Git – Version Control

# Git is a version control system, found locally on your computer. “Records changes to a file or a set of files over time so that you can recall specific versions later”. Free and open source.

# GitHub is a web-based hosting service for software development projects that use the Git revision control system.

Source control: allows synchronised changes to files across different sources. Syncs to the repository and ‘pulls’ in the made changes. Could pull a repository from the internet ‘GitHub’ to a local repo on your computer.

When you create a new repository in GitHub, always initialise with a readme file (allows you to clone from the repo straight away). Copy the clone URL (on the right hand side, starting with https:). In the command line, first navigate to the working directory. This is defaulted as C:\Users\emea249, but I want to use the H: drive. To change drives simply type h:, then change directories using cd. So cd Documents, cd Huntington’s Disease, cd GitHub.

Now we’re in the correct directory, clone the newly made repository using the command ‘git clone (paste URL of new repo here)’. This take the repo and clones it, storing it in a new local directory on your computer (with the same repo name you created in GitHub). Use ‘dir’ to check directories. Move to newly created repo using cd (name of repo) and check the git status using the command ‘got status’. If the branch is up-to-date, it should state “nothing to commit, working directory clean”.

In the command line: the term “push” pushes from your local computer to the global GitHub repo, whereas “pull” pulls from GitHub to your computer. Important that you are in the correct directory.

Add a new file to the repository on your computer, e.g. a word document called ‘GitHub.docx’. Now if we check status ‘git status’, it says “nothing added to commit but untracked files present”. Use ‘git add (name of file)’, e.g. git add GitHub.docx. Now if we check git status, it says “new file: GitHub.docx”. Now type ‘git commit –m “(type message here enclosed by “”)”” ‘. This locks the file into the repository contained on your machine/computer, but it is not yet synchronised to the global GitHub repo. To synchronise with GitHub, type ‘git push’. This pushes anything on our machine and pushes it up to github.com. You will need to type in your username ‘emily5’ and password ‘usual 4 numbered lowercase’. Check the file has been committed online.

If somebody else cloned my repository to their local computer, they could clone the new file using ‘git pull’. This pulls down from GitHub to their computer, however, it only works if they’re in the correct repo/ working directory.

If we make changes to the file and then look at ‘git status’, we can see that the file has been modified. To update the new file, and push it to the global repo: first type ‘git commit –m “instructions on pushing modified files”’, then ‘git push’. Check the file is now on GitHub.com.

To clone a pre-made repository from GutHub: i.e. clone the repository for the Statistical Inference John Hopkins coursera course. First create a repo on GitHub, termed Statistical-Inference-JH and initialise with a readme file. Then clone the repo to my computer as before, so we have a new directory called ‘Statistical-Inference-JH’ under within the ‘GitHub’ directory. Then type ‘git clone https://github.com/bcaffo/courses.git’ (which is the URL for the entire JH repo). These files will now be saved locally to my computer.

# >add >commit >push

# Workspace Index Local repo Remote repo

git config --list : lists current git configurations

mkdir : makes new directory

cd : changes to said directory

git init : makes a git repo in current (new directory)

ls -a : list hidden files in directory

git status : current git status 'nothing to commit'

gedit : opens text editor 'gedit', write text and save

git status : now says 'name of file', 'use git add...' etc

git add : followed by 'name of file'. Says initial commit...

git commit -m : followed by message of a useful description of what you did (commits file to local repo, not GitHub)

# When we run git commit, git takes everything we have told it to save by using git add and stores a copy permanently inside the special .git directory (hidden but found using ls -a). This permanent copy is called a commit and has a short identifier i.e. f22b25e.

git status : nothing to commit, working directory clean

git log : lists all commits made to a repo in reverse order, includes the commits full identifier.

ls : lists the files in the wd

# Then create a repo in GitHub (sensible to use same repo/directory name). Back in terminal (still in newly created directory).

git remote add origin <http://github.com/emily5/(nameofrepo).git>

git remote -v : checks the command has worked

git push origin master : pushes the changes from the local repo (on computer) to the repo on GitHub

# Have to type in username i.e. emily5 and password

# Make another change to file and save. Then push to remote repo (GitHub), as before. So git add, git commit -m, git push origin master, git remote -v

git add : adds all new files

git add -u : updates files

git add -A : does both of the above (should do these before next commit)