What is Git?

Git is a distributed version control system (DVCS) used to track changes in files, especially in software development. It allows multiple developers to collaborate on a project, keep track of changes, and revert to previous versions if needed.

Git was created by Linus Torvalds in 2005 for Linux kernel development. It is widely used for managing source code efficiently.

Key Features of Git

Distributed System – Every developer has a complete copy of the project, not just the latest version.

Version Control – Tracks changes, making it easy to revert to previous versions.

Branching & Merging – Developers can work on separate branches and merge changes when ready.

Collaboration – Teams can work on the same project using platforms like GitHub, GitLab, and Bitbucket.

Security – Uses cryptographic hash functions (SHA-1) to ensure data integrity.

Speed & Efficiency – Fast performance even with large projects.

Git: Core Concepts & Command Lines

Git is a distributed version control system that helps developers track changes in source code during software development. It allows multiple people to collaborate on a project efficiently.

Core Concepts of Git

Repository (Repo): A directory that contains all the project files and the complete history of changes.

Commit: A snapshot of changes made to the files in the repository.

Branch: A separate line of development that allows working on features independently.

Merge: Combining changes from one branch into another.

Remote: A version of your repository stored on a server (e.g., GitHub, GitLab).

Clone: Creating a copy of a remote repository on your local machine.

Pull & Push: Fetching changes from a remote repo (pull) and sending local changes to the remote (push).

Staging Area: A place where changes are reviewed before committing them.

Common Git Commands

Basic Commands

git init → Initialize a new Git repository.

git clone <repo\_url> → Clone a remote repository.

git status → Check the current status of files.

git add <file> → Add a file to the staging area.

git commit -m "message" → Commit changes with a message.

git log → View commit history.

Branching & Merging

git branch → List branches.

git branch <branch\_name> → Create a new branch.

git checkout <branch\_name> → Switch to a branch.

git merge <branch\_name> → Merge a branch into the current branch.

Remote Repositories

git remote add origin <repo\_url> → Connect local repo to a remote repo.

git push origin <branch> → Push changes to the remote repository.

git pull origin <branch> → Fetch and merge changes from the remote repo.

Undoing Changes

git reset --hard <commit\_hash> → Reset to a specific commit and discard all changes.

git revert <commit\_hash> → Revert a commit without deleting history.