Computing Lab #1: Computing Essentials

Summary: Setup and familiarize yourself with essential tools we will be using throughout the semester.

Points: 10

Before doing this lab, please make sure your computer is correctly setup. See the handout *Setting up your computer for scientific computing*

I. TASK #1: INTRODUCTION TO THE TERMINAL (3 POINTS)

Learning objective/task: Using the terminal (and nothing else!) please analyze word frequency in the Python code Tweepy, a Python library for accessing the Twitter API (this library allows you to grab tweets and analyze them to make word clouds like this one: https://21centuryedtech.files.wordpress.com/2012/03/wordcloud1.jpg). You are encouraged to use the internet and any Unix help (for example "man wc" to find out what the program "wc" does) to solve this problem.

1. Open up the terminal.

2. When you open the terminal, by default you are placed into your home folder. Check this fact by using the Unix command pwd (= print working directory)

   ```
   >>> pwd
   ```

   Note: ">>>" signifies the command-line prompt and does not need to be typed. While we're at it, use the Unix command ls (= list) to inspect what files and subfolders are in your home directory.

   ```
   >>> ls
   ```

3. Now, as a sanity check, open up your file explorer (the GUI you typically use to view files) and check that you see all the files which are listed with the ls command.

4. Use mkdir (= make directory) to create a directory called "work". To do this type the following at the command-line prompt and hit enter:

   ```
   >>> mkdir work
   ```

   and then use ls to confirm the new folder exists

   ```
   >>> ls
   ```

   Finally, use your file explorer to find the new folder called work.

5. Use cd (= change directory) to change into "work", and use ls to check that its empty

   ```
   >>> cd work
   >>> ls
   ```

6. Before continuing, notice that you can go "up one directory level" (the directory directly above where you are) by using this command (NOTE: "#" means a shell comment – so do not type "#" or what follows it)

   ```
   >>> cd .. # the double periods, "..", is always the directory above where you are
   >>> pwd
   ```

   Now move back into your work folder. All of the following steps should be carried out in the folder "work"

   ```
   >>> cd work
   >>> pwd # you should be in the work folder
   ```

7. Use wget (= web get) or curl to download the tweepy source code https://github.com/tweepy/tweepy/archive/v3.5.0.tar.gz

   ```
   >>> wget https://github.com/tweepy/tweepy/archive/v3.5.0.tar.gz   # Linux
   >>> curl -L -O https://github.com/tweepy/tweepy/archive/v3.5.0.tar.gz   # windows or Mac
   ```

   and check that you have a new file in your work directory.

8. Use tar to unpack the file

   ```
   >>> tar -xzf v3.5.0.tar.gz
   ```

   Above, the strange "-xzf" is called a program argument, telling the program tar how to behave.

9. If tar is successful it will create a few folder, probably called tweepy-3.5.0. There will be a lot of files in this folder. You can use the file explore to take a look.

10. Use grep (grep is a command-line tool to search for files) to the find the files containing the phrase "destroy_friendship"

    ```
    >>> grep -r "destroy_friendship" tweepy-3.5.0
    ```

11. The answer to this first task is the name of the files containing the phrase "destroy_friendship". Either screen shot your answer or create a text file and add the file names.

II. TASK #2: INTRODUCTION TO GIT AND BITBUCKET (3 POINTS)

Learning objective/task: You are required to submit all code via the version control system Git. You will do so by "pushing" updates to your remote code repository hosted by Bitbucket. This problem will get you setup for this. By the end you should be familiar with the basics of git, Bitbucket, and the git commands init, clone, pull, push, add, commit, log and status.

Git (https://git-scm.com/): "Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency." Distributed version control system means that various people all have copies of a code repository (a code repository is a directory/archive storing the entire code project) and there is no "master" repository. From Git's point of view, all code repository copies are treated equally.

Git tutorials (to be viewed after class):

- Introduction video: https://www.youtube.com/watch?v=HVsySz-h9r4
- Getting started guide: https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control

- Interactive, browser-based, tutorial: https://try.github.io/
- More comprehensive tutorial: https://www.atlassian.com/git/tutorials/
- **Part a** Setup git, bitbucket, and cloning the class project folder.
    1. Make sure you have git installed on your computer. Installation tips were discussed in class, and there are numerous resources available on the internet. To check you have a working git installation, open up the command line terminal and type

        ```
        >>> git --version
        ```

        to find out which version of git you have installed. Record the version as your answer to part a.
    2. Create a Bitbucket account on https://bitbucket.org/. Make sure to register with an edu email address for extra features. Please email me with your Bitbucket account name as soon as possible.
    3. Once you have access to Bitbucket I will add you to our class team. After receiving confirmation from me, log into Bitbucket where you should now see a list of projects (called repositories in git-speak) which includes "umassd-mth280-2022". Click it, and copy and paste the "clone this repository" url into the terminal (which appears after selecting the "+" item on the left-side menu). Clone the repo by executing this command in the terminal.
    4. Consider moving the new folder umassd-mth280-2022 to your Desktop. This will make it easy to find.
    5. Change into the newly created directory and inspect the git log with the following command:

        ```
        >>> git log
        ```

        Record the first log message as your answer to part a
- **Part b** Setting up your personal git project with Bitbucket
    1. Go back to the main Bitbucket page where you'll see a list of projects, and find the git repository which matches your name. You should see two repositories: yours and "umassd-mth280-2022". **Contact me immediately if you are able to view anyone else's folder!**
    2. Click on your git repository, and clone this project to your desktop. The instructions are essentially the same as those from part a.

    At the end of this second task, you should now have two Bitbucket projects located on the Desktop.

III. TASK #3: SIMPLE PYTHON PROGRAM (4 POINTS)

Learning objective/task: Write and run a simple python program.

Imagine dropping a ball off of a building. Newton's second law of motion predicts the distance the ball has fallen, y (in meters), as a function of time, t (in seconds), according to

$$y = \frac{1}{2} g t^2 \,, \tag{1}$$

where $g = 9.81\ m/s^2$ is the *gravity of Earth.*

1. Using your favorite text editor (use a “plain text” editor like notepad, text wrangler, or sublime), and write the following three lines

```
t = 1.0 # time
y = .5 * 9.81 * (t**2) # vertical position
print(y)
```

2. Save this file as ball.py. You’ve just written a python program!
3. Run the python program by executing the following in the terminal

```
>>> ls      # COMMENT: make sure you see "ball.py" in the folder
>>> python ball.py  # Your answer should be 4.905
```

IV. UPLOAD YOUR WORK TO GIT

Congratulations! You’ve completed the lab. Now lets upload your work to git.

In what follows, “/path/to/your/project” should be replaced with wherever your git project is located on your computer (the Desktop is the best spot).

1. Open up a terminal and change into the git folder you created in Task #2.

```
>>> cd /path/to/your/project
```

2. Create a new directory to place your work

```
>>> mkdir lab1
```

3. Move your answers to Task #1 (name of the Tweepy files containing the word “destroy_friendship”), Task #2 (parts a and d), and Task #3 (the file ball.py) into the new lab1 directory.
4. Make git aware of these new files by doing

```
>>> ls      # COMMENT: make sure you see "lab1" in the folder
>>> git add lab1
```

5. Commit the files to the git project along with a commit message:

```
>>> git commit -m "Lab 1 answers"
```

6. Upload the files to Bitbucket

```
>>> git push
```

7. Finally, open the web browser and go to your bitbucket page. Check the files have been uploaded.