Press enter to submit commands

> git status

# On branch master  
#  
# Initial commit  
#  
nothing to commit (create/copy files and use "git add" to track)

Success!

$ git status

# On branch master  
#  
# Initial commit  
#  
# Untracked files:  
# (use "git add <file>..." to include in what will be committed)  
#  
# octocat.txt  
nothing added to commit but untracked files present (use "git add" to track)

Success!

$ git add octocat.txt

Nice job, you've added octocat.txt to the Staging Area

$ git status

# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
# (use "git rm --cached <file>..." to unstage)  
#  
# new file: octocat.txt  
#

Success!

$ git commit -m "Add cute octocat story"

[master (root-commit) 20b5ccd] Add cute octocat story  
1 file changed, 1 insertion(+)  
create mode 100644 octocat.txt

Success!

$ git add '\*.txt'

Success!

$ git commit -m 'Add all the octocat txt files'

[master 3852b4d] Add all the octocat txt files  
4 files changed, 4 insertions(+)  
create mode 100644 blue\_octocat.txt  
create mode 100644 octofamily/baby\_octocat.txt  
create mode 100644 octofamily/momma\_octocat.txt  
create mode 100644 red\_octocat.txt

Success!

$ git log

commit 3852b4db1634463d0bb4d267edb7b3f9cd02ace1  
Author: Try Git <try\_git@github.com>  
Date: Sat Oct 10 08:30:00 2020 -0500  
  
Add all the octocat txt files  
  
commit b652edfd888cd3d5e7fcb857d0dabc5a0fcb5e28  
Author: Try Git <try\_git@github.com>  
Date: Sat Oct 10 08:30:00 2020 -0500  
  
Added cute octocat story

Success!

$ git remote add origin https://github.com/try-git/try\_git.git

Success!

$ git push -u origin master

Branch master set up to track remote branch master from origin.

Success!

$ git pull origin master

Part 2.

GitHub is a revision control software which utilizes both online and local repositories for one or more users to work on projects. GitHub was launched in April 2008. Some similar platforms include Subversion, BitKeeper, and Bazaar. A platform such as this can be very useful when collaborating with multiple users. For instance, Git will not allow you to push an update to the repository unless you have the most recent version of the project. This way, users will not be overwriting one another’s modifications. It can also be useful simply for storage. If your computer breaks and you lose your local repository, you can copy the online repository and get back to work. These platforms are also useful in the idea of forking a repository. Users can fork a repository, that is create a new branch of revisions from the main branch of revisions. This allows users working on the project to go in different directions without affecting the main branch of the project. It also allows users not involved with the project to make their own modifications and then request they be merged with the master branch later on.

Part 4.

Repository – The space in which all of the files are stored for your project

Commit – The state of your project at the time the commit was created

Push – A command in which the current state of your repository is merged into the online repository

Branch – An isolated tree of revisions. It is a copy of a repository that does not affect that repository in any way when a user makes updates

Fork – Forking is the process of copying a repository at any given time so that you can make your own edits without affecting the main branch

Merge – The act of applying a local repository’s updates to the main branch

Clone – The act of creating an exact copy of a repository at any given time

Pull – The act of copying all the data from one or more repositories and attempting to merge it with your current branch

Pull request – This lets other users working on the repository know you have pushed to a repository. The users can build upon that or revert your push.

Part 6.

In order to do this section, I first forked the paceuniversity/courses repository. Then, I cloned the repository I had forked to my computer. From there, I edited the README.md file and pushed it to the repository I had forked. After that, I was able to send a pull request to the master repository.

Part 10.

I’ve used GitHub to create and manage personal projects as long as collaborate with group projects for class. I’ve mostly used it for personal projects which were mostly tutorials. Since I was the only collaborator in almost all of those projects, there was no need for me to make additional branches. There was one project I did for a class with another student. We only used one branch for editing because we had minimal updates and always built off of each other’s updates.