



# Basic Git Commands

Command	Description
<code>git init</code>	To initialize git repository
<code>git add file.txt</code>	To add file.txt in Staging area
<code>git add .</code>	To add all files in Staging area
<code>git status</code>	To know the current status of our git
<code>git commit -m "Initial Commit"</code>	To commit changes in git repository
<code>git commit</code>	To write long description in Code Editor
<code>git commit -am "Updating the file"</code>	To directly Commit code without writing <code>git add</code> command
<code>git ls-files</code>	To get list of all files which are in Staging area
<code>git rm file.txt</code>	To remove file.txt from local & staging area both
<code>git rm --cached -r bin/</code>	remove bin directory all files from the staging area
<code>git diff</code>	return the difference between working directory & staging area code
<code>git difftool</code>	open the difference between working directory & staging area code in diff-tool
<code>git difftool --staged</code>	open the difference between staging area & commit code in diff-tool
<code>git status -s</code>	return short status
<code>git log</code>	return the list of all commits with all details
<code>git log --oneline</code>	return the list of all commits with id & message only
<code>git log --oneline --reverse</code>	same as above command but in reverse order (old to new)
<code>git restore --staged Chapter1.txt</code>	unstage the Chapter1.txt file but changes in the working directory will stay the same
<code>git restore Chapter1.txt</code>	restore working area file with staging area file (all working area changes in Chapter1 file will go)

Command	Description
<code>git restore .</code>	restore all working area file with staging area files
<code>git clean -fd</code>	remove all untracked files
<code>git restore --source=89b38b7 Chapter1.txt</code>	restore the Chapter1.txt file from this Commit "89b38b7"