

Version Control with Git: Task-Oriented Hands-on Lab

Objectives:

By the end of the session, students will be able to:

- 1. Install and configure Git on Windows**
- 2. Work with local Git repositories**
- 3. Use Git commands to manage source code versions**
- 4. Create and use remote GitHub repositories**
- 5. Simulate team collaboration with Git branches and pull requests**

Task 0: Install Git on Windows

- 1. Visit <https://git-scm.com/download/win>**
- 2. Download and run the installer with default settings**
- 3. Open Command Prompt and verify:**

```
git --version
```

Task 1: Configure Git

```
git config --global user.name "Your Name"
```

```
git config --global user.email "your@email.com"
```

```
git config --list
```

Task 2: Initialize Local Repository

```
mkdir my-git-project
```

```
cd my-git-project
```

```
git init
```

```
echo "Hello Git World" > readme.txt
```

```
git status
```

Task 3: Stage and Commit Changes

```
git add readme.txt  
git commit -m "Initial commit with readme"  
echo "Added second line." >> readme.txt  
git diff
```

Task 4: View History and Revert

```
git log  
git checkout readme.txt
```

Optional:

```
git reset --soft HEAD~1
```

Task 5: Create GitHub Repository

1. Go to <https://github.com>
2. Click `+` > New repository > Name: `my-git-project`
3. Do not initialize with README
4. Create repository and copy the URL

Task 6: Connect Local Repo to GitHub

```
git remote add origin https://github.com/yourname/my-git-project.git  
git branch -M main  
git push -u origin main
```

Task 7: Create and Merge Branches

```
git checkout -b feature1  
echo "Feature 1 added" > feature.txt
```

```
git add feature.txt  
git commit -m "Added feature1"  
git checkout main  
git merge feature1  
git push
```

Task 8: Two-Person Collaboration Workflow

Roles: Student A and Student B

Step 1: Student A

- Creates GitHub repo and pushes code
- Adds Student B as a collaborator via GitHub settings

Step 2: Student B

- Accepts invite

```
git clone https://github.com/studentA/my-git-project.git  
cd my-git-project  
git checkout -b feature-branch  
echo "Code added by Student B" > newfile.txt  
git add newfile.txt  
git commit -m "Student B: added new feature"  
git push origin feature-branch
```

Step 3: Student A

- Reviews and merges pull request on GitHub UI

Step 4: Student A

git pull origin main

Optional Bonus: Both students edit the same file and resolve merge conflicts.

Deliverables:

- **GitHub repo link**
- **Screenshot of git log and pull requests**
- **Code files with individual contributions**