

LET'S GO GIT

Chai (☺) pe Charcha

WHAT IS GIT?

Git is a **distributed** VCS (Version control system) that helps us manage versions of files.

THE FEATURES

FEATURES?

1. Version Control
2. Branching and Merging
3. Distributed System
4. Snapshots
5. Speed and Performance

BRANCH?

BRANCHING

A lightweight, movable pointer to a specific commit.

HEAD: A pointer pointing to the latest commit of current branch.

^(Caret): Refers to the first parent of the commit

- HEAD[^] or HEAD^{^1}: first parent of HEAD commit
- HEAD^{^2}: second parent of HEAD commit (only for merge)
- HEAD^{^^}: grandparent commit (first parent's first parent)

BRANCHING

~(Tilde): Used to navigate a specified number of commits in commit history following the first parent.

- HEAD~: first parent of HEAD commit (same as HEAD^)
- HEAD~2: grandparent commit (same as HEAD^^)
- HEAD~3: great-grandparent commit

MERGE V/S REBASE

MERGE AND REBASE

Merge: Used to combine branched by creating a new merge commit.

- Fast-forward: current branch is behind the target and has no unique commits, just move pointer forward to target
- Three-way: create a new commit that combines changes from both branches
- Syntax: `git merge <branch-name>`

Rebase: Used to move/combine a series of commits to a new base commit. Rewrites history!

GIT PULL

PULL

Used to fetch changes from remote repo and integrate them to your current branch.

- A combination of “git fetch” and “git merge” (default).
- “git pull -rebase”: fetches changes from remote, rebase the current branch on top of the fetched branch.
- Other strategies:
 - ours: current first
 - recursive (default)
 - octopus
 - subtree

REMOTES?

REMOTES?

Remote is a reference to a repository (typically hosted on a network)

- origin is the default name of primary remote repository
- ``git remote add <remote-name> <repo-url>``

RESOLVING CONFLICTS

CONFLICTS

- Check for `<<<<<<`, `====`, `>>>>>>`

- <<<<<< HEAD

Your changes here

====

Incoming changes here

>>>>> branch-name

CHERRYPICK

CHERRY-PICK

- Allows to apply specific commit from one branch to another
- Syntax: `git cherry-pick <commit-hash>`
- `git cherry-pick <commitA>^..<commitB>` (inclusive from A to B)
- `git cherry-pick <commit1> <commit2> <commit3>`

GIT V/S GITHUB

GITHUB V/S GIT?

Github is a web-based platform that uses Git as a VCS and provides a User Interface for version control and other features.

Other tools:

- Gitlab
- Bitbucket

GIT RESET

GIT RESET?

Used to undo changes in your working directory and staging area

- `--soft`: moves HEAD to a specified commit, but does not touch the staging area. Keeps changes staged after the reset point
- `--mixed` (default): resets the staging area to match the commit. Changes in working directory are kept unstaged
- `--hard`: resets both staging and working directory

GIT STASH

GIT STASH?

Used to temporarily save changes in your working directory and staging area, without committing them.

- `git stash`
- `git stash pop`
- `git stash apply: `git stash apply stash@{1}``
- `git stash list`

ANYTHING ELSE?