# Assignment 2.

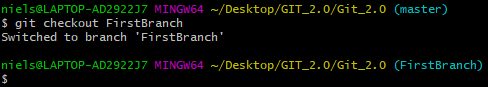
## Creating a branch (2.1.1)

I have the “testing repository” already. I’ll just be branching on it since I already have it. To create a branch the command “git branch {branch name}” is used. “Branch name” can be anything you want, just make sure its descriptive.



## Moving around (2.1.2)

To work with the branch you must move to it first on GitBash. This is done with the command “git checkout {location name}”. The location name is the branch or version you want to work on. So “master” for the master branch (which is the default), or the name of other branches created. In this case I’ll be going to “FirstBranch” as that is the branch I created.

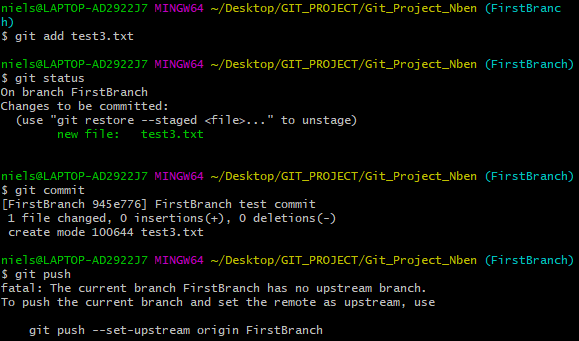


Now it has switched to “FirstBranch”, you can tell by the name at the end in this case it went from “(master)” to “(FirstBranch)”.

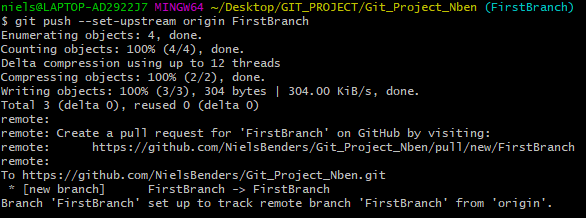
## Updating the repository (2.1.3)

As of yet what I have done in this assignment has all been local, so the repository online is still outdated and has no branch. So I’m going to add something to the branch to make it different from the master branch and update the repository.

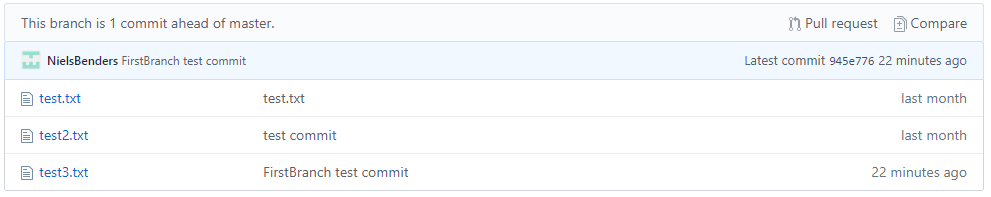
As this repository is just for private testing purposes I added a new text file named “Test3.txt”



As you can see, the standard “add -> commit -> push” will not work when working with a branch that does not exist yet online, but GitBash has you covered. GitBash will come with suggestions on how to properly push this commit, in this case adding some parameters to the push command. So I will be using “git push –set-upstream origin FirstBranch”.

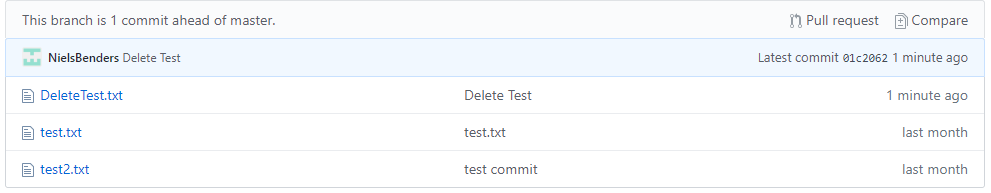


And now the remote repository has been updated.

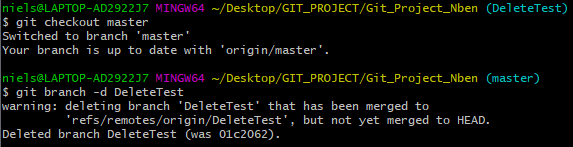


## Deleting branches (2.1.4)

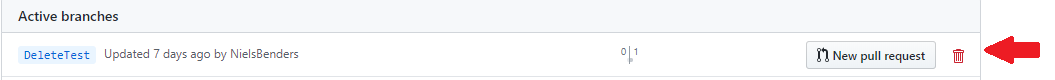
Finally, we have the deleting of branches, it pretty simple. But to show this and not lose any of my reference material I am going to be using a new branch specifically made to show this.



To delete a branch the command you use “git -d branch {Branch Name}”.



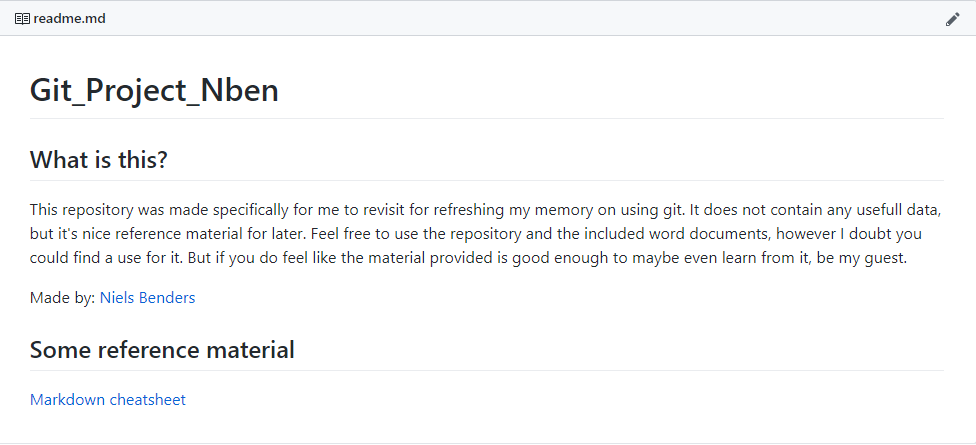
However, the branch will stay online. To delete it from the remote repository you go to the GitHub website and click this button in the branches tab of your repository.



I did leave the branch in the repository as this is handy reference material for later.

## Adding a readme (2.2)

A readme is a file which explains what the repository is for or how it’s files can be used. But since this is a reference repository, I made this:



This was done in MarkDown language which is similar to MarkUp language. Full reference [here](https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet).

## Adding a .gitignore (2.3)

.gitignore is a file used to exclude certain files from a commit in case you do not need to upload it, but do not want to filter it out by hand every time. I did add the .gitignore but since the documentation did not go further into using it (and I do not really understand the GitHub documentation). I will not be doing anything with it.

