**GitHub Practice Instructions**

Developer #1

1. Create a GitHub repo and clone it down to your computer
2. Add a file called myFile.txt and add some text to the file
3. Make your first commit
4. Push your commit up to your GitHub repo
5. Invite Developer #2 to collaborate with you

Developer #2

1. Accept the email invitation to collaborate on a repo with Developer #1
2. Clone the repo down to your computer
3. Make a branch for your changes and switch to it
4. Add a new file called secondFile.txt and add some text to it
5. Commit your changes
6. Push your commit up to your GitHub repo
7. On GitHub, open up a pull request for Developer #1 to look at your changes and approve them to be merged into the main branch

Developer #1

1. Review and approve the pull request

Developer #2

1. Merge your branch and close the pull request

**A cheat sheet at a glance:**

| cd <directory name> | Changes your working directory |
| --- | --- |
| ls | Lists files and folders in your current directory |
| touch <filename> | Creates a file named filename |
| mkdir <dirname> | Creates a directory called dirname |
| nano <filename> | Opens filename in the nano text editor |
| echo <text> | Outputs the specified text in the terminal |
| cat <filename> | Outputs the contents of filename in the terminal |
| > | Character to take input from the left and use it to overwrite contents of the file specified on the right |
| >> | Character to take input from the left and append it to contents of the file specified on the right |
| git init | Initializes the local git repository (only need to do this once) |
| git add <file\_name> | Adds the changes of the specified file to the staging area |
| git commit -m “committing” | Commits all staged changes to that point with the specified commit message (“committing” in this case) |
| git pull | Pulls the latest changes from your remote repository (GitHub, BitBucket, etc.) |
| git status | Checks the status of your current changes and what is and is not in the staging area (great command to use all the time) |
| git log | Shows latest previous commits with a unique commit stamp, the author of those commits, and the date and time the commit was made |
| git checkout -b <branch\_name> | Creates and changes to a specific branch name |
| git checkout <branch\_name> | Changes to an existing branch |
| git branch <branch\_name> | Creates a new branch, but does not switch to it |
| git branch -d <branch\_name> | Deletes a specific existing branch (be very careful with this command) |
| git push origin <branch\_name> | Pushes all the latest changes to an existing branch in your remote repository |
| git merge <branch\_name> | Merge another branch into your current/active branch |
| git diff <source\_branch> <target\_branch> | Shows the current differences between the chosen branches (good for when you want to merge to know where there are conflicts) |