**Git**

Git is a source control management(SCM) tool

* Version control tool (VCT)
* Track the changes made to your source code (file/folder)
* Revert back the changes
* Maintain the history of changes
* It is also called as distributed version control system
* Collabration between the team members
* Branching
* We work locally and well as remotely
* CI & CD

The popular git servers:

-Github

-Gitlab

-Bitbucket

In Git all the information/code/file/folder is stored in a place called Repository.

It also has the info about the changes that has made. (.git)

1.Download git application

>git version

You can use git by GUI(graphical user interface) and CLI(command line interface)

To work with CLI

* Command prompt
* Git bash
* Terminal
* Power shell

On-time configuration

-user name

-user email id

> git config

>git config - -global - -list

>git config --global user.name "user\_name"

>git config --global user.email "user\_mail\_id"

3 locations

Working area – Staging area – git repository

Two popular ways to get started using git

1.Creating a new repository and sharing it with others for collaboration

a.Initialize a repository (.git folder)

> git init (initialize the folder as git repo)

>git init foldername

>dir /ls 🡪 list the file and folders

>git status

b.Add the files to the staging area

>touch filename.extension

>ls

> cat filename.extension 🡪 to check the code in the file

> mv old\_filename new\_filename

>git add specific\_filename

>git add \*.html

>git add . (add all the files to staging area)

>git rm - -cached <filename> 🡪 to remove file from staging area

>git reset

>git reset <filename/foldername>

>git clean -fd 🡪 remove untracked file from working area (delete)

c.Commit the files

Note: Every commit will have a unique Id

>git commit -m “commit message”

>git log 🡪 to get the commit messages

>git log - - oneline

>git log --stat

>git commit - - amend -m “updated message”

d.Create a remote repo in gitlab

>git remote -v : list configured remote repos

>git remote add origin repourl (to add a new remote repos)

>git remote set-url location\_name <https://newlocation>

>git remote rm location\_name

e.To send the changes/add the code to remote repo

>git push location\_name master

f.If you do further changes in the repo

Repeat b,c,e

2.Using an existing shared remote repository

a.clone the repo

> git clone repo url 🡪 only once

>git pull location\_name master

b.To do the further changes in the repo, following steps need to be repeated

Repeat b,c,e

>git init

------------------

>git add .

>git commit -m “message”

>git push location master

>git pull location master

Branching

-------------

Default branch – main branch – master

>git branch : shows the current branch

>git branch branch\_name 🡪 create the new branch

>git checkout branch\_name 🡪 to switch to new branch

>git checkout -b branch\_name 🡪 to create a branch and switch to branch

>git merge branch\_name 🡪 merge the branch

>git branch -d branch\_name 🡪 delete the branch in the local

>git push location\_name - -delete branch\_name 🡪 delete the branch in the remote

.gitignore

To ignore the files

Forking

------------

SSH key (Secure shell)

-------------------------------

>ssh-keygen