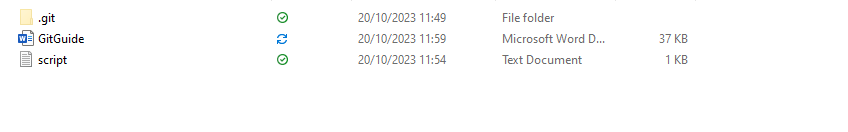
**GitHub**

**Noel Byrne 20/10/23**

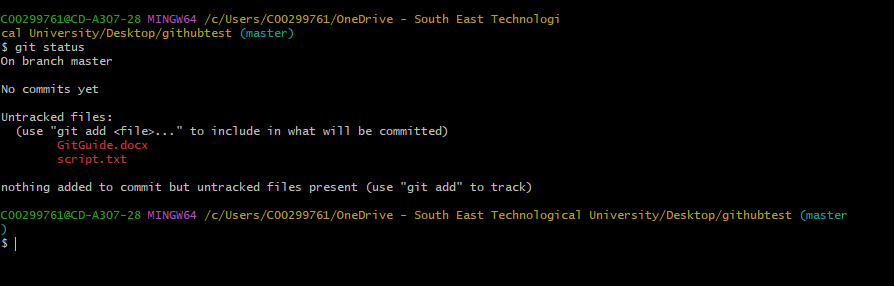
# **Init:**

git init creates a new repository called .git. It is created locally and not on github.

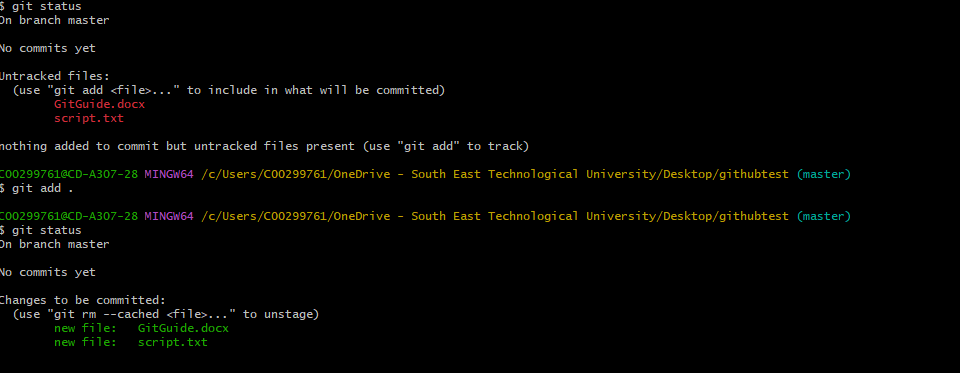
****

# **Status:**

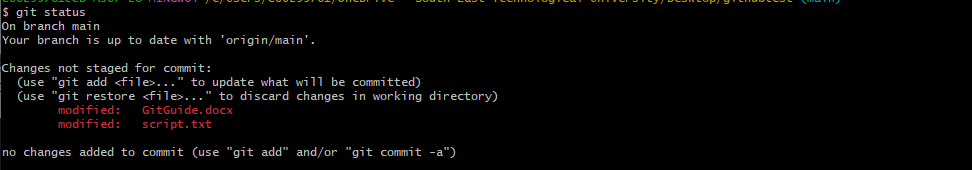
Use git status before and after adding files. Git status lists all the files in the current branch. Files in red are dirty files that aren’t committed. Once committed the files will be green.



# **Add:**

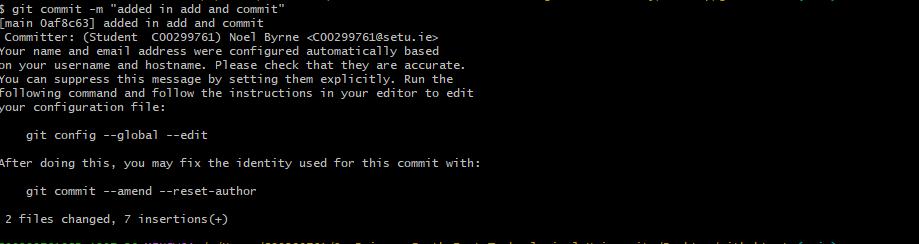
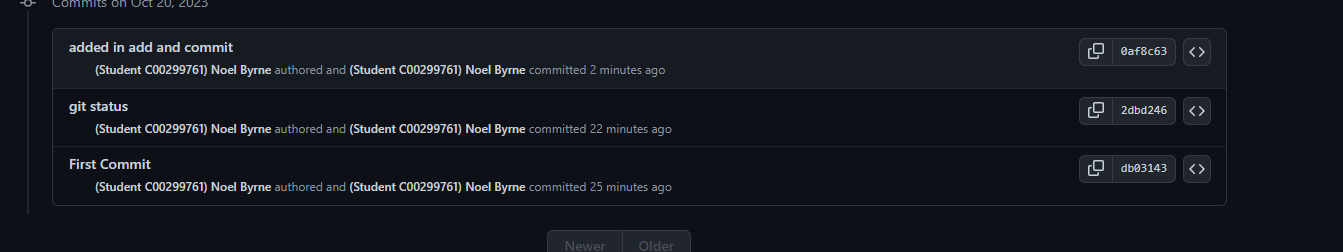
Git add . adds all files to the branch. It changes the files colour from red to green that haven’t been added yet/ have been changed.

As you can see the file colour has changed after we added the files.



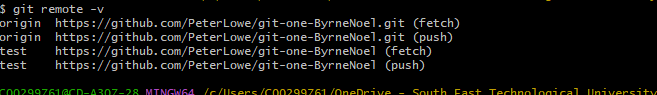
After adding more to the files and saving them they have turned back to red.

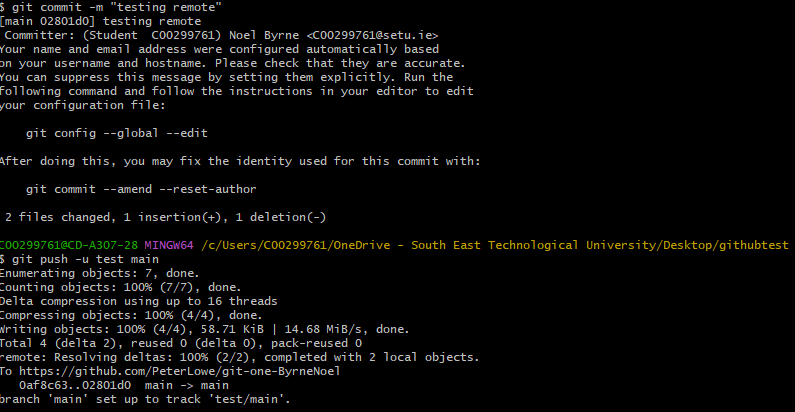
# **Commit:**

Git commit adds a new revision point in your work. By doing git commit -m “”, you can enter a name for your commit. The name is usually something new you have added into your files/code.

The name I gave the commit has now been added.

# **Remote:**

Git remote can be used to list all the repositories on the server. You can add a remote repository by typing “git remote add <name> <url>”. You can then commit and push to that repository instead of always committing to origin/master.

I created a repository called test and I committed + pushed to it instead of origin.

# **Clone:**

A screen shot of a computer

Description automatically generatedGit clone allows you to take a copy of your work and clone it to your computer somewhere else. That copy is stored on your computer, and you can send the updated work back when you’re done. Here is me cloning my work from Friday.

# **Push:**

A screenshot of a computer

Description automatically generatedGit push is used to push the work you have committed to the main branch. It updates the remote node with the current. This is me pushing my work to the main branch.