Lab 1
(Based on lab by Katherine St. John at https://stjohn.github.io/teaching/cmp108/s17/lab6.html)

Topics: Reading CSV files and basic plotting using Pandas

Commands introduced:
`df = pd.read_csv("filename.txt")` : opens and reads the CSV file `filename.txt` into the Pandas dataframe `df`
Options: `skiprows = 3` : doesn't include the first 3 rows
`index_col = ['column_name']`: uses the values in the column called column_name as the row (index) name
`transposed_df = df.T` : creates a new dataframe that transposes (swaps rows and columns) `df`

`df.plot()` : default plot of dataframe `df`
Options: `title = "title for plot"` : adds title to plot

Motivating Question: How has the population of the New York boroughs changed over the years?

Part A

Get the Data

We will first use a slightly easier dataset (originally created by Prof. Katherine St. John from Wikipedia data) than the NYC Open Data one. Download it here: nycHistPop.csv

1) nycHistPop.csv is a CSV (comma-separated values) file, which stores data from table, using commas to distinguish between the columns. Open nycHistPop.csv in both Excel and TextEdit, and compare the two views of the file. *How do you think any empty cell in the middle of the table would be represented in a CSV file? You can create a new Excel spreadsheet with this situation, and save it as a CSV file to check your guess.*

2) Open a new file in IDLE. In the file, type:

```
import matplotlib.pyplot as plt
import pandas as pd
```

These commands load the plotting (Matplotlib) and data science (Pandas) libraries that we will use in this program into the computer's working memory. We can refer to Matplotlib using `plt` and Pandas using `pd` in our program.

3) Save your file as lab1a.py and make a note of what directory/folder you save it in.

4) Move nycHistPop.csv into the same directory as lab1a.py.

5) First we want to link the file with a variable called `pop` in our Python program. This also opens the file for our program to use. Do this by typing:

```
pop = pd.read_csv('nycHistPop.csv',skiprows=5)
```

The command `pd.read_csv("filename.txt")` will work for any CSV file, where you put the name of the file in quotes in between the parentheses. This assumes the CSV file is in the same directory as your program. `skiprows=5` means the first 5 rows of the file (which are not part of the table) are not included.

6) Try running your program. *What happens? What happens if you now type* `pop` *or* `print(pop)` *in the shell?*

Plot the Data

7) On the next line in lab1a.py, create and show the plot of the data by typing:

```
pop.plot(x = "Year")
plt.show()
```

8) Run your program. *What happens?* You should see a plot of the data. You can save the plot by clicking the save button on the bottom of the pop up window.

Challenges:

- What happens if you leave off the `x = "Year"`? Why?
- What happens if you add in `x = "Year", y = "Bronx"`?

Part B

We will repeat the above using the NYC Open Data borough population set. This data set is structured differently, so we will have to transpose it (switch the rows and columns) to plot it.

Get the Data

1) Go to the main page for the data here. The data set is called New York City Population by Borough, 1950-2040. This page gives information about the data set, including when it was uploaded and a summary of the data.

2) Download a CSV file of some of the data by:

a) In the upper right corner, click on the button "View Data". This leads to a page that looks somewhat similar to an Excel spreadsheet.

b) Click on the "Manage" button in the upper right. Then click on Show & Hide Columns. Unselect (click the check mark) the following columns:

i)Age Group

ii) year - Boro share of NYC total for all years

Click the "Apply" button at the bottom when you are finished. *What happened?*

6) We have now cleaned our data, and can download it. Click the “Export” button in the upper right. Under “Download As”, click “CSV”.

7) There should now be a file called “New_York_City_Population_by_Borough__1950_-_2040.csv” in the “Downloads” folder. Rename it to be called “nycPopOpen.csv” . How is this file different from `nycHistPop.csv` from Part a?

Explore the Data

1) As in part A, create a new Python file called lab1b.py, and import the matplotlib and Pandas libraries. Move the new .csv file into the same directory as your Python file.

2) This .csv file does not contain extra lines at the top, but it has the years as columns and the boroughs as rows. We will need to switch them for the plotting command to work properly, so we need to tell the computer the the Borough names are names for the rows by adding the instruction `index_col = ['Borough']`. Thus, we can read the file into the variable pop2 using the command:
```
pop2 = pd.read_csv("nycPopOpen.csv",index_col = ['Borough'])
```

3) Check that the file was read in correctly by running your program and typing `pop2` or `print(pop2)` in the shell.

4) Next we need to transpose pop2 by adding the following line to our program:
```
pop2T = pop2.T
```
This stores the transposed dataframe in a new variable `pop2T`.

5) Run lab1b.py, and typing pop2 and pop2T at the shell. *What’s the difference? Is it what you expect?*

6) As in part A, plot pop2T. This time, don’t include `x = 'Year'` (since the year column is not labeled). *Are there more or less changes in the borough populations than in the years 1850-1940 than in the years 1950-2040? (Refer to your graph from part A)*

7) We can add a title to our plot by changing the plotting command to:
```
pop2T.plot(title="NYC Borough Population 1950-2040")
```
Over the next few weeks, we will learn some other techniques for improving the look of our plots.