

6. Collaboration and Remote Repositories

Write the command to merge “feature-branch” into “master” while providing a custom commit message for the merge.

Step 1: Create a new repository in GitHub

- Go to [GitHub](#) → Create a **new repository** (e.g., gitlab6-merge-demo).
- Inside the repo, click “**Add file → Create new file**”, name it `demo.txt`.
- Add some sample content like:
`Hello, this is the main branch content.`
- Click **Commit changes** to save the file in the `main` branch.

Step 2: Clone the Repository to Local System

```
cd Desktop
mkdir gitlabs6
cd gitlabs6
git clone https://github.com/<your-username>/gitlab6-merge-demo.git
cd gitlab6-merge-demo
```

Step 3: Create and switch to a new branch

```
git checkout -b feature-branch
```

Step 4 : Make changes in the feature branch

- Edit the file `demo.txt` and add:
`Hello from feature branch version`

- Then run:

```
git add demo.txt
git commit -m "Added feature branch changes"
```

Step 5 : Push this new branch to GitHub

When you create a new branch locally, Git doesn't automatically link it to the remote repository. So you must push it manually the first time using:

```
git push -u origin feature-branch
```

After running this, check your GitHub repo →

You'll now see your **feature-branch** listed under the **branches** dropdown.

Step 6: Switch back to main branch and make sure it's up-to-date

```
git checkout master  
git pull origin master
```

Step 7: Merge feature branch into main

Now merge the feature-branch into main with a custom commit message:

```
git merge feature-branch -m "Merged feature-branch into main with updates"
```

The changes from feature-branch are now combined into master.

Step 8: Verify the merge

- To view recent commits: **git log - -oneline**
- To check file content: **cat demo.txt**

Step 9: Push merged Main branch to GitHub

```
git push -u origin main
```

Final Verification

- Go to GitHub → open your repository.
- Check the main branch — the changes from feature-branch should now appear.
- You can also check the commit history to see the custom merge message.