

# **Source Code Lab**

**Name: Nithyashree P**

**Branch: B.Tech**

**Section: CSE**

**Course: Source Code Lab**

**SEN: A86605223007**

**Submitted To: Dr. Monit Kapoor**

**Date of Submission: 06 June 2025**



## INDEX

SI	Topic	Date	Pg No
1	<b>Git &amp; GitHub Fundamentals-</b> Install Git & GitHub- Basic Linux Commands	<b>31-01-2025</b>	
2	<b>Committing &amp; Viewing History-</b> Understanding Commits- Viewing Git Logs	<b>14-02-2025</b>	
3	<b>Branching &amp; Version Comparison-</b> Creating & Managing Branches- Using git diff	<b>07-03-2025</b>	
4	<b>Remote Repositories-</b> Pushing Local Repos to GitHub	<b>16-05-2025</b>	
5	<b>Collaboration &amp; Conflict Resolution-</b> Inducing & Resolving Conflicts- Opening & Closing Pull Requests	<b>21-05-2025</b>	

# 1. Git & GitHub Fundamentals

## Basic Linux Commands

- `ls` – List directory contents
  - `ls -ah` – List all files including hidden ones
  - `pwd` – Print working directory
  - `cd ..` – Move to the parent directory
  - `mkdir` – Create a new directory
  - `cd <folder>` – Change into a directory
- 

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~  
$ cd Desktop
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop  
$ dir  
May16  calculator  example  project
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop  
$ cd calculator
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)  
$ ls  
calculator_code.py
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)  
$ ls -ah  
./  ../  .git/  calculator_code.py
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)  
$ pwd  
/c/Users/Admin/Desktop/calculator
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)  
$ mkdir Cache
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)  
$ ls  
Cache/  calculator_code.py
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)  
$
```

## File Operations

- vi <filename> – Open file in vi editor
- Press i to insert
- Press Esc to exit insert mode
- Use :wq to save and quit
- cat <filename> – Display contents of a file
- ls -l – Show file permissions and details
- history – Show command history
- clear – Clear the terminal screen

```
MINGW64:/c/Users/Admin/Desktop/calculator

Admin@DESKTOP-3IU8VC8 MINGW64 ~
$ cd Desktop

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop
$ dir
May16  calculator  example  project

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop
$ cd calculator

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)
$ ls
calculator_code.py

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)
$ ls -ah
./  ../  .git/  calculator_code.py

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)
$ pwd
/c/Users/Admin/Desktop/calculator

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)
$ mkdir Cache

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)
$ ls
Cache/  calculator_code.py

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/calculator (master)
$ vi newfile
```

```
this is my first line
~
~
~
~
~
~
~
~
```

## Git Initialization

- `git init` – Initialize a Git repository
- `git status` – Show the working tree status
- `git add <filename>` – Stage a file for commit
- `git commit -m "message"` – Commit changes with a message
- Configure Git with your email and username

MINGW64:/c/Users/Admin/Desktop/labfile

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop
$ cd labfile
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile
$ vi F1
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile
$ git init
Initialized empty Git repository in C:/Users/Admin/Desktop/labfile/.git/
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git status
On branch master
```

No commits yet

```
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  F1
```

nothing added to commit but untracked files present (use "git add" to track)

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git add .
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git status
On branch master
```

No commits yet

```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
  new file:   F1
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ gti commit -m"First commit"
bash: gti: command not found
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git commit -m"First commit"
[master (root-commit) 637d896] First commit
 1 file changed, 1 insertion(+)
 create mode 100644 F1
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ |
```

## 2. Committing & Viewing History

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git log --oneline
9fe8ebe (HEAD -> master) Second commit
637d896 First commit
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git diff^C
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git diff 637d896 9fe8ebe
```

```
@@ -0,0 +1,3 @@
```

```
+
```

```
+As I said I'll be adding more files.
```

```
+This is the second file I'm adding.
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$
```

### 3.

## Branching & Version Comparison

### 1. Branching:

- \* `git branch [branchname]`: Create a new branch
- \* `git checkout [branchname]`: Switch to a different branch
- \* `git checkout master`: Switch back to the master branch
- \* **Concept:** Any branch and the master branch can have completely different codebases after many commits in their respective branches.
- \* **HEAD:** A special pointer that points to the last commit in the current branch.

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git branch master2

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git checkout master2
Switched to branch 'master2'

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ ls
F1  F2

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ vi F1

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ vi F2

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git add .

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git remote

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git checkout master
M      F1
M      F2
Switched to branch 'master'

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$
```

---



4.

## 2. Comparing Changes:

\* git diff: Show changes between commits, commit and working tree, etc.

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git log --oneline
9fe8ebe (HEAD -> master, master2) Second commit
637d896 First commit

Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$
```

## Push Local Repository to GitHub (Remote)

### Linking Local to Remote Repository:

- \* git remote: Helps connect your local repository to a remote repository
- \* git remote add origin "[address of repo GitHub]": Add a new remote repository (e.g., github.com/kavya4kavya/Repo\_1). origin is a common variable name but can be anything.
- \* git push -u origin master: Push local master branch to the origin remote. The -u flag sets the upstream. This is used when there's only one branch.
- \* git push -u origin [branchname]: Push a specific local branch to the remote.

```
MINGW64:/c/Users/Admin/Desktop/labfile
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git remote add origin https://github.com/shreysherikar/Labfile
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git remote
origin
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 585 bytes | 97.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/shreysherikar/Labfile
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ |
```

## 5.

### Induce Conflicts, Open and Close Pull Requests

#### 1. Git Merging & Conflicts:

- \* git merge: Integrate changes from one branch into another.
- \* Conflicts arise during merging when Git cannot automatically reconcile diverging changes.
- \* Use a merge tool to resolve conflicts. cv
- \* git mergetool: Command to launch the configured merge tool.

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git branch
* master
  master2
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master)
$ git checkout master2
M      F1
M      F2
Switched to branch 'master2'
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ ls ah
ls: cannot access 'ah': No such file or directory
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ ls
F1  F2
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git merge
fatal: No remote for the current branch.
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git merge master2
Already up to date.
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git merge master
Already up to date.
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$ git commit -m "Merged two branches"
[master2 76ea9d0] Merged two branches
 2 files changed, 2 insertions(+)
```

```
Admin@DESKTOP-3IU8VC8 MINGW64 ~/Desktop/labfile (master2)
$
```

---

```
MINGW64/c/Users/kavya/Project_3
ky5hgt4r3ewd5tred
fikumjnyhbgvdf
mvscededdeg

kjfnv kdnkd
kvnkdjnvfk
mk ngjfnbjg

abcdefgd
rtiwhfhdks,m cjnvfk
mertgrtbgfdcew

./hello_LOCAL_1027.txt [dos] (12:12 28/05/2025) 2,1 All
./hello_BASE_1027.txt [dos] (12:12 28/05/2025) 2,1 All
./hello_REMOTE_1027.txt [dos] (12:12 28/05/2025) 2,1 All

<<<<<< HEAD
ky5hgt4r3ewd5tred
fikumjnyhbgvdf
mvscededdeg
~~~~~
abcdefgd
rtiwhfhdks,m cjnvfk
mertgrtbgfdcew
>>>>> test1

hello.txt [dos] (12:12 28/05/2025)
"hello.txt" [dos] 9L, 1378 2,1 All
```

**2. Gitignore:** \*. gitignore file: Specifies intentionally untracked files that Git should ignore.

\* Files listed in .gitignore will not be tracked by Git and will not be mirrored on GitHub.

**3. Pull Requests:** \* Opening a pull request

