Name: ____________________

Date: ____________________

GitHub Homework

QBIO Multi-omic Data Analysis

Background:

GitHub is a very widely used resource to save and share code. Here are some key GitHub definitions.

- Repository: A project folder for all your code and code related documents.
  - TWO "versions" of a repository exist: one on your computer where you can make edits, and one that is saved on your GitHub account.
  - You must manually save all your changes to your GitHub account.
- "Pushing" or "Committing" changes: To "push means to save your edits and send them up to your web-based GitHub account. (See class slides on how to push your edits to GitHub)

Activity:

1. In class, you "cloned" our course materials repository. This means you now have a local version of the class web-based folder. But how do you get new files that we share to the repository? Once files are added to the web-based repository, you must manually "pull" them into your local version.
2. User your terminal and appropriate `cd` commands to navigate to your local version of the course repository.
3. Use `ls` to view the contents of your local version of the class repository.
4. Open the web-based version of the repository. Is it the same as your local version?
5. Type `git pull` to "pull" any changes or new files in the web-based GitHub to your local computer.
6. Use `ls` to verify that a new file has been added to your local version and it now matches the web-based repository.
7. Now that you have the new file, what if you want to edit and then save your edits to GitHub?
   1. THINK ABOUT IT: What would happen if every student tried to edit the file and then save the same file to the class repository? These are called "merge conflicts."
8. To avoid this, you will need to copy any file you want to edit to your own personal, local GitHub repository, and push it to your individual web-based repository, NOT the shared class repository.
   1. First, use `mkdir` to create a week2_hw folder in your personal repository.
   2. Next, use the `scp` command to copy the desired file into this folder from the class repository. Review the previous homework for details on this command.
   3. Eventually, these will be coding files. For now, open the `getting_to_know_you.txt` file in Nano from your *personal* repository, NOT the class repository.
9. Answer the fun questions in Nano! Remember to save and exit Nano.
10. Use the commands you learned in class to add, commit, and push the entire Homework folder to your GitHub account.
11. To submit this folder and file as homework, follow these directions.

Homework Check:

- To push your local changes to GitHub, use the following sequence of commands:
  1. ____________ to view any unsaved changes.
  2. ____________ to save *all* files, or ____________ to save a specific file.
  3. ____________ to commit files for saving. Use '-m' to include a message.
  4. ____________ to push your changes to GitHub.
- The following steps need to be taken to edit any class assignments posted in the class GitHub:
  1. Navigate to your local version of the class repository.
  2. Use ____________ to "pull" any new files to your local version.
  3. Use the ________ command to copy new files to your personal repository.
  4. Edit the files in your personal repository.
  5. Add, commit, and push the files to your PERSONAL GitHub repository.
- Remember the command_dictionary.html file you made last week? Update the GitHub commands and their definitions into that file for future reference.