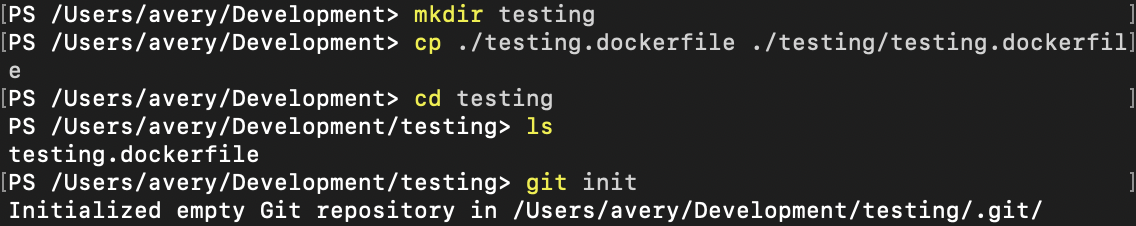
At the end of the day, Git is a tool that allows us to store data, which could be applications, files, and a number of other artifacts with its repository structure, and the most common way for us to do this is via the command line tools:

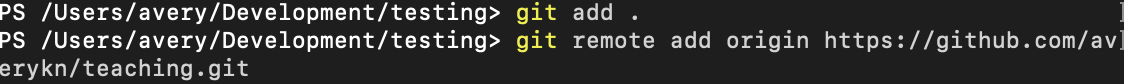


With those installed, we can then start the process of managing whatever data we’d like to control/track changes for, and we’ll first do that in a designated directy using *git init*;

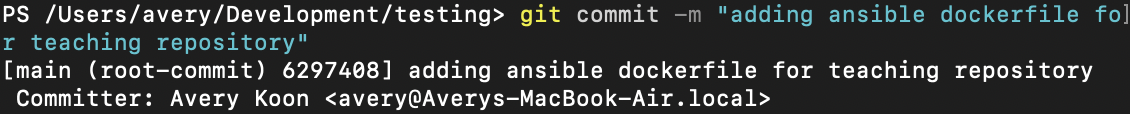
In this case, we’re doing a simple Dockerfile, named testing.dockerfile, for a containerized ansible deployment, moving into the desired directory and initializing, which tells git where to look for changes



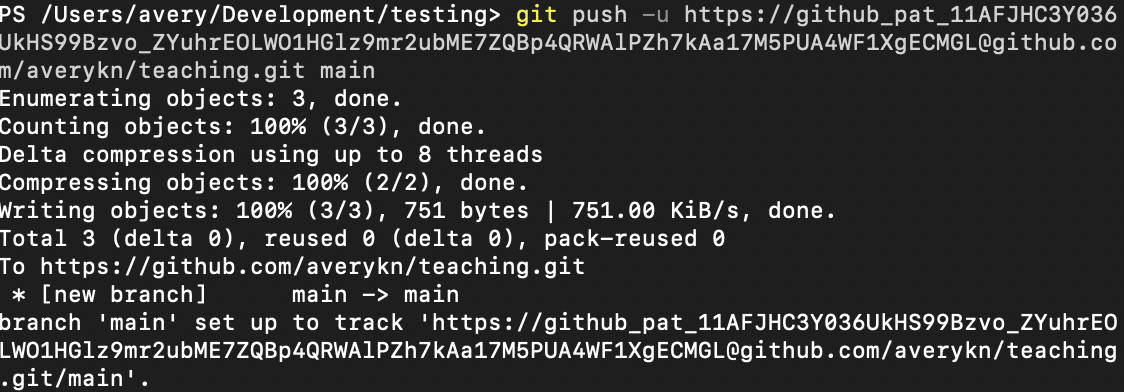
Then we’ll use git add to tell git, in this example, to add anything that’s in the current directory to the list of things to be considered for control/tracking and add our remote repository:



Once that’s been done, we’ll be able to include our commit context, via *git commit- M “$yournoteshere”* to let git know the context of the changes made:



And we can push to our public repository with git push, using a token, generated from <https://github.com/settings/tokens>, and using the syntax *git push -u https://$yourtoken@$yourgithubrepo*:



Et voila we can see our updated code in our remote repository:

