**Construct a branch, do some changes in the branch, and merge it with master (or trunk)**

**Part 1: Branching**

**1.1 Create a new branch**

bash

git branch GitNewBranch

**1.2 List all branches (local & remote)**

bash

git branch -a

\* marks the branch you are currently on.

**1.3 Switch to the newly created branch**

bash

git checkout GitNewBranch

**1.4 Create a new file and add content**

bash

echo "This is a file in GitNewBranch" > branchfile.txt

**1.5 Stage the file**

bash

git add branchfile.txt

**1.6 Commit the changes**

bash

git commit -m "Added branchfile.txt in GitNewBranch"

**1.7 Verify status**

bash

git status

**Part 2: Merging**

**2.1 Switch back to master branch**

bash

git checkout master

**2.2 List differences between master and GitNewBranch (CLI)**

bash

git diff master GitNewBranch

**2.3 View visual differences using P4Merge**

bash

git mergetool

Ensure P4Merge is installed and configured as Git’s merge tool.

**2.4 Merge GitNewBranch into master**

bash

git merge GitNewBranch

**2.5 View commit history in graphical form**

bash

git log --oneline --graph --decorate

**2.6 Delete the merged branch**

bash

git branch -d GitNewBranch

**2.7 Verify status after merge**

bash

git status

**Part 3: Creating Branch & Merge Requests in GitLab**

**3.1 Push the new branch to GitLab**

bash

git push origin GitNewBranch

**3.2 Create a Branch Request in GitLab**

* Go to your project in GitLab.
* Navigate to **Repository → Branches**.
* Click **New branch** or locate GitNewBranch.

**3.3 Create a Merge Request in GitLab**

* Go to **Merge Requests** in your GitLab project.
* Click **New Merge Request**.
* Select **Source branch**: GitNewBranch  
  **Target branch**: master
* Click **Compare branches and continue**.
* Add title, description, and click **Create Merge Request**.