Git🡺 version control system

GitHub🡺code hosting service

Hosting platform which host our repositories.

Three Stage Architecture

1. Working Directory--- your current directory
2. Staging Area---next git-commit directory
3. Git directory----all versions code

To set user setup

git config --global user.name "danish"

To display the user

$ git config –list

To check the status

$ Git status

To make current directory as git repo

$git init

To add single file into staging area

$ git add <filename>

To add all files into staging area

$ git add –a

To unstage the file

$ git restore –-staged <<filename>>

To commit files from staging area

$ git commit -m “comment”

To restore the deleted content of files

$ git checkout -- <<filnemae>>

To restore all the files from the last commit

$ git checkout -- f

To check how commits

$ git log

To git log status

$ git log stat

To get git log removed lines

$ git log -p

To align log files in one line

$ git log –pretty=oneLine

To format git log

$ git log –pretty=format:”%h---%an”

To compare staging area with working area

Before staging

$ git diff

After staging

$ git diff –staged

To untrack all the files and delete all the files as well as unlink git rep

$ rm -rf .git

To clone any url

$ git clone <<URL>> “comment”

To make new file in the current directory

$ touch <<filename>>

To ignore error files in the director

Make .getignore in the directory and inside .getignore file enter the file name

$ touch .getignore

$ touch error.log

When you modify error.log file .getignore fill will ignore the error file and won’t give

Any error message.

Also to remove/ignore specific file with the same extension

Put \*.log in getignore

To remove directory

Put <<dirName>>/ in .getignore

To track outer director

Put /<<dirName>>/ in .getignore

To delete any file

$git rm <<filename>>

To rename any file

$ git mv <<filename from>> <<filename To>>

To skip the staging are

$ git -a -m “<<string>>”

File Status lifecycle

Untracked---initially or after making it git repo

Unmodified --- after staging and tracking the file

Modified ---After changing content of files ( if you modify any file then you have to stage it again)

Staged ---After staging ,after modification

To connect to Remote Repositories

$ git remote add origin git@github.com:danishriyaz/gitTutorialDemo.git

How to create SSH key

<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

Enter this in Github shell

$ ssh-keygen -t ed25519 -C [your\_email@example.com](mailto:your_email@example.com)

Press enter enter,

To start ssh agent enter this command

eval "$(ssh-agent -s)"

add the identity

ssh-add ~/.ssh/id\_ed25519

enter this command to generate ssh key

tail < ~/.ssh/id\_ed25519.pub

after that copy the SSH id and paste in git repo in setting-SSH -ADD SSH