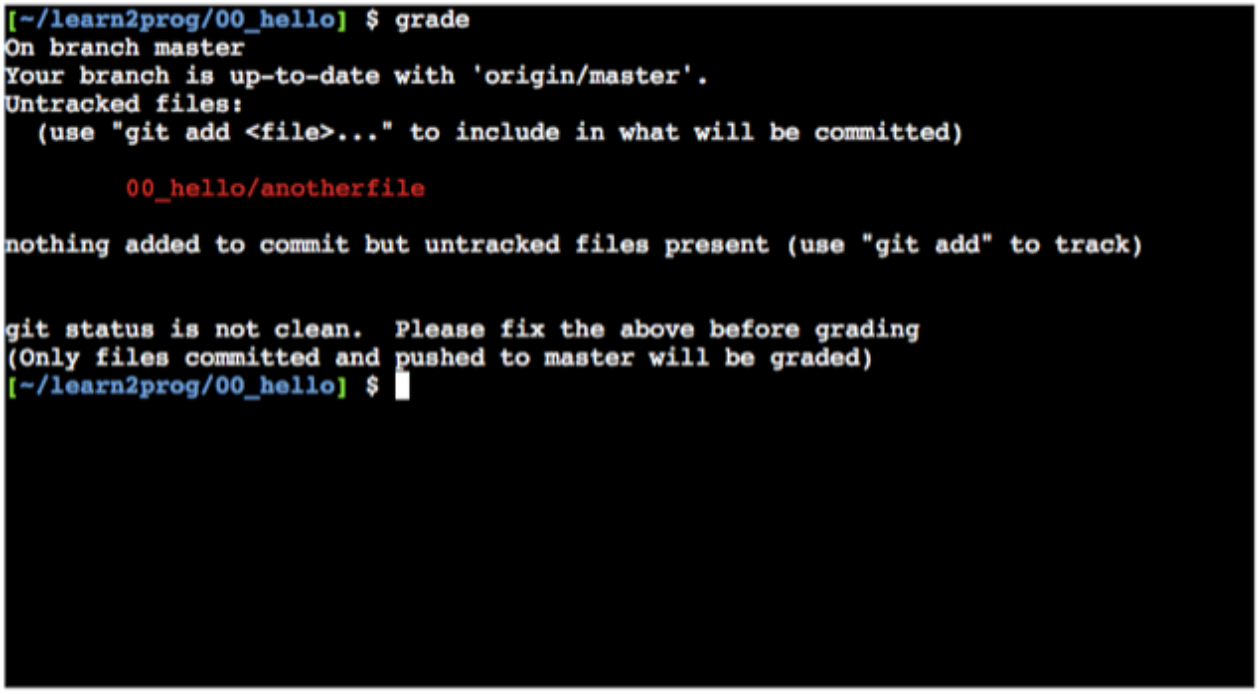
While you may not need this information right now, we want to provide it to you before you are likely to run into these common problems in the Practice Programming Environment. If you do, remember that you can come back to this reading for help.

## Common Problem 1: “git status is not clean” when trying to “grade”

If you have files which are not put into git, then when you grade you will encounter an error message like this:




The grade command requires that ALL files be either placed into git and pushed, or explicitly ignored. It does this so that you do not run into confusion from having only part of your work graded (the grader will only see the files that are put into git and pushed).

So how do you fix this error? Either add the files listed in the error message into git and then git push (if you want to keep them), or remove those files (if you do not). If you are unsure, then go ahead and add the files. If you want to add the files, use git add, for the filenames in question:

*$ git add filename1 filename2 ...*

(The *$* character above is your terminal prompt. You don't need to type it in.) In the above example, we would do

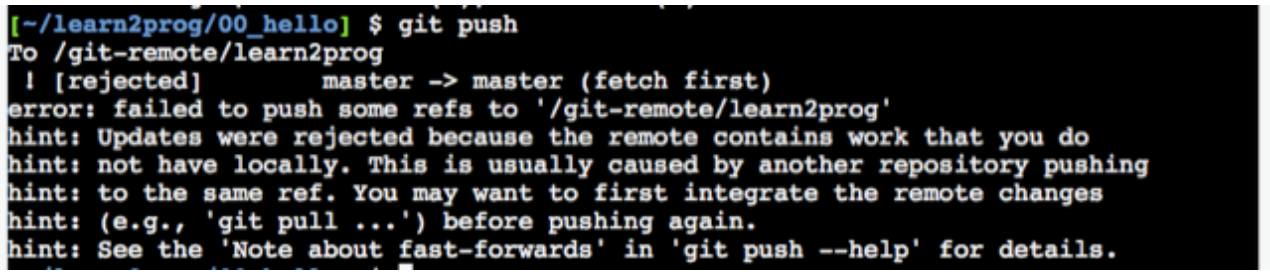
*$ git add anotherfile*

Then use git commit to commit the files. Then do git push. Now you can do *grade* .

If you want to remove the files, use *rm filename* to remove the file.

## Common Problem 2: “git push” gives you a “[rejected]” message

If you try to do “git push” when there are changes that you do not have, you will get a message like this one:




This typically happens when you have run “grade” (which puts the grade report into your remote repository) but have not done “git pull” to receive that grade report yet. The solution to this problem is to just run

*$ git pull*

which will pull in the remote changes. After that, you should be able to run “git push” again.

## Common Problem 3: how do I delete files like #hello.txt#? Why are they created?

These files are created by Emacs when you fail to exit Emacs properly. If you see these files appearing, you can either tab to autocomplete or use the escape character (\#) to delete these files. See example below.

*$ rm \#hello.txt#*

## Common Problem 4: I didn't change files, but git is showing them as modified

When opening an assignment, you may notice that *git status* shows many of your files from previous assignments as modified, even though you haven't edited them. This is because when you don't use your PPE for a while, Coursera shuts it down, and when you access it the next time it starts the environment up again. As part of this startup process, it modifies the Unix permissions of all files in your home folder to be read-write-execute (rwx) for all users. Files you create via Emacs and other programs in this course are typically created as read-write for you, and read-only for other users. Git tracks changes to permissions, and therefore shows the files as modified because their permissions were changed. As detailed in Problem 1, this "unclean" status prevents you from running *grade* .

So how to fix this? There's no harm in adding these permission changes to git as they occur (and should only be necessary once per file):

*cd /learn2progcd/learn2prog git status* <== (make sure you want to add all these files/changes to git) *gitadd.gitadd. git commit -m "Permission changes"*