

Computer Science CS134 (Spring 2020)

Shikha Singh & Iris Howley

Laboratory 1

Introduction to the Python/git workflow

(Monday: due 11pm Wednesday, Tuesday: due 11pm Thursday)

Objective. To become comfortable working with Python and git.

This week we will spend a little time on some little programs. These programs will give you a taste of what it is like to work with Python in a computer scientist's environment. Beneath the Mac O/S is a Unix-based, extensible operating system. We'll be using git and gitlab to check out and turn in our work. We'll introduce you to Atom, an editor suitable for a wide variety of tasks. Finally, we'll get our first exposure to python3, the primary tool of this semester's study. Your job, this week, is to begin developing a working knowledge of these systems. It will be worth it.

New Environment Setup.

The following should be done the first time you use any new environment. Over the semester, you may use machines in CS labs, OIT machines, or, perhaps, your own computer. You can tell if you're setup for use in this environment by looking for a cs134 folder in your Home folder on the Mac (Shift-Command-H). If it doesn't exist, you need to set up the environment as follows:

1. Click on the Terminal application (identified by a >_ symbol) in the dock. If the application is not in the dock, try searching it on the Mac (Command-Option-Space); if unsuccessful you must install it. We will use the Terminal frequently as part of our workflow.
2. Create a directory that will hold all of your CS134 work. Throughout the course we will assume this directory is called cs134.¹

```
mkdir cs134
```

3. Now, descend into the cs134 directory (folder).

```
cd cs134
```

4. Now that we're set-up, we're going to retrieve the files we need to complete this week's lab. This will create a new directory in your cs134 folder with the name lab01.
5. Log-on to <https://evolene.cs.williams.edu> using your CS account credentials on a web browser. When asked for a password, use the password associated with your CS account.
6. On the homepage after you log in, you should see a link to a repository named cs134-s20/lab01/22xyz3 (where the last part is your username), click on it.
7. On the right side near the top is a blue button that is a drop-down menu that says Clone. Click on it and select the Copy URL to clipboard button under the Clone with HTTPS text. This copies the URL of the git repository for this lab.

¹You can name it anything you like, but in the future you'll have to remember to change instructions so they work with your particular setup.

8. Return to the Terminal application and type `git clone` followed by the URL of the lab project you just copied (you can paste on Mac by pressing Command-V) followed by the name of the lab `lab01`. Your line should look something like this:

```
git clone https://evolene.cs.williams.edu/cs134-s20/lab01/22xyz3.git lab01/
```

Here, `evolene` is the gitlab server dedicated to holding all of our collective work. The `cs.williams.edu` is the Computer Science IP domain at Williams. The `cs134-s20` reference is the course home directory, `lab01` is a directory we use for organizing student files by lab number, and `22xyz3.git` is the name of your repository on the server.

Finally, we need an editor in which we will write our scripts. We will use Atom—you should see the application in the dock (identified by a green atom symbol). Again, if it is not there, you can search for it on the Mac using the shortcut Command-Option-Space. Atom is pre-installed on all Lab machines; if using a personal machine you must install it from <https://atom.io/>.

Now, you're ready to begin work.

This week's lab. Make sure you've followed the instructions for setting up your environment. This week we'll focus on managing the workflow associated with the course.

Open the editor Atom by clicking on the application. You will see some welcome tabs, which you can ignore or close (there's a checkbox on the Welcome tab to hide these tabs in the future). On the top left, click on File and select Add Project Folder from the dropdown menu. Navigate to the Lab01 folder within your `cs134` directory and enter. You should see the folder added to your Project tab on the left.

Click on the `hello.py` file in the left pane. It will open up in the editor pane on the right. The file is empty except for comments (starting with `#`). You will edit this file to accomplish the following two tasks.

1. **Task 1.** Modify the script to write your first python program which prints out Hello World!. To test your work, open the terminal and execute: `python3 hello.py`. After completing a part of your lab, such as Task 1, it is usually a good idea to submit your partial work by committing and pushing as described under **Submit your work** below.
2. **Task 2.** We will continue modifying `hello.py` to accept input from the user and do some arithmetic with it. In this task, you must accept as input from the user a value that represents number of seconds. Convert this value to its equivalent in minutes and seconds and print it.

Below are two examples of how your code should behave when executed, when the user inputs are 146 and 42 respectively.

```
Hello World!  
Enter number of seconds: 146  
146 seconds is 2 minutes and 26 seconds
```

```
Hello World!  
Enter number of seconds: 23  
23 seconds is 0 minutes and 23 seconds
```

3. **Task 3.** Every week you will need to certify that your work is your own. Open up the `honorcode.txt` in Atom and type out the Honor Code statement, along with your full name. Be sure to Stage your changes to `honorcode.txt`, Commit to `master`, and Push just as you did with your python work.

Submitting your work. At any point, you can push your work to the server as follows. First, make sure you have saved all your edits to `hello.py`. Atom does not automatically save any changes, so it is important to frequently save your work.

1. From the top menu, go to Packages and select Github and click on Toggle Git Tab. A new pane will appear on the right with the title Git.
2. You should see `hello.py` under Unstaged Changes in the Git pane. Right click on the file name and select Stage, which will move the file to the Staged Changes section below.
3. Once all your edited files are staged (in this case `hello.py` and `honorcode.txt`), you are ready to commit your work. Write a brief but descriptive commit message in the text box at the bottom of the Git pane and click on Commit to `master`. This commits your work locally and you do not need to be connected to the internet to commit.
4. After you commit a Push button will appear at the bottom of the pane. Clicking on it will push your work to the CS server. You need to be connected to the internet for this step. After you push your finished work, you are done!

Important. You must complete Tasks 1, 2, and 3 and push your work to the server by 11 pm this Wednesday (for Monday labs) or 11pm Thursday (for Tuesday labs).

Working from off-campus. Our gitlab servers are only available from the Williams network. If you are off campus and wish to do CS134 work, you will need a Virtual Private Network (VPN) connection to campus to do so. OIT has instructions on how to do this, here:

<https://oit.williams.edu/help-guides/wifi-and-wired-connections/vpn/>