

Lab 2 Github and File Transfer

UWYO COSC 2030

1 Lab: Introduction to Github, Git, File Transfer, and Pi's

Github is used for version control, it keeps a running record of your project so you are able to revert changes, work across several systems, and work with each other. Below are some terminal commands that will be needed for this lab:

- `git clone [url]` -clone a repo down from git hub
- `git pull` -pull updates down to your repo
- `git add [file]` -prepare files to push up files. `git add .` to add all files
- `git commit -m "example"` -commit file changes with a message. Message inside of the quotes
- `git push` -finalize the push up
- `wget [url]` -pull a file from url
- `scp` -secure copy, copy a file from a local host to another over an ssh connection or vice versa

2 README.MD

2.1 Every lab must be submitted with a readme in the following format

```
Your name  
Help given/received
```

If your name is not in the readme your submission WILL have points removed, even if your GitHub username contains your name. If you do not put your name in the readme, it will be penalized.

3 Part 1: Github GUI

You will begin to learn the basics of Github following these steps:

- Step 1. Follow this link to start your github classroom assignment: <https://classroom.github.com/a/-SgPYKZg>
- Step 2. Go to your repo once it has been created and save the clone link.
- Step 3. Click on Add File ↴ “upload existing file”
- Step 4. Make a simple c++ program that prints “Hello, World!” named “labTwoTest.cpp” and upload it
- Step 5. Go back to Github and add a description
- Step 6. Click commit changes
- Step 7. Now your files should be uploaded!
- Step 8. Create a personal authentication token by going to your user account settings
- Step 9. At the bottom of the left sidebar click developer settings
- Step 10. You’ll be creating a classic personal authentication token
- Step 11. Put whatever you want in the note
- Step 12. For the expiration date set it so that it expires after the class ends
- Step 13. Check all of the boxes then generate your code
- Step 14. SAVE this code somewhere you won’t lose it!
- Step 15. Use this authentication code as your github password in later steps.

4 Part 2: Git Command Line

You will begin to learn the basics of Git (command line):

- Step 1. Ssh into a linux machine
- Step 2. Using the url you saved above, clone your repo
- Step 3. Enter your username and your authentication token as your password when prompted
- Step 4. Check to make sure your file made it by using the ls command. Go into that folder once you’ve seen it is there.
- Step 5. Using nano (or vim) edit your cpp document, save, and return to the base git directory
- Step 6. Using the commands above add, commit, and push your repo back up
- Step 7. Go to github.com, login, and your files should be there!

5 Part 3: Using the Raspbery Pi's

You will learn how to reserve and log into the raspbery pi's:

- Step 1. Go to the course webpage
- Step 2. Click on the "Pi's" link on the left sidebar
- Step 3. Claim a raspbery Pi
- Step 4. Sign into the Pi by following the instruction's in the lab FAQ OR using PuTTY
- Step 5. Username is "padawan", password is "wyoming"
- Step 6. Run your Hello World program on the raspbery Pi
- Step 7. Once you've successfully run your program, you are done!