Mr. Bradley’s git workflow:

1. git pull

This updates your master branch with any changes that other people have pushed.

1. git branch myCoolStuff

git checkout myCoolStuff

This creates a branch called myCoolStuff for you to make useful changes in.

1. git add <any files you created or changed>

This tells git you want it to commit the changes you’re made to the files you’ve listed.

1. git commit

This creates a commit with your changes to the files you’ve added.

1. git checkout master

This changes which branch you’re looking at to master in your local version of the repository – any changes you’ve commited to myCoolStuff will disappear from the file system (but will reappear when you checkout myCoolStuff).

1. git pull

This updates your master branch with any changes that other people have pushed – we need to do this again because people may have changed things while you were working.

1. git checkout myCoolStuff

git rebase master

This appends the changes you’ve made in myCoolStuff to come after whatever other changes people have made in master in a nice line.

1. git checkout master

git merge myCoolStuff

If you’ve done everything correctly, this should be a FAST FORWARD merge. This updates master to include the changes you made in myCoolStuff.

1. git push

This pushes the changes you’ve made to the global repository (in this case, on Github). This will fail if your master branch is not up to date (which occurs when someone pushed changes while you were doing steps 7 or 8). If this fails, you should reset master to before your changes then repeat from step 6.

1. git branch –d myCoolStuff

This deletes the branch you created to make your changes. We don’t need it anymore since those changes are in master. When you want to make more changes, start from step 1. Remember, you can have as many branches as you’d like and name them whatever you want.