Introduction to Git

What is Git?

- Git is a distributed version control system that helps track changes in source code during software development.

Basic Features of Git

- **Version Control:** Track changes in files over time.
- **Branching and Merging:** Develop new features in isolated branches.
- **Distributed System:** Each developer has a complete repository copy.
- **Staging Area:** Intermediate area for changes before committing.

- **Commit History:** Record of all changes with unique IDs.
- **Collaboration:** Enables multiple developers to work on a project.
- **Backup:** Every developer has a backup of the repository.

Setting up Git

- Command: git config --global user.name "Your Name"
- Command: git config --global user.email “ab@gmail.com"

Initializing a Git Repository

- Command: git init
- Initialize a new Git repository.

Cloning a Repository

- Command: git clone [URL]
- Create a copy of an existing repository.

Adding Files to the Staging Area

- Command: git add [file]
- Add file changes to the staging area.

- Command: git add --all
- Add all files.

Committing Changes

- Command: git commit -m 'message’
- Record changes to the repository.

Checking status

- Command: git status
- Display the state of the working directory.

Viewing Commit History

- Command: git log
- Show commit history.

Revert

- Command: git revert <commit-hash>
- To take a previous commit and add it as a new commit, keeping the log intact.

Creating a New Branch

- Command: git branch<branch-name>
- List, create, or delete branches.

Switching Branches

- Command: git checkout [branch]
- Switch between branches.

Merging Branches

- Command: git checkout <branch to merge into>
- Command: git merge <branch to merge from>
- Merge branches

Pulling Changes from a Remote Repository

- Command: git pull
- Fetch and merge changes from a remote repository.

Pushing Changes to a Remote Repository

- Command: git push
- Push local changes to a remote repository.

Fetch

- Command: git fetch
- Downloads the updates but does not merge them.

Stashing Changes

- Command: git stash
- Temporarily saves changes that you don’t want to commit yet.

- Viewing Stashes
- Command: git stash list

- Applies the most recent stash back to directory.
- Command: git stash apply

Removing Files from Git

- Command: git rm --cached <file-name>
- Removing Files from Git without deleting locally.

Rebase

- git rebase <base-branch>
- This command used to reapply your changes (commits) on top of another base commit.

Tag

- git tag --a
- Used to form annotated tags.

Resolve a conflict in Git

- **Identify Conflicting Files:** Locate files that have conflicts due to changes from different branches**.**

- **Resolve Conflicts:** Edit the files to reconcile conflicting changes and prevent future conflicts.

- **Stage Resolved Files:** Use git add to move the resolved files to the staging area.

- **Commit the Changes:** Finalize the process by committing the updated files with git commit.