**Lab Exercise 2– Working with Branches in Git**

**Objective**

To understand and practice creating, switching, merging, and managing branches in a Git local repository.

**Prerequisites**

* Git installed on your system (git --version)
* A local Git repository initialized (you can create one with git init myrepo && cd myrepo)

**Lab Steps**

**1. Create a New Repository and Initial Commit**

mkdir git-branch-lab

cd git-branch-lab

git init

echo "Master branch content" > master.txt

git add master.txt

git commit -m "Initial commit: Added master.txt"

You now have a master branch with one commit.

**2. Create a New Branch**

git branch feature-1

git branch

git branch lists all branches. The star (\*) indicates the current branch.

**3. Switch Between Branches**

git checkout feature-1

echo "Feature 1 content" > feature1.txt

git add feature1.txt

git commit -m "Added feature1.txt in feature-1 branch"

You committed changes in feature-1 without affecting master.

**4. Merge a Branch into Master**

git checkout master

git merge feature-1

The changes from feature-1 are merged into master.

**5. Create and Merge Another Branch**

git checkout -b feature-2

echo "Feature 2 content" > feature2.txt

git add feature2.txt

git commit -m "Added feature2.txt in feature-2 branch"

git checkout master

git merge feature-2

This time we used git checkout -b to create and switch to feature-2 in one step.

**7. Delete a Branch**

git branch -d feature-1

git branch -d feature-2

git branch

Deletes merged branches and lists remastering ones.