GIT

Distributed Version Control System

Why?

Allows groups of people to work on the same documents (often code) at the same time, and without stepping on each other's toes

Repository

A directory where Git has been initialized to start version controlling your files.

git init

"initialize", or create, a new repository in your current directory

Turning that directory into a git repository

Configure

git config --global user.name "myname"

git config --global user.email "myname@abc.com"

git status

status of your working directory

git add

Move files from untracked to staging area

git reset

Remove a file or files from the staging area to untracked state

git commit

git commit -m 'commit message'

A "commit" is a snapshot of our repository. This way if we ever need to look back at the changes we've made (or if someone else does), we will see a nice timeline of all changes.

git log

A journal that remembers all the changes we've committed so far, in the order we committed them

Remote Repository

GitHub

https://github.com

Connecting GitHub Repo

git remote add origin <url_to_remote_repo>

git push

Saving local changes to remote repo

git push -u origin master

git pull

Pulling changes made from others from Remote

git pull origin master

git branch

Duplicating the code, to do experiment! Without affecting the working code

git branch

Creating a branch

git branch my_exp_branch

git branch

Changing from current branch to new branch

git checkout my_branch

git merge

merging your changes from the new_branch into the master branch

git branch

Deleting a branch

git branch -d my_branch

git rm

Removing files from git

git rm <filename>

Reference

- https://try.github.io/levels/1/challenges/1

- https://git-scm.com/

- https://github.com/

Dharshan

dharshandj@gmail.com

darshan@fsmk.org

dharshan4help@gmail.com

LinkedIn - Dharshan Bharathur

Github – darshand

Telegram - @dhar5han