# What is Git ? what are the basic commands

## About Git

- Git is a distributed version control system designed for tracking changes in source code during software development.
- Created by Linus Torvalds in 2005 and has maintained by Junio Harmano since then
- allows multiple developers to collaborate on projects efficiently,and also tracking who made the changes
- It will Manage projects with Repositories, and Clone a project to work on a local copy
- maintaining a history of changes and enabling seamless integration of modifications.

## Basic Commands of Git

- **git init**: Initializes a new Git repository in the current directory.
- **git clone [url]**: Copies an existing Git repository from a specified URL to the local machine.
- **git add [file]**: Adds specified file(s) to the staging area to prepare for a commit.
- **git commit -m "[message]"**: Records changes staged in the index with a descriptive message.
- **git status**: Displays the current status of the repository, including tracked/untracked files and changes to be committed.

- **git pull**: Fetches changes from a remote repository and merges them into the current branch.
- **git push**: Uploads local changes to a remote repository.
- **git branch**: Lists all existing branches in the repository.
- **git checkout [branch]**: Switches to the specified branch.
- **git merge [branch]**: Combines changes from the specified branch into the current branch.
- **git log**: Displays a chronological list of commits in the repository.
- **git remote -v**: Lists the URLs of the remote repositories associated with the local repository.
- **git diff**: Shows the differences between the working directory, staging area, and the last commit.
- **git fetch**: Retrieves updates from a remote repository without merging them into the local branch.
- **git reset [file]**: Removes changes to a file from the staging area, but keeps the changes in the working directory.
- **git stash**: temporarily shelves (or stashes) changes you've made to your working copy so you can work on something else, and then come back and re-apply them later on.
- **git cherrypick**: is the act of picking a commit from a branch and applying it to another.