

## WHAT IS A GIT?

Formal:

Version Control system that tracks changes in computer files and also coordinates work on files with multiple people.

It is distributed. Git does not necessarily rely on central server to store all versions of a project's file.

Rather users clone copy of repo. We can host this clone on Github.

## DIFFERENCE BETWEEN GIT AND GITHUB

Git is an open-source, version control tool created in 2005 by Linux developers. GitHub is a company founded in 2008 that makes tools which integrate with git.

You do not need GitHub to use git, but you cannot use GitHub without using git.

Git is a software. GitHub is a service.

Git is a command-line tool. GitHub is a graphical user interface

Git is installed locally on the system. GitHub is hosted on the web.

Git is maintained by linux. GitHub is maintained by microsoft.

***GitHub is focused on centralized source code hosting. In nutshell, GitHub is a hosting service for Git repositories.***

## WHY USE GIT:

UNDO MISTAKES

BRANCHING

DISTRIBUTED

COMMUNITY

Can make various versions of app/web.

## LAYERS OF GIT:

There are 3 layers inside git namely:

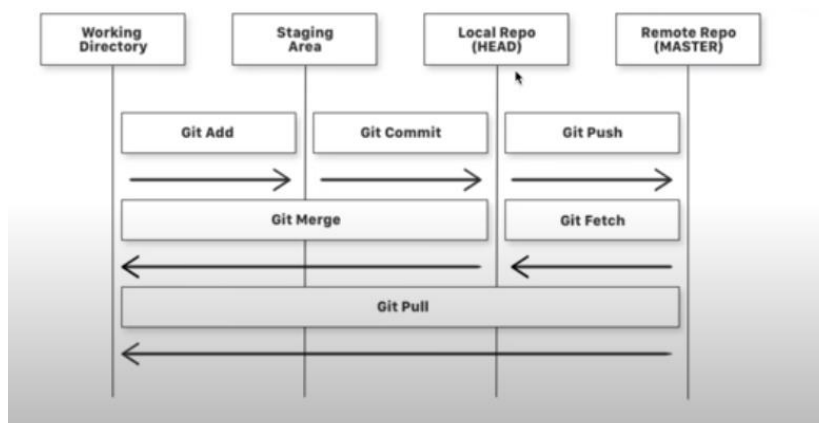
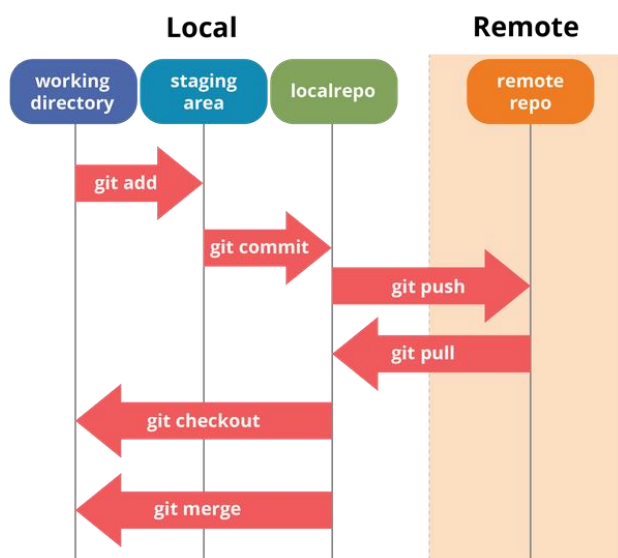
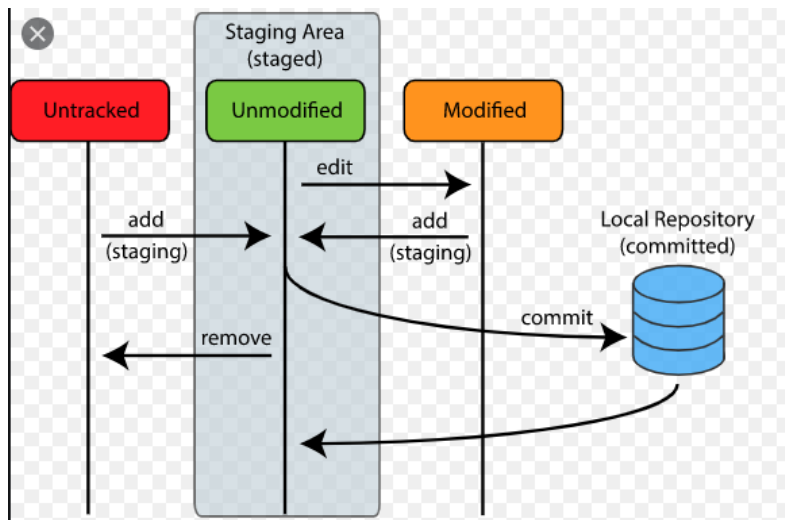
***Untracked:*** This is working directory where all the files are placed as usual.

***Staging area*** is also known as ready state area. To reach from untracked to unmodified area we need to add files from untracked area to first staging area to make the files ready for commit. It is done using ***git add*** command.

***Local Repository*** is known as local repo which has been initialised by git and all the contents of working directory are placed in git as local repo. To reach from staging to local repo we use ***git commit*** command.

***Remote Repository:*** This is where all the working directories are stored online as we push data from local repo online. This is done using ***git push*** command

## DIAGRAMS OF LAYERS OF GIT ALONG WITH THE COMMANDS:



## **GIT COMMANDS:**

- i. **Git init:** It is used to initialise a git repository for a new or existing project or working directory.
- ii. **Git add <filename>:** This command is used to add one or more files to staging area as shown in layers.
- iii. **Git status:** It is used to check the updates of the files along with the changes made along with the files that are yet required to be added or committed.
- iv. **Git commit -m "message":** This command moves the files from staging area to the head of the local repository.
- v. **Git rm –cached <filename>:** This command is used to remove files from the staging area that is it is used to unstage the files.
- vi. **Git branch <branch\_name>:** This command is used to create a new branch with name as branch\_name.
- vii. **Git checkout <branch\_name>:** This is the command through which we can switch from one branch to other
- viii. **Git merge <branch\_name>:** It merges branch into the current branch.

## **HANDS-ON:**

CREATE REPO ON GITHUB AND ON FOLDERS

Index.html

Tags:

H1,h2,p,img,textarea, input, label etc...Hosting.