`git status` - find out what's going on in the current directory (assuming it is tracked by git, of course)

`git clone [repo url from github/bitbucket]` - grab the repository and put it on your machine in a directory, tracked

`git pull [remote_name] [branch_name]` - pull down any changes that are in the repository stored in the cloud, in the [branch_name] branch; usually, git will ask you to commit any changes after you do this -- if you're doing simple stuff and not branching, the default will be master

`git add [filename]` - add this file to the next commit (add all changed files with git add *)

`git commit -m "commit message goes here"` - commit the changes you've made; you can make multiple commits before you push! also, try to leave yourself helpful messages when you make commits (if you like to do really verbose commit messages, you can leave off the -m flag and the message, and it will pull up whatever your default editor is on the current machine; if it's vi/vim and you aren't a vi user, :q! is how you quit without saving, and then you can look up how to set nano as your default text editor ;))

`git push [remote_name] [branch_name]` - push any commits you've made up to the repo in the cloud; you can specify any branch, but 'master' is going to be the default

`git branch [branch_name]` - make a branch with name [branch_name]

`git checkout [branch_name]` - switch to branch [branch_name]

`git stash` -- I'm including this because many people love it, but you'll want to look it up to get the usage right. I ... don't really use this. I had to, one time, when I was working on a project with multiple feature branches, and I made my own life pretty hard, honestly

`git remote` -- this is the key to working with multiple repositories; if you have forked someone else's repository, and you're going to want to make pull requests (updates to their code), you'll want to add their remote, in addition to your own cloud-based repo; traditionally, theirs is called "upstream", and yours is called "origin", and whenever you want to make changes, first you want to make sure your local (on your machine) repository is up to date with theirs by issuing the command git pull upstream [whichever branch you're working on]