**GIT**

Setting up Git

**$git config --global user.name “<username>”**

**$git config --global user.email <email id>**

To start a repository

1. First create a directory

**$mkdir <directory name>**

**$cd <directory name>**

1. Then initialize git repository

To initialize a Git repository

**$git init**

To see repository status change

**$git status**

To add a file to the staging area

**$git add <filename. Extension>**

E.g.: git add test.txt

To add all the modified files at once

**$git add --all**

To add all text files in a docs directory

**$git add docs/\*.txt**

To store staged changes with a message

**$git commit -m <”message”>**

Note: staged files are files we have told git that are ready to be committed.

To add many files of the same type

**$git add ‘\*.txt’**

To list all the changes we've committed so far, in the order we committed them.

**$git log**

To push our local *repo* to the GitHub server we'll need to add a remote repository.

**$git remote add <remote-name> <repository URL>**

To push our local changes to our **origin (remote-name)** repo (on GitHub).

**$git push –u <origin> <default local branch name>**

Note: -u tells git to remember the parameters, so that next time we can simply run $git push

To can check for changes on our GitHub repository and pull down any new changes by running

**$git pull <origin> <default local branch name>**

To look at what is different from our last commit

**$git diff**

For most recent commit, we can use HEAD pointer

**$gif diff HEAD**

To see the changes you just staged

**$git diff –staged**

To create branch (copy) of the code

**$git branch <branch name>**

When developers are working on a feature or bug they'll often create a copy (aka. branch) of their code they can make separate commits to. Then when they're done they can merge this branch back into their main master branch.

To switch branches

**$git checkout <branch>**

To remove files

**$git rm ‘<filename. Extension>’**

To merge branches to master

**$git merge <branch name>**

To delete a branch

**$git branch –d <branch name>**