Shaun Ziman

210262877

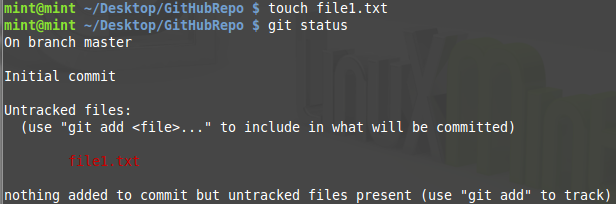
Class 3B

GitHub Assignment 1

1. Set up a GitHub account and address. Make sure that you choose a meaningful name that you will be able to easily recall. For Example my GitHub account and address is http://github.com/boniface
2. Log into the Linux Image VM Player and create your unique folder and run the git commands for the following tasks and record the screen shots of the results in a word document that you need to submit.
3. \*\*\*Initialise git repository

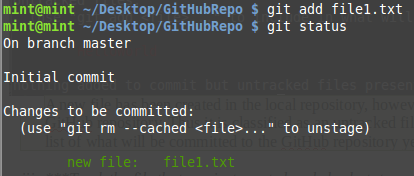


1. \*\*\*Create a file in your repository, check the status and give the explanation of the status

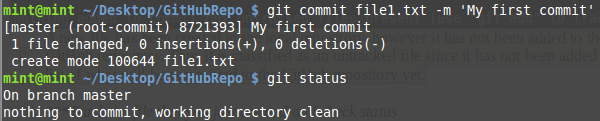


A new file has been created in the local repository, however it has not yet been added to the GitHub repository. Thus it is classified as an untracked file since it has not been added to the list of what will be committed (stage).

1. \*\*\*Track the file that you just created and check status

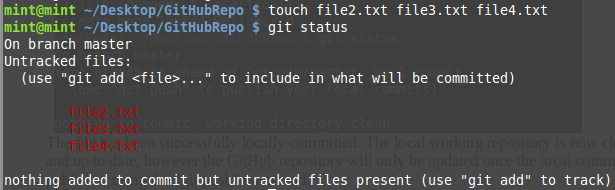


1. \*\*\*Commit the file to your repository, check the status and give an explanation



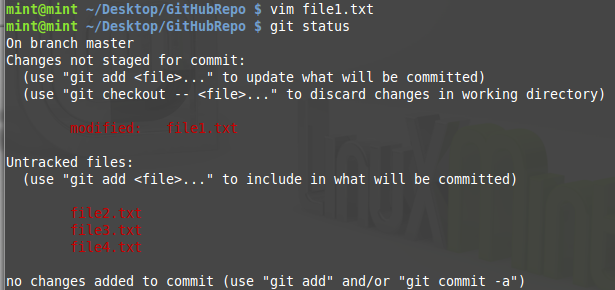
The file has successfully been locally committed. The local working repository is now clean and up to date, however the GitHub repository will only be updated once the local commit is published using the “git push” command.

1. \*\*\*Create three more files and check the status and explain why the files are in the states they are in

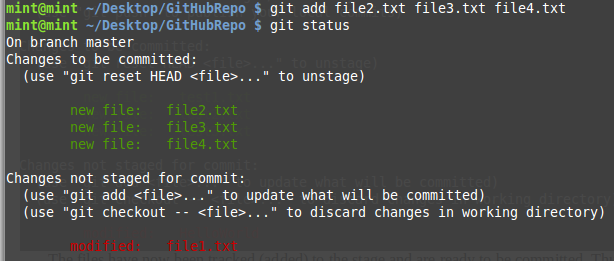


The files have been created in the local repository but not yet tracked. Thus listed in red text.

1. \*\*\*Make Changes to the file you committed and check the status

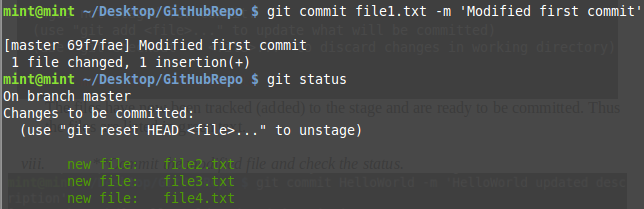


1. \*\*\*Now track the three files you added above and give an explanation of your status output

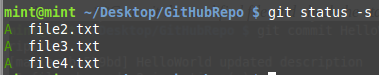


The files have now been tacked (added) to the stage and are ready to be committed. Thus the files are listed in green text.

1. \*\*\*Commit the modified file and check the status

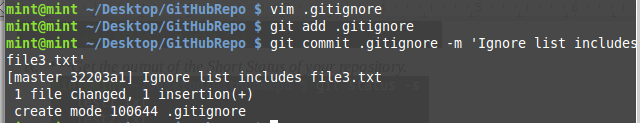


1. \*\*\*Get the output of the Short Status of your repo

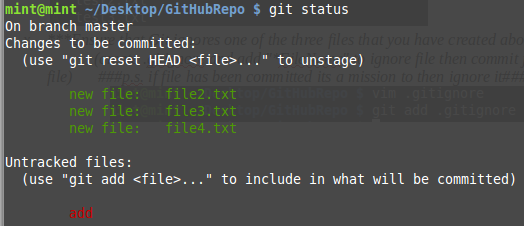


1. \*\*\*Ensure that Git ignores one of the three files that you have created above

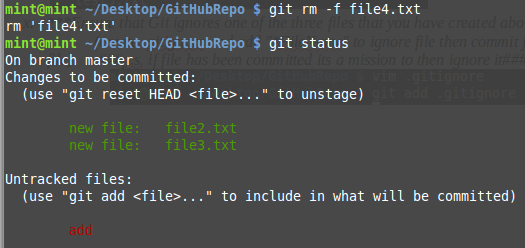
(create .gitignore and add "\*FileName" to ignore file then commit just the .gitignore file) ###p.s. if file has been committed it is a mission to then ignore it###



1. \*\*\*Get the status of the changes that you made to your repository

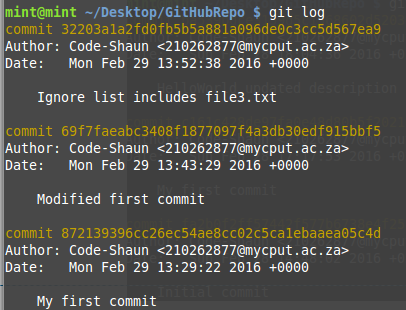


1. \*\*\*Remove one of the three files that you had created and are tracking. Get the status and give an explanation



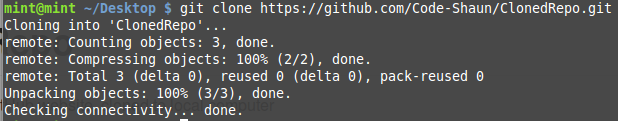
The file has been successfully removed and thus does not appear in the list.

1. \*\*\*Get the output of your commit History



3. This section requires you to use your GitHub account that you just created

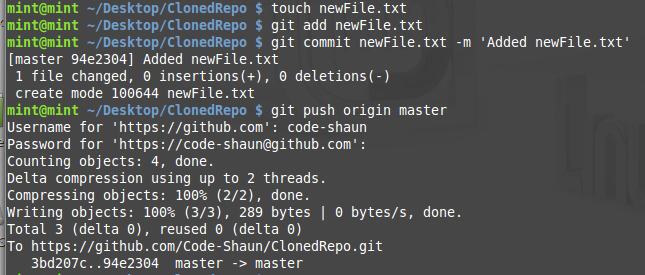
1. In your GitHub account, create a repository and clone it onto your PC and ensure that you put it in a different directory



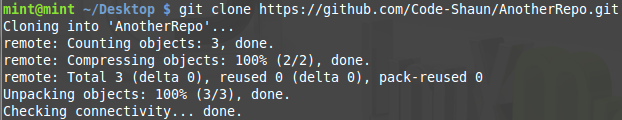
1. Show your full remotes on your local machine using a git command and take a snapshot



1. Add a file to your repo and push the file to GitHub



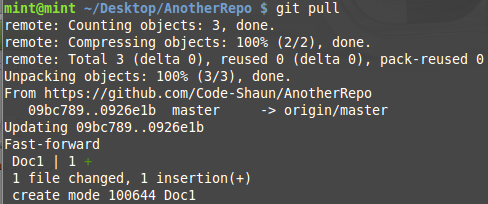
1. Create another Git Repo and add any other file to the remote repository and again show full remotes for your repository



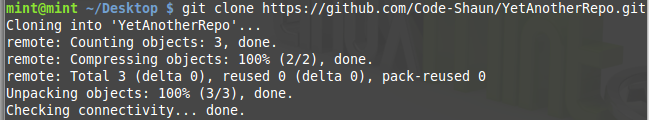




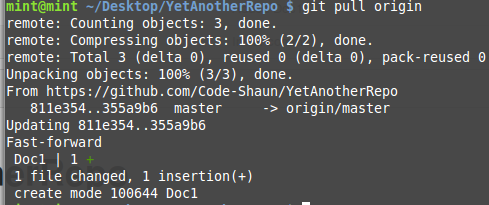
1. Get the contents of the remote repository in and merge it into your local repo



1. Create another git repository and add another repo repository.



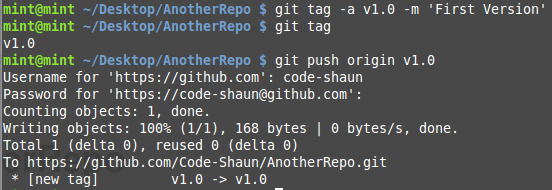
1. Get the content using one git command so that the result is a fully merged repo



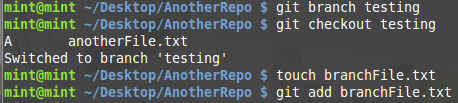
1. Remove the remote from your repository

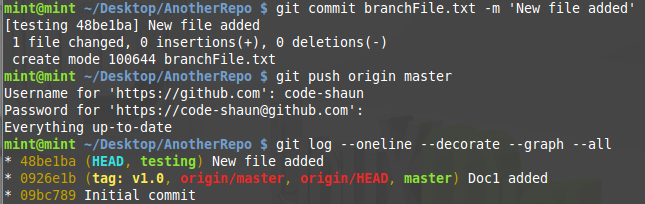


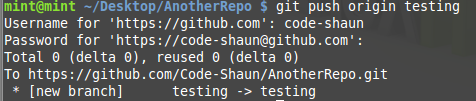
1. Demonstrate the ability to tag your repo with version numbers and commit them to GitHub

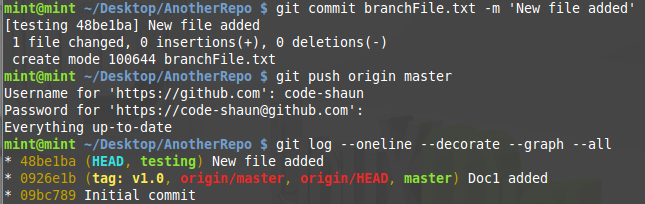


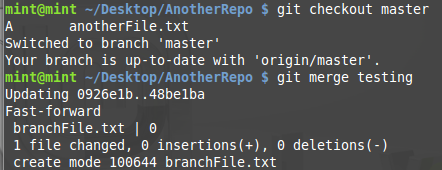
1. Read the chapter on branching and demonstrate the ability to create branches and integrating the merge. GitHub will produce graphs for you.

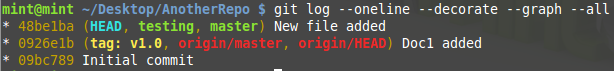












1. Demonstrate Rebasing on another repo that you create on GitHub

