**Git** : It is a version control system. It is used to locally track changes in your project/folder and push and pull changes from remote repositories like GitHub, BitBucket,GitLab etc.

**GitLab** : Services that allow to host your project on a remote repo & to have additional features to help in SDLC and CI&CD.

e.g Managing,Sharing,Wiki,Bug Tracking,CI&CD

**Steps** :

Step 1: Download git from [https://git-scm.com/](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbjRkTjRZLW5XOWZJVU5SZ294aFBGWU53S1g3QXxBQ3Jtc0tub05ldVZmNE8yTWxzX3JBSUxaTzZFZkNRdzBHamZWM0JFRG13LUJFNWdOZlQ5V2JKNl9udzhUUFdqd0JReFlIRTdZNHlUZHd5LUFrTEI1eURVRWY0bDVfY05rcG9BNTVTTkhrNHVqNENGMVlLX1dKMA&q=https%3A%2F%2Fgit-scm.com%2F) and install git

Step 2: Check git is installed on your system

git --version

Step 3: Run following git commands

git config --global user.name “xxxx”

git config --global user.name

git config --global user.email “xxxx”

git config --global user.email

git config --global --list

Step 4: Create a demo project/folder & add to git

Step 5: Goto cmd OR terminal OR git bash CD to the location of the folder and run following commands

git init : git initialization

git status : status of the current folder

git add . : to add untracked changes

git commit -m “msg” : committing with the commit message to local repo

git push -u “url” master : pushing the changes to remote repo

Step 6: Check project (files) added on GitLab

**Fork** : A fork is a copy of the project.Forking a project or repo allows you to make changes without affecting the original project.

Step 1: Login to GitLab and goto your project

Step 2: Click on Fork button

If you get a message No available namespaces to fork the project.

After making changes to the fork project you can merge the changes to the original project using Merge Request

**Secured Shell (SSH)** :

It is used for authentication. By getting the SSH key, we can connect to our gitlab server without using username and password each time.

Step 1: Run command

ssh-keygen

On Mac - run command on terminal

On Windows - use putty or git bash

Step 2: Login to GitLab

Gotot account ＞ Settings ＞ SSH Keys

Step 3: Copy contents of id\_rsa.pub and Add Key

Step 4: Verify SSH key is added

**Gitlab Runner :**

* used in GitLab CI
* Open-source continuous integration service included with GitLab
* used to run jobs & send results back to GitLab

Step 1: Install GitLab Runner

<https://docs.gitlab.com/runner/>

[https://docs.gitlab.com/runner/instal...](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbmlGMjIyN1JqN3pyaGZSc21NZWJpOE1MMTZ6QXxBQ3Jtc0tsb2hfSmkyaGtYeE5zWjh3TnBaLUxMSTFaUmlqQ2ZLaDdOMG8xckdWN1pOdENOTnhPc3pHMWZabDlIR0o3a2RSQkcwZHpxZ2pZNXE5ZjJ6OFlsbVJyd3VXMjhqd3gxUGl3Mlc2bnUtd09QRF80MnJWWQ&q=https%3A%2F%2Fdocs.gitlab.com%2Frunner%2Finstall%2Findex.html)

gitlab-runner.exe install

gitlab-runner --version

Step 2: Register GitLab Runner (process to bind runner with gitlab instance) <https://docs.gitlab.com/runner/regist...> gitlab-runner.exe register

Step 3: Start GitLab Runner gitlab-runner.exe start

Step 4: Check runner is started in the project

