Brooke Russell

Problem Set #2

1. Assume all of the directories below are valid directories (e.g., they exist and we have access to them). What would the following print out?

cd /usrs/INFO3401/../homework/./problem1  
pwd

* The command “cd” will change the current working directory into the specified directory. The command “pwd” will display the current working directory. So, pwd will print /usrs/INFO3401/../homework/./problem1.

1. What set of commands would you use to achieve the following:  
   You've found yourself in a situation where your python program is trying to parse files in a directory and is throwing an error. You think it might be a permissions issue (i.e., you don't have permission to access the files in that directory) and need to verify the permissions on all files in that directory. The directory is in your home directory under the "datasets/activedata" directory. Make no assumptions about what directory you are currently working in.
   * In order to list the permissions of all the files in the activedata directory, I would first navigate directly there using “cd ~/datasets/activedata” and then use the “ls -l” command to see user permissions on all files in the current directory. You could then use a sudo command to effectively change the permissions on any file.
2. Briefly describe what the following set of commands would achieve. What process would happen and what would be printed to the command line?

cd ~  
mkdir ./problem\_set\_1  
touch submission.txt  
cd ..  
pwd

* “cd ~” will bring us to our home directory. “mkdir.problem\_set\_1” will make a new directory under our home directory called “problem\_set\_1”. “touch submission.txt” will create a blank file named “submission.txt” in our “problem\_set\_1” directory. “cd ..” will then change the current directory to the parent directory. So, “pwd” will print our working directory, which is “problem\_set\_1”.

1. What set of commands would you use to achieve the following:  
   Copy a document called config.txt from your home directory to the root directory. Then, create a new directory in the root called preferences. Within preferences, copy prefs.txt from a matching directory called preferences in your home directory. Finally, determine that the start and end of the config.txt document in both root and home match.
   * To copy a document called config.txt from your home directory to the root directory I would use the command “cp ~ ./config.txt > cd/”
   * To create a new directory in the root called preferences, I would use the command “cd/” then “mkdir preferences”
   * “cd preferences” “cp prefs.txt preferences”
   * “head –n 1 config.txt” “tail –n 1 config.txt”
2. You have accidentally moved a file (~/Documents/datafile.csv) to your current directory rather than copying it. What command did you use to do that? What command should you have used to do that? What command might you use to undo it?
   * I used the command “mv data.file.csv Documents”
   * I should of used the command “cp data.file.csv Documents”
   * The command that I might use to undo it is “rm datafile.csv”
3. Download and unzip [asciify-master.zip](https://canvas.colorado.edu/courses/60694/files/13400065/download?wrap=1) from Canvas. Then, navigate to the directory and use the command line to run asciify.py. What does this script do? Include a screenshot of your results.
   * This script prints the Github mascot formed by characters.
   * 
4. Install the Delorean Python package. What command did you use? Then, from the command line, launch Python. Use it to execute the following commands:

from delorean import Delorean

EST = "US/Eastern"

d = Delorean(timezone=EST)  
print(d)

* I used the command “pip install Delorean”

What prints to the command line?

* Delorean(datetime=datetime.datetime(2020, 2, 10, 22, 45, 20, 322772), timezone='US/Eastern')

1. We can scrape webpage content or download collections of data files using wget. NASA, the USDA, and the NCBI all recommend using wget to download data from their repositories because it is faster and allows you to download data in bulk. We'll experiment with that on a smaller scale: use the command line to download data on the World Bank's budgetary expenditures at [https://finances.worldbank.org/api/views/yu93-ayrw/rows.csv?accessType=DOWNLOAD (Links to an external site.)](https://finances.worldbank.org/api/views/yu93-ayrw/rows.csv?accessType=DOWNLOAD) [Links to an external site.](https://finances.worldbank.org/api/views/yu93-ayrw/rows.csv?accessType=DOWNLOAD). What command did you use? Make sure to include this file as part of your submission.
   * I used the command “wget <https://finances.worldbank.org/api/views/yu93-ayrw/rows.csv?accessType=DOWNLOAD>”
   * File: attach file from desktop
2. The above budgetary data contains both individual items and aggregated totals across different sectors. Let's distill the data down to different aggregated totals. In this dataset, you can use the keyword "Total" to identify only those rows containing aggregate budget items. Use the command line to identify these rows. What command did you use? How many rows did this find (hint: the -c option can be helpful here)?
   * I used the command “cat rows.csv?accessType=DOWNLOAD.17 | grep Total –c”
   * I found 33 rows
3. Now, print the lines containing the word "Total" to a file to "distilledExpenditures.csv". What command did you use? Include this file as part of your submission.

* The command I used was “grep -h Total rows.csv?accessType=DOWNLOAD.17 > distilledExpenditures.csv” then “emacs distilledExpenditures.csv”
* Attach the file