Project Population

Assignment Overview
(learning objectives)
This assignment will give you more experience on the use of:

1. Functions
2. File input and output
3. try-except

The goal of this project is to analyze population data. Use of advanced data structures such as list, sets, and dictionaries is **prohibited**.

Assignment Background

We take a file of population data that is organized by columns. You will read each line of the file and figure out the maximum increase in population between columns for that continent (that line). You will also find the maximum increase overall, i.e. for all continents.

Here is a file. The first two header lines of the file will always be identical (in particular, the years will be the same). For the data lines the first field (name) is 15 characters wide followed by seven fields where each is 6 characters wide (right-justified, digits only). The field width specification allows you to use slicing to extract values from each line.

```
        Continent Population by Year (millions)
  Continent     1750  1800  1850  1900  1950  2000  2050
   Africa        106   107   111   133   221   767  1766
    Asia         502   635   809   947  1402  3634  5268
  Australia        2     2     2     6    13    30    46
   Europe        163   203   276   408   547   729   628
North America      2     7    26    82   172   307   392
South America     16    24    38    74   167   511   809
```

You will examine successive columns to determine the percentage change from one column to the next. Consider Africa from 1900 to 1950. You use the two values from those years, 133 and 221, to calculate the change:

```
     change = (221-133)/133
```

You need to examine each pair of values for Africa, that is 106 & 107, 107 & 111, 111 & 133, etc. and determine the maximum change for Africa. Display that as a percent. See samples below for display formatting.

Project Description

Your program must meet the following specifications:

1. At program start prompt the user for a file to be analyzed
2. You have to use the five functions in the provided projPopu.py skeleton. You have to implement and use the

a. `open_file()`: Returns the file pointer to the file opened by asking user for the file name. Error checking is required. That is, keep prompting until a valid file is entered. Use `'Error. Please try again.'` for the error message.

b. `print_headers()`: Print the headers for the output. See samples.

c. `calc_delta(line,col)`: Takes one line (string) of data from the data file as an argument along with a column index (int) in the line (where the continent's name is "column 0", the first data value is in "column 1", etc.).
You will examine successive columns to determine the percentage change from one column to the next. Consider Africa from 1900 to 1950. You use the two values from those years, 133 and 221, to calculate the change:

   change = (221-133)/133

   To extract the two values, 133 and 221, you need to construct a slice whose values are a function of the column number. You will use the fact that the continent name is in a 15-character wide field and that each value is in a 6-character wide field.
   Return change as a float.
   (Hint: I developed the slice values by writing a short program that had a string that was one line of the file and then I looped through the string using slicing to print the pairs of values, e.g. for Africa I printed 106 & 107, 107 & 111, 111 & 133, etc. Once I could do that, I knew that I could slice out the desired values and then the rest was easy.)

d. `format_display_line (continent,year,change)`: Takes as arguments the continent name (string), year (int), and change (float) and returns a formatted string. The continent name will be in a field of 26 characters left justified, the year and its predecessor (year-50) are printed with a hyphen between in a field width of 9 and no spaces on either side of the hyphen, the change will be displayed as a percent with no fractional values including the percent sign in a field width of 10. See samples below. For example, if the parameters are (`'Antarctica', 2018, 0.12567`), the string returned will be:

   ```
   Antarctica                1968-2018       13%
   ```

e. `main()`: Takes no input. Returns nothing. Call the functions from here. Close the file here.

Assignment Notes

1) To clarify the project specifications, sample output is appended to the end of this document.
2) The algorithm for finding a maximum is as follows (finding minimum is similar):
   a) Initialize the maximum to be a small number, e.g. zero. This is usually done before a loop.
   b) If a new value is greater than the existing maximum, assign maximum to be the new value. Also, you may want to update associated values, e.g the year of the new maximum.
3) We provide a projPopu.py program for you to start with.
4) Use of advanced data structures such as list, sets, and dictionaries is prohibited.
5) If you "hard code" answers, you will receive a grade of zero for the whole project. An example of hard coding is to simply print an average rather than calculating an average and then printing the calculated average.

Test Cases

Function Test calc_delta
(calculates change for all six pairs in the input line.)

```
Input line:   Antarctica      160   213   275   418   537   727   625
change:    0.33125
change:    0.29107981220657275
change:    0.52
change:    0.284688995215311
change:    0.3538175046554935
change:    -0.14030261348005502
```

Function Test format_display_line

```
continent,max_delta_year,max_delta = 'Antarctica', 2018, .12567
```

returns

```
Antarctica                1968-2018       13%
```

Test 1 (data.txt)

```
Enter a file name: data.txt

     Maximum Population Change by Continent

Continent                     Years     Delta
Africa                    1950-2000      247%
Asia                      1950-2000      159%
Australia                 1850-1900      200%
Europe                    1850-1900       48%
North America             1800-1850      271%
South America             1950-2000      206%

Maximum of all continents:
North America             1800-1850      271%
```