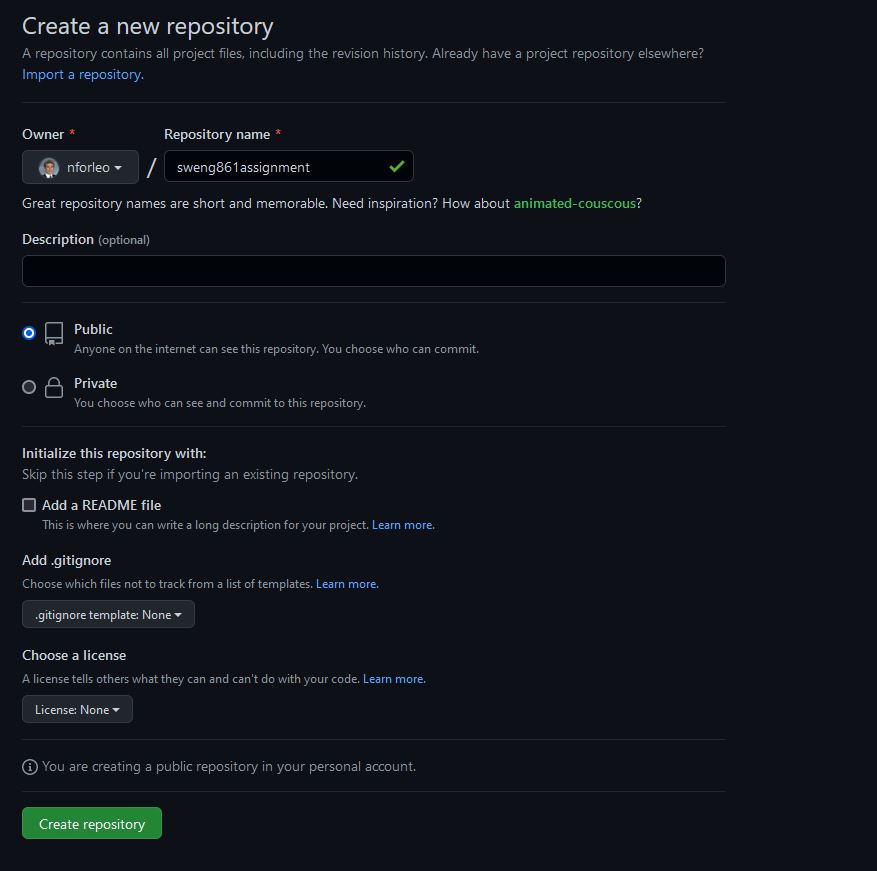
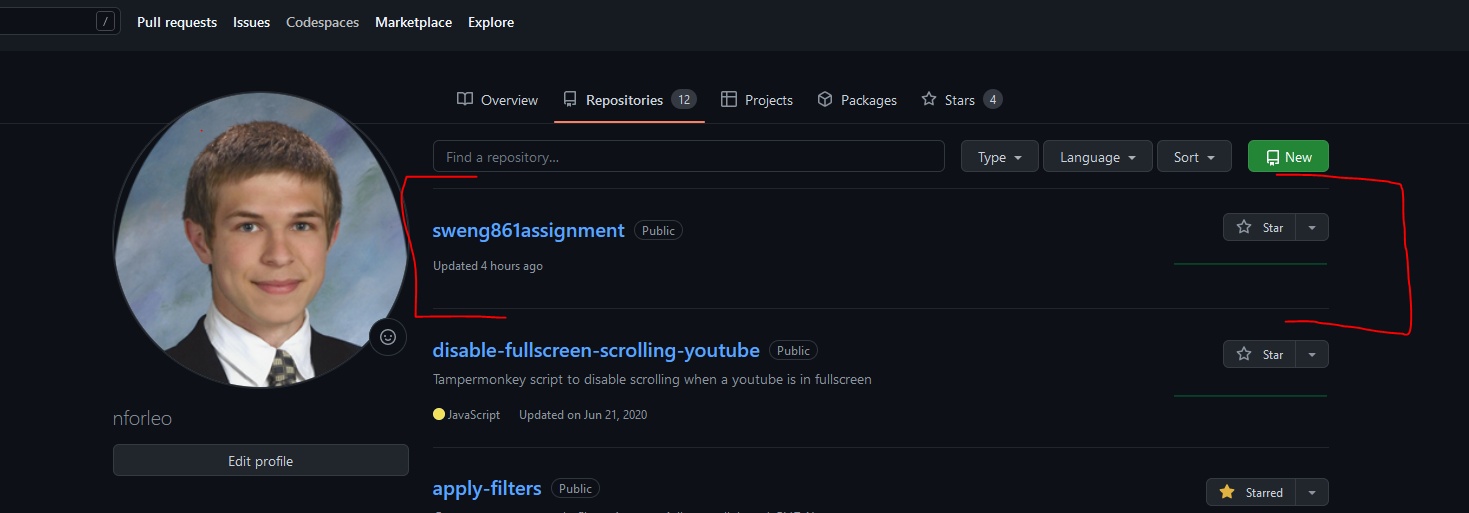
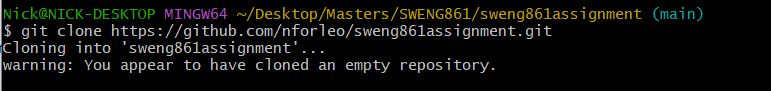
# Task1

Step 6:

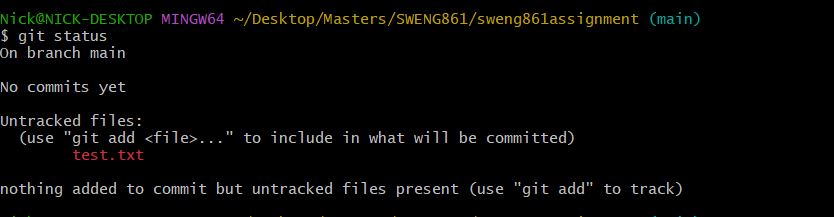




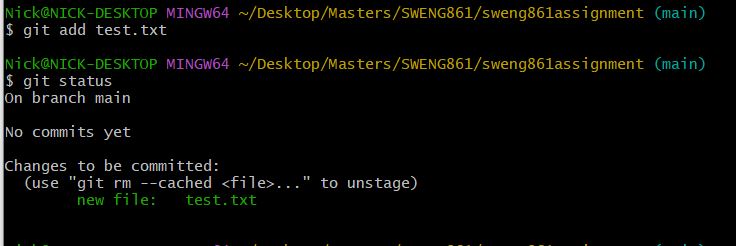
Step 8:



Step 12:



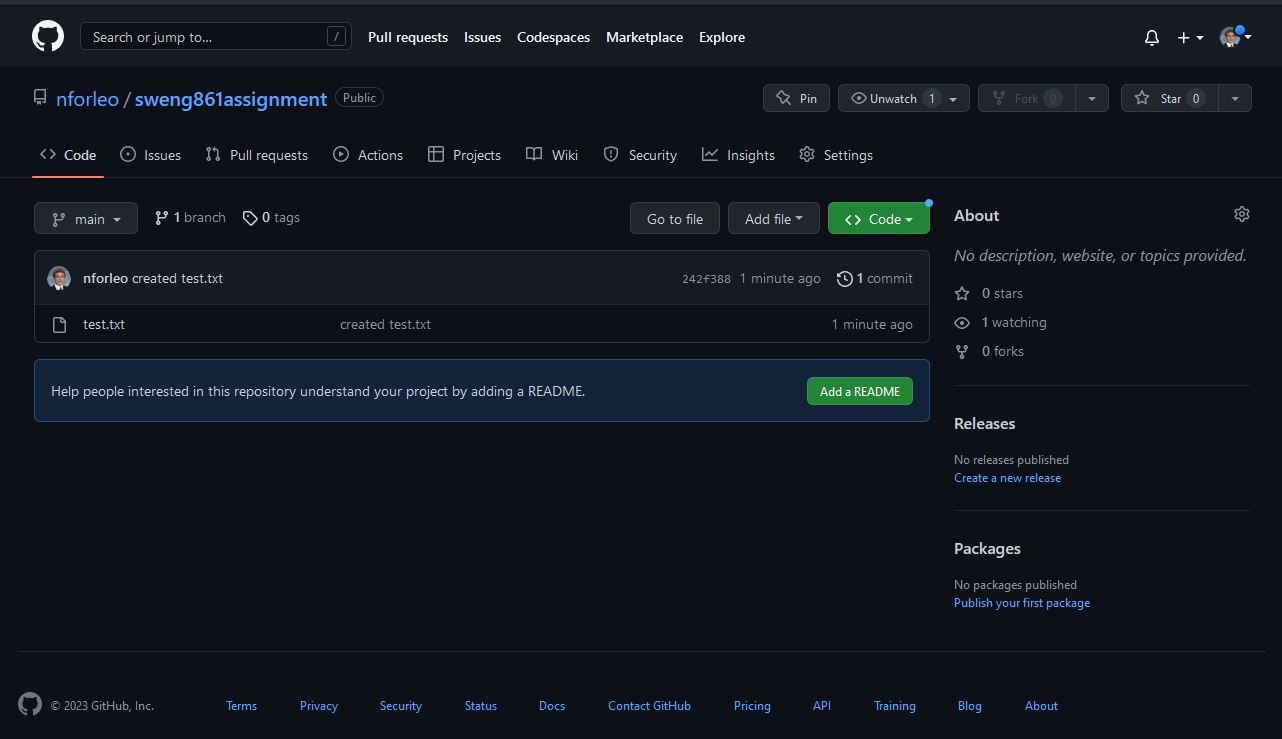
Step 14:



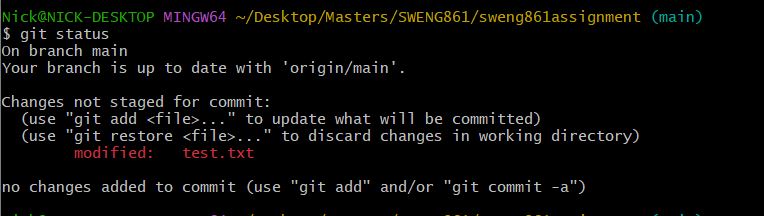
Step 16:



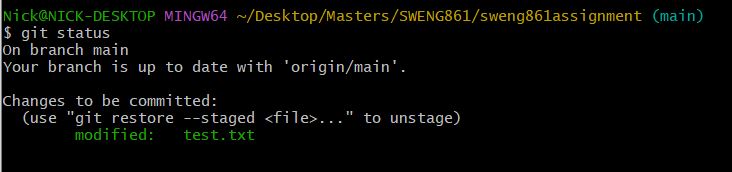
Step 18:



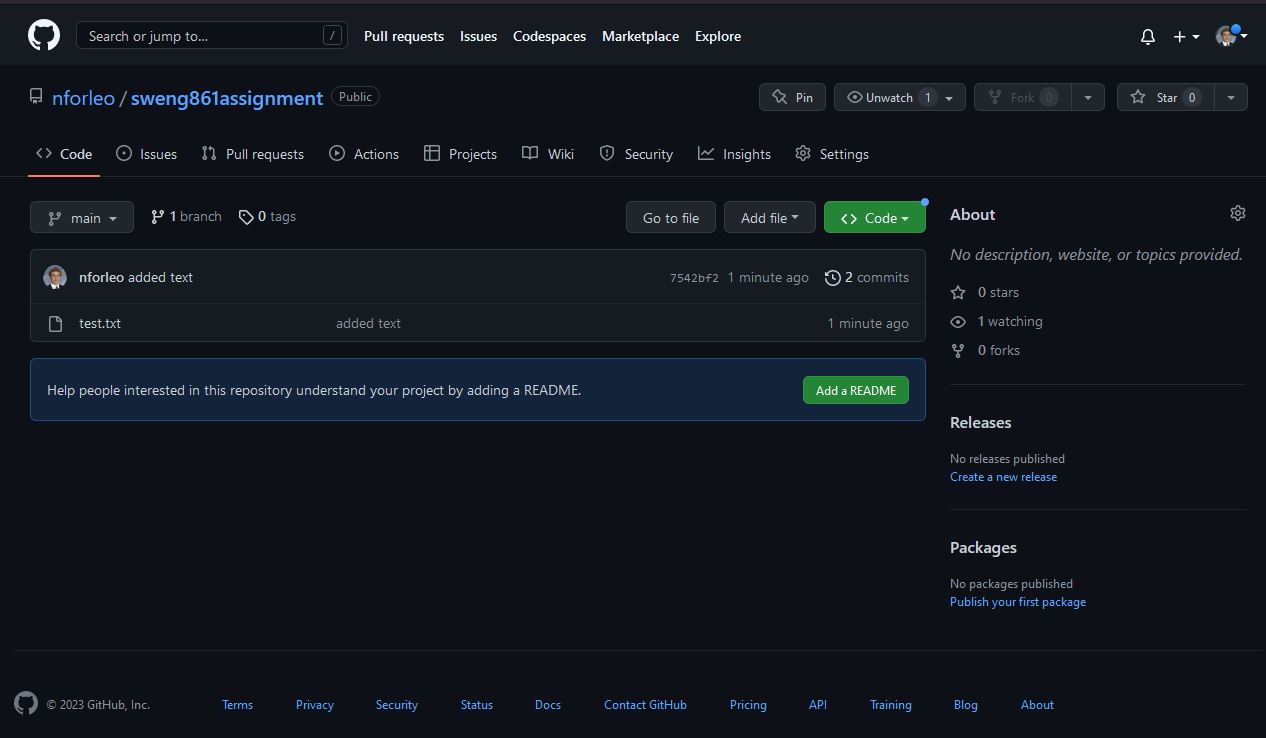
Step 20:



Step 22:



Step 25:

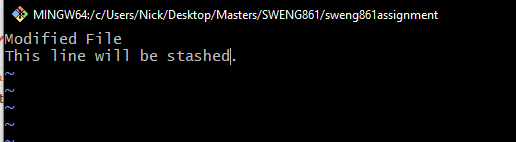


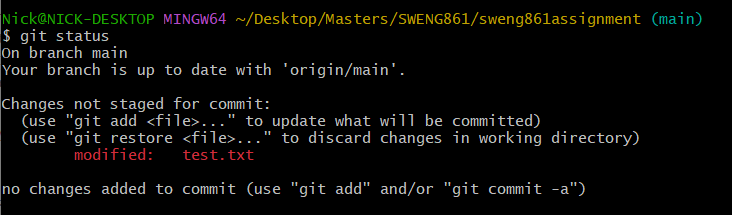
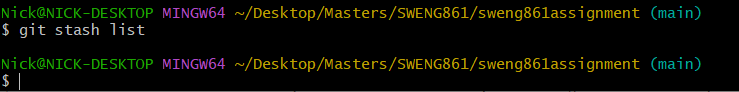
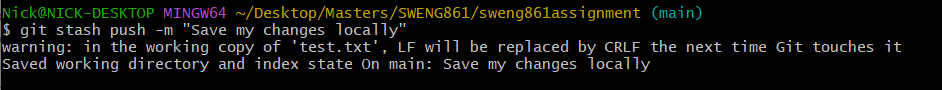
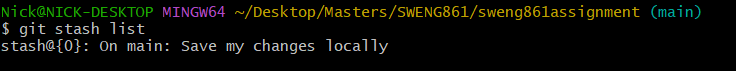
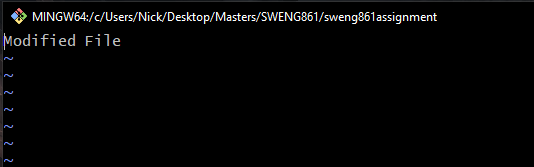
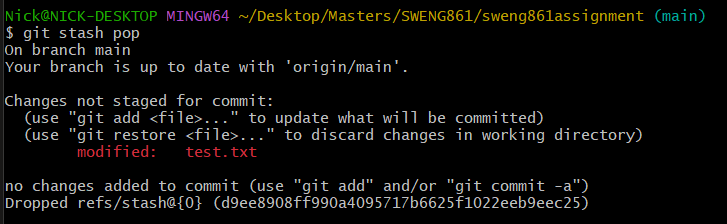
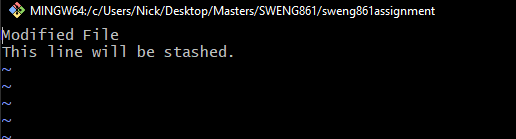
# 

# Task 2

One useful feature of git is git stash. It will store any changes you have made locally and then revert your working directory back to the latest commit. It’s used when you want to save your changes, but go back to a clean working directory or if you need to switch branches without making a commit. I use this feature mostly when I need to verify changes on another branch, but I am in the middle of making changes and do not want to commit my changes.

1. Using the test.txt file from task 1, take note of the initial (clean) state of the file
2. Make some changes and then save the file. I will add a new line that says, “This line will be stashed”



1. Run `git status` to see that the file was modified
2. Run `git stash list` to see there are no changes saved in your stash
3. Run `git stash push –m “Save my changes locally”` to stash your change(s). You will see your changes have been stashed locally
4. Run `git stash list` again to see your stashed changes
5. Open test.txt and you will see that your file has been reverted to its clean state
6. You are now free to change branches or make new changes
7. To apply your changes on top of the working directory from your stash, you can use `git stash pop`. It will also remove the changes from your stash.
8. Your file now has your uncommitted changes again.
9. Some other useful options/features of git stash include:
   1. `git stash pop –index <index>` where index is the number in the curly brackets (stash@{index}). Helpful for when you have multiple stashes
   2. `git stash clear` will delete all the stashes
   3. `git stash apply` does the same thing as pop except that it will not delete from the stash