Command line practice

1. What is the absolute path of your home directory?
2. Are there any other user accounts on your computer? What are they?
3. Do you have Microsoft Office installed? Google Chrome? Identify the absolute path of where these applications are stored. What other programs are stored in this directory?
4. Navigate to your home directory. Create a directory called 'PythonClass' using a relative path. Create a directory in your home directory called 'PythonClass2' using an absolute path. Verify both directories were created using the ls command. Now remove both directories using the rm command
5. Identify all of the directories in your root directory
6. Identify directories that contain directories, text files, and symbolic links. Use ls -la to determine a way to distinguish between these three files.
7. While not covered in Lecture, grep is an extremely useful command for searching for text strings. On Canvas, I have included an example file that you can search through called Demographics.csv. This file lists demographic information by county. This information could be useful to predict how many people are expected to be hospitalized if Covid-19 spreads through a county. We know that Covid-19 has a strong age-dependence in determining the likelihood that someone is hospitalized or dies from it if they are infected.
    a. Read the manual page in order to learn how to use grep. Use grep in order to search for the demographics specific to Fulton County, where we are located now. Make sure you only print out information for Fulton County from Georgia.
    b. The default behavior of grep is to print out information to standard out (which shows on the screen). If you want to redirect this output to a file, it is very easy to using ">" after the command and then supplying the name of the file that you want to print to. Create a file that just contains the information for Fulton County and a file that just contains the information for the county you were born in.
    c. wc is a command that you can use to count the number of characters, words, or lines in a file. Use this command to count how many of each are in the Demographics.csv file along with the two custom files you created. Identify flags that you can send to wc to restrict the output to just characters, words, or lines.
    d. Identify a flag using grep to identify all of the ages/counties that do not have a person alive (i.e. the last column = 0). Create a new file where these 0 lines are filtered out of the file.
8. Piping: piping is a method to combine two programs. You can take the output of one program and send it into the input of a second program. For example, if you want to combine grep and wc you can do it like so:

```
$ grep ",South Carolina," Demographics.csv | grep "America,0,"
```

This command first searches the Demographics.csv file for all the data from counties in South Carolina. It then send the output of the first command to grep to further filter it for the number of individuals in each county that are between 0-1 year old. The "|" character is what tells zsh to do this. This way we have a single line for each county.

    a. Use a pipe to create three commands. 1) How many symbolic links are in a given directory? 2. How many subdirectories are in a given directory? 3. How many files are in a given directory? Use the answers from 6 to help

b. Identify a command that lists the number of counties that are in the state of South Carolina (This can be done using 2 separate pipes). There should be 46 counties. Does your number agree? If not what extra data is included or what counties are missing?
c. Based upon your answer in 8b, identify the demographics information for the state of Georgia.

9. pico is a command that you can use as a text editor. It takes the name of the file you want to use as an argument. If the file exists, it opens the file up for editing. If it doesn't exist, it creates a brand new file.

    Some useful commands to interact with pico can be found here: http://www.diochnos.com/tips/terminal/pico_tutorial.pdf

    a. Create a file called temp.txt using pico that contains the line: "This is my first text file". Save the file and exit the editor. Use more to verify the file saved correctly.
    b. Open the Demographics.csv file in the pico editor. Play around with some of the pico commands to see how they work.

10. Use echo to print out the values of some of your environmental variables listed here:

https://geek-university.com/linux/common-environment-variables/