan in df_gun_purchase dataset with zeros

df_gun_purchase.fillna(0,inplace=True)
df_gun_purchase
```

Out[22]:

	state	month	permit	permit_recheck	handgun	long_gun	other	multiple	ac
0	Alabama	2017-09	16717.0	0.0	5734.0	6320.0	221.0	317	
1	Alaska	2017-09	209.0	2.0	2320.0	2930.0	219.0	160	
2	Arizona	2017-09	5069.0	382.0	11063.0	7946.0	920.0	631	
3	Arkansas	2017-09	2935.0	632.0	4347.0	6063.0	165.0	366	
4	California	2017-09	57839.0	0.0	37165.0	24581.0	2984.0	0	
5	Colorado	2017-09	4356.0	0.0	15751.0	13448.0	1007.0	1062	
6	Connecticut	2017-09	4343.0	673.0	4834.0	1993.0	274.0	0	
7	Delaware	2017-09	275.0	0.0	1414.0	1538.0	66.0	68	
8	Florida	2017-09	10784.0	0.0	39199.0	17949.0	2319.0	1721	
9	Georgia	2017-09	12074.0	0.0	10933.0	7982.0	315.0	494	
10	Hawaii	2017-09	946.0	0.0	0.0	0.0	0.0	0	
11	Idaho	2017-09	5162.0	0.0	3058.0	5241.0	187.0	205	
12	Illinois	2017-09	15712.0	71432.0	18290.0	10201.0	0.0	814	
13	Indiana	2017-09	18241.0	0.0	16093.0	11332.0	1123.0	597	
14	Iowa	2017-09	5847.0	1217.0	151.0	2640.0	27.0	1	
15	Kansas	2017-09	1567.0	3.0	4518.0	5025.0	306.0	297	
16	Kentucky	2017-09	378384.0	0.0	8112.0	7543.0	253.0	543	
17	Louisiana	2017-09	1827.0	0.0	10495.0	11573.0	635.0	776	
18	Maine	2017-09	783.0	0.0	3026.0	4220.0	179.0	186	
19	Maryland	2017-09	2424.0	0.0	3389.0	4897.0	168.0	34	
20	Massachusetts	2017-09	7160.0	0.0	4749.0	2808.0	449.0	158	
21	Michigan	2017-09	16571.0	19.0	8654.0	10676.0	379.0	163	
22	Minnesota	2017-09	25645.0	0.0	4862.0	12677.0	346.0	273	

	state	month	permit	permit_recheck	handgun	long_gun	other	multiple	ac
23	Mississippi	2017-09	1362.0	0.0	6260.0	6035.0	206.0	405	
24	Missouri	2017-09	791.0	0.0	16993.0	14395.0	1050.0	991	
25	Montana	2017-09	1076.0	0.0	2395.0	4878.0	140.0	216	
26	Nebraska	2017-09	3036.0	113.0	110.0	1989.0	11.0	4	
27	Nevada	2017-09	1952.0	0.0	3992.0	2509.0	251.0	237	
28	New Hampshire	2017-09	2795.0	0.0	4410.0	3248.0	132.0	3	
29	New Jersey	2017-09	0.0	0.0	3985.0	3040.0	140.0	0	
...	...	...	...	...	...	...	...	...	
11320	Massachusetts	1998-11	0.0	0.0	4.0	39.0	0.0	0	
11321	Michigan	1998-11	579.0	0.0	3.0	443.0	0.0	0	
11322	Minnesota	1998-11	9.0	0.0	27.0	280.0	0.0	3	
11323	Mississippi	1998-11	0.0	0.0	286.0	491.0	0.0	7	
11324	Missouri	1998-11	0.0	0.0	116.0	458.0	0.0	4	
11325	Montana	1998-11	0.0	0.0	101.0	98.0	0.0	2	
11326	Nebraska	1998-11	88.0	0.0	1.0	96.0	0.0	1	
11327	Nevada	1998-11	0.0	0.0	75.0	76.0	0.0	8	
11328	New Hampshire	1998-11	0.0	0.0	8.0	46.0	0.0	1	
11329	New Jersey	1998-11	0.0	0.0	20.0	53.0	0.0	2	
11330	New Mexico	1998-11	0.0	0.0	86.0	121.0	0.0	4	
11331	New York	1998-11	0.0	0.0	40.0	279.0	0.0	0	
11332	North Carolina	1998-11	524.0	0.0	87.0	695.0	0.0	4	
11333	North Dakota	1998-11	0.0	0.0	20.0	38.0	0.0	0	
11334	Ohio	1998-11	0.0	0.0	502.0	434.0	0.0	16	
11335	Oklahoma	1998-11	0.0	0.0	259.0	361.0	0.0	13	

	state	month	permit	permit_recheck	handgun	long_gun	other	multiple	ad
11336	Oregon	1998-11	0.0	0.0	153.0	186.0	0.0	5	
11337	Pennsylvania	1998-11	0.0	0.0	5.0	8.0	0.0	4	
11338	Rhode Island	1998-11	0.0	0.0	13.0	23.0	0.0	2	
11339	South Carolina	1998-11	0.0	0.0	0.0	6.0	0.0	0	
11340	South Dakota	1998-11	0.0	0.0	8.0	66.0	0.0	0	
11341	Tennessee	1998-11	0.0	0.0	19.0	85.0	0.0	3	
11342	Texas	1998-11	0.0	0.0	1384.0	1349.0	0.0	60	
11343	Utah	1998-11	0.0	0.0	98.0	169.0	0.0	0	
11344	Vermont	1998-11	0.0	0.0	23.0	35.0	0.0	0	
11345	Virginia	1998-11	0.0	0.0	14.0	2.0	0.0	8	
11346	Washington	1998-11	1.0	0.0	65.0	286.0	0.0	8	
11347	West Virginia	1998-11	3.0	0.0	149.0	251.0	0.0	5	
11348	Wisconsin	1998-11	0.0	0.0	25.0	214.0	0.0	2	
11349	Wyoming	1998-11	8.0	0.0	45.0	49.0	0.0	5	

11350 rows × 27 columns

In [23]:
```
# convert the datatype of the 'multiple' column to float

df_gun_purchase['multiple'] = df_gun_purchase['multiple'].astype(float)
df_gun_purchase['multiple'].dtypes
```

Out[23]:
```
dtype('float64')
```

In [24]:
```
# sum all columns in the df_gun_purchase excluding 'totals' column

float_cols = df_gun_purchase.select_dtypes(include=['float']).columns.values.t
olist()
df_gun_purchase['new_totals'] = df_gun_purchase[float_cols].sum(axis=1)
df_gun_purchase['new_totals'].head()
```

Out[24]:
```
0     32019.0
1      6303.0
2     28394.0
3     17747.0
4    123506.0
Name: new_totals, dtype: float64
```

In [25]:
```
# Confirm if the sum of all the columns equals to the totals column

df_gun_purchase['totals'] = df_gun_purchase['totals'].astype(float)
equality =df_gun_purchase['totals'] == df_gun_purchase['new_totals']
equality.unique()
```

Out[25]:
```
array([ True])
```

In [26]:
```
# drop the new_totals column

df_gun_purchase.drop("new_totals", axis=1, inplace=True)
df_gun_purchase.head()
```

Out[26]:

	state	month	permit	permit_recheck	handgun	long_gun	other	multiple	admin	prep
0	Alabama	2017-09	16717.0	0.0	5734.0	6320.0	221.0	317.0	0.0	
1	Alaska	2017-09	209.0	2.0	2320.0	2930.0	219.0	160.0	0.0	
2	Arizona	2017-09	5069.0	382.0	11063.0	7946.0	920.0	631.0	0.0	
3	Arkansas	2017-09	2935.0	632.0	4347.0	6063.0	165.0	366.0	51.0	
4	California	2017-09	57839.0	0.0	37165.0	24581.0	2984.0	0.0	0.0	

5 rows × 27 columns

In [27]:
```
# Split the data in the month column into months and year. This will make it e
asier to group the total column by year and then, by day
df_gun_purchase['year'] = df_gun_purchase['month'].str[0:4]
df_gun_purchase['month'] = df_gun_purchase['month'].str[-2:]
df_gun_purchase['month']
```

Out[27]:
```
0        09
1        09
2        09
3        09
4        09
         ..
11345    11
11346    11
11347    11
11348    11
11349    11
Name: month, Length: 11350, dtype: object
```

In [28]:
```
# check if there are missing values
df_gun_purchase.isnull().sum()
```

Out[28]:
```
state                        0
month                        0
permit                       0
permit_recheck               0
handgun                      0
long_gun                     0
other                        0
multiple                     0
admin                        0
prepawn_handgun              0
prepawn_long_gun             0
prepawn_other                0
redemption_handgun           0
redemption_long_gun          0
redemption_other             0
returned_handgun             0
returned_long_gun            0
returned_other               0
rentals_handgun              0
rentals_long_gun             0
private_sale_handgun         0
private_sale_long_gun        0
private_sale_other           0
return_to_seller_handgun     0
return_to_seller_long_gun    0
return_to_seller_other       0
totals                       0
year                         0
dtype: int64
```

In [29]:
```
# check if there are duplicates in the df_gun_purchase dataset
df_gun_purchase.duplicated().sum()
```

Out[29]: 0

In [30]:
```
# check the structure of the df_gun_purchase dataset
df_gun_purchase.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11350 entries, 0 to 11349
Data columns (total 28 columns):
state                        11350 non-null object
month                        11350 non-null object
permit                       11350 non-null float64
permit_recheck               11350 non-null float64
handgun                      11350 non-null float64
long_gun                     11350 non-null float64
other                        11350 non-null float64
multiple                     11350 non-null float64
admin                        11350 non-null float64
prepawn_handgun              11350 non-null float64
prepawn_long_gun             11350 non-null float64
prepawn_other                11350 non-null float64
redemption_handgun           11350 non-null float64
redemption_long_gun          11350 non-null float64
redemption_other             11350 non-null float64
returned_handgun             11350 non-null float64
returned_long_gun            11350 non-null float64
returned_other               11350 non-null float64
rentals_handgun              11350 non-null float64
rentals_long_gun             11350 non-null float64
private_sale_handgun         11350 non-null float64
private_sale_long_gun        11350 non-null float64
private_sale_other           11350 non-null float64
return_to_seller_handgun     11350 non-null float64
return_to_seller_long_gun    11350 non-null float64
return_to_seller_other       11350 non-null float64
totals                       11350 non-null float64
year                         11350 non-null object
dtypes: float64(25), object(3)
memory usage: 2.4+ MB
```

In [31]:
```
# convert the values in state column to lowercase and replace space with under
score so that it will be easy to join the two dataframes by state.
new_state = []
for state in df_gun_purchase["state"]:
    state = state.strip().lower().replace(" ", "_")
    new_state.append(state)
df_gun_purchase["state"] = new_state
```

In [32]:
```
# Group total permits by state, year, month and create a table for analysis an
d visualisation
new_gun_data = df_gun_purchase[["year",'month', 'state', 'totals']]
total_permit = new_gun_data.groupby(["state", 'year', "month"])["totals"].sum
().reset_index(name="totals")
total_permit.head()
```

Out[32]:

	state	year	month	totals
0	alabama	1998	11	1062.0
1	alabama	1998	12	35506.0
2	alabama	1999	01	18049.0
3	alabama	1999	02	20583.0
4	alabama	1999	03	19424.0

In [33]:
```
# Merge the year and month columns. Value grouping has been done
total_permit['year_month'] = total_permit["year"].astype(str)+"-"+total_permit
["month"]
total_permit.drop(["year","month"], axis=1, inplace=True)
total_permit.head()
```

Out[33]:

	state	totals	year_month
0	alabama	1062.0	1998-11
1	alabama	35506.0	1998-12
2	alabama	18049.0	1999-01
3	alabama	20583.0	1999-02
4	alabama	19424.0	1999-03

In [34]:
```
#Group df_gun_purchase by year

total_permit_by_year = new_gun_data.groupby(['year', "month"])["totals"].sum()
.reset_index(name="totals")
total_permit_by_year.head()
```

Out[34]:

	year	month	totals
0	1998	11	21174.0
1	1998	12	870202.0
2	1999	01	585569.0
3	1999	02	689867.0
4	1999	03	741234.0

In [35]:
```
# Merge the year and month columns. Value grouping has been done

total_permit_by_year['year_month'] = total_permit_by_year["year"].astype(str)+
"-"+total_permit_by_year["month"]
total_permit_by_year.drop(["year","month"], axis=1, inplace=True)
total_permit_by_year.head()
```

Out[35]:

	totals	year_month
0	21174.0	1998-11
1	870202.0	1998-12
2	585569.0	1999-01
3	689867.0	1999-02
4	741234.0	1999-03

In [36]:
```
# Create a data table with state as columns , year_month as index and totals a
s values

gun_data_piv2 = total_permit.pivot(index='year_month', columns="state", values
= "totals").reset_index()
gun_data_piv2.head()
```

Out[36]:

state	year_month	alabama	alaska	arizona	arkansas	california	colorado	connecticut	delawa
0	1998-11	1062.0	145.0	379.0	589.0	2101.0	622.0	80.0	5
1	1998-12	35506.0	3840.0	17074.0	21163.0	65344.0	23176.0	6790.0	208
2	1999-01	18049.0	2278.0	12859.0	11953.0	56953.0	19503.0	6265.0	112
3	1999-02	20583.0	2413.0	14546.0	15348.0	57471.0	22239.0	8069.0	107
4	1999-03	19424.0	3206.0	14992.0	13720.0	68327.0	17287.0	7877.0	131

5 rows × 51 columns

In [37]:
```
#drop the year_month column for the new pivot table
gun_data_piv3 = gun_data_piv2.copy("gun_data_piv2")
gun_data_piv3.drop("year_month", axis=1, inplace=True)
```

In [38]:
```
# drop the last 21 rows in the census dataset
df_census_data.drop(df_census_data.tail(21).index, inplace=True)
df_census_data.tail()
```

Out[38]:

	Fact	Fact Note	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Conr
59	Nonminority-owned firms, 2012	NaN	272,651	51,147	344,981	189,029	1,819,107	442,365	:
60	Veteran-owned firms, 2012	NaN	41,943	7,953	46,780	25,915	252,377	51,722	
61	Nonveteran-owned firms, 2012	NaN	316,984	56,091	427,582	192,988	3,176,341	469,524	:
62	Population per square mile, 2010	NaN	94.4	1.2	56.3	56	239.1	48.5	
63	Land area in square miles, 2010	NaN	50,645.33	570,640.95	113,594.08	52,035.48	155,779.22	103,641.89	4

5 rows × 52 columns

In [39]:
```
# drop the fact note column because the values are NaN and irrelevant for anal
ysis
df_census_data.drop('Fact Note', axis=1, inplace=True)
df_census_data.head()
```

Out[39]:

	Fact	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Connecticut	Delav
0	Population estimates, July 1, 2016, (V2016)	4,863,300	741,894	6,931,071	2,988,248	39,250,017	5,540,545	3,576,452	952
1	Population estimates base, April 1, 2010, (V2...	4,780,131	710,249	6,392,301	2,916,025	37,254,522	5,029,324	3,574,114	897
2	Population, percent change - April 1, 2010 (es...	1.70%	4.50%	8.40%	2.50%	5.40%	10.20%	0.10%	6.(
3	Population, Census, April 1, 2010	4,779,736	710,231	6,392,017	2,915,918	37,253,956	5,029,196	3,574,097	897
4	Persons under 5 years, percent, July 1, 2016, ...	6.00%	7.30%	6.30%	6.40%	6.30%	6.10%	5.20%	5.:

5 rows × 51 columns

In [40]:
```
# convert the column names to lowercase and replace the spaces with underscore
df_census_data.rename(columns=lambda x:x.strip().lower().replace(" ", "_"), in
place=True)
```

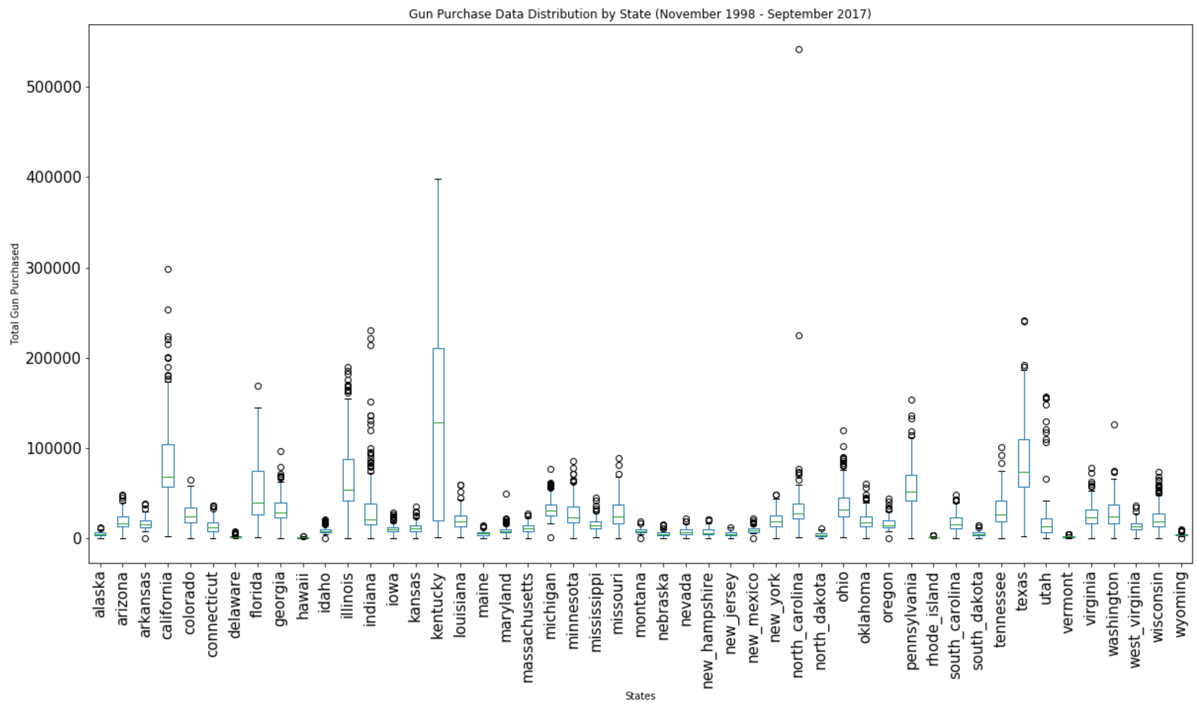
In [41]:
```
new_percent_change = []
for values in df_census_data.iloc[2]:
    values = values.strip().replace("%", "")
    new_percent_change.append(values)
df_census_data.iloc[2] = new_percent_change
```

Exploratory Data Analysis

Research Question 1 - What is the distribution of gun purchases across all states between November 1998 and September 2017?

```
In [42]: #new_gun_data.groupby('state').plot(kind='boxplot');

state_list1 = gun_data_piv2.columns[2:].to_list()
state_list1
ax_5 = gun_data_piv2.boxplot(column=state_list1, figsize=(20,10),rot = 90, gri
d=False, fontsize=15);
ax_5.set(xlabel = "States", ylabel="Total Gun Purchased", title= 'Gun Purchase
Data Distribution by State (November 1998 - September 2017)');
```



From the diagram above, none of the data is normally distributed. Most states, except Kentucky, have outliers. Most states, except Connecticut,Louisiana, New york, and West Virginia have a right-skewed distribution.

Kentucky had the longest boxplot with no outliers. This indicated that gun purchases are more consistent and spread out across all months than in any other states. The median value for Kentucky is higher than the maximum value for other states, except Washinton, Utah, Texas, Pennyslavania, Indiana, Illinois, Florida, California, and North Carolina

North Carolina has two outliers. One of the outliers is an extreme value that stood out from others. This shows that NC had a gun purchase of 54,178 in March 2014, the highest gun purchase among purchase within the period under review.

Research Question 2 - Which states had the highest and the lowest gun purchases between November 1998 and September 2017?

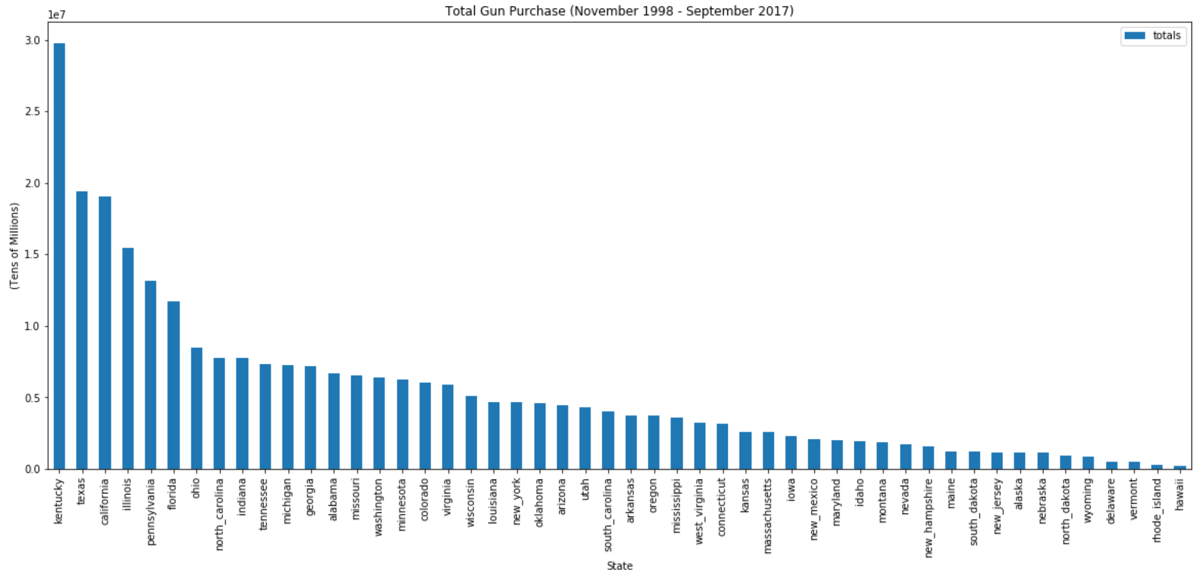
In [44]:
```
#Group the total gun purchases by states and plot a bar chat

total_permit2 = new_gun_data.groupby(["state"])["totals"].sum().reset_index(na
me="totals")
sorted_total_permit2 = total_permit2.sort_values(by='totals', ascending=False)

highest_gun_purchase = sorted_total_permit2['totals'].max()
lowest_gun_purchase = sorted_total_permit2['totals'].min()
print(highest_gun_purchase,lowest_gun_purchase)
```

```
29762434.0 197580.0
```

In [43]:
```
ax = sorted_total_permit2.plot( x="state", y="totals", kind="bar",figsize=(20,
8), title = "Total Gun Purchase (November 1998 - September 2017)");
ax.set(xlabel = "State", ylabel = "(Tens of Millions)");
```

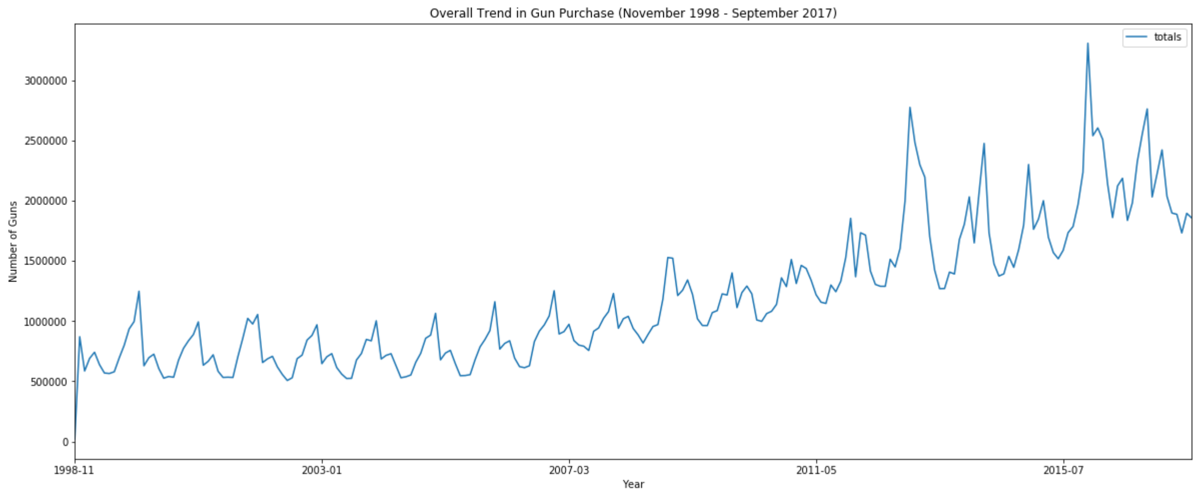


Between November 1998 and September 2017, Kentucky had the highest gun purchase with the total of number 29,762,434. Hawaii had the lowest gun purchases with the total number of 197,580.

Research Question 3 : What is the overall trend in gun purchase?

In [74]:
```
# Plot the line chart to show the overall trend in gun purchase

ax2 = total_permit_by_year.plot( x="year_month", y="totals", kind="line",figsi
ze=(20,8), title = "Overall Trend in Gun Purchase (November 1998 - September 2
017)");
ax2.set(xlabel = "Year", ylabel='Number of Guns');
```



Gun purchase is on the increase. Total gun purchase as of November 1998 was 21,174 and as of September 2017, total gun purchase had increased to 1,856,214. Some spikes can be observed in the line chart below, and an analysis of the dataset shows that gun purchases drops drastically in January of every year followed by growth in gun purchase in subsequent month. December 2015 had the highest total gun purchase with the value of 3,306,435.

In [45]:
```
# Check the month with the highest gun purchase

total_permit_by_year[total_permit_by_year['totals'] == total_permit_by_year.to
tals.max()]
```

Out[45]:

	totals	year_month
205	3306435.0	2015-12

In [46]:
```
#Review the months that are responsible for the observed spikes in the line ch
art above

total_permit_by_year
```

Out[46]:

	totals	year_month
0	21174.0	1998-11
1	870202.0	1998-12
2	585569.0	1999-01
3	689867.0	1999-02
4	741234.0	1999-03
5	638299.0	1999-04
6	568781.0	1999-05
7	564521.0	1999-06
8	579263.0	1999-07
9	694469.0	1999-08
10	798738.0	1999-09
11	934384.0	1999-10
12	995378.0	1999-11
13	1247812.0	1999-12
14	628579.0	2000-01
15	695878.0	2000-02
16	725088.0	2000-03
17	606212.0	2000-04
18	526074.0	2000-05
19	539560.0	2000-06
20	533493.0	2000-07
21	675825.0	2000-08
22	773629.0	2000-09
23	835879.0	2000-10
24	888128.0	2000-11
25	993141.0	2000-12
26	633466.0	2001-01
27	665891.0	2001-02
28	719949.0	2001-03
29	583322.0	2001-04
...	...	...
197	1696255.0	2015-04
198	1569096.0	2015-05
199	1517014.0	2015-06
200	1587710.0	2015-07

	totals	year_month
201	1734239.0	2015-08
202	1785086.0	2015-09
203	1967570.0	2015-10
204	2234871.0	2015-11
205	3306435.0	2015-12
206	2538652.0	2016-01
207	2603227.0	2016-02
208	2507513.0	2016-03
209	2134410.0	2016-04
210	1859080.0	2016-05
211	2121693.0	2016-06
212	2185520.0	2016-07
213	1835093.0	2016-08
214	1981896.0	2016-09
215	2324740.0	2016-10
216	2552584.0	2016-11
217	2761141.0	2016-12
218	2030400.0	2017-01
219	2222804.0	2017-02
220	2420835.0	2017-03
221	2035418.0	2017-04
222	1896910.0	2017-05
223	1886240.0	2017-06
224	1731550.0	2017-07
225	1894569.0	2017-08
226	1856214.0	2017-09

227 rows × 2 columns

Research Question 4 - Which states had the highest and the lowest percent change in gun purchase between April 2010 and July 2016?

In [47]:
```
# Get the indexes of July 2016 and April 2010 and extract the data for the per
iod

print(gun_data_piv2[gun_data_piv2['year_month']=="2016-07"].index[0])
print(gun_data_piv2[gun_data_piv2['year_month']=="2010-04"].index[0])
```

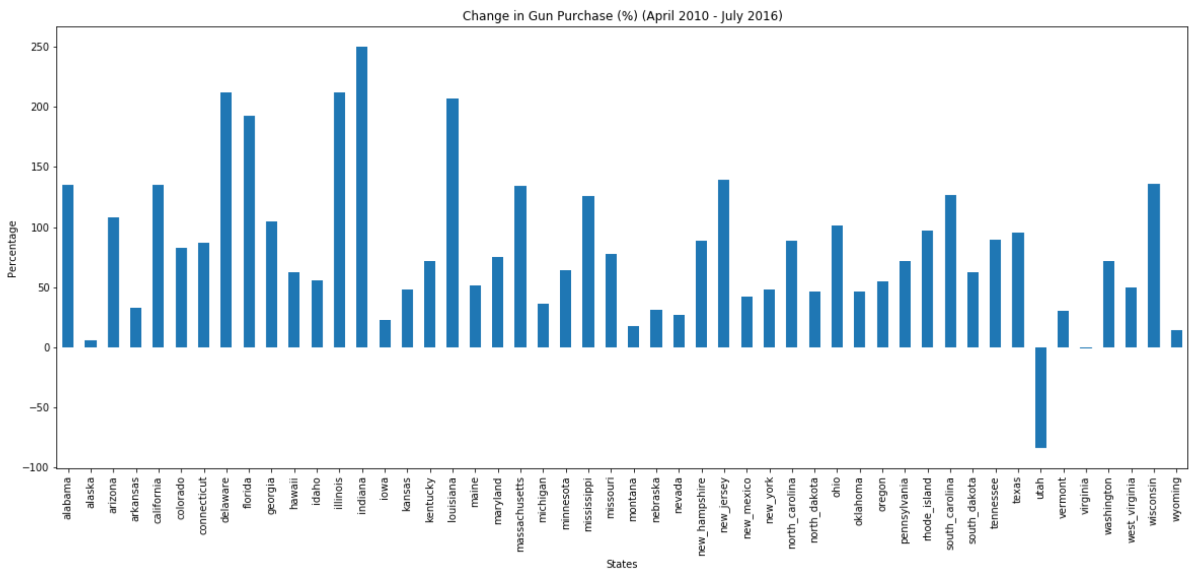
```
212
137
```

In [48]:
```
#Calculate the percentage change in total gun purchased between April 2010 and
July 2016

gun_data_piv3.loc['percent_change(%)'] = ((gun_data_piv3.iloc[212] / gun_data_
piv3.iloc[137])-1)*100
gun_data_piv3_min = gun_data_piv3.loc['percent_change(%)'].max()
```

In [49]:
```
# Plot the chat

ax3 = gun_data_piv3.loc['percent_change(%)'].plot(kind="bar" , figsize=(20,8),
title = "Change in Gun Purchase (%) (April 2010 - July 2016)");
ax3.set(xlabel = "States");
ax3.set(ylabel = "Percentage");
```



In [50]:
```
# Get the min and max values for the percent change in gun purchase

max_change = gun_data_piv3.loc['percent_change(%)'].max()
min_change = gun_data_piv3.loc['percent_change(%)'].min()
print(max_change, min_change)
```

```
250.11097019657575 -83.90361181450028
```

Indiana had the highest percent change in gun purchase with the value of 250.12%. Utah had the lowest percent change in gun purchase with the value of -83.90% for the period between April 2010 and July 2016.

Research Question 5 - Are the changes in population size responsible for the changes in gun purchase?

In [51]:
```
# Convert data type from object to float
population_change = df_census_data.iloc[[2],1:]

new_float = []
for values in df_census_data.iloc[[2],1:].values:
    values = values.astype(float)
    new_float.append(values)

df_census_data.iloc[[2],1:] = new_float

census_sub = df_census_data.iloc[[2]].set_index("fact")
census_sub
```

Out[51]:

	alabama	alaska	arizona	arkansas	california	colorado	connecticut	delaware	flo
fact									
Population, percent change - April 1, 2010 (estimates base) to July 1, 2016, (V2016)	1.7	4.5	8.4	2.5	5.4	10.2	0.1	6	

1 rows × 50 columns

In [52]:
```
#Merge the population change from census dataset with gun purchase change in t
he gun dataset
gun_sub = gun_data_piv3.loc[['percent_change(%)']]

new_dataframe = pd.concat([census_sub,gun_sub], sort=False)
new_dataframe
```

Out[52]:

state	alabama	alaska	arizona	arkansas	california	colorado	connecticut	dela
Population, percent change - April 1, 2010 (estimates base) to July 1, 2016, (V2016)	1.7	4.5	8.4	2.5	5.4	10.2	0.1	
percent_change(%)	135.328	5.95851	108.083	33.0632	135.564	82.5321	86.8798	2

2 rows × 50 columns

In [53]:
```
change_df = new_dataframe.transpose()
change_df
```

Out[53]:

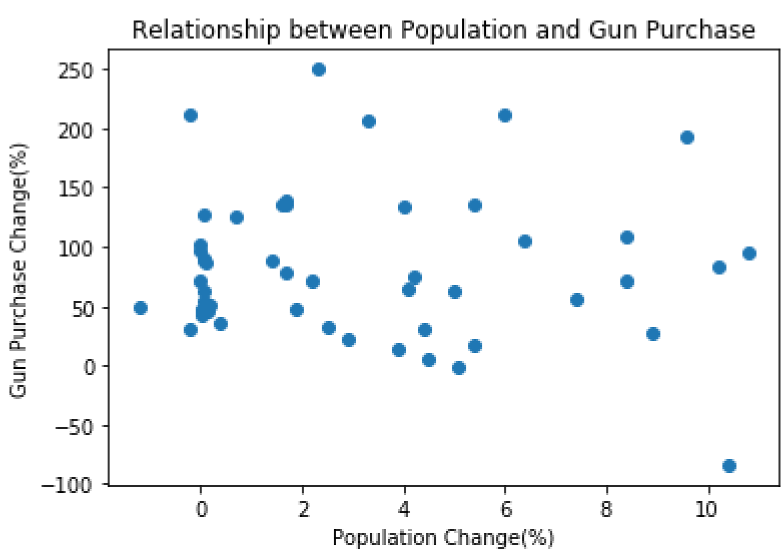
state	Population, percent change - April 1, 2010 (estimates base) to July 1, 2016, (V2016)	percent_change(%)
alabama	1.7	135.328
alaska	4.5	5.95851
arizona	8.4	108.083
arkansas	2.5	33.0632
california	5.4	135.564
colorado	10.2	82.5321
connecticut	0.1	86.8798
delaware	6	212.3
florida	9.6	192.583
georgia	6.4	104.376
hawaii	5	62.513
idaho	7.4	55.5413
illinois	-0.2	211.942
indiana	2.3	250.111
iowa	2.9	22.8086
kansas	1.9	48.3891
kentucky	2.2	71.8656
louisiana	3.3	207.059
maine	0.2	51.8234
maryland	4.2	74.8856
massachusetts	4	134.111
michigan	0.4	36.6164
minnesota	4.1	64.5782
mississippi	0.7	125.799
missouri	1.7	77.4012
montana	5.4	17.9515
nebraska	4.4	31.1036
nevada	8.9	26.8022
new_hampshire	1.4	89.0899
new_jersey	1.7	139.004
new_mexico	0.011	42.0979
new_york	0.019	48.2618
north_carolina	0.064	88.7373
north_dakota	0.127	46.8062

	Population, percent change - April 1, 2010 (estimates base) to July 1, 2016, (V2016)	percent_change(%)
state		
ohio	0.007	101.673
oklahoma	0.046	46.1746
oregon	0.068	54.7814
pennsylvania	0.006	71.4203
rhode_island	0.003	97.4979
south_carolina	0.073	126.646
south_dakota	0.063	62.3767
tennessee	0.048	89.318
texas	10.8	95.667
utah	10.4	-83.9036
vermont	-0.2	30.2484
virginia	5.1	-1.27557
washington	8.4	72.1563
west_virginia	-1.2	50.1878
wisconsin	1.6	136.306
wyoming	3.9	14.2537

In [54]:
```
# Plot a scatter graph of the relationship

x = new_dataframe.iloc[0].values
y = new_dataframe.iloc[1].values

plt.scatter(x, y)
plt.title('Relationship between Population and Gun Purchase')
plt.xlabel('Population Change(%)')
plt.ylabel('Gun Purchase Change(%)');
```



The scatter plot shows a negative relationship but the strength of the relationship is weak because the data points are dispersed and not clustered together. The correlation coefficient is -0.06611259. Therefore, changes in the population is not really responsible for the increase in gun purchase.

Conclusions

Total gun purchase between April 2010 and July 2016 has increased. This increase is not correlated with the changes in the population. Indiana with the highest percent change in gun purchase of 250.11% only experienced a 2.5% increase in population growth. Also, Utah with the lowest percent change in gun purchase of -83.90% experienced a 10.4% increase in population growth. The correlation coefficient of -0.066 shows there is a weak negative relationship between change in population and gun purchase.

While increase in crime rate could explain the increase in gun purchase, this data is not provided. Information about the crime rate can be obtained from U.S. Crime Rate & Statistics.

Despite the upward trend, January of every year always experience a significant drop in the number of gun purchase. Tight budget due to the holiday shopping in the prior month(December) could be responsible for this drop. A review of consumer spending, or personal consumption expenditures (PCE) data by the US Bureau of Economic Analysis could provide insight on consumer spending in January.

Limitations

The major limitation was combining the data from the two tables based on the state. This is because the state data for the census dataset is located on the columns axis but the state data for the gun purchase data is located on the row axis.

In [49]:
```
from subprocess import call
call(['python', '-m', 'nbconvert', 'Investigate_a_Dataset.ipynb'])
```

Out[49]: 0