git revert: Creates a new commit that undoes the changes made in a previous commit.

git cherry-pick: Applies a single commit from one branch to another branch, allowing developers to selectively apply changes made in one branch to another branch.

The **git squash** command is used to combine multiple commits into a single commit.

git fetch: Retrieves changes made to a remote repository without merging them with the local repository.

git reset: Resets the Git repository to a previous commit, discarding any changes made after that commit.

git stash: Temporarily saves changes that are not ready to be committed, allowing developers to switch to another branch or commit without losing their work.

git rebase: Changes the base commit of a branch, allowing developers to integrate changes made in one branch into another branch.

git init: Initializes a new Git repository in a directory, creating an empty repository with the necessary files to track changes to code.

git clone: Copies an existing Git repository to a local machine or remote server, creating a copy of the repository that can be modified and pushed back to the original repository.

git add: Adds changes made to a file to the staging area, preparing them to be committed.

git commit: Creates a new snapshot of the changes made to the files in the staging area, saving the changes to the local repository along with a commit message describing the changes.

git push: Sends the committed changes to a remote repository, updating the codebase for all developers working on the same project.

git pull: Retrieves changes made to a remote repository and merges them with the local repository, updating the codebase with the latest changes made by other developers.

git branch: Creates a new branch in the Git repository, allowing developers to work on new features or experiments without affecting the main codebase.

git checkout: Switches between different branches or commits in the Git repository, allowing developers to work on different parts of the codebase without affecting the current state of the repository.

git merge: Combines changes made in one branch with changes made in another branch, integrating the changes into a single branch.

git log: Displays a history of commits made in the Git repository, including commit messages, author information, and timestamps.