Git – git are the commands used for source code management. tracking changes in the source code, enabling multiple developers to work together on non-linear development. Open source and free source control management or what's referred to as SCM.

Github – hosting platform where you can collaborate with others on your Git repositories.

1. git config - - global user.name “Victor Santos”
2. git config - - global user.email “victor.santos@kodego.ph”
3. git config - - global init.defaultBranch main
4. git config -h
5. git help config //opens up a manual from your localhost
6. clear
7. cd C:\Users\vic-s\kodegobootcamp\activity_one
8. git init
9. git status

// test track and untrack the files
1. git add file_name
2. git status
3. git rm –cached file_name
4. git status

// to create .gitignore to ignore or hide or exclude files in tracking
1. open a notepad
2. type:
    1. #ignore all text
    2. *.txt
    3. save the file with a filename .gitignore into the same directory

3. git staus

// to track all files
1. git add -A, git add –all, git add .
2. git status

// to commit all the tracked files
1. git commit -m ”message here”
2. git status // to check

//changing or updating the file code, would put it back to WORKING environment
1. Edit a file code and check the changes via git status to locate the modified files
2. git diff   //to compare the previous changes
3. git add file_name //to put the file in STAGING environment

//if not ready to commit
1. git restore - - staged file_name
2. git commit -a -m “updated the files”   //this bypass the Staging

Things to understand and remember the 3 layers are WORKING FILES, STAGING and COMMIT

//deleting a file

1. git rm file_name
2. git status //git recognised about the deleted file
3. git restore file_name //to restore that file
4. file will be back

//renaming a filename

1. git mv “KCC Logo.png” “Primary Logo.png”
2. git commit -m “we changed the file name”

//review all the commits we made

1. git log
2. git log –oneline

//to check the changes inside the commit, dig into the specifics

1. git log -p
2. use keyboard arrow down to scroll
3. to exit the view press Q in the keyboard
4. git help log //to open the manul copy about log command

// to go back to the previous commit

1. git log - -oneline
2. git reset hashtag