**Git**

**Git: is a source code hosting platform for version control and collaboration. It lets programmer to work together on projects anywhere. It is designed for tracking changes in source code during software development.**

**GitHub:** acts as remote storage for Git repositories and provides a simple and neat way for people to collaborate and contribute to development projects.

**Branching:** is the way to work on different versions of repository at one time. By default, your repository has one branch named *master* which is considered to be the definitive branch. When we create a branch off the *master* branch. We are making a copy or snapshot of the master as it was at that point in time,

Setting up >

* Download and Install latest version of Git.
* Set username in Git > open Git bash

Git config –global user.name “ username”

* To set email

Git config –global user.email “youremail@yahoo.com”

* Next: authenticating with GitHub from Git (can be done by HTTPS OR SSH) open Git Bash, use below commands

ssh -keygen -t rsa -b 4096 -C “youremail@yahoo.com”

* Got to GitHub and the ssh key from rsa\_pub file

**Git basic commands**

**git init** ---->To initialize a local repository, it will create a .git folder in the local machine.

**git add < file name>** ----> To add files and folders to staging area.

**git add .** ----> To add all files and folders to staging area.

**git status** ----> To check the status of the staging area.

**git commit -m “ must add comments”** ----> to commit changes to Index.

**git remote** ----> to check the origin/remote repository.

**git remote add origin < https/------- .git>** ----> to link the local repository to remote repository.

**git pull origin master** ----> To pull code from remote repository to local repository.

**git push -u origin master** ----> To push code from local repository to remote repository

**touch <filename>** ----> to create a tile in local repository.

**git –version** ----> to find the version and check whether properly installed.

**Use escape and :wq** ----> to exit from commit screen.

**Branching**

**git branch <branchname>** ----> to create a branch

**git checkout <branchname>** ----> to move to the branch

**git checkout -b <branchname>** ---🡪 creates branch branchname

git push -u origin master “comments” ---🡪 to push to origin master

**git merge <branchname>** ----> to merge branch with master.\*we must be master branch.

**Very important git command to resolve push:**

**Git pull origin master – - allow-unrelated-histories** 🡪 if it rejects push

**Rebase**

**Git rebase <branchname>** ----> similar to merge but it is pull + merge.

**Global Configuration**

**git config - - global user.name “ your name”**

**git config - - global user.email “your email “**

**ssh-keygen** ----> to generate ssh key.