Day Objectives

1st June 2019

- Practice on Income Data set
  - Define functions for the following data points
    - Average Income of all states from 2005 to 2013
    - State with highest average income in the last three years
    - State with lowest average income from 2007 to 2010(inclusive)
    - Print the list of all states in the same line with average income less than California
    - Print the names of states based on descending order of income in the year 2009
    - State with the lowest recorded income from 2005 to 2013

Average Income of all states from 2005 to 2013

```
In [11]:   # Fucntion to find the Average income of all states from 2005 to 2013
           def avg(df):
               s=0
               c=0
               for i in range(len(df.values)):
                   for j in range(2,len(df.columns)):
                       s=s+df.values[i][j]
                       c=c+1
               print(s//c)
           avg(incomedf)
```

48525

```
In [ ]:
```

Print the names of states based on descending order of income in the year 2009

```
In [12]:   # Print the names of states based on descending order of income in the year 2009
           def desc_2009(df):
               li=[]
               u=[]
               s=[]
               for i in range(len(df.values)):
                   for j in range(len(df.columns)):
                       a=df.values[i][6]
                       if a not in li:
                           li.append(a)
               u=sorted(li,reverse=True)
               print(u)

           desc_2009(incomedf)
```

```
[61604, 56134, 45739, 39980, 36538]
```

```
In [13]:   # Print the names of states based on descending order of income in the year 2009
           def desc_2009(df):
               li={}
               u=[]
               for i in range(len(df.values)):
                   for j in range(6,len(df.columns)-4):
                       li[df.values[i][1]]=df.values[i][j]
               u=sorted(li.values(),reverse=True)
               for i in u:
                   for item in li.items():
                       if item[1]==i:
                           print(item[0],':',i)
               return

           desc_2009(incomedf)
```

```
Alaska : 61604
California : 56134
Arizona : 45739
Alabama : 39980
Arkansas : 36538
```

State with lowest average income from 2007 to 2010(inclusive)

```
In [14]:   # State with lowest average income from 2007 to 2010(inclusive)
           def LowestavgIncome(df):
               li={}
               u=[]
               for i in range(len(df.values)):
                   s=0
                   c=0
                   for j in range(4,len(df.columns)-3):
                       s=s+df.values[i][j]
                       c=c+1
                   li[df.values[i][1]]=s//c
               print(li)
               print("\n")
               lis=min(li.values())
               for item in li.items():
                   if lis==item[1]:
                       print(item[0],':',lis)
           LowestavgIncome(incomedf)
```

{'Alabama': 41900, 'Alaska': 61608, 'Arizona': 46691, 'Arkansas': 38876, 'Calif
ornia': 55791}

Arkansas : 38876

State with highest average income in the last three years

```
In [15]:   # State with highest average income in the last three years

           def HighestavgIncome(df):
               li={}
               u=[]
               for i in range(len(df.values)):
                   s=0
                   c=0
                   for j in range(8,len(df.columns)):
                       s=s+df.values[i][j]
                       c=c+1
                   li[df.values[i][1]]=s//c
               print(li)
               print("\n")
               lis=max(li.values())
               for item in li.items():
                   if lis==item[1]:
                       print(item[0],':',lis)
           HighestavgIncome(incomedf)
```

{'Alabama': 42478, 'Alaska': 60738, 'Arizona': 48755, 'Arkansas': 40079, 'Calif
ornia': 55971}

Alaska : 60738

State with the lowest recorded income from 2005 to 2013

In [16]:
```python
# State with the lowest recorded income from 2005 to 2013


def LowestavgIncome2005_2013(df):
    u=[]
    for i in range(len(df.values)):

        for j in range(2,len(df.columns)):
            u.append(df.values[i][j])
    a=min(u)
    for i in range(len(df.values)):
        for j in range(2,len(df.columns)):
            if df.values[i][j]==a:
                b=df.values[i][1]
    print(b,':',a)
LowestavgIncome2005_2013(incomedf)
```

```
Arkansas : 36538
```

Print the list of all states in the same line with average income less than California

In [17]:
```python
# Fucntion to print the list of all the states in the same line with average inc
def AvgLessCalifornia(df):
    li={}
    for i in range(len(df.values)):
        s=0
        c=0
        for j in range(2,len(df.columns)):
            s=s+df.values[i][j]
            c=c+1
        li[df.values[i][1]]=s//c
    print(li)
    print('\n')
    for item in li.items():
        if item[1]<55350:
            print(item[0],':',item[1])

AvgLessCalifornia(incomedf)
```

```
{'Alabama': 41126, 'Alaska': 60106, 'Arizona': 47214, 'Arkansas': 38828, 'Calif
ornia': 55350}


Alabama : 41126
Arizona : 47214
Arkansas : 38828
```

In [ ]:

Read CSV File

In [18]:
```python
# Function to read csv data into a Data Frame
# retruns the DataFrame Object

import pandas as pd
    # comma seprated values all spreads are csv files
def readCSVdata(filepath):
    return pd.read_csv(filepath)
filepath='DataFiles\income.csv'
readCSVdata(filepath)
```

Out[18]:

	GEOID	State	2005	2006	2007	2008	2009	2010	2011	2012	2013
0	04000US01	Alabama	37150	37952	42212	44476	39980	40933	42590	43464	41381
1	04000US02	Alaska	55891	56418	62993	63989	61604	57848	57431	63648	61137
2	04000US04	Arizona	45245	46657	47215	46914	45739	46896	48621	47044	50602
3	04000US05	Arkansas	36658	37057	40795	39586	36538	38587	41302	39018	39919
4	04000US06	California	51755	55319	55734	57014	56134	54283	53367	57020	57528

In [ ]:

Print Data Frame Columns

In [19]:
```python
incomedf=readCSVdata(filepath)
# Fucntion to print all columns names in a single line

# GEOID State 2005 2006 2007 2008 2009 2010 2011 2012 2013

def printDataFrameColumns(df):
    columns=df.columns
    for column in columns:
        print(column,end=" ")
    return
printDataFrameColumns(incomedf)
```

```
GEOID State 2005 2006 2007 2008 2009 2010 2011 2012 2013
```

In [ ]:

Access data frame Row

In [20]:
```python
# Function to access a row based on a unique
def accessDataFrameRow(df,key):
    for row in df.values:
        if key in row:
            for item in row:
                print(item,end=" ")
    return
accessDataFrameRow(incomedf,'Alaska')
```

04000US02 Alaska 55891 56418 62993 63989 61604 57848 57431 63648 61137

In [ ]: