

## 1. Basic Git Commands

Command	Description
<code>git init</code>	Initializes a new Git repository in the current directory.
<code>git clone &lt;repository&gt;</code>	Creates a copy of an existing repository from a remote source.
<code>git status</code>	Shows the current state of the working directory and staging area.
<code>git add &lt;file&gt;</code>	Stages a file to be committed.
<code>git add .</code>	Stages all changes in the current directory.
<code>git commit -m "message"</code>	Commits the staged changes with a message.
<code>git commit -a -m "message"</code>	Stages and commits all modified files with a message.

## 2. Branching and Merging

Command	Description
<code>git branch</code>	Lists all branches in the repository.
<code>git branch &lt;branch_name&gt;</code>	Creates a new branch.
<code>git checkout &lt;branch_name&gt;</code>	Switches to the specified branch.
<code>git checkout -b &lt;branch_name&gt;</code>	Creates and switches to a new branch.
<code>git merge &lt;branch_name&gt;</code>	Merges the specified branch into the current branch.
<code>git branch -d &lt;branch_name&gt;</code>	Deletes the specified branch.

## 3. Remote Repositories

Command	Description
<code>git remote -v</code>	Lists remote repositories associated with the local repo.
<code>git remote add origin &lt;URL&gt;</code>	Adds a new remote repository.
<code>git push origin &lt;branch_name&gt;</code>	Pushes the branch to the remote repository.
<code>git push -u origin &lt;branch_name&gt;</code>	Pushes and sets the upstream branch for future pushes.
<code>git pull</code>	Fetches and integrates changes from the remote repository.
<code>git fetch</code>	Fetches changes from the remote without merging.

## 4. Viewing History

Command	Description
<code>git log</code>	Shows the commit history.
<code>git log --oneline</code>	Displays a simplified, one-line commit history.
<code>git log --graph</code>	Shows a graph of the commit history.
<code>git show &lt;commit_hash&gt;</code>	Shows details of a specific commit.
<code>git diff</code>	Displays changes between the working directory and the index.
<code>git diff &lt;branch1&gt;..&lt;branch2&gt;</code>	Shows differences between two branches.

## 5. Undoing Changes

Command	Description
<code>git reset &lt;file&gt;</code>	Unstages a file from the staging area.
<code>git reset --hard &lt;commit&gt;</code>	Resets to a specific commit and discards all changes.
<code>git reset --soft &lt;commit&gt;</code>	Resets to a specific commit but keeps changes staged.
<code>git checkout -- &lt;file&gt;</code>	Discards changes in the working directory for a file.
<code>git revert &lt;commit&gt;</code>	Reverts a specific commit.

## 6. Stashing

Command	Description
<code>git stash</code>	Saves changes that are not ready to be committed.
<code>git stash list</code>	Lists all stashed changes.
<code>git stash apply</code>	Applies the most recent stash without deleting it.
<code>git stash drop</code>	Removes a specific stash from the list.

## 7. Tagging

Command	Description
<code>git tag &lt;tag_name&gt;</code>	Creates a new tag.
<code>git tag -a &lt;tag_name&gt; -m "message"</code>	Creates an annotated tag with a message.
<code>git push origin &lt;tag_name&gt;</code>	Pushes the tag to the remote repository.

## 8. Collaboration and Workflow

Command	Description
<code>git rebase &lt;branch_name&gt;</code>	Re-applies commits on top of another base branch.
<code>git cherry-pick &lt;commit&gt;</code>	Applies a specific commit from another branch.
<code>git blame &lt;file&gt;</code>	Shows who made each change in a file.

## 9. Git Configuration

Command	Description
<code>git config --global user.name "Your Name"</code>	Sets the global username.
<code>git config --global user.email "you@example.com"</code>	Sets the global email.
<code>git config --list</code>	Lists all Git configuration settings.