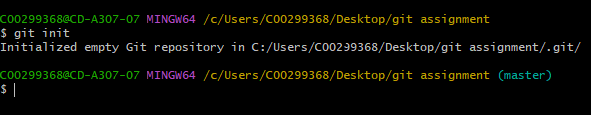
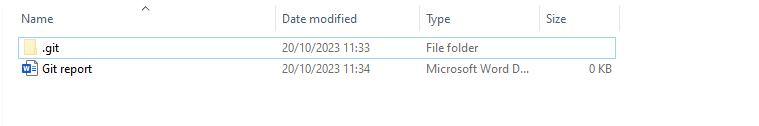
**git Init**

Creates a new repository which is stored only in 1 location which is the local PC

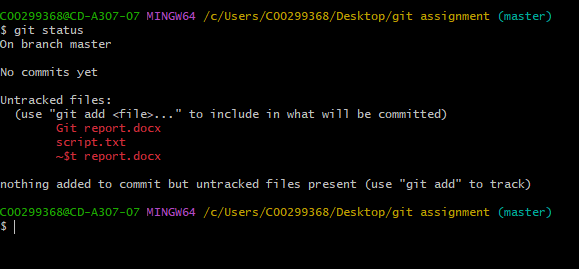


This creates a hidden .git file in the repository which can be viewed by going into the view tab and checking the “hidden files” box.



**git Status**

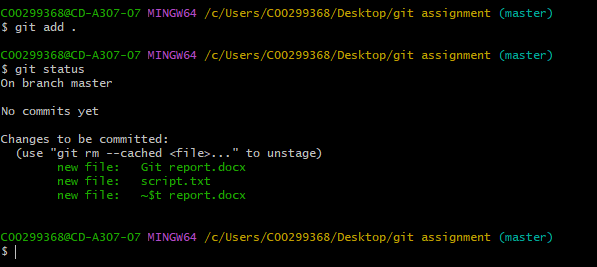
Shows the current status of your git repository and any files that are not staged or the ones that are staged



Red files are new files which are in the repository which have not yet been staged and will not be saved when you execute the “git commit” command.

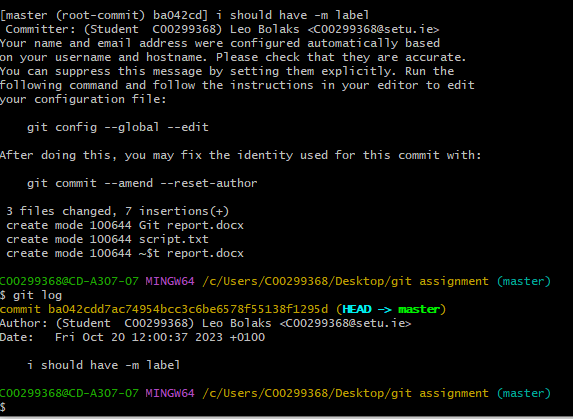
**git ADD**

Git add command stages the files in the repository and they will be displayed in green, this can be checked by re-using the git status command, now when you do a commit, these files will be saved and you will be able to see them in the github repository.



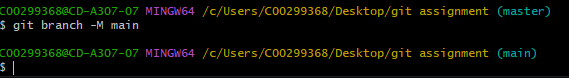
**git COMMIT**

The commit command is basically a save which creates a new node which you can go back to if there has been a mistake further up the nodes, the commit command saves all the staged files (the ones in green) to the repository



**git BRANCH**

The branch command I used was “git branch -M main” this command renames the repository from “master” to “main” but different variations of this command can create new branches, edit them, or delete them.



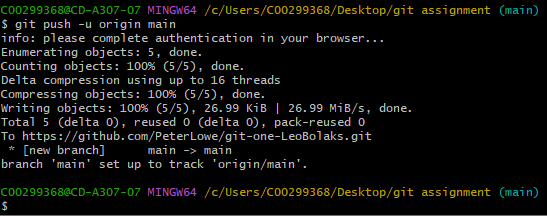
**git REMOTE**

The remote command allows you to connect to a repository somewhere in the cloud, in this case the name of the cloud repository is remote, you can now push all the commited files into the repository



**git PUSH**

The push command pushes all of the files you have commited to the repository which is not on your local machine, in this case I pushed all files I have commited to the repository named “origin”



**git CHECKOUT**

the checkout command allows me to compare whether the node I’m working on is the same or different from the main directory in this case my branch is up to date with the origin/main repository.

