Lab 3

In this lab, we will perform analysis on data in several CSV files.

The Scenario:

You are working on a research project that is investigating changes in population within the United States. You have recently read the Forbes article "The States People Are Fleeing" (by Susan Adams, 1/13/2015) and are wondering the extent to which these trends are reflected in population data. You have found some data from the Census Bureau website, which you have downloaded a single zip file (`popdata.zip`). In this file is a folder containing a comma separated value (CSV) file for each state and the District of Columbia (D.C.). The file `SUB-EST2013.pdf` describes the fields within each file.

NOTE: For some of you, the file might unzip with a .csv.txt extension. `pd.read_csv('Alabama.csv.txt')` will still read the file like normal. Also, some of you may need to use the extension `pd.read_csv('Alabama.csv.txt',encoding='iso-8859-1')`

In your analysis, you are interested in changes in population between 2010 and 2013. Specifically, we define the following:

- the "difference": `diff = POPESTIMATE2013 - POPESTIMATE2010`
- the "percent difference": `diffpct = diff / POPESTIMATE2010`

You need to be able to answer the following questions:

1. For each state (and D.C.), what was the change in population from 2010 to 2013? (note: state population is given by (geographic summary level or SUMLEV code 040)
2. Considering all states (and D.C.), and for a specified integer value k,
   - Which k states (including D.C.) had the greatest *increase* in population? List them in order (e.g., 1,2,... k).
   - Which ones had the greatest *decrease* in population? List them in order (e.g., 1,2,... k).
   - Which k ones had the greatest *percent increase* in population? List them in order (e.g., 1,2,... k).
   - Which ones had the greatest *percent decrease* in population? List them in order (e.g., 1,2,... k).
3. For each state (including D.C.), consider county-level information given by geographic summary level or SUMLEV code 050)
   - Which k counties experienced the greatest *increase* in population during this period? List them in order (e.g., 1,2,... k).
   - Which k counties experienced the greatest *decrease* in population during this period? List them in order (e.g., 1,2,... k).
   - Which k counties experienced the greatest *percent increase* in population during this period? List them in order (e.g., 1,2,... k).
   - Which k counties experienced the greatest *percent decrease* in population during this period? List them in order (e.g., 1,2,... k).

Note: For the sake of sorting, you can consider *negative* increases (or negative decreases as well) in the top k elements if less than k elements experience an actual increase (or decrease).

Working with your research colleagues, you have decided to write a simple Python program that will help you answer these questions.

Your Task:

Your task is to write a Python program named `CalcPop.py` (please use that exact name!) that takes two command line arguments, representing (1) the path to the folder containing all of the CSV files (i.e., `popdata`), and (2) the parameter k. Your program should open each of the CSV files in the specified folder, read its data, and calculate answers to the above questions.

For example, for the state of Alabama one obtains the following results when $k = 3$: (the format of the printout need not match this exactly)

```
 Alabama
Total population change:  48152

 top 3 in growth for Alabama
                diff
NAME
Baldwin County  12317
Madison County  10688
Lee County      10083

 bottom 3 growth for Alabama
                  diff
NAME
Montgomery County -3146
Macon County      -1853
Dallas County     -1844

 top 3 in percent growth for Alabama
                  diffpct
NAME
Russell County    0.119051
Lee County        0.071587
Limestone County  0.067502

 bottom 3 in percent growth for Alabama
                diffpct
NAME
Macon County   -0.086022
Coosa County   -0.057511
Lowndes County -0.050984
```

Notes:

- A skeleton of the program `CalcPop.py` has been provided for you. Use it!
- You can report state-level results either before or after the state-by-state results.
- You do not need to consider changes in population for cities, towns, unincorporated areas, or other smaller jurisdictional areas.
- The fact that D.C. is represented as having only a single county should not break your program.
- Note: the command line argument specifying the path to the `popdata` folder could be either an absolute path or a relative path. Your program should work for either case.
- **Your program should work dynamically with the CSV files available in the specified folder. That is, the list of files to read should be generated dynamically from the contents of the specified folder. That way, if a file for a given state is missing, your program should simply skip it. Do not hardcode the list of names for states or D.C..**
- Have your program print its output to standard out, so that it can be redirected into a text file.
- There is no strict specification for output format, but make the text readable please.

Hints:

- The `calc_pop` function within `CalcPop.py` takes argument `dirname` . This is input by the user at the command line and specifies the path to the `popdata` folder. Within the `calc_pop` function, use the `os` module to change into the directory given by the user in the `dirname` variable. Do not hard code the path to the `popdata` folder.
- Use the `glob` module as illustrated in the shell scripting lecture to find all CSV files in the `popdata` folder once you have used the `os` module to navigate to that folder.
- I suggest you use Pandas to read the csv files and to do all of the calculations. This is a good chance to practice with Pandas, which will be the primary subject of the second exam.

Submit all source code and a text file containing your output for $k = 3$. Please use comments in your code. Expect that the instructors will run your program from the command line in addition to looking at your results.