How to: Git Essential Commands

To commit your recent changes it is necessary to complete the following three steps:

1. *git add (file1).r (file2).r (file3).txt*

You have to navigate into the directory containing the files you would like to add. You can also use *git add ** command to add all of the files inside the directory you are currently in.

2. *git status* (make sure the files you want are added in correctly)
3. *git commit -m "(your commit message)"*
4. *git push -u origin master* (after the first push, you can just type **git push**)

If you are adding multiple new files from different subdirectories you may use:

1. git add -A (or *git add --all*) (from the top repository directory)
2. *git commit -m "describe committed materials"*
3. *git push origin master* (after the first push, you can just type **git push**)

To get recent changes:

1. *git pull* (or *git pull origin master*)
2. If you have a merge conflict type:
   a. *git stash*
   b. *git pull origin master*
   c. IMPORTANT: check the code that was pulled (it will be in the git pull log) in a text editor and make sure that it is not a better (more recent version) than your own code
   d. If your code is more recent run: git stash apply
      i. If merge conflicts still appear, they need to be fixed manually by modifying the file that is causing the conflict
   e. If it is not, you can run git stash drop

When you don't know what you're doing or something goes wrong or just for fun do this:
git status

Non-Essential Commands:

git rm --cached fileName.r	this will remove a file from your git repository
git stash	saves changes in the local
git merge	merges branches of a directory (use this if conflicts exist)
git status	shows current changes that you have made using git ex: any files that have been added
git reset	undo any git actions that have been made including commits or adds but not pushes