**GIT HANDSON-3**

**Checkout my Gitlab:** [**https://gitlab.com/nikithapotluri/GitDemo/-/tree/master?ref\_type=heads**](https://gitlab.com/nikithapotluri/GitDemo/-/tree/master?ref_type=heads)

**1. Branching and Merging in Git**  
Branching allows developers to work separately on features or fixes without affecting the main code (usually the master branch).  
Merging combines changes from one branch (like a feature) into the main branch.

**2. Creating a Branch in GitLab**

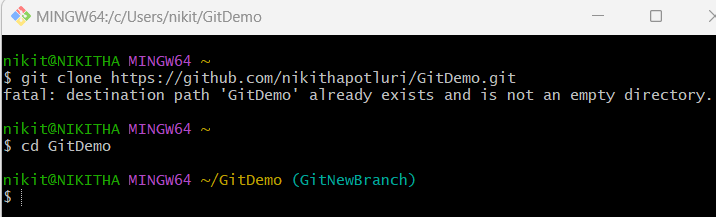
* Login and open your repo.
* Go to the **Branches** section.
* Click **New Branch**.
* Enter a name (e.g., fix-typo) and base branch (usually master).
* Click **Create Branch**.

**3. Creating a Merge Request in GitLab**  
A merge request lets you combine your changes into another branch. Steps:

* Push your changes. GitLab prompts for a merge request. Or go to **Merge Requests** → **New Merge Request**.
* Select source and target branches.
* Add a title, description, and (optionally) reviewers.
* Click **Create Merge Request**.

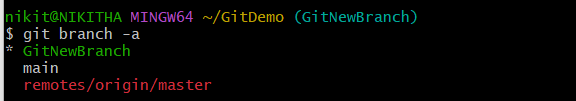
**LAB STEPS:  
  
PART 1: Branching in Git**

**Step 1:** Clone your GitHub repository to your local system

****

**Step 2:** Create a new branch called GitNewBranch

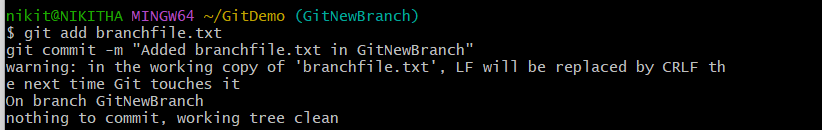
**  
  
Step 3:** List all branches (local and remote)



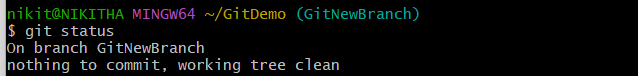
**Step 4:** Create a new file in the branch

****

**Step 5:** Stage and commit the file

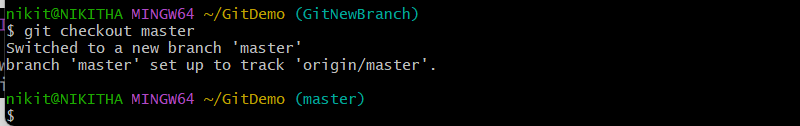
****

**Step 6:** Check status

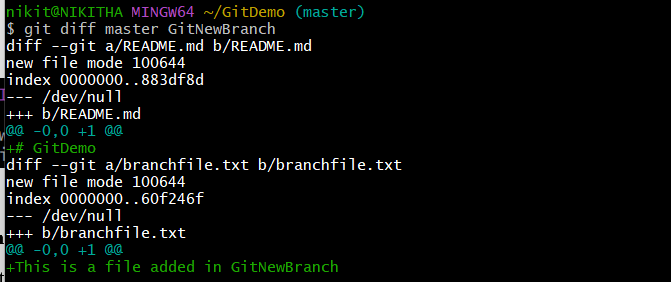
****

**PART 2: Merging in Git**

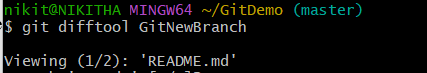
**Step 7:** Switch back to master (or main) branch

****

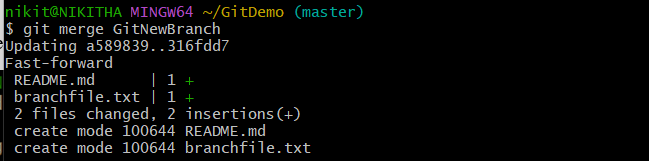
**Step 8:** See differences between master and GitNewBranch (CLI)

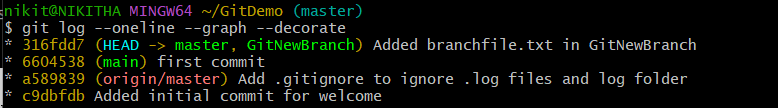


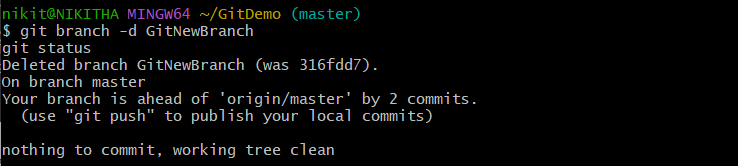
**Step 9:** See visual diff using P4Merge

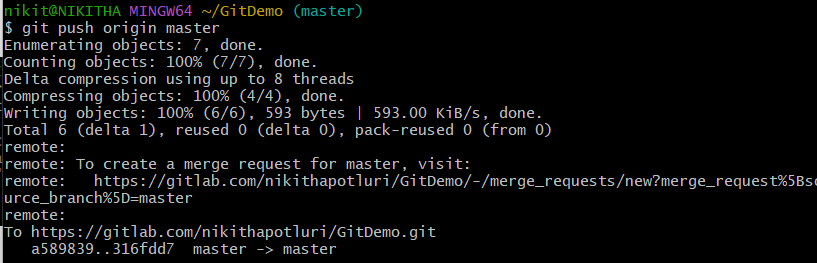

**Step 10:** Merge branch to main

  
  
**Step 11:** View log graphically

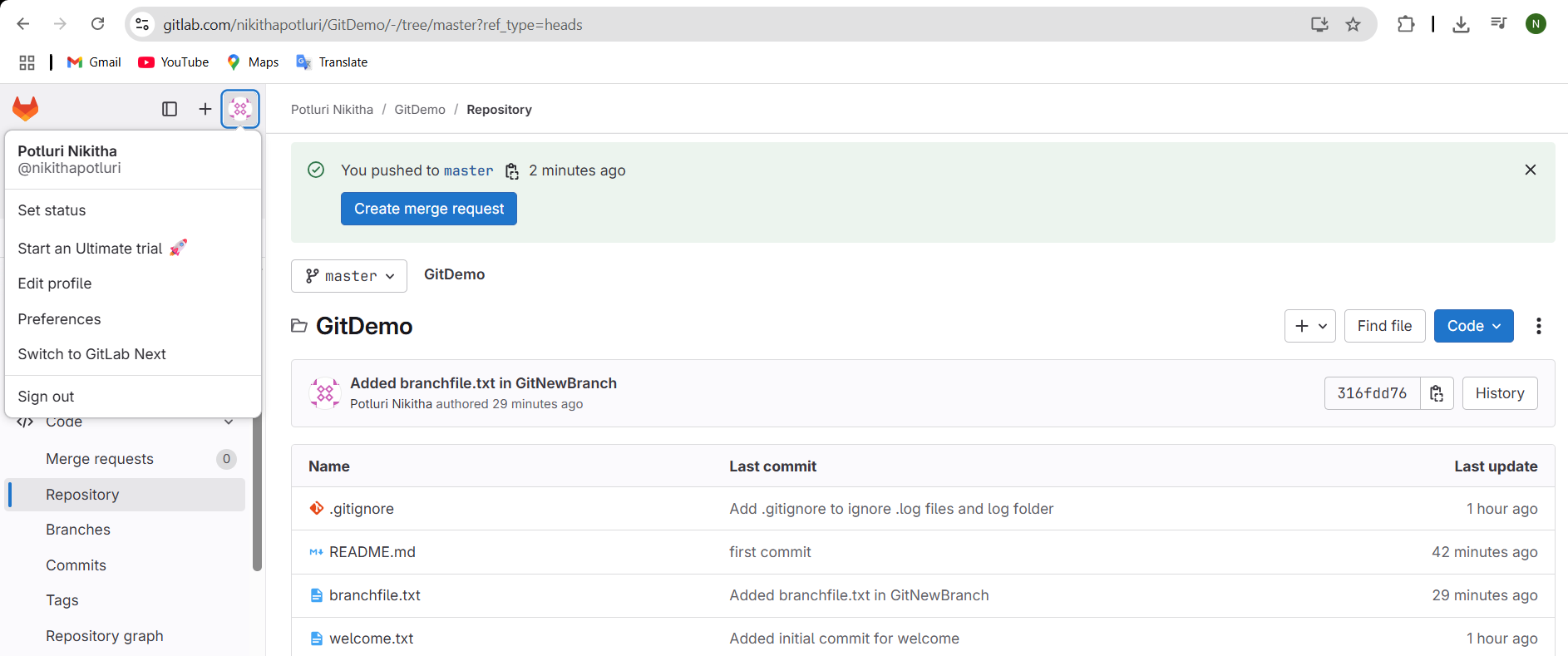
  
  
**Step 12:** Delete merged branch



**Step 13 :** Push your changes to GitHub



**Final “GitDemo” Repository:**

****