FBI_Gun_Date

December 20, 2021

1 Project: Investigate a Dataset - [FBI_Gun_Data]

1.1 Table of Contents

Introduction

1.1.1 Dataset Description

In this notebook, we will analyze data for two related data sets:

1- Gun Data (NICS Data), this data comes from the FBI's National Instant Criminal Background Check System, it is a system that contains people who are allowed to own weapons and people who have a criminal history. Our data is columns that contains the number of firearm checks by month, state, type, and totals.

2-Census Data: It contains several variables at the state level.variables may be have one data point per state , or have data for more than one year, columns(fact,fact note,states).

1.1.2 Questions:

1- How much growth rate in gun registration in each state?
2- What is the over all trend in gun phrchases?
3- How many persons are permitted to own gun in last date?
4- How many persons are permitted versus not permitted to own gun in Arizona state in last date?
5- How much quantity of each types of guns?
6- in which state did men owned firms by maximum number of gun in 2012?
7- How many gun registered in all states in 2017?
8- how many gun registered in each state i 2000?
9- What is the average of guns registered in each states since 1998?
10- How many guns registered in Texas in 2017?
11- What is the total revenue of all fims in all states in 2012?

```
In [1]: #import packages
        import pandas as pd
        import numpy as np
        %matplotlib inline
        import matplotlib.pyplot as plt
        import datetime
```

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

Data Wrangling

```
In [3]: # Loading DataFrames
        # read_csv to read and load csv file
        # read_excel to read and load xlsx file
        df_census = pd.read_csv('US_Census_Data.csv')
        df_gun = pd.read_excel('gun_data.xlsx')
```

```
In [4]: # print out first 5 rows of census data
        df_census.head()
```

```
Out[4]:                                                Fact Fact Note    Alabama  \
        0      Population estimates, July 1, 2016,  (V2016)       NaN  4,863,300
        1  Population estimates base, April 1, 2010,  (V2...       NaN  4,780,131
        2  Population, percent change - April 1, 2010 (es...       NaN      1.70%
        3                 Population, Census, April 1, 2010       NaN  4,779,736
        4  Persons under 5 years, percent, July 1, 2016, ...       NaN      6.00%

            Alaska    Arizona   Arkansas  California   Colorado Connecticut Delaware  \
        0  741,894  6,931,071  2,988,248  39,250,017  5,540,545   3,576,452  952,065
        1  710,249  6,392,301  2,916,025  37,254,522  5,029,324   3,574,114  897,936
        2    4.50%      8.40%      2.50%       5.40%     10.20%       0.10%    6.00%
        3  710,231  6,392,017  2,915,918  37,253,956  5,029,196   3,574,097  897,934
        4    7.30%      6.30%      6.40%       6.30%      6.10%       5.20%    5.80%

            ...     South Dakota Tennessee       Texas       Utah  Vermont   Virginia  \
        0   ...           865454   6651194  27,862,596  3,051,217  624,594  8,411,808
        1   ...           814195   6346298  25,146,100  2,763,888  625,741  8,001,041
        2   ...            0.063     0.048      10.80%     10.40%   -0.20%      5.10%
        3   ...           814180   6346105  25,145,561  2,763,885  625,741  8,001,024
        4   ...            0.071     0.061       7.20%      8.30%    4.90%      6.10%

          Washington West Virginia  Wisconsin  Wyoming
        0  7,288,000     1,831,102  5,778,708  585,501
        1  6,724,545     1,853,011  5,687,289  563,767
        2      8.40%        -1.20%      1.60%    3.90%
        3  6,724,540     1,852,994  5,686,986  563,626
        4      6.20%         5.50%      5.80%    6.50%

        [5 rows x 52 columns]
```

```
In [5]: # info() to show some general information of data
        # there are missing values
        # there are incorrect data types
        # shape(85,52)
        df_census.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: 34.6+ KB
```

```
In [6]: # see number of columns that contain missing values (NaN)
        df_census.isnull().any().sum()
```

```
Out[6]: 52
```

```
In [7]: # number of missing values
        df_census.isnull().sum().sum()
```

```
Out[7]: 1062
```

```
In [8]: # which columns have missing and count it
        df_census.isnull().sum()
```

```
Out[8]: 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 [9]:
```
# there are three duplicates values
df_census.duplicated().sum()
```

Out[9]: 3

In [10]:
```
# do the same with df_gun
df_gun.head()
```

Out[10]:

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

```
   multiple  admin  prepawn_handgun   ...    returned_other  rentals_handgun  \
0       317    0.0             15.0   ...               0.0              0.0
1       160    0.0              5.0   ...               0.0              0.0
2       631    0.0             13.0   ...               0.0              0.0
3       366   51.0             12.0   ...               0.0              0.0
4         0    0.0              0.0   ...               0.0              0.0

   rentals_long_gun  private_sale_handgun  private_sale_long_gun  \
0               0.0                   9.0                   16.0
1               0.0                  17.0                   24.0
2               0.0                  38.0                   12.0
3               0.0                  13.0                   23.0
4               0.0                   0.0                    0.0

   private_sale_other  return_to_seller_handgun  return_to_seller_long_gun  \
0                 3.0                       0.0                        0.0
1                 1.0                       0.0                        0.0
2                 2.0                       0.0                        0.0
3                 0.0                       0.0                        2.0
4                 0.0                       0.0                        0.0

   return_to_seller_other  totals
0                     3.0   32019
1                     0.0    6303
2                     0.0   28394
3                     1.0   17747
4                     0.0  123506

[5 rows x 27 columns]
```

In [11]:
```
# show general information
# missing values
# incorrect data types
# shape(12485,27)
df_gun.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 [12]:
```
# number of columns has missing values
df_gun.isnull().any().sum()
```

Out[12]: 23

In [13]:
```
# number of missing values
df_gun.isnull().sum().sum()
```

Out[13]: 154595

In [14]:
```
# which columns contain missing value
df_gun.isnull().sum()
```

Out[14]:
```
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 [15]:
```
# check duplicates value
# there are no duplicates
df_gun.duplicated().sum()
```

Out[15]: 0

1.1.3 Data Cleaning

In [16]:
```
# fixing data type
# we need to convert type of states column in census data from object type to float typ
# but state data separated by (,) then when we try conver this data using extract and a
# data will converted as minimum float number
# to solve this we remove sympol like , and % then convert it
# but astype() method can't convert string to float(its NaN value)
# first we fill NaN with type of same data because float object has no attibuute split
# after convert, we get NaN again then we fill it again by mean, that is better.

#for column in df_census.columns[2:]:
#    df_census[column]=df_census[column].str.extract('(\d+)').astype(float)
#    df_census[column]= pd.to_numeric(df_census[column],errors='coerce')
#    df_census[column]=df_census[column].replace(',','.').str.extract('(\d)').astype(fl

df_census.fillna('1,111',inplace=True)

for column in df_census.columns[2:]:
    df_census[column]=df_census[column].apply(lambda x:x.split()[0].replace(',',''))
    df_census[column]=df_census[column].apply(lambda x:x.split()[0].replace('%',''))
    df_census[column]=df_census[column].str.extract('(\d+)').astype(float)

df_census.replace('1,111',np.NaN, inplace=True)
df_census.replace(1111.0, np.NaN, inplace=True)
```

In [17]:
```
# confirm changes
df_census.head(66)
```

Out[17]:
```
                                                 Fact  \
0       Population estimates, July 1, 2016,  (V2016)
1   Population estimates base, April 1, 2010,  (V2...
2   Population, percent change - April 1, 2010 (es...
3                   Population, Census, April 1, 2010
4   Persons under 5 years, percent, July 1, 2016, ...
5       Persons under 5 years, percent, April 1, 2010
6   Persons under 18 years, percent, July 1, 2016,...
7      Persons under 18 years, percent, April 1, 2010
8   Persons 65 years and over, percent,  July 1, 2...
9   Persons 65 years and over, percent, April 1, 2010
10   Female persons, percent,  July 1, 2016,  (V2016)
11            Female persons, percent, April 1, 2010
12      White alone, percent, July 1, 2016,  (V2016)
13  Black or African American alone, percent, July...
14  American Indian and Alaska Native alone, perce...
15      Asian alone, percent, July 1, 2016,  (V2016)
16  Native Hawaiian and Other Pacific Islander alo...
17  Two or More Races, percent, July 1, 2016,  (V2...
18  Hispanic or Latino, percent, July 1, 2016,  (V...
19  White alone, not Hispanic or Latino, percent, ...
20                              Veterans, 2011-2015
21           Foreign born persons, percent, 2011-2015
22            Housing units,  July 1, 2016,  (V2016)
23                        Housing units, April 1, 2010
24       Owner-occupied housing unit rate, 2011-2015
25  Median value of owner-occupied housing units, ...
26  Median selected monthly owner costs -with a mo...
27  Median selected monthly owner costs -without a...
28                     Median gross rent, 2011-2015
29                           Building permits, 2016
..                                                ...
36  With a disability, under age 65 years, percent...
37  Persons  without health insurance, under age 6...
38  In civilian labor force, total, percent of pop...
39  In civilian labor force, female, percent of po...
40  Total accommodation and food services sales, 2...
41  Total health care and social assistance receip...
42       Total manufacturers shipments, 2012 ($1,000)
43     Total merchant wholesaler sales, 2012 ($1,000)
44              Total retail sales, 2012 ($1,000)
45                Total retail sales per capita, 2012
46  Mean travel time to work (minutes), workers ag...
47  Median household income (in 2015 dollars), 201...
48  Per capita income in past 12 months (in 2015 d...
```

```
49                       Persons in poverty, percent
50              Total employer establishments, 2015
51                           Total employment, 2015
52              Total annual payroll, 2015 ($1,000)
53      Total employment, percent change, 2014-2015
54           Total nonemployer establishments, 2015
55                                  All firms, 2012
56                            Men-owned firms, 2012
57                          Women-owned firms, 2012
58                       Minority-owned firms, 2012
59                    Nonminority-owned firms, 2012
60                        Veteran-owned firms, 2012
61                     Nonveteran-owned firms, 2012
62                Population per square mile, 2010
63                 Land area in square miles, 2010
64                                        FIPS Code
65                                              NaN

                                      Fact Note    Alabama     Alaska  \
0                                           NaN  4863300.0   741894.0
1                                           NaN  4780131.0   710249.0
2                                           NaN        1.0        4.0
3                                           NaN  4779736.0   710231.0
4                                           NaN        6.0        7.0
5                                           NaN        6.0        7.0
6                                           NaN       22.0       25.0
7                                           NaN       23.0       26.0
8                                           NaN       16.0       10.0
9                                           NaN       13.0        7.0
10                                          NaN       51.0       47.0
11                                          NaN       51.0       48.0
12                                          (a)       69.0       66.0
13                                          (a)       26.0        3.0
14                                          (a)        0.0       15.0
15                                          (a)        1.0        6.0
16                                          (a)        0.0        1.0
17                                          NaN        1.0        7.0
18                                          (b)        4.0        7.0
19                                          NaN       65.0       61.0
20                                          NaN   363170.0    69323.0
21                                          NaN        3.0        7.0
22                                          NaN  2230185.0   310658.0
23                                          NaN  2171853.0   306967.0
24                                          NaN       68.0       63.0
25                                          NaN   125500.0   250000.0
26                                          NaN     1139.0     1827.0
27                                          NaN      345.0      554.0
28                                          NaN      717.0     1146.0
```

```
29                                            NaN      15001.0      1503.0
..                                            ...          ...         ...
36                                            NaN         11.0         8.0
37                                            NaN         10.0        15.0
38                                            NaN         58.0        67.0
39                                            NaN         53.0        65.0
40                                            (c)    7576462.0   2221335.0
41                                            (c)   26039632.0   6375483.0
42                                            (c)  124809759.0         NaN
43                                            (c)   57746565.0   5216303.0
44                                            (c)   58564965.0  10474275.0
45                                            (c)      12145.0     14320.0
46                                            NaN         24.0        19.0
47                                            NaN      43623.0     72515.0
48                                            NaN      24091.0     33413.0
49                                            NaN         17.0         9.0
50  Includes data not distributed by county.      98540.0     20907.0
51  Includes data not distributed by county.    1634391.0    267999.0
52  Includes data not distributed by county.   67370353.0  15643303.0
53  Includes data not distributed by county.          1.0         0.0
54                                            NaN     322025.0     55521.0
55                                            NaN     374153.0     68032.0
56                                            NaN     203604.0     35402.0
57                                            NaN     137630.0     22141.0
58                                            NaN      92219.0     13688.0
59                                            NaN     272651.0     51147.0
60                                            NaN      41943.0      7953.0
61                                            NaN     316984.0     56091.0
62                                            NaN         94.0         1.0
63                                            NaN      50645.0    570640.0
64                                            NaN          1.0         2.0
65                                            NaN          NaN         NaN

       Arizona    Arkansas  California    Colorado  Connecticut  \
0    6931071.0   2988248.0  39250017.0   5540545.0    3576452.0
1    6392301.0   2916025.0  37254522.0   5029324.0    3574114.0
2          8.0         2.0         5.0        10.0          0.0
3    6392017.0   2915918.0  37253956.0   5029196.0    3574097.0
4          6.0         6.0         6.0         6.0          5.0
5          7.0         6.0         6.0         6.0          5.0
6         23.0        23.0        23.0        22.0         21.0
7         25.0        24.0        25.0        24.0         22.0
8         16.0        16.0        13.0        13.0         16.0
9         13.0        14.0        11.0        10.0         14.0
10        50.0        50.0        50.0        49.0         51.0
11        50.0        50.0        50.0        49.0         51.0
12        83.0        79.0        72.0        87.0         80.0
13         4.0        15.0         6.0         4.0         11.0
```

14	5.0	1.0	1.0	1.0	0.0
15	3.0	1.0	14.0	3.0	4.0
16	0.0	0.0	0.0	0.0	0.0
17	2.0	2.0	3.0	3.0	2.0
18	30.0	7.0	38.0	21.0	15.0
19	55.0	72.0	37.0	68.0	67.0
20	505794.0	220953.0	1777410.0	391725.0	199331.0
21	13.0	4.0	27.0	9.0	13.0
22	2961003.0	1354762.0	14060525.0	2339118.0	1499116.0
23	2844526.0	1316299.0	13680081.0	2212898.0	1487891.0
24	62.0	66.0	54.0	64.0	67.0
25	167500.0	111400.0	385500.0	247800.0	270500.0
26	1343.0	1019.0	2155.0	1577.0	2067.0
27	380.0	327.0	500.0	419.0	833.0
28	913.0	677.0	1255.0	1002.0	1075.0
29	35578.0	9474.0	102350.0	38974.0	5504.0
..	...	...	...	...	...
36	8.0	12.0	6.0	7.0	7.0
37	11.0	9.0	8.0	8.0	5.0
38	59.0	58.0	63.0	67.0	67.0
39	54.0	53.0	57.0	62.0	62.0
40	13996635.0	4307264.0	90830372.0	13617654.0	9542068.0
41	37055881.0	15792628.0	248953592.0	29488161.0	29573119.0
42	51243473.0	62712925.0	512303164.0	50447098.0	55160095.0
43	69437272.0	31256110.0	666652186.0	77034971.0	161962244.0
44	84716542.0	36815256.0	481800461.0	67815200.0	51632467.0
45	12927.0	12483.0	12665.0	13073.0	14381.0
46	24.0	21.0	28.0	24.0	25.0
47	50255.0	41371.0	61818.0	60629.0	70331.0
48	25848.0	22798.0	30318.0	32217.0	38803.0
49	16.0	17.0	14.0	11.0	9.0
50	136352.0	65175.0	908120.0	161737.0	89232.0
51	2295186.0	1003113.0	14325377.0	2253795.0	1503102.0
52	102671393.0	39451191.0	856954246.0	117539555.0	92555072.0
53	2.0	1.0	3.0	3.0	1.0
54	451951.0	198380.0	3206958.0	480847.0	272809.0
55	499926.0	231959.0	3548449.0	547352.0	326693.0
56	245243.0	123158.0	1852580.0	284554.0	187845.0
57	182425.0	75962.0	1320085.0	194508.0	106678.0
58	135313.0	35982.0	1619857.0	85849.0	56113.0
59	344981.0	189029.0	1819107.0	442365.0	259614.0
60	46780.0	25915.0	252377.0	51722.0	31056.0
61	427582.0	192988.0	3176341.0	469524.0	281182.0
62	56.0	56.0	239.0	48.0	738.0
63	113594.0	52035.0	155779.0	103641.0	4842.0
64	4.0	5.0	6.0	8.0	9.0
65	NaN	NaN	NaN	NaN	NaN

	Delaware	...	South Dakota	Tennessee	Texas	\
0	952065.0	...	865454.0	6651194.0	27862596.0	
1	897936.0	...	814195.0	6346298.0	25146100.0	
2	6.0	...	0.0	0.0	10.0	
3	897934.0	...	814180.0	6346105.0	25145561.0	
4	5.0	...	0.0	0.0	7.0	
5	6.0	...	0.0	0.0	7.0	
6	21.0	...	0.0	0.0	26.0	
7	22.0	...	0.0	0.0	27.0	
8	17.0	...	0.0	0.0	12.0	
9	14.0	...	0.0	0.0	10.0	
10	51.0	...	0.0	0.0	50.0	
11	51.0	...	0.0	0.0	50.0	
12	70.0	...	0.0	0.0	79.0	
13	22.0	...	0.0	0.0	12.0	
14	0.0	...	0.0	0.0	1.0	
15	4.0	...	0.0	0.0	4.0	
16	0.0	...	0.0	0.0	0.0	
17	2.0	...	0.0	0.0	1.0	
18	9.0	...	0.0	0.0	39.0	
19	62.0	...	0.0	0.0	42.0	
20	71213.0	...	63742.0	462414.0	1539655.0	
21	8.0	...	0.0	0.0	16.0	
22	426149.0	...	383838.0	2919671.0	10753629.0	
23	405885.0	...	363438.0	2812133.0	9977436.0	
24	71.0	...	0.0	0.0	62.0	
25	231500.0	...	140500.0	142100.0	136000.0	
26	1537.0	...	1210.0	1181.0	1432.0	
27	445.0	...	433.0	359.0	460.0	
28	1018.0	...	655.0	764.0	882.0	
29	5804.0	...	5686.0	36157.0	165853.0	
..	...	...	...	...	...	
36	8.0	...	0.0	0.0	8.0	
37	6.0	...	0.0	0.0	18.0	
38	63.0	...	0.0	0.0	64.0	
39	59.0	...	0.0	0.0	57.0	
40	2148437.0	...	1873699.0	12499013.0	54480811.0	
41	7003251.0	...	6211731.0	42383683.0	145035130.0	
42	22597384.0	...	16882647.0	139960482.0	702603073.0	
43	5628914.0	...	20411059.0	111718421.0	691242607.0	
44	14456001.0	...	13791827.0	91641605.0	356116376.0	
45	15763.0	...	16550.0	14194.0	13666.0	
46	25.0	...	16.0	24.0	25.0	
47	60509.0	...	50957.0	45219.0	53207.0	
48	30554.0	...	26747.0	25227.0	26999.0	
49	11.0	...	0.0	0.0	15.0	
50	24852.0	...	26511.0	133344.0	569091.0	
51	397385.0	...	353540.0	2507205.0	10239710.0	

```
52  21305227.0   ...     13812997.0  110481280.0  521095797.0
53         1.0   ...            0.0          0.0          3.0
54     60734.0   ...        64006.0     495703.0    2205149.0
55     73418.0   ...        81314.0     550453.0    2356748.0
56     38328.0   ...        42418.0     302249.0    1251696.0
57     23964.0   ...        23722.0     195694.0     866678.0
58     14440.0   ...         4101.0     105234.0    1070392.0
59     54782.0   ...        74228.0     434025.0    1224845.0
60      7206.0   ...         8604.0      59379.0     213590.0
61     60318.0   ...        66219.0     469392.0    2057218.0
62       460.0   ...           10.0        153.0         96.0
63      1948.0   ...        75811.0      41234.0     261231.0
64        10.0   ...           46.0         47.0         48.0
65         NaN   ...            NaN          NaN          NaN

          Utah    Vermont    Virginia  Washington  West Virginia  \
0    3051217.0   624594.0   8411808.0   7288000.0      1831102.0
1    2763888.0   625741.0   8001041.0   6724545.0      1853011.0
2         10.0        0.0         5.0         8.0            1.0
3    2763885.0   625741.0   8001024.0   6724540.0      1852994.0
4          8.0        4.0         6.0         6.0            5.0
5          9.0        5.0         6.0         6.0            5.0
6         30.0       19.0        22.0        22.0           20.0
7         31.0       20.0        23.0        23.0           20.0
8         10.0       18.0        14.0        14.0           18.0
9          9.0       14.0        12.0        12.0           16.0
10        49.0       50.0        50.0        50.0           50.0
11        49.0       50.0        50.0        50.0           50.0
12        91.0       94.0        70.0        80.0           93.0
13         1.0        1.0        19.0         4.0            3.0
14         1.0        0.0         0.0         1.0            0.0
15         2.0        1.0         6.0         8.0            0.0
16         1.0        NaN         0.0         0.0            NaN
17         2.0        1.0         2.0         4.0            1.0
18        13.0        1.0         9.0        12.0            1.0
19        78.0       93.0        62.0        69.0           92.0
20    134332.0    44708.0    706539.0    564864.0       150021.0
21         8.0        4.0        11.0        13.0            1.0
22   1054164.0   329525.0   3491054.0   3025685.0       886640.0
23    979709.0   322539.0   3364939.0   2885677.0       881917.0
24        69.0       71.0        66.0        62.0           72.0
25    215900.0   217500.0    245000.0    259500.0       103800.0
26      1428.0     1535.0      1711.0      1731.0          966.0
27       388.0      641.0       433.0       511.0          293.0
28       887.0      895.0      1116.0      1014.0          643.0
29     22662.0     1771.0     31132.0     44077.0         2544.0
..         ...        ...         ...         ...            ...
36         6.0       10.0         7.0         8.0           14.0
```

37	9.0	4.0	10.0	6.0	6.0
38	67.0	66.0	64.0	63.0	53.0
39	59.0	63.0	60.0	58.0	49.0
40	4789281.0	1564272.0	17795901.0	14297278.0	4036333.0
41	14521857.0	4457996.0	47705003.0	43966889.0	12259395.0
42	50046429.0	9315494.0	96389872.0	131530601.0	24553072.0
43	30927885.0	6450076.0	86613641.0	83313366.0	14295437.0
44	38024486.0	9933751.0	110002385.0	118924049.0	22637923.0
45	13317.0	15868.0	13438.0	17243.0	12201.0
46	21.0	22.0	27.0	26.0	25.0
47	60727.0	55176.0	65015.0	61062.0	41751.0
48	24686.0	29894.0	34152.0	31762.0	23450.0
49	10.0	11.0	11.0	11.0	17.0
50	75463.0	21121.0	197384.0	182913.0	36993.0
51	1203954.0	266363.0	3198718.0	2602408.0	565435.0
52	51453266.0	10615093.0	165788897.0	149258789.0	22159084.0
53	4.0	2.0	1.0	2.0	1.0
54	216280.0	60312.0	576446.0	444135.0	88136.0
55	251419.0	75827.0	653193.0	541522.0	114435.0
56	132163.0	41270.0	353012.0	262650.0	63112.0
57	76269.0	23417.0	236290.0	187677.0	39065.0
58	24423.0	2354.0	185043.0	92807.0	5777.0
59	218826.0	70491.0	450109.0	426697.0	104785.0
60	18754.0	8237.0	76434.0	49331.0	12912.0
61	219807.0	63317.0	548439.0	461401.0	94960.0
62	33.0	67.0	202.0	101.0	77.0
63	82169.0	9216.0	39490.0	66455.0	24038.0
64	49.0	50.0	51.0	53.0	54.0
65	NaN	NaN	NaN	NaN	NaN

	Wisconsin	Wyoming
0	5778708.0	585501.0
1	5687289.0	563767.0
2	1.0	3.0
3	5686986.0	563626.0
4	5.0	6.0
5	6.0	7.0
6	22.0	23.0
7	23.0	24.0
8	16.0	15.0
9	13.0	12.0
10	50.0	48.0
11	50.0	49.0
12	87.0	92.0
13	6.0	1.0
14	1.0	2.0
15	2.0	1.0
16	0.0	0.0

```
17          1.0          2.0
18          6.0         10.0
19         81.0         84.0
20     381940.0      48505.0
21          4.0          3.0
22    2668444.0     270600.0
23    2624358.0     261868.0
24         67.0         69.0
25     165800.0     194800.0
26       1402.0       1348.0
27        532.0        386.0
28        776.0        789.0
29      19274.0       1727.0
..          ...          ...
36          8.0          8.0
37          6.0         13.0
38         67.0         67.0
39         63.0         62.0
40   10303256.0    1644844.0
41   40680625.0    3291478.0
42  177728926.0   10783794.0
43   77066883.0    5597891.0
44   78201822.0    9446043.0
45      13656.0      16388.0
46         21.0         18.0
47      53357.0      58840.0
48      28340.0      29803.0
49         11.0         11.0
50     139500.0      21040.0
51    2503532.0     219881.0
52  112406494.0   10094010.0
53          2.0          NaN
54     341935.0      48140.0
55     432980.0      62427.0
56     236252.0      30039.0
57     133859.0      19344.0
58      40507.0       4077.0
59     379934.0      55397.0
60      39830.0       6470.0
61     370755.0      51353.0
62        105.0          5.0
63      54157.0      97093.0
64         55.0         56.0
65          NaN          NaN

[66 rows x 52 columns]
```

In [18]:
```
# fixing NaN values
```

```
# first one to fill numerical NaN with mean
# second one to fill string NaN with any word like non
# and confirm changes
df_census.fillna(df_census.mean(), inplace=True)
df_census.fillna('non',inplace=True)
df_census.isnull().any().sum()
```

Out[18]: 0

In [19]:
```
# drop duplicates and confirm changes
df_census.drop_duplicates(inplace=True)
df_census.duplicated().sum()
```

Out[19]: 0

In [20]:
```
# convert time in month column to datetime
# confirm changes
df_gun['month'] = pd.to_datetime(df_gun['month'])
df_gun['month'].dtype
```

Out[20]: dtype('<M8[ns]')

In [21]:
```
# fixing missing value and confirm changes
df_gun.fillna(df_gun.mean(), inplace=True)
df_gun.isnull().any().sum()
```

Out[21]: 0

In [22]:
```
# confirm census data after cleaning
df_census.head(66)
```

Out[22]:
```
                                                 Fact  \
0        Population estimates, July 1, 2016,  (V2016)
1   Population estimates base, April 1, 2010,  (V2...
2   Population, percent change - April 1, 2010 (es...
3                   Population, Census, April 1, 2010
4   Persons under 5 years, percent, July 1, 2016, ...
5       Persons under 5 years, percent, April 1, 2010
6   Persons under 18 years, percent, July 1, 2016,...
7      Persons under 18 years, percent, April 1, 2010
8   Persons 65 years and over, percent,  July 1, 2...
9   Persons 65 years and over, percent, April 1, 2010
10   Female persons, percent,  July 1, 2016,  (V2016)
11            Female persons, percent, April 1, 2010
12       White alone, percent, July 1, 2016,  (V2016)
13  Black or African American alone, percent, July...
14  American Indian and Alaska Native alone, perce...
15       Asian alone, percent, July 1, 2016,  (V2016)
16  Native Hawaiian and Other Pacific Islander alo...
```

```
17  Two or More Races, percent, July 1, 2016,  (V2...
18  Hispanic or Latino, percent, July 1, 2016,  (V...
19  White alone, not Hispanic or Latino, percent, ...
20                                Veterans, 2011-2015
21           Foreign born persons, percent, 2011-2015
22             Housing units,  July 1, 2016,  (V2016)
23                       Housing units, April 1, 2010
24        Owner-occupied housing unit rate, 2011-2015
25  Median value of owner-occupied housing units, ...
26  Median selected monthly owner costs -with a mo...
27  Median selected monthly owner costs -without a...
28                       Median gross rent, 2011-2015
29                             Building permits, 2016
..                                                ...
36  With a disability, under age 65 years, percent...
37  Persons  without health insurance, under age 6...
38  In civilian labor force, total, percent of pop...
39  In civilian labor force, female, percent of po...
40  Total accommodation and food services sales, 2...
41  Total health care and social assistance receip...
42       Total manufacturers shipments, 2012 ($1,000)
43     Total merchant wholesaler sales, 2012 ($1,000)
44                Total retail sales, 2012 ($1,000)
45                 Total retail sales per capita, 2012
46  Mean travel time to work (minutes), workers ag...
47  Median household income (in 2015 dollars), 201...
48  Per capita income in past 12 months (in 2015 d...
49                         Persons in poverty, percent
50                 Total employer establishments, 2015
51                             Total employment, 2015
52                Total annual payroll, 2015 ($1,000)
53        Total employment, percent change, 2014-2015
54             Total nonemployer establishments, 2015
55                                    All firms, 2012
56                              Men-owned firms, 2012
57                            Women-owned firms, 2012
58                         Minority-owned firms, 2012
59                      Nonminority-owned firms, 2012
60                           Veteran-owned firms, 2012
61                        Nonveteran-owned firms, 2012
62                  Population per square mile, 2010
63                   Land area in square miles, 2010
64                                          FIPS Code
65                                                non

                                 Fact Note      Alabama        Alaska  \
0                                      non    4863300.0  7.418940e+05
1                                      non    4780131.0  7.102490e+05
```

2	non	1.0	4.000000e+00
3	non	4779736.0	7.102310e+05
4	non	6.0	7.000000e+00
5	non	6.0	7.000000e+00
6	non	22.0	2.500000e+01
7	non	23.0	2.600000e+01
8	non	16.0	1.000000e+01
9	non	13.0	7.000000e+00
10	non	51.0	4.700000e+01
11	non	51.0	4.800000e+01
12	(a)	69.0	6.600000e+01
13	(a)	26.0	3.000000e+00
14	(a)	0.0	1.500000e+01
15	(a)	1.0	6.000000e+00
16	(a)	0.0	1.000000e+00
17	non	1.0	7.000000e+00
18	(b)	4.0	7.000000e+00
19	non	65.0	6.100000e+01
20	non	363170.0	6.932300e+04
21	non	3.0	7.000000e+00
22	non	2230185.0	3.106580e+05
23	non	2171853.0	3.069670e+05
24	non	68.0	6.300000e+01
25	non	125500.0	2.500000e+05
26	non	1139.0	1.827000e+03
27	non	345.0	5.540000e+02
28	non	717.0	1.146000e+03
29	non	15001.0	1.503000e+03
..	...	...	...
36	non	11.0	8.000000e+00
37	non	10.0	1.500000e+01
38	non	58.0	6.700000e+01
39	non	53.0	6.500000e+01
40	(c)	7576462.0	2.221335e+06
41	(c)	26039632.0	6.375483e+06
42	(c)	124809759.0	6.965095e+05
43	(c)	57746565.0	5.216303e+06
44	(c)	58564965.0	1.047428e+07
45	(c)	12145.0	1.432000e+04
46	non	24.0	1.900000e+01
47	non	43623.0	7.251500e+04
48	non	24091.0	3.341300e+04
49	non	17.0	9.000000e+00
50	Includes data not distributed by county.	98540.0	2.090700e+04
51	Includes data not distributed by county.	1634391.0	2.679990e+05
52	Includes data not distributed by county.	67370353.0	1.564330e+07
53	Includes data not distributed by county.	1.0	0.000000e+00
54	non	322025.0	5.552100e+04

```
55                                              non     374153.0  6.803200e+04
56                                              non     203604.0  3.540200e+04
57                                              non     137630.0  2.214100e+04
58                                              non      92219.0  1.368800e+04
59                                              non     272651.0  5.114700e+04
60                                              non      41943.0  7.953000e+03
61                                              non     316984.0  5.609100e+04
62                                              non         94.0  1.000000e+00
63                                              non      50645.0  5.706400e+05
64                                              non          1.0  2.000000e+00
65                                              non    5644810.4  6.965095e+05

         Arizona      Arkansas    California      Colorado   Connecticut  \
0   6.931071e+06  2.988248e+06  3.925002e+07  5.540545e+06  3.576452e+06
1   6.392301e+06  2.916025e+06  3.725452e+07  5.029324e+06  3.574114e+06
2   8.000000e+00  2.000000e+00  5.000000e+00  1.000000e+01  0.000000e+00
3   6.392017e+06  2.915918e+06  3.725396e+07  5.029196e+06  3.574097e+06
4   6.000000e+00  6.000000e+00  6.000000e+00  6.000000e+00  5.000000e+00
5   7.000000e+00  6.000000e+00  6.000000e+00  6.000000e+00  5.000000e+00
6   2.300000e+01  2.300000e+01  2.300000e+01  2.200000e+01  2.100000e+01
7   2.500000e+01  2.400000e+01  2.500000e+01  2.400000e+01  2.200000e+01
8   1.600000e+01  1.600000e+01  1.300000e+01  1.300000e+01  1.600000e+01
9   1.300000e+01  1.400000e+01  1.100000e+01  1.000000e+01  1.400000e+01
10  5.000000e+01  5.000000e+01  5.000000e+01  4.900000e+01  5.100000e+01
11  5.000000e+01  5.000000e+01  5.000000e+01  4.900000e+01  5.100000e+01
12  8.300000e+01  7.900000e+01  7.200000e+01  8.700000e+01  8.000000e+01
13  4.000000e+00  1.500000e+01  6.000000e+00  4.000000e+00  1.100000e+01
14  5.000000e+00  1.000000e+00  1.000000e+00  1.000000e+00  0.000000e+00
15  3.000000e+00  1.000000e+00  1.400000e+01  3.000000e+00  4.000000e+00
16  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
17  2.000000e+00  2.000000e+00  3.000000e+00  3.000000e+00  2.000000e+00
18  3.000000e+01  7.000000e+00  3.800000e+01  2.100000e+01  1.500000e+01
19  5.500000e+01  7.200000e+01  3.700000e+01  6.800000e+01  6.700000e+01
20  5.057940e+05  2.209530e+05  1.777410e+06  3.917250e+05  1.993310e+05
21  1.300000e+01  4.000000e+00  2.700000e+01  9.000000e+00  1.300000e+01
22  2.961003e+06  1.354762e+06  1.406052e+07  2.339118e+06  1.499116e+06
23  2.844526e+06  1.316299e+06  1.368008e+07  2.212898e+06  1.487891e+06
24  6.200000e+01  6.600000e+01  5.400000e+01  6.400000e+01  6.700000e+01
25  1.675000e+05  1.114000e+05  3.855000e+05  2.478000e+05  2.705000e+05
26  1.343000e+03  1.019000e+03  2.155000e+03  1.577000e+03  2.067000e+03
27  3.800000e+02  3.270000e+02  5.000000e+02  4.190000e+02  8.330000e+02
28  9.130000e+02  6.770000e+02  1.255000e+03  1.002000e+03  1.075000e+03
29  3.557800e+04  9.474000e+03  1.023500e+05  3.897400e+04  5.504000e+03
..           ...           ...           ...           ...           ...
36  8.000000e+00  1.200000e+01  6.000000e+00  7.000000e+00  7.000000e+00
37  1.100000e+01  9.000000e+00  8.000000e+00  8.000000e+00  5.000000e+00
38  5.900000e+01  5.800000e+01  6.300000e+01  6.700000e+01  6.700000e+01
39  5.400000e+01  5.300000e+01  5.700000e+01  6.200000e+01  6.200000e+01
```

```
40  1.399664e+07  4.307264e+06  9.083037e+07  1.361765e+07  9.542068e+06
41  3.705588e+07  1.579263e+07  2.489536e+08  2.948816e+07  2.957312e+07
42  5.124347e+07  6.271292e+07  5.123032e+08  5.044710e+07  5.516010e+07
43  6.943727e+07  3.125611e+07  6.666522e+08  7.703497e+07  1.619622e+08
44  8.471654e+07  3.681526e+07  4.818005e+08  6.781520e+07  5.163247e+07
45  1.292700e+04  1.248300e+04  1.266500e+04  1.307300e+04  1.438100e+04
46  2.400000e+01  2.100000e+01  2.800000e+01  2.400000e+01  2.500000e+01
47  5.025500e+04  4.137100e+04  6.181800e+04  6.062900e+04  7.033100e+04
48  2.584800e+04  2.279800e+04  3.031800e+04  3.221700e+04  3.880300e+04
49  1.600000e+01  1.700000e+01  1.400000e+01  1.100000e+01  9.000000e+00
50  1.363520e+05  6.517500e+04  9.081200e+05  1.617370e+05  8.923200e+04
51  2.295186e+06  1.003113e+06  1.432538e+07  2.253795e+06  1.503102e+06
52  1.026714e+08  3.945119e+07  8.569542e+08  1.175396e+08  9.255507e+07
53  2.000000e+00  1.000000e+00  3.000000e+00  3.000000e+00  1.000000e+00
54  4.519510e+05  1.983800e+05  3.206958e+06  4.808470e+05  2.728090e+05
55  4.999260e+05  2.319590e+05  3.548449e+06  5.473520e+05  3.266930e+05
56  2.452430e+05  1.231580e+05  1.852580e+06  2.845540e+05  1.878450e+05
57  1.824250e+05  7.596200e+04  1.320085e+06  1.945080e+05  1.066780e+05
58  1.353130e+05  3.598200e+04  1.619857e+06  8.584900e+04  5.611300e+04
59  3.449810e+05  1.890290e+05  1.819107e+06  4.423650e+05  2.596140e+05
60  4.678000e+04  2.591500e+04  2.523770e+05  5.172200e+04  3.105600e+04
61  4.275820e+05  1.929880e+05  3.176341e+06  4.695240e+05  2.811820e+05
62  5.600000e+01  5.600000e+01  2.390000e+02  4.800000e+01  7.380000e+02
63  1.135940e+05  5.203500e+04  1.557790e+05  1.036410e+05  4.842000e+03
64  4.000000e+00  5.000000e+00  6.000000e+00  8.000000e+00  9.000000e+00
65  6.042079e+06  3.162765e+06  4.686571e+07  5.907422e+06  6.449429e+06

        Delaware      ...      South Dakota    Tennessee         Texas  \
0   9.520650e+05      ...      8.654540e+05  6.651194e+06  2.786260e+07
1   8.979360e+05      ...      8.141950e+05  6.346298e+06  2.514610e+07
2   6.000000e+00      ...      0.000000e+00  0.000000e+00  1.000000e+01
3   8.979340e+05      ...      8.141800e+05  6.346105e+06  2.514556e+07
4   5.000000e+00      ...      0.000000e+00  0.000000e+00  7.000000e+00
5   6.000000e+00      ...      0.000000e+00  0.000000e+00  7.000000e+00
6   2.100000e+01      ...      0.000000e+00  0.000000e+00  2.600000e+01
7   2.200000e+01      ...      0.000000e+00  0.000000e+00  2.700000e+01
8   1.700000e+01      ...      0.000000e+00  0.000000e+00  1.200000e+01
9   1.400000e+01      ...      0.000000e+00  0.000000e+00  1.000000e+01
10  5.100000e+01      ...      0.000000e+00  0.000000e+00  5.000000e+01
11  5.100000e+01      ...      0.000000e+00  0.000000e+00  5.000000e+01
12  7.000000e+01      ...      0.000000e+00  0.000000e+00  7.900000e+01
13  2.200000e+01      ...      0.000000e+00  0.000000e+00  1.200000e+01
14  0.000000e+00      ...      0.000000e+00  0.000000e+00  1.000000e+00
15  4.000000e+00      ...      0.000000e+00  0.000000e+00  4.000000e+00
16  0.000000e+00      ...      0.000000e+00  0.000000e+00  0.000000e+00
17  2.000000e+00      ...      0.000000e+00  0.000000e+00  1.000000e+00
18  9.000000e+00      ...      0.000000e+00  0.000000e+00  3.900000e+01
19  6.200000e+01      ...      0.000000e+00  0.000000e+00  4.200000e+01
```

```
20  7.121300e+04       ...       6.374200e+04  4.624140e+05  1.539655e+06
21  8.000000e+00       ...       0.000000e+00  0.000000e+00  1.600000e+01
22  4.261490e+05       ...       3.838380e+05  2.919671e+06  1.075363e+07
23  4.058850e+05       ...       3.634380e+05  2.812133e+06  9.977436e+06
24  7.100000e+01       ...       0.000000e+00  0.000000e+00  6.200000e+01
25  2.315000e+05       ...       1.405000e+05  1.421000e+05  1.360000e+05
26  1.537000e+03       ...       1.210000e+03  1.181000e+03  1.432000e+03
27  4.450000e+02       ...       4.330000e+02  3.590000e+02  4.600000e+02
28  1.018000e+03       ...       6.550000e+02  7.640000e+02  8.820000e+02
29  5.804000e+03       ...       5.686000e+03  3.615700e+04  1.658530e+05
..           ...       ...                ...           ...           ...
36  8.000000e+00       ...       0.000000e+00  0.000000e+00  8.000000e+00
37  6.000000e+00       ...       0.000000e+00  0.000000e+00  1.800000e+01
38  6.300000e+01       ...       0.000000e+00  0.000000e+00  6.400000e+01
39  5.900000e+01       ...       0.000000e+00  0.000000e+00  5.700000e+01
40  2.148437e+06       ...       1.873699e+06  1.249901e+07  5.448081e+07
41  7.003251e+06       ...       6.211731e+06  4.238368e+07  1.450351e+08
42  2.259738e+07       ...       1.688265e+07  1.399605e+08  7.026031e+08
43  5.628914e+06       ...       2.041106e+07  1.117184e+08  6.912426e+08
44  1.445600e+07       ...       1.379183e+07  9.164160e+07  3.561164e+08
45  1.576300e+04       ...       1.655000e+04  1.419400e+04  1.366600e+04
46  2.500000e+01       ...       1.600000e+01  2.400000e+01  2.500000e+01
47  6.050900e+04       ...       5.095700e+04  4.521900e+04  5.320700e+04
48  3.055400e+04       ...       2.674700e+04  2.522700e+04  2.699900e+04
49  1.100000e+01       ...       0.000000e+00  0.000000e+00  1.500000e+01
50  2.485200e+04       ...       2.651100e+04  1.333440e+05  5.690910e+05
51  3.973850e+05       ...       3.535400e+05  2.507205e+06  1.023971e+07
52  2.130523e+07       ...       1.381300e+07  1.104813e+08  5.210958e+08
53  1.000000e+00       ...       0.000000e+00  0.000000e+00  3.000000e+00
54  6.073400e+04       ...       6.400600e+04  4.957030e+05  2.205149e+06
55  7.341800e+04       ...       8.131400e+04  5.504530e+05  2.356748e+06
56  3.832800e+04       ...       4.241800e+04  3.022490e+05  1.251696e+06
57  2.396400e+04       ...       2.372200e+04  1.956940e+05  8.666780e+05
58  1.444000e+04       ...       4.101000e+03  1.052340e+05  1.070392e+06
59  5.478200e+04       ...       7.422800e+04  4.340250e+05  1.224845e+06
60  7.206000e+03       ...       8.604000e+03  5.937900e+04  2.135900e+05
61  6.031800e+04       ...       6.621900e+04  4.693920e+05  2.057218e+06
62  4.600000e+02       ...       1.000000e+01  1.530000e+02  9.600000e+01
63  1.948000e+03       ...       7.581100e+04  4.123400e+04  2.612310e+05
64  1.000000e+01       ...       4.600000e+01  4.700000e+01  4.800000e+01
65  1.203696e+06       ...       1.195122e+06  8.342865e+06  4.004406e+07

            Utah       Vermont      Virginia    Washington  West Virginia  \
0   3.051217e+06  6.245940e+05  8.411808e+06  7.288000e+06   1.831102e+06
1   2.763888e+06  6.257410e+05  8.001041e+06  6.724545e+06   1.853011e+06
2   1.000000e+01  0.000000e+00  5.000000e+00  8.000000e+00   1.000000e+00
3   2.763885e+06  6.257410e+05  8.001024e+06  6.724540e+06   1.852994e+06
4   8.000000e+00  4.000000e+00  6.000000e+00  6.000000e+00   5.000000e+00
```

5	9.000000e+00	5.000000e+00	6.000000e+00	6.000000e+00	5.000000e+00
6	3.000000e+01	1.900000e+01	2.200000e+01	2.200000e+01	2.000000e+01
7	3.100000e+01	2.000000e+01	2.300000e+01	2.300000e+01	2.000000e+01
8	1.000000e+01	1.800000e+01	1.400000e+01	1.400000e+01	1.800000e+01
9	9.000000e+00	1.400000e+01	1.200000e+01	1.200000e+01	1.600000e+01
10	4.900000e+01	5.000000e+01	5.000000e+01	5.000000e+01	5.000000e+01
11	4.900000e+01	5.000000e+01	5.000000e+01	5.000000e+01	5.000000e+01
12	9.100000e+01	9.400000e+01	7.000000e+01	8.000000e+01	9.300000e+01
13	1.000000e+00	1.000000e+00	1.900000e+01	4.000000e+00	3.000000e+00
14	1.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00	0.000000e+00
15	2.000000e+00	1.000000e+00	6.000000e+00	8.000000e+00	0.000000e+00
16	1.000000e+00	7.208260e+05	0.000000e+00	0.000000e+00	1.710532e+06
17	2.000000e+00	1.000000e+00	2.000000e+00	4.000000e+00	1.000000e+00
18	1.300000e+01	1.000000e+00	9.000000e+00	1.200000e+01	1.000000e+00
19	7.800000e+01	9.300000e+01	6.200000e+01	6.900000e+01	9.200000e+01
20	1.343320e+05	4.470800e+04	7.065390e+05	5.648640e+05	1.500210e+05
21	8.000000e+00	4.000000e+00	1.100000e+01	1.300000e+01	1.000000e+00
22	1.054164e+06	3.295250e+05	3.491054e+06	3.025685e+06	8.866400e+05
23	9.797090e+05	3.225390e+05	3.364939e+06	2.885677e+06	8.819170e+05
24	6.900000e+01	7.100000e+01	6.600000e+01	6.200000e+01	7.200000e+01
25	2.159000e+05	2.175000e+05	2.450000e+05	2.595000e+05	1.038000e+05
26	1.428000e+03	1.535000e+03	1.711000e+03	1.731000e+03	9.660000e+02
27	3.880000e+02	6.410000e+02	4.330000e+02	5.110000e+02	2.930000e+02
28	8.870000e+02	8.950000e+02	1.116000e+03	1.014000e+03	6.430000e+02
29	2.266200e+04	1.771000e+03	3.113200e+04	4.407700e+04	2.544000e+03
..	...	...	...	...	...
36	6.000000e+00	1.000000e+01	7.000000e+00	8.000000e+00	1.400000e+01
37	9.000000e+00	4.000000e+00	1.000000e+01	6.000000e+00	6.000000e+00
38	6.700000e+01	6.600000e+01	6.400000e+01	6.300000e+01	5.300000e+01
39	5.900000e+01	6.300000e+01	6.000000e+01	5.800000e+01	4.900000e+01
40	4.789281e+06	1.564272e+06	1.779590e+07	1.429728e+07	4.036333e+06
41	1.452186e+07	4.457996e+06	4.770500e+07	4.396689e+07	1.225940e+07
42	5.004643e+07	9.315494e+06	9.638987e+07	1.315306e+08	2.455307e+07
43	3.092788e+07	6.450076e+06	8.661364e+07	8.331337e+07	1.429544e+07
44	3.802449e+07	9.933751e+06	1.100024e+08	1.189240e+08	2.263792e+07
45	1.331700e+04	1.586800e+04	1.343800e+04	1.724300e+04	1.220100e+04
46	2.100000e+01	2.200000e+01	2.700000e+01	2.600000e+01	2.500000e+01
47	6.072700e+04	5.517600e+04	6.501500e+04	6.106200e+04	4.175100e+04
48	2.468600e+04	2.989400e+04	3.415200e+04	3.176200e+04	2.345000e+04
49	1.000000e+01	1.100000e+01	1.100000e+01	1.100000e+01	1.700000e+01
50	7.546300e+04	2.112100e+04	1.973840e+05	1.829130e+05	3.699300e+04
51	1.203954e+06	2.663630e+05	3.198718e+06	2.602408e+06	5.654350e+05
52	5.145327e+07	1.061509e+07	1.657889e+08	1.492588e+08	2.215908e+07
53	4.000000e+00	2.000000e+00	1.000000e+00	2.000000e+00	1.000000e+00
54	2.162800e+05	6.031200e+04	5.764460e+05	4.441350e+05	8.813600e+04
55	2.514190e+05	7.582700e+04	6.531930e+05	5.415220e+05	1.144350e+05
56	1.321630e+05	4.127000e+04	3.530120e+05	2.626500e+05	6.311200e+04
57	7.626900e+04	2.341700e+04	2.362900e+05	1.876770e+05	3.906500e+04

58	2.442300e+04	2.354000e+03	1.850430e+05	9.280700e+04	5.777000e+03
59	2.188260e+05	7.049100e+04	4.501090e+05	4.266970e+05	1.047850e+05
60	1.875400e+04	8.237000e+03	7.643400e+04	4.933100e+04	1.291200e+04
61	2.198070e+05	6.331700e+04	5.484390e+05	4.614010e+05	9.496000e+04
62	3.300000e+01	6.700000e+01	2.020000e+02	1.010000e+02	7.700000e+01
63	8.216900e+04	9.216000e+03	3.949000e+04	6.645500e+04	2.403800e+04
64	4.900000e+01	5.000000e+01	5.100000e+01	5.300000e+01	5.400000e+01
65	3.142726e+06	7.208260e+05	8.711424e+06	8.875524e+06	1.710532e+06

	Wisconsin	Wyoming
0	5.778708e+06	5.855010e+05
1	5.687289e+06	5.637670e+05
2	1.000000e+00	3.000000e+00
3	5.686986e+06	5.636260e+05
4	5.000000e+00	6.000000e+00
5	6.000000e+00	7.000000e+00
6	2.200000e+01	2.300000e+01
7	2.300000e+01	2.400000e+01
8	1.600000e+01	1.500000e+01
9	1.300000e+01	1.200000e+01
10	5.000000e+01	4.800000e+01
11	5.000000e+01	4.900000e+01
12	8.700000e+01	9.200000e+01
13	6.000000e+00	1.000000e+00
14	1.000000e+00	2.000000e+00
15	2.000000e+00	1.000000e+00
16	0.000000e+00	0.000000e+00
17	1.000000e+00	2.000000e+00
18	6.000000e+00	1.000000e+01
19	8.100000e+01	8.400000e+01
20	3.819400e+05	4.850500e+04
21	4.000000e+00	3.000000e+00
22	2.668444e+06	2.706000e+05
23	2.624358e+06	2.618680e+05
24	6.700000e+01	6.900000e+01
25	1.658000e+05	1.948000e+05
26	1.402000e+03	1.348000e+03
27	5.320000e+02	3.860000e+02
28	7.760000e+02	7.890000e+02
29	1.927400e+04	1.727000e+03
..	...	...
36	8.000000e+00	8.000000e+00
37	6.000000e+00	1.300000e+01
38	6.700000e+01	6.700000e+01
39	6.300000e+01	6.200000e+01
40	1.030326e+07	1.644844e+06
41	4.068062e+07	3.291478e+06
42	1.777289e+08	1.078379e+07

```
43  7.706688e+07  5.597891e+06
44  7.820182e+07  9.446043e+06
45  1.365600e+04  1.638800e+04
46  2.100000e+01  1.800000e+01
47  5.335700e+04  5.884000e+04
48  2.834000e+04  2.980300e+04
49  1.100000e+01  1.100000e+01
50  1.395000e+05  2.104000e+04
51  2.503532e+06  2.198810e+05
52  1.124065e+08  1.009401e+07
53  2.000000e+00  6.921724e+05
54  3.419350e+05  4.814000e+04
55  4.329800e+05  6.242700e+04
56  2.362520e+05  3.003900e+04
57  1.338590e+05  1.934400e+04
58  4.050700e+04  4.077000e+03
59  3.799340e+05  5.539700e+04
60  3.983000e+04  6.470000e+03
61  3.707550e+05  5.135300e+04
62  1.050000e+02  5.000000e+00
63  5.415700e+04  9.709300e+04
64  5.500000e+01  5.600000e+01
65  8.099572e+06  6.921724e+05

[66 rows x 52 columns]
```

In [23]:
```
# confirm nics data after cleaning
df_gun.head()
```

Out[23]:
```
       month       state   permit  permit_recheck  handgun  long_gun   other  \
0 2017-09-01     Alabama  16717.0             0.0   5734.0    6320.0   221.0
1 2017-09-01      Alaska    209.0             2.0   2320.0    2930.0   219.0
2 2017-09-01     Arizona   5069.0           382.0  11063.0    7946.0   920.0
3 2017-09-01    Arkansas   2935.0           632.0   4347.0    6063.0   165.0
4 2017-09-01  California  57839.0             0.0  37165.0   24581.0  2984.0

   multiple  admin  prepawn_handgun   ...     returned_other  rentals_handgun  \
0       317    0.0             15.0   ...                0.0              0.0
1       160    0.0              5.0   ...                0.0              0.0
2       631    0.0             13.0   ...                0.0              0.0
3       366   51.0             12.0   ...                0.0              0.0
4         0    0.0              0.0   ...                0.0              0.0

   rentals_long_gun  private_sale_handgun  private_sale_long_gun  \
0               0.0                   9.0                   16.0
1               0.0                  17.0                   24.0
2               0.0                  38.0                   12.0
3               0.0                  13.0                   23.0
```

```
4               0.0                    0.0                     0.0

   private_sale_other  return_to_seller_handgun  return_to_seller_long_gun  \
0                 3.0                       0.0                        0.0
1                 1.0                       0.0                        0.0
2                 2.0                       0.0                        0.0
3                 0.0                       0.0                        2.0
4                 0.0                       0.0                        0.0

   return_to_seller_other  totals
0                     3.0   32019
1                     0.0    6303
2                     0.0   28394
3                     1.0   17747
4                     0.0  123506

[5 rows x 27 columns]
```

In [24]:
```
# show some general properties
df_census.describe()
```

Out[24]:
```
            Alabama        Alaska       Arizona      Arkansas    California  \
count  8.200000e+01  8.200000e+01  8.200000e+01  8.200000e+01  8.200000e+01
mean   5.644810e+06  6.965095e+05  6.042079e+06  3.162765e+06  4.686571e+07
std    1.778849e+07  2.217958e+06  1.739900e+07  9.604165e+06  1.408072e+08
min    0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
25%    2.325000e+01  2.050000e+01  2.625000e+01  2.150000e+01  3.250000e+01
50%    4.713400e+04  2.152400e+04  8.192450e+04  3.867650e+04  2.040780e+05
75%    5.644810e+06  6.965095e+05  6.042079e+06  3.162765e+06  4.686571e+07
max    1.248098e+08  1.564330e+07  1.026714e+08  6.271292e+07  8.569542e+08

           Colorado   Connecticut      Delaware       Florida       Georgia  \
count  8.200000e+01  8.200000e+01  8.200000e+01  8.200000e+01  8.200000e+01
mean   5.907422e+06  6.449429e+06  1.203696e+06  1.905439e+07  1.101339e+07
std    1.784105e+07  2.189089e+07  3.797209e+06  5.638734e+07  3.246234e+07
min    0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
25%    2.400000e+01  2.275000e+01  2.275000e+01  2.625000e+01  2.725000e+01
50%    7.323900e+04  4.745800e+04  1.986350e+04  1.376200e+05  7.715000e+04
75%    5.907422e+06  6.449429e+06  1.203696e+06  1.905439e+07  1.101339e+07
max    1.175396e+08  1.619622e+08  2.259738e+07  3.370745e+08  1.748394e+08

           ...    South Dakota     Tennessee         Texas          Utah  \
count      ...    8.200000e+01  8.200000e+01  8.200000e+01  8.200000e+01
mean       ...    1.195122e+06  8.342865e+06  4.004406e+07  3.142726e+06
std        ...    3.573824e+06  2.490916e+07  1.262991e+08  9.412073e+06
min        ...    0.000000e+00  0.000000e+00  0.000000e+00  1.000000e+00
25%        ...    0.000000e+00  0.000000e+00  3.600000e+01  3.100000e+01
50%        ...    2.511650e+04  5.229900e+04  1.897215e+05  4.270650e+04
```

```
75%          ...      1.195122e+06  8.342865e+06  4.004406e+07  3.142726e+06
max          ...      2.041106e+07  1.399605e+08  7.026031e+08  5.145327e+07

            Vermont      Virginia    Washington  West Virginia     Wisconsin  \
count  8.200000e+01  8.200000e+01  8.200000e+01   8.200000e+01  8.200000e+01
mean   7.208260e+05  8.711424e+06  8.875524e+06   1.710532e+06  8.099572e+06
std    2.019675e+06  2.588748e+07  2.672587e+07   4.717679e+06  2.579573e+07
min    0.000000e+00  0.000000e+00  0.000000e+00   0.000000e+00  0.000000e+00
25%    3.950000e+01  2.925000e+01  2.750000e+01   3.100000e+01  2.400000e+01
50%    2.226900e+04  7.072450e+04  6.375850e+04   3.051550e+04  4.693200e+04
75%    7.208260e+05  8.711424e+06  8.875524e+06   1.710532e+06  8.099572e+06
max    1.061509e+07  1.657889e+08  1.492588e+08   2.455307e+07  1.777289e+08

            Wyoming
count  8.200000e+01
mean   6.921724e+05
std    1.996627e+06
min    0.000000e+00
25%    2.425000e+01
50%    2.542150e+04
75%    6.921724e+05
max    1.078379e+07

[8 rows x 50 columns]
```

In [25]:
```
# show some general properties
df_gun.describe()
```

Out[25]:
```
              permit  permit_recheck        handgun       long_gun  \
count   12485.000000    12485.000000   12485.000000   12485.000000
mean     6413.629404     1165.956364    5940.881107    7810.847585
std     23729.495816     2736.848174    8611.677589    9302.758891
min         0.000000        0.000000       0.000000       0.000000
25%         0.000000     1165.956364     868.000000    2079.000000
50%       522.000000     1165.956364    3067.000000    5130.000000
75%      4338.000000     1165.956364    7277.000000   10374.000000
max    522188.000000   116681.000000  107224.000000  108058.000000

              other      multiple         admin  prepawn_handgun  \
count  12485.000000  12485.000000  12485.000000     12485.000000
mean     360.471636    268.603364     58.898090         4.828021
std      895.634628    783.185073    604.257419        10.023040
min        0.000000      0.000000      0.000000         0.000000
25%      163.000000     15.000000      0.000000         0.000000
50%      360.471636    125.000000      0.000000         1.000000
75%      360.471636    301.000000      0.000000         4.828021
max    77929.000000  38907.000000  28083.000000       164.000000
```

```
       prepawn_long_gun  prepawn_other      ...        returned_other  \
count      12485.000000   12485.000000      ...          12485.000000
mean           7.834156       0.165591      ...              1.027548
std           15.130888       0.676584      ...              1.672013
min            0.000000       0.000000      ...              0.000000
25%            0.000000       0.000000      ...              1.027548
50%            3.000000       0.165591      ...              1.027548
75%            7.834156       0.165591      ...              1.027548
max          269.000000      49.000000      ...             64.000000

       rentals_handgun  rentals_long_gun  private_sale_handgun  \
count     12485.000000      12485.000000          12485.000000
mean          0.076768          0.087273             14.936000
std           0.178589          0.172556             33.418596
min           0.000000          0.000000              0.000000
25%           0.076768          0.087273             14.936000
50%           0.076768          0.087273             14.936000
75%           0.076768          0.087273             14.936000
max          12.000000         12.000000           1017.000000

       private_sale_long_gun  private_sale_other  return_to_seller_handgun  \
count           12485.000000        12485.000000              12485.000000
mean               11.602909            1.030182                  0.402020
std                25.458626            2.096565                  0.643964
min                 0.000000            0.000000                  0.000000
25%                11.602909            1.030182                  0.402020
50%                11.602909            1.030182                  0.402020
75%                11.602909            1.030182                  0.402020
max               777.000000           71.000000                 28.000000

       return_to_seller_long_gun  return_to_seller_other         totals
count               12485.000000            12485.000000   12485.000000
mean                    0.441818                0.105987   21595.725911
std                     0.717129                0.181592   32591.418387
min                     0.000000                0.000000       0.000000
25%                     0.441818                0.105987    4638.000000
50%                     0.441818                0.105987   12399.000000
75%                     0.441818                0.105987   25453.000000
max                    17.000000                4.000000  541978.000000

[8 rows x 25 columns]
```

In [26]:
```
# show data in Fact column
df_census['Fact']
```

Out[26]:
```
0           Population estimates, July 1, 2016,  (V2016)
1      Population estimates base, April 1, 2010,  (V2...
2      Population, percent change - April 1, 2010 (es...
```

```
3                  Population, Census, April 1, 2010
4     Persons under 5 years, percent, July 1, 2016, ...
5         Persons under 5 years, percent, April 1, 2010
6     Persons under 18 years, percent, July 1, 2016,...
7        Persons under 18 years, percent, April 1, 2010
8     Persons 65 years and over, percent,  July 1, 2...
9     Persons 65 years and over, percent, April 1, 2010
10     Female persons, percent,  July 1, 2016,  (V2016)
11               Female persons, percent, April 1, 2010
12          White alone, percent, July 1, 2016,  (V2016)
13    Black or African American alone, percent, July...
14    American Indian and Alaska Native alone, perce...
15          Asian alone, percent, July 1, 2016,  (V2016)
16    Native Hawaiian and Other Pacific Islander alo...
17    Two or More Races, percent, July 1, 2016,  (V2...
18    Hispanic or Latino, percent, July 1, 2016,  (V...
19    White alone, not Hispanic or Latino, percent, ...
20                                Veterans, 2011-2015
21             Foreign born persons, percent, 2011-2015
22               Housing units,  July 1, 2016,  (V2016)
23                         Housing units, April 1, 2010
24           Owner-occupied housing unit rate, 2011-2015
25    Median value of owner-occupied housing units, ...
26    Median selected monthly owner costs -with a mo...
27    Median selected monthly owner costs -without a...
28                         Median gross rent, 2011-2015
29                               Building permits, 2016
                          ...
52                Total annual payroll, 2015 ($1,000)
53          Total employment, percent change, 2014-2015
54               Total nonemployer establishments, 2015
55                                      All firms, 2012
56                                Men-owned firms, 2012
57                              Women-owned firms, 2012
58                           Minority-owned firms, 2012
59                        Nonminority-owned firms, 2012
60                            Veteran-owned firms, 2012
61                         Nonveteran-owned firms, 2012
62                     Population per square mile, 2010
63                      Land area in square miles, 2010
64                                            FIPS Code
65                                                  non
66    NOTE: FIPS Code values are enclosed in quotes ...
68                                          Value Notes
69                                                    1
71                                           Fact Notes
72                                                  (a)
73                                                  (b)
```

```
74                                                    (c)
76                                            Value Flags
77                                                      -
78                                                      D
79                                                      F
80                                                     FN
81                                                    non
82                                                      S
83                                                      X
84                                                      Z
Name: Fact, Length: 82, dtype: object
```

In [27]:
```
# check columns in df_gun to understand data
df_gun.columns
```

Out[27]:
```
Index(['month', 'state', 'permit', 'permit_recheck', 'handgun', 'long_gun',
       'other', 'multiple', 'admin', 'prepawn_handgun', 'prepawn_long_gun',
       'prepawn_other', 'redemption_handgun', 'redemption_long_gun',
       'redemption_other', 'returned_handgun', 'returned_long_gun',
       'returned_other', 'rentals_handgun', 'rentals_long_gun',
       'private_sale_handgun', 'private_sale_long_gun', 'private_sale_other',
       'return_to_seller_handgun', 'return_to_seller_long_gun',
       'return_to_seller_other', 'totals'],
      dtype='object')
```

Exploratory Data Analysis

In [28]:
```
# summary(it takes long time but you can run it !!!!)

# pd.plotting.scatter_matrix(df_gun, figsize=(50,50));
```

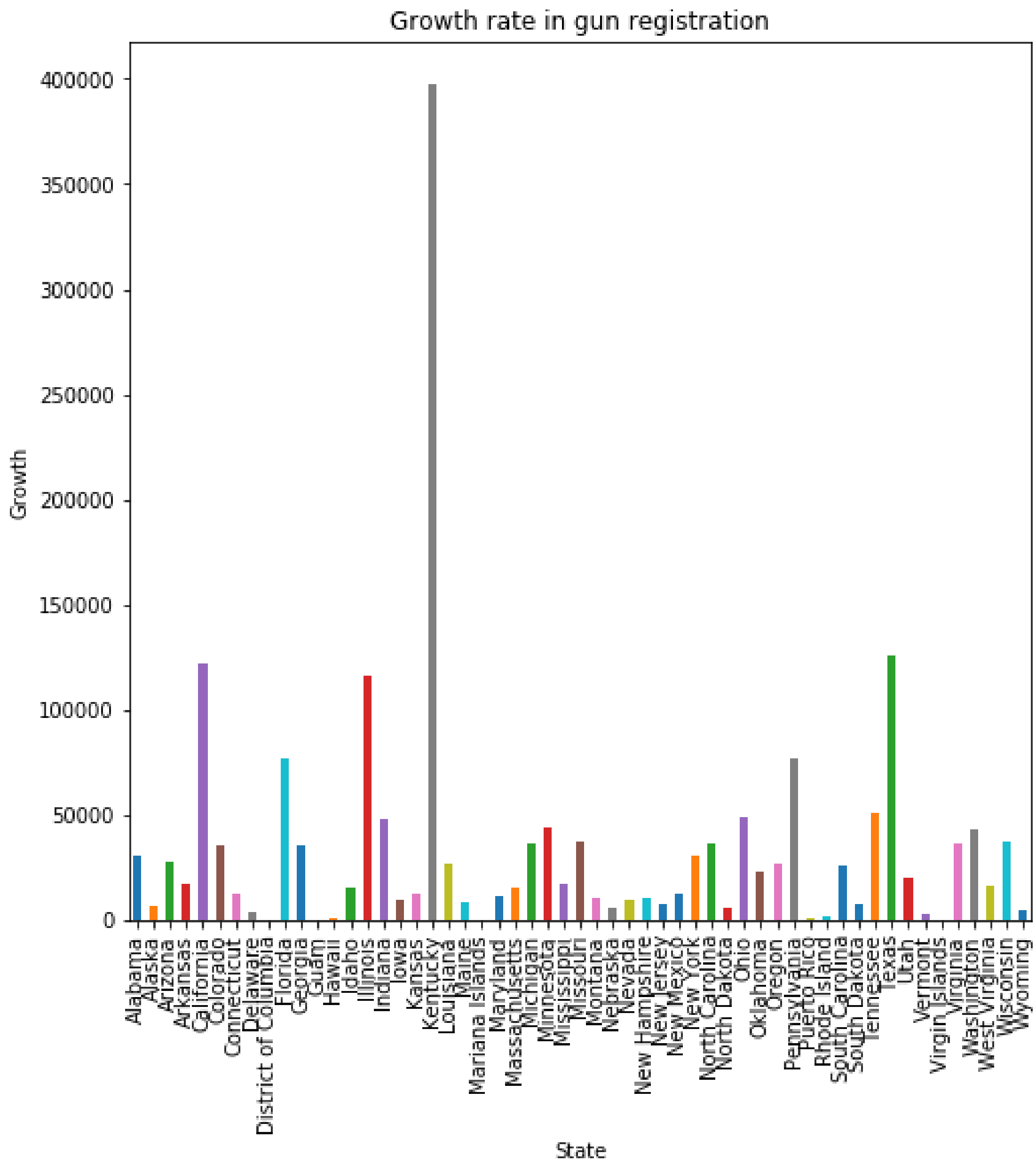
Research Question 1 (How much growth in gun registration in each state?)

In [29]:
```
# grouped data by month,state to show total registration
filt = df_gun.groupby(['month','state'])['totals'].sum()
```

In [30]:
```
# calculate growth rating by subtract total registration in current date from total reg
growth = filt.loc[df_gun['month'].max()] - filt.loc[df_gun['month'].min()]
```

In [31]:
```
# visualize result
growth_plot = growth.plot(kind='bar', figsize=(8,8), title='Growth rate in gun registra
growth_plot.set_xlabel('State')
growth_plot.set_ylabel('Growth')
```

Out[31]:
```
Text(0,0.5,'Growth')
```



In [32]:
```
# show max and min growth
growth[growth==growth.max()],growth[growth==growth.min()]
```

Out[32]:
```
(state
 Kentucky    397866
 Name: totals, dtype: int64, state
 Virgin Islands    9
 Name: totals, dtype: int64)
```

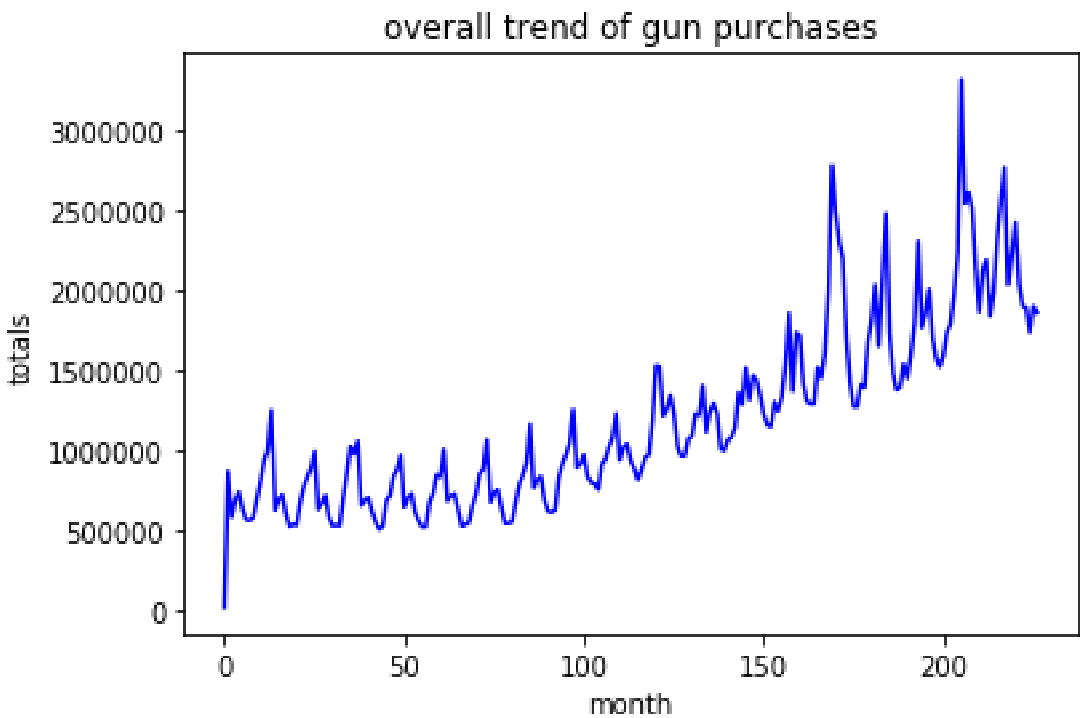
Maximum growth registration in (Kentucky) equal (397866)

Minimum growth registration in (Virgin) equal (9)

Research Question 2 (what is the overall trend of gun purchases?)

In [33]:
```
# grouped data to show totals respect month
df_gun_trend = df_gun.groupby('month')['totals'].sum()
#df_gun.month.dt.year.unique()
```

In [34]:
```
# visualize result using matplotlib
ind = np.arange(len(df_gun_trend))

plt.plot(ind,df_gun_trend, color='blue')
plt.title('overall trend of gun purchases')
plt.ylabel('totals')
plt.xlabel('month')
plt.figsize=(10,10)
```



overall trend of purchases is constantly increasing.

Research Question 3 (how many persons are permitted to own gun in last date)

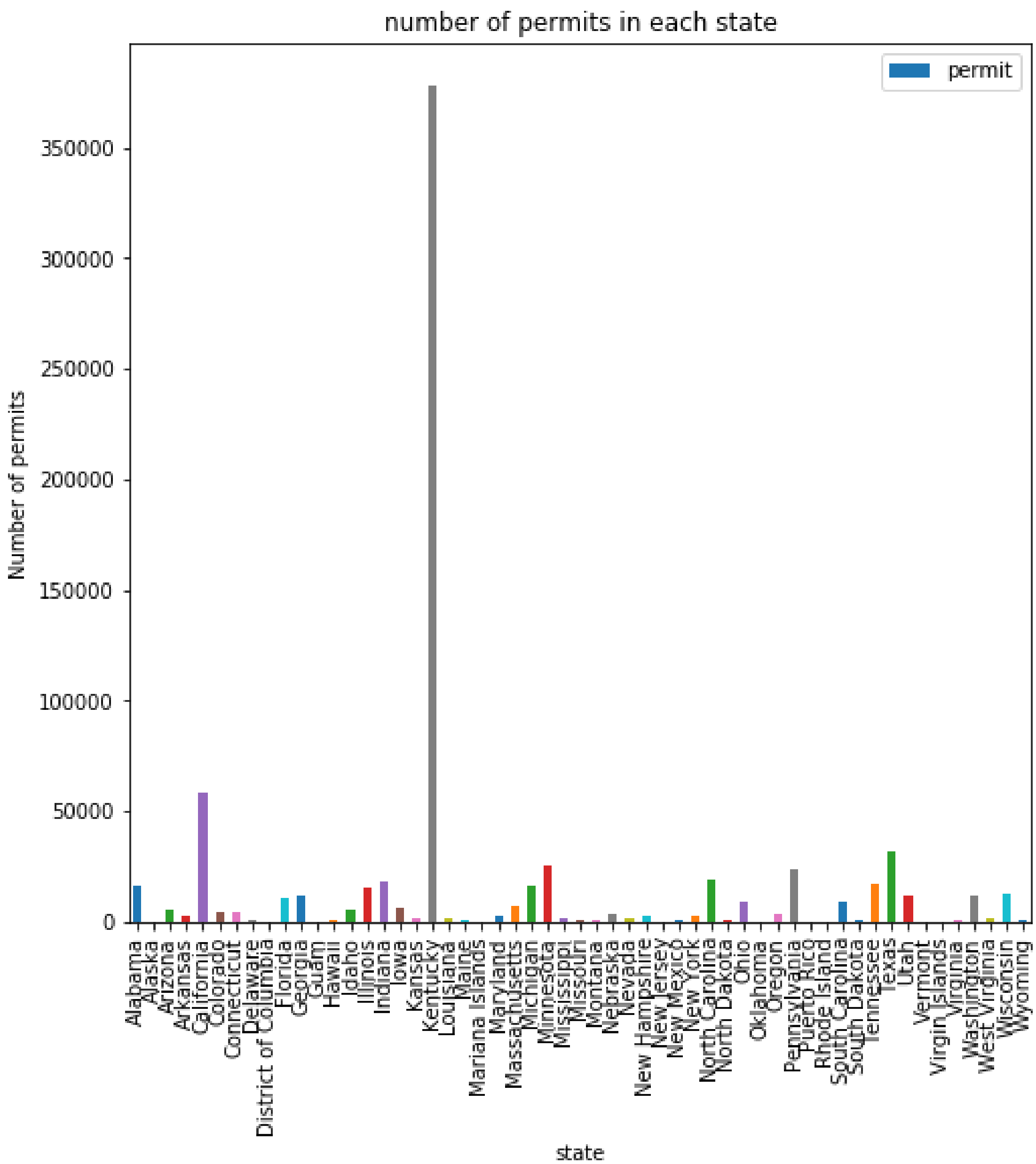
In [35]:
```
# see last date
df_gun.month.max()
```

Out[35]: Timestamp('2017-09-01 00:00:00')

In [36]: # filter data by last date
```
df_permit = df_gun.query('month == "2017-09"')[['permit','permit_recheck','state']]
```

In [37]: # visualize result
```
permit_plot = df_permit.plot(x='state', y='permit', kind='bar',figsize=(8,8),title='num
permit_plot.set_ylabel('Number of permits')
```

Out[37]: Text(0,0.5,'Number of permits')



In [38]: # how many permitted in all state except Kentucky state
```
df_permit.permit.sum() - df_permit.permit.max()
```

Out[38]: 383238.0

Kentucky state has high permit.

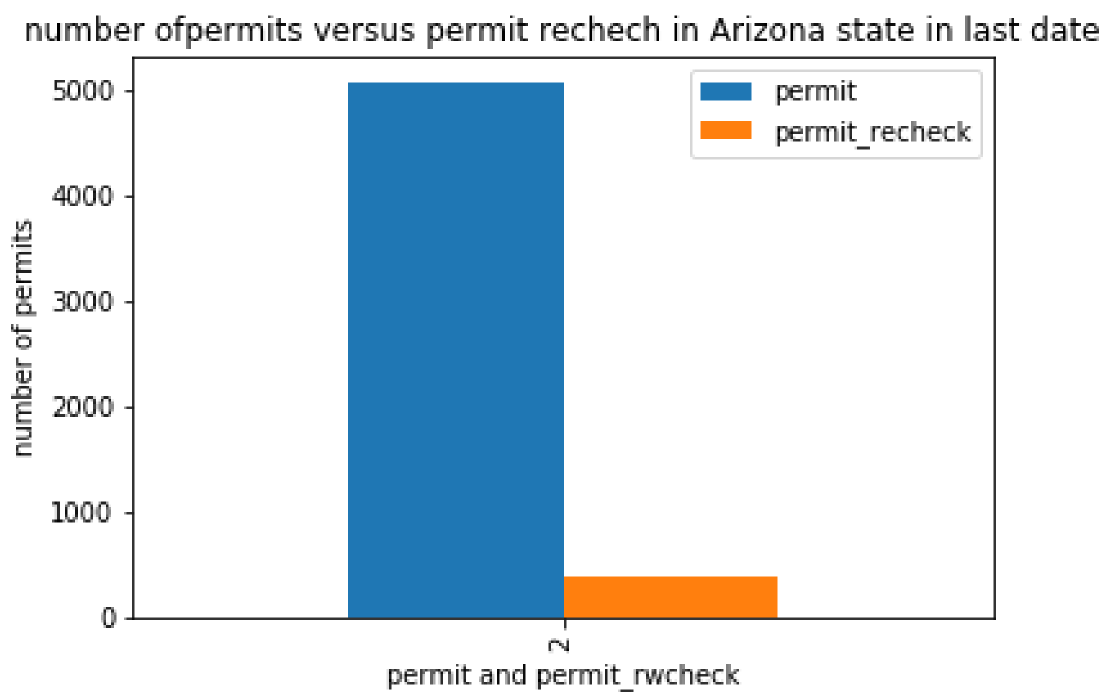
persons permitted to own gun in (Kentucky state) equal to approximately person permitted to own gun in (ALL) other states

Research Question 4 (How many persons are permitted versus not permitted in (Arizona state) in last date ?)

In [39]:
```
# filter data by Arizona state
Arizona_filt = df_permit.query('state == "Arizona"')
```

In [40]:
```
# visualize result
Arizona_permit_plot = Arizona_filt.plot(kind='bar', title='number ofpermits versus perm
Arizona_permit_plot.set_ylabel('number of permits')
Arizona_permit_plot.set_xlabel('permit and permit_rwcheck')
```

Out[40]: Text(0.5,0,'permit and permit_rwcheck')



In [41]:
```
Arizona_filt.sum()
```

Out[41]:
```
permit                5069
permit_recheck         382
state              Arizona
dtype: object
```

total permit in Arizona in 09-2017: 5069

total permut_recheck in Arizona in 09-2017: 382

Research Question 5 (quantity types of guns)

In [42]:
```
# select type of guns and sum
df_gun_type = df_gun.iloc[:,4:26].sum()
df_gun_type
```

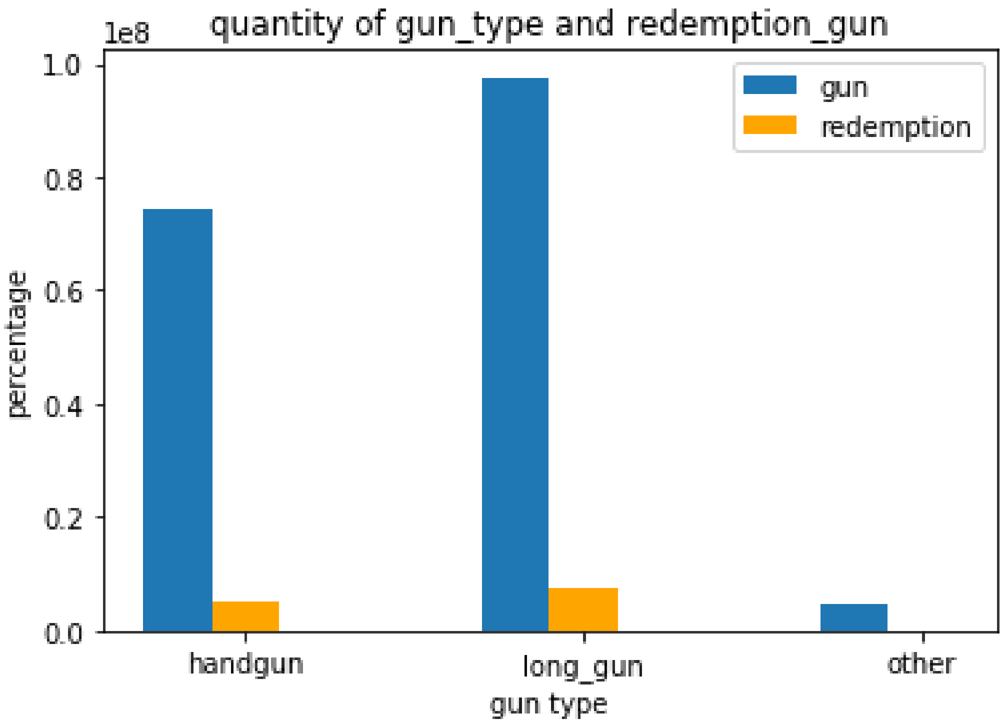
Out[42]:
```
handgun                      7.417190e+07
long_gun                     9.751843e+07
other                        4.500488e+06
multiple                     3.353513e+06
admin                        7.353427e+05
prepawn_handgun              6.027785e+04
prepawn_long_gun             9.780943e+04
prepawn_other                2.067409e+03
redemption_handgun           5.093511e+06
redemption_long_gun          7.482665e+06
redemption_other             2.266339e+04
returned_handgun             3.697149e+05
returned_long_gun            9.435633e+04
returned_other               1.282894e+04
rentals_handgun              9.584444e+02
rentals_long_gun             1.089600e+03
private_sale_handgun         1.864760e+05
private_sale_long_gun        1.448623e+05
private_sale_other           1.286182e+04
return_to_seller_handgun     5.019222e+03
return_to_seller_long_gun    5.516100e+03
return_to_seller_other       1.323244e+03
dtype: float64
```

In [43]:
```
# index to use it in visualization using matplotlib
# labels to use it in visualization using matplotlin
# function return selected rwos of type

ind=np.arange(3)
labels=['handgun','long_gun','other']
def return_rwo(a,b):
    return df_gun_type.iloc[a:b]
```

In [44]:
```
# visualize guns with redemption guns alone because they have high quantaty

plt.bar(ind,return_rwo(0,3),width=0.2,label='gun')
plt.bar(ind+0.2,return_rwo(8,11),color='orange',width=0.2,label='redemption')
plt.legend()
plt.xticks(ind+0.2,labels);
```

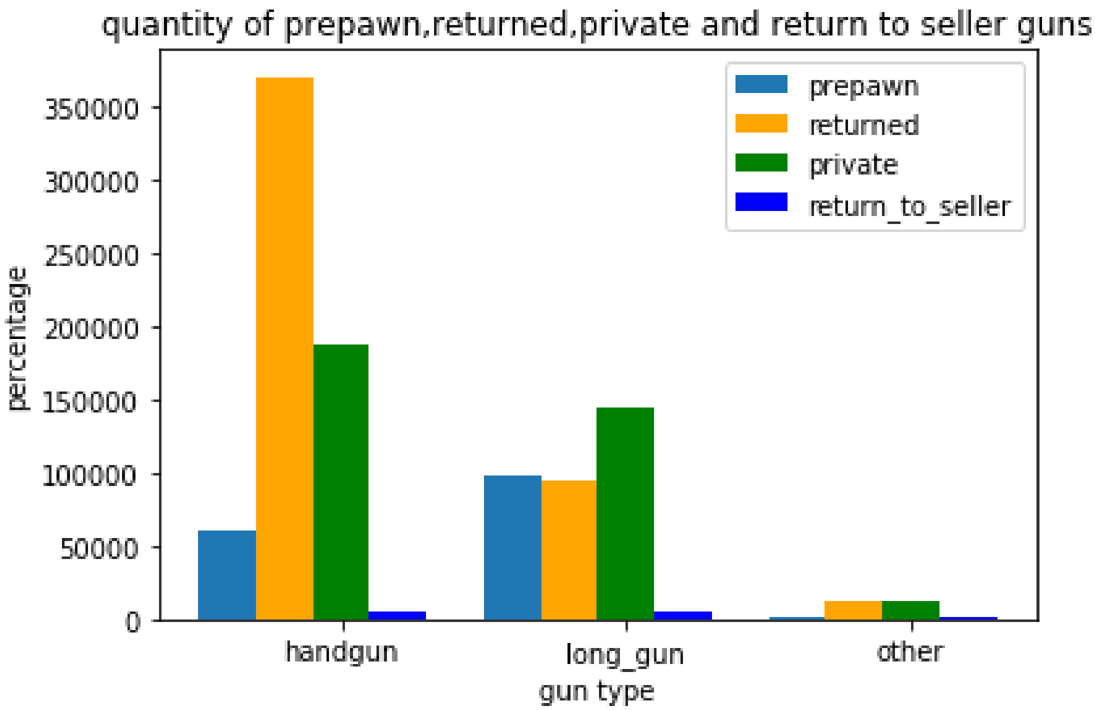
```
plt.xlabel('gun type')
plt.ylabel('percentage')
plt.title('quantity of gun_type and redemption_gun')
```

Out[44]: Text(0.5,1,'quantity of gun_type and redemption_gun')



In [45]:
```
# visualize other types

plt.bar(ind,return_rwo(5,8),width=0.2,label='prepawn')
#plt.bar(ind+0.2,return_rwo(8,11),width=0.2,color='red')
plt.bar(ind+0.2,return_rwo(11,14),width=0.2,color='orange',label='returned')
#plt.bar(np.arange(2)+0.6,return_rwo(14,16),width=0.2,color='pink')
plt.bar(ind+0.4,return_rwo(16,19),width=0.2,color='green',label='private')
plt.bar(ind+0.6,return_rwo(19,22),width=0.2,color='blue',label='return_to_seller')

plt.legend()

plt.xticks(ind+0.4,labels);
plt.xlabel('gun type')
plt.ylabel('percentage')
plt.title('quantity of prepawn,returned,private and return to seller guns')
```

Out[45]: Text(0.5,1,'quantity of prepawn,returned,private and return to seller guns')



long_gun have maximum quantity in (gun_type, redemption, prepawn)

hand_gun have maximum quantity in (returned, private, return_to_seller)

Research Question 6 (In which state did men owned firms by maximum number of gun in 2012?)

In [46]:
```
# select rwo 56 that contain data who men_owned firms(2012)
# max_: max number of gun
# state: list contain states have the maximum number of gun
# for loop: loop for compare column number to maximum number and store column name havi

df_men_owned_firms = df_census.iloc[56:57,2:]

max_ = df_men_owned_firms.max(axis=1)
state=[]
for column in df_men_owned_firms.columns:
    if (df_men_owned_firms[column] == max_).any():
        max_ = df_men_owned_firms[column]
        state.append(column)
state,max_
```

Out[46]:
```
(['California'], 56    1852580.0
 Name: California, dtype: float64)
```

maximum number of gund that men owned firms in 2012 owned in Nebraska state

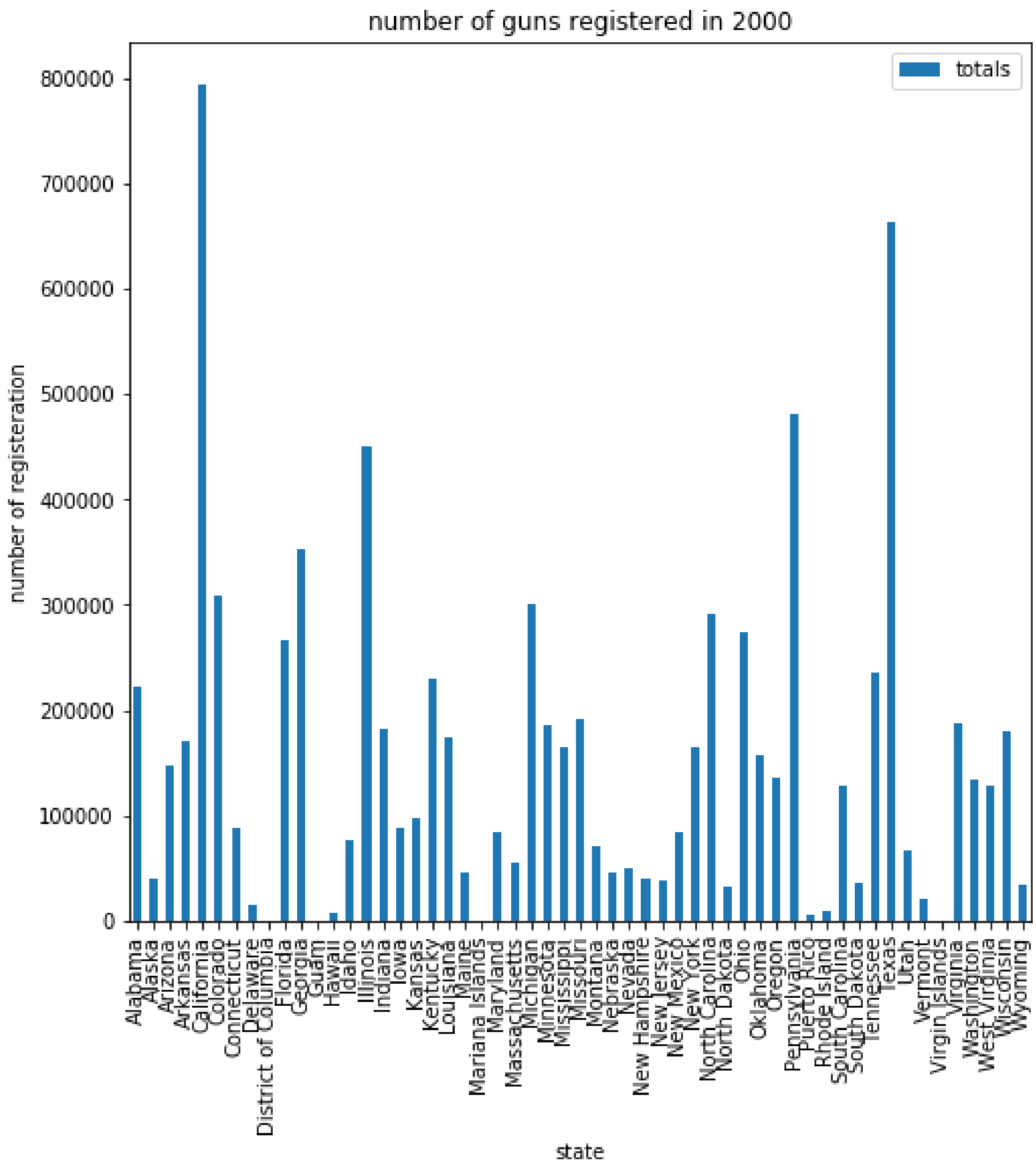
Research Question 7 (How many guns registered in 2017 in all states?)

```
In [47]: # copy df_gun
         # append year column in new dataframe from month column
         # filter data by year 2017 and select only totals registeration
         # calculate sum of totals in 2017 dataframe

         df_copy = df_gun.copy()
         df_copy['year'] = df_copy['month'].dt.year
         df_copy.query('year == "2017"')['totals'].sum()

Out[47]: 17990528
```

there are 17990528 guns regustered in all state in 2017

Research Question 8 (How many guns registered in 2000 in each state?)

```
In [48]: # filter data by year 2000 and select only state and totals columns
         # grouped data by state
         # visualize result

         filter_2000 = df_copy.query('year == "2000"')[['state','totals']]
         group = filter_2000.groupby('state').sum()
         filter_2000_plot = group.plot(kind='bar',figsize=(8,8), title='number of guns registere
         filter_2000_plot.set_ylabel('number of registeration')

Out[48]: Text(0,0.5,'number of registeration')
```




```
In [49]: group.max()
```

```
Out[49]: totals    794506
         dtype: int64
```

California has the highest number of guns registration in 2000

Research Question 9 (Average guns registered in each state since 1998)

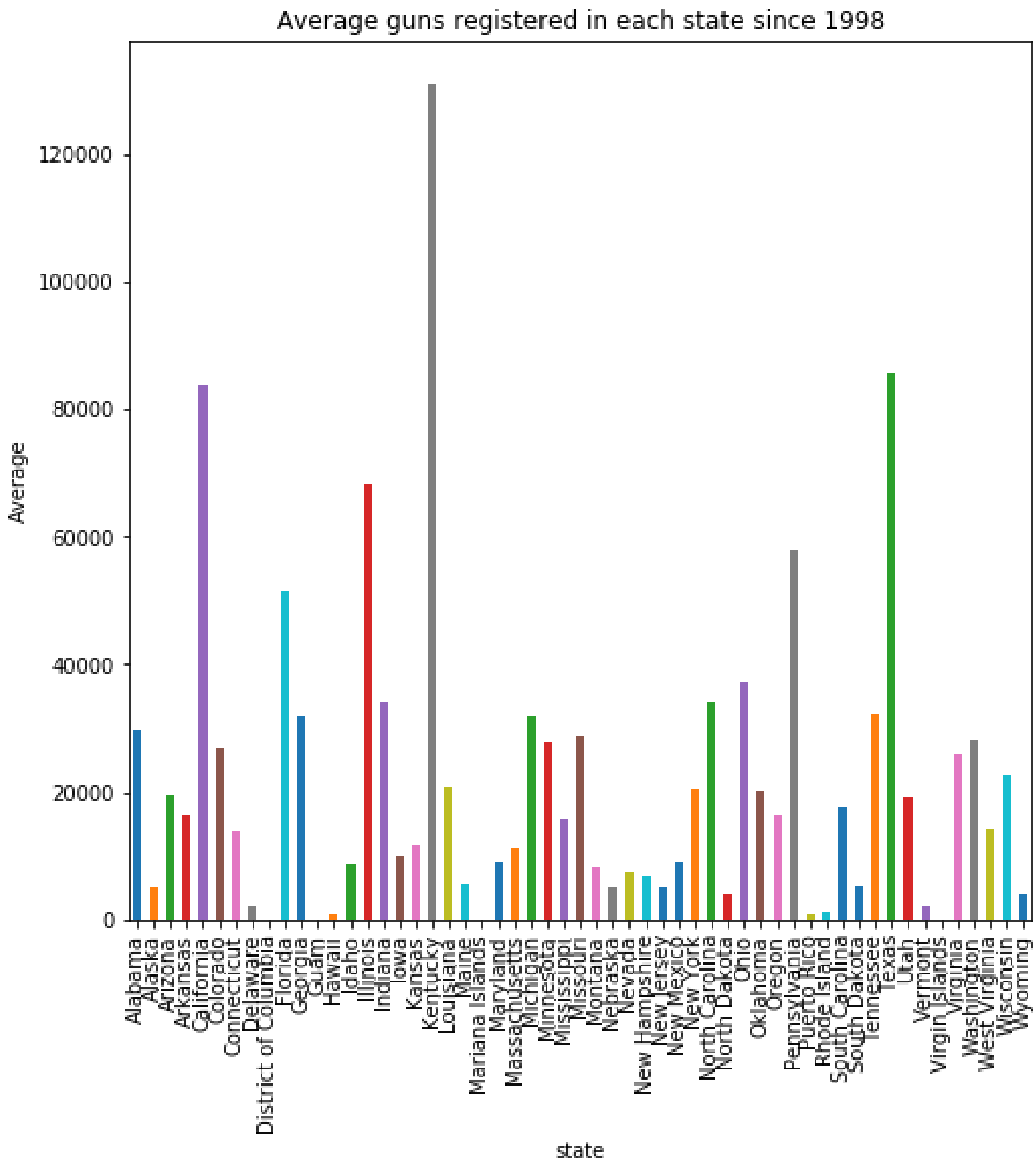
```
In [50]: # filter data by state and select only totals column
         # claculate mean in totals column in all date
```

```
total_filter = df_gun.groupby('state')['totals'].mean()
total_filter.mean()
```

Out[50]: 21595.725911093308

In [51]:
```
# visualize result
total_filter_plot = total_filter.plot(kind='bar',figsize=(8,8), title='Average guns reg
total_filter_plot.set_ylabel('Average')
```

Out[51]: Text(0,0.5,'Average')



Kentucky has the highest average of gun registration since 1998

California and Texas have the same average of gun registration

Research Question 10 (How many guns registered in 2017 in Texas)?

```
In [52]: # filter data by year 2017 and Texas state and select only totals column
         # calculate sum of totals

         df_copy.query('year=="2017" and state=="Texas"')['totals'].sum()
```

```
Out[52]: 1074971
```

there are 1074971 gun registrations in Texas in 2017

Research Question 11 (What is the total revenue of all firms in all state in 2012?)

```
In [53]: # select row 55 in df_census that contain revenue of firms in 2012
         # calculate sum of revenue

         df_census.iloc[55,2:].sum()
```

```
Out[53]: 27744592.0
```

27744592 is the total revenue of all firms in all states in 2012

Conclusions

from the previous analyzes we draw some important conclusions in several aspects:

growth in gun registration

> 1- the highest growth rate in gun registration in KENTUCKY state (growth = 397899) from 1998 to 2017.
> 2- the lowest growth rate in VIRGIN state (growth = 9).
> 3- growth rate in CALIFORNIA,ILINOIS and TEXAS veryvclise and ranks third of growth.

registration

> 1- there are 17990528 total guns registered in all states in 2017.
> 2- CALIFORNIA has the highest number of guns registration in 2000.
> 3- KENTUCKY has the highest average of gun registration since 1998.
> 4- CALIFORNIA and TEXAS have the same average of gun registration.
> 5- there are 1074971 gun registered in TEXAS in 2017.

overall trend

> overall trend of gun purchases is constantly increasing and qickly.
> **permits**
> 1- KENTUCKY state has the highest number ofpermits.
> 2- number of permits in KENTUCKY is equal approximately number of permits in all other states.
> 3- number of permits in ARIZONA in 09-2017 is 5069.
> 4- number of permits recheck in ARIZONA in 09-2017 is 382.

quantities of guns types

> 1- long_gun has the highest quantity in (gun type,redemption,and prepawn)
> 2- hand_gun has the highest quantity in (returned,private, and return_to_seller) > **other**
> 1- the highest number of guns that men owned firms buy in 2017was in NEBRASCA state.
> 2- the total revenue of all firms in all states in 2012 equal 27744592.

1.1.4 Limitations

1- in census data, all columns contain missing values, and the majority of columns contain 20 missing value out of 85 rows. meaning that approximately 25% of the data are missing and this is an exaggeration.

2- there are 154595 missing in the gun data, this represents about 45% of the total values, this is avery large number even if we try to complete it using appropriate methods, the result will still be somewhat inaccurate.

3- in gun data, there are months of specific years that have no been recorded, and there also states that have no been recorded.

4- the numeric values in census data separated by a comma, this makes converting them into float more difficult, and when doing this in the best way, there are completely inaccurate values and this changes the accuracy of the result.

In [54]:
```
from subprocess import call
call(['python', '-m', 'nbconvert', 'Investigate_a_Dataset.ipynb'])
```

Out[54]: 255