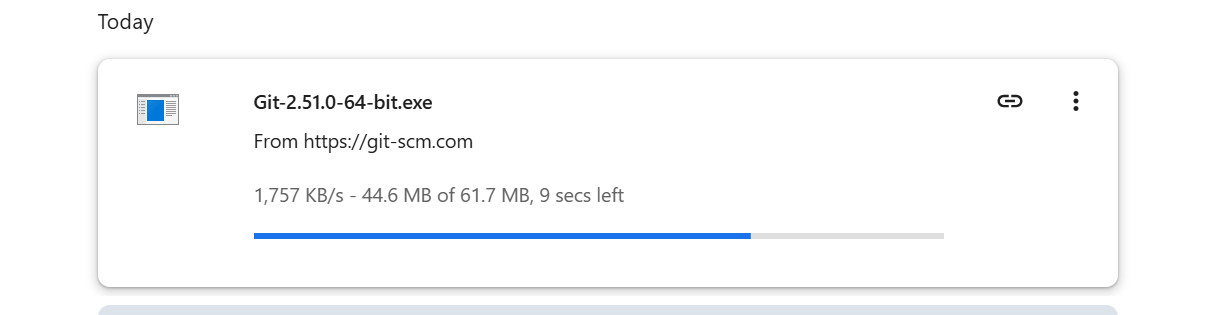
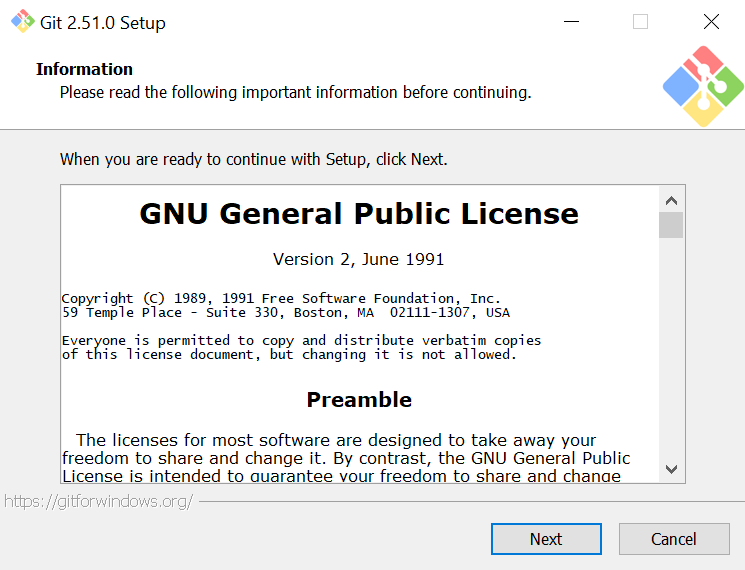
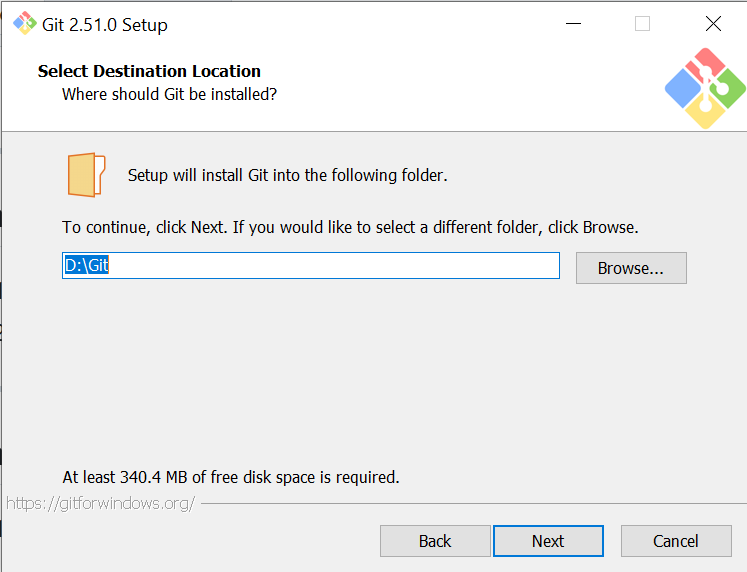
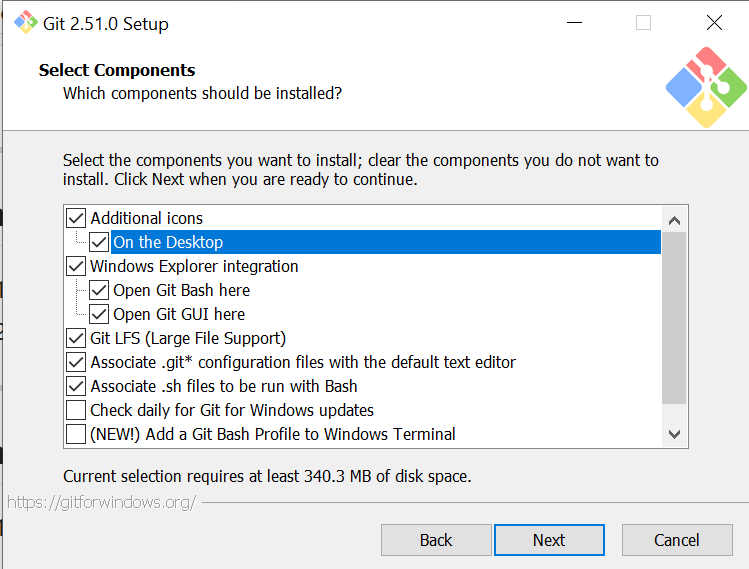
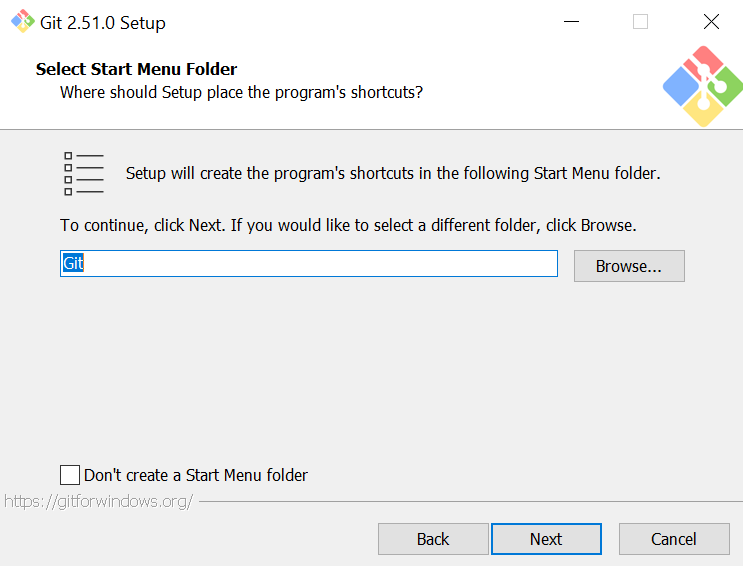
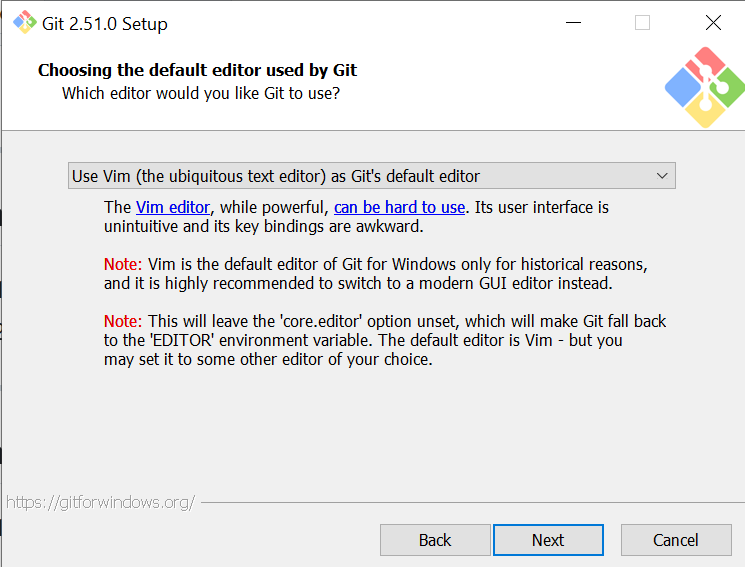
**Step 1: Install Git**

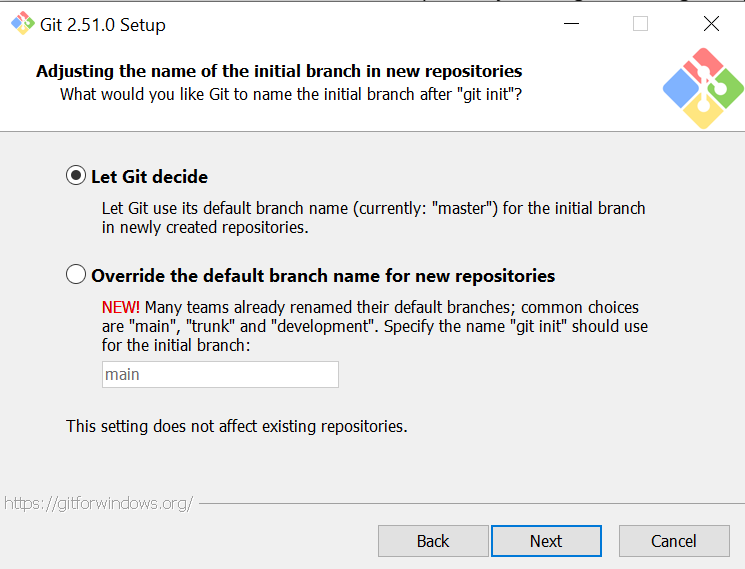


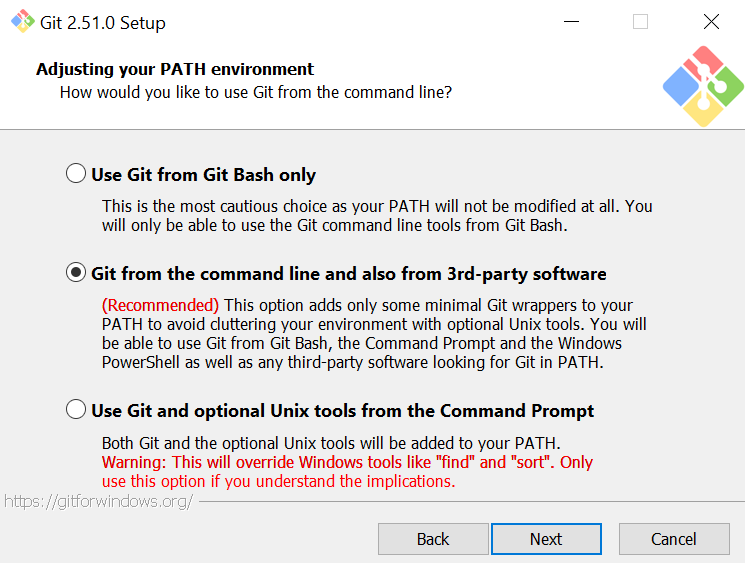


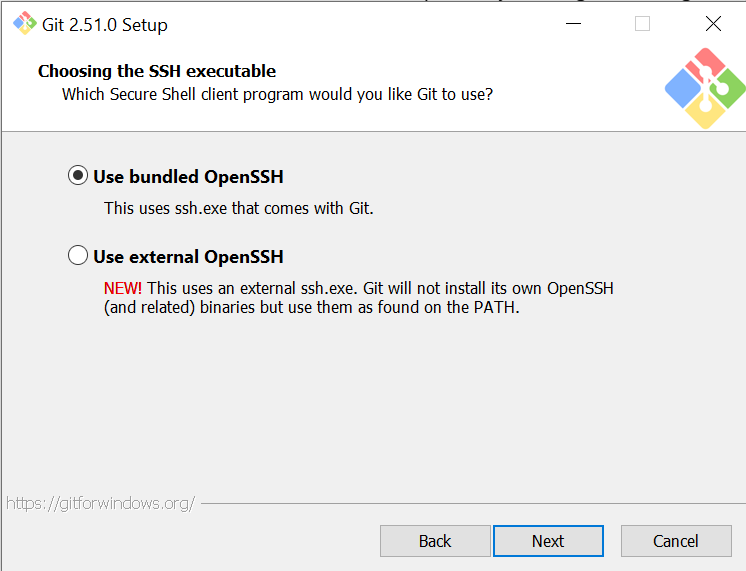


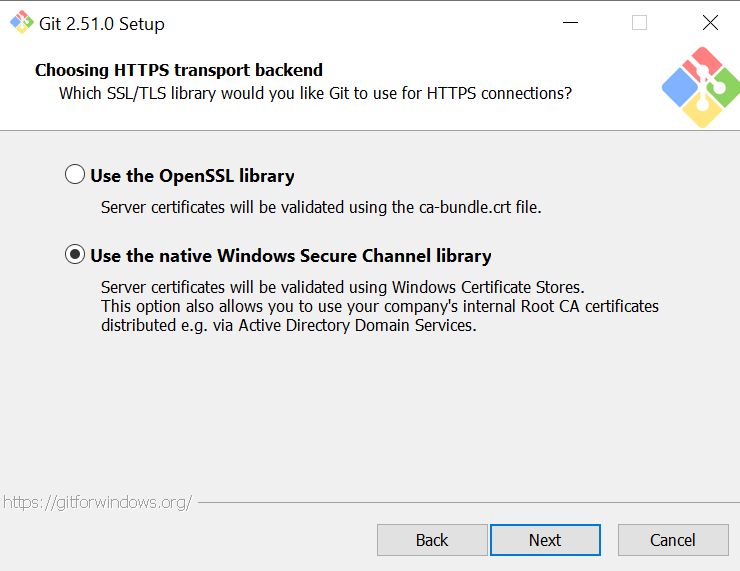


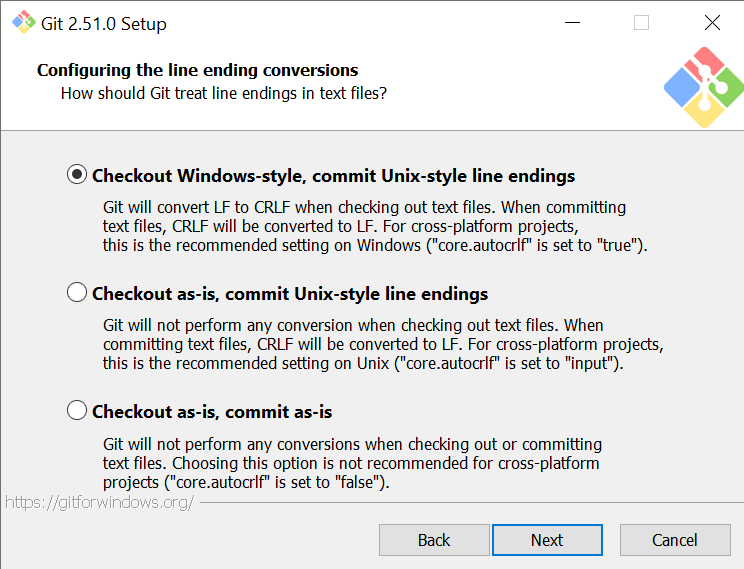


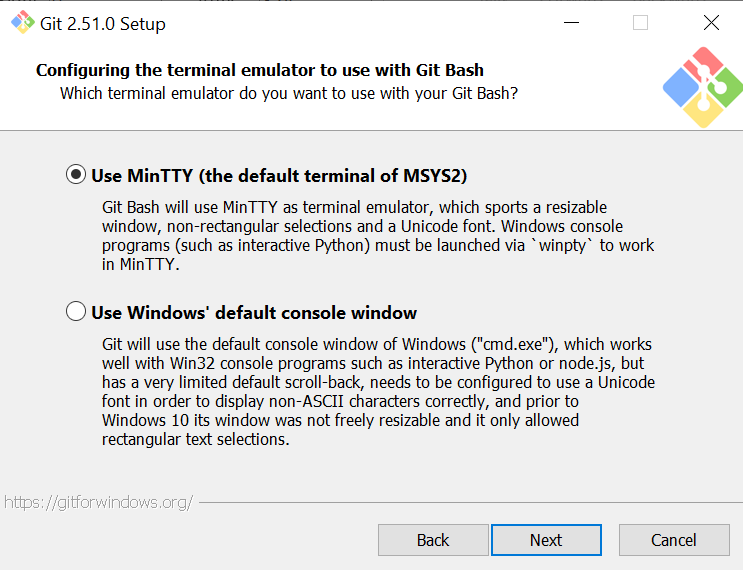


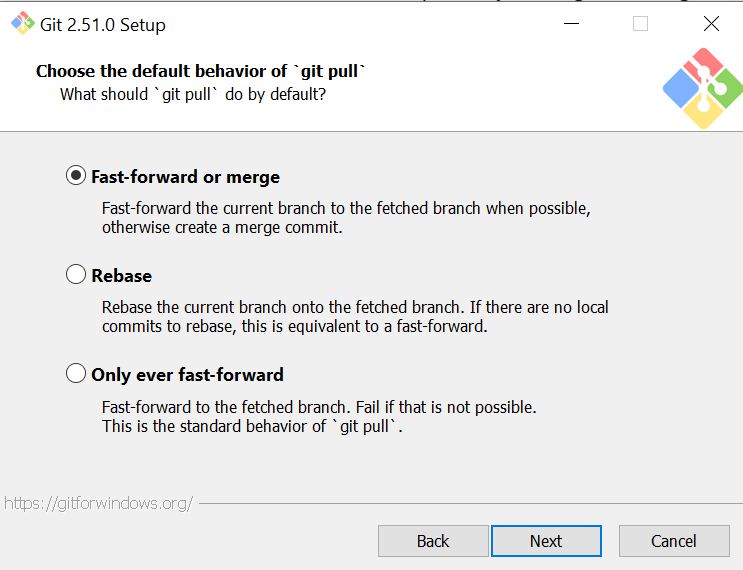


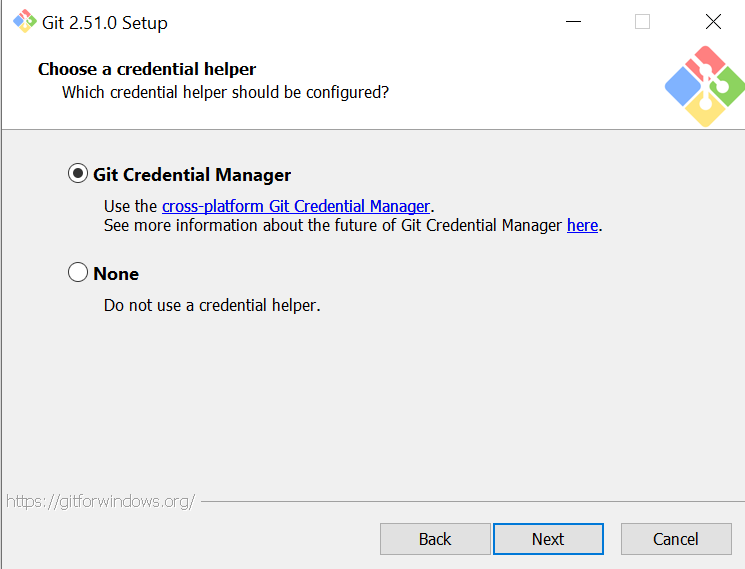


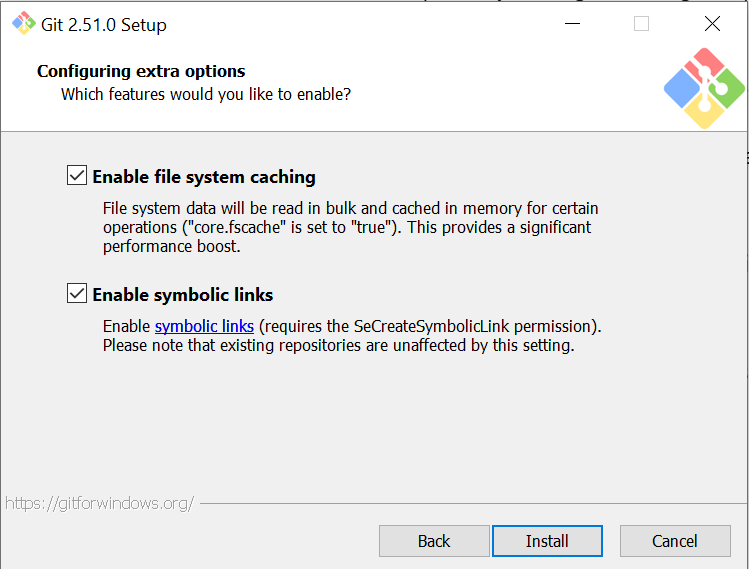


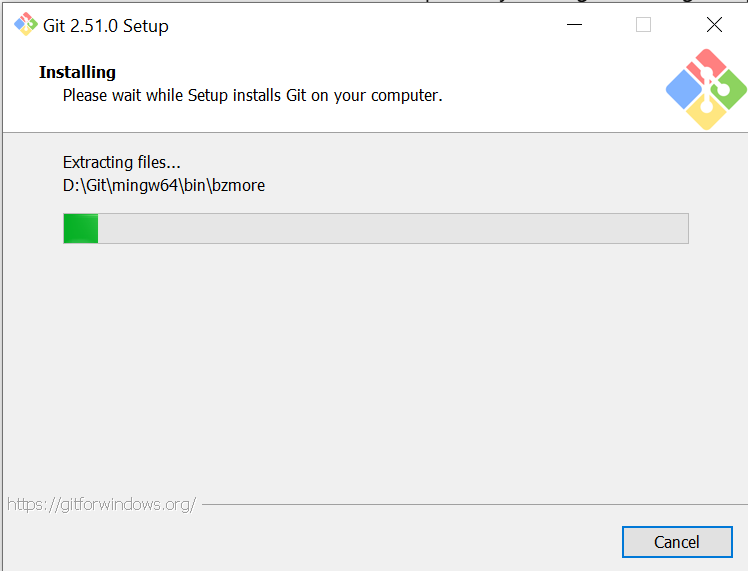


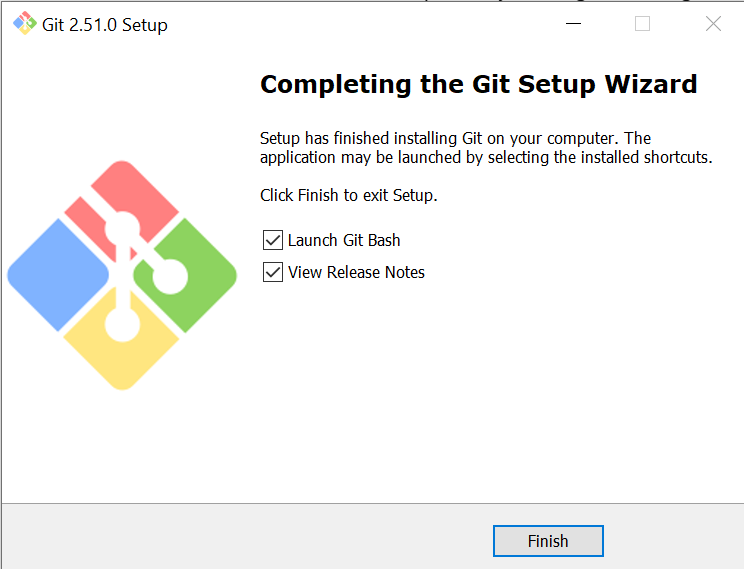












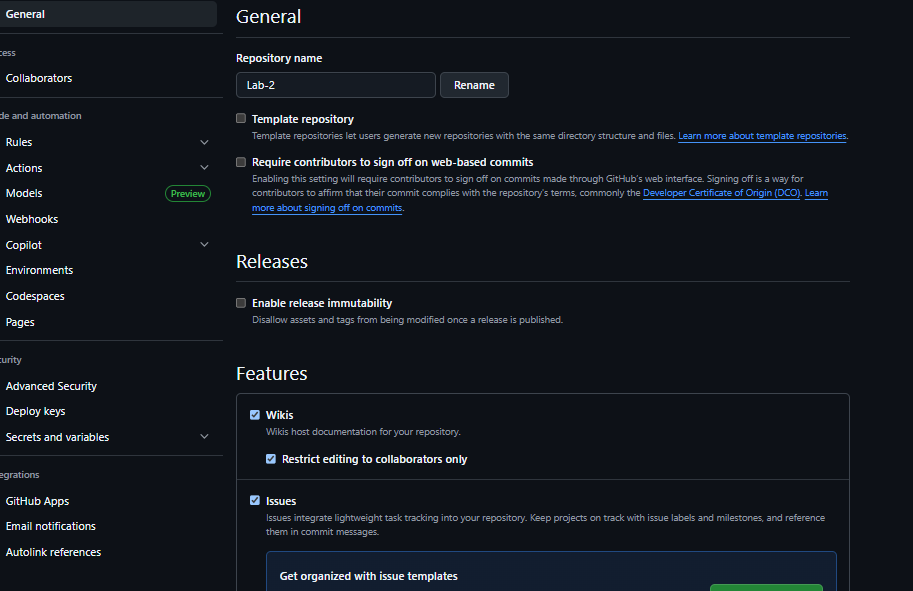
1. After installation, open **Git Bash** or **PowerShell** and check:

**GIT VERSION:**



## **Task 1: Create Private GitHub Repository**

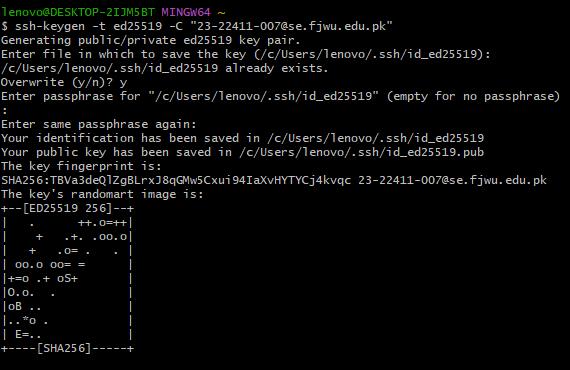
Go to GitHub → Click **+ → new repository**. Name it **Lab2**. Choose **Private**. Click **Create Repository**.



## **Task 2: Connect Repository via SSH**

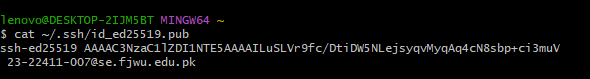
### Step 1: Generate SSH key

1. Press Enter for default location → Enter passphrase or leave empty.

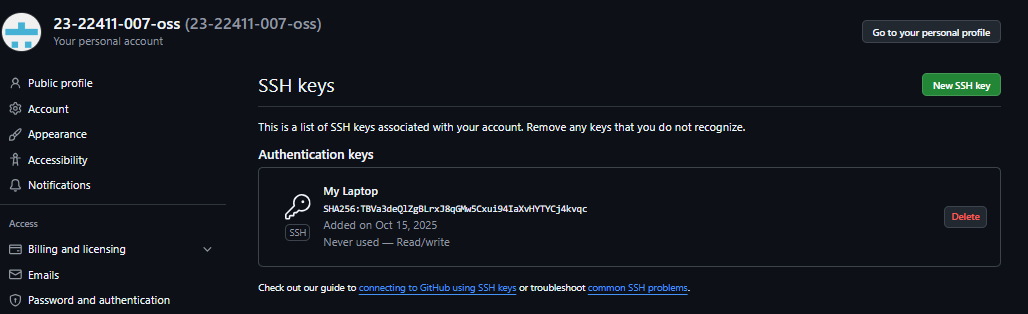


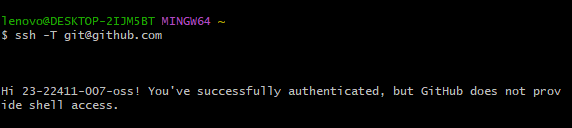
### Step 2: Add SSH key to GitHub

1. Run:



1. Copy the key. On GitHub → **Settings → SSH and GPG Keys → New SSH Key** → paste it.

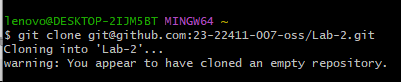




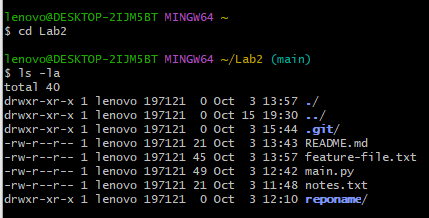
### Step 3: Clone repository using SSH

### 1. Get the SSH URL from GitHub

**2. Clone the Repository**

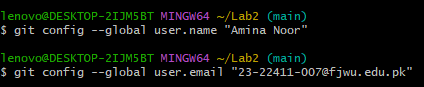


**3. Verify the Clone Worked**

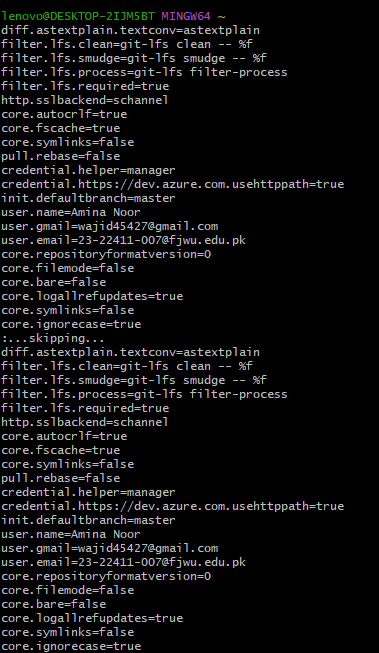


## 🔹 **Task 3: Configure Git Username & Email**

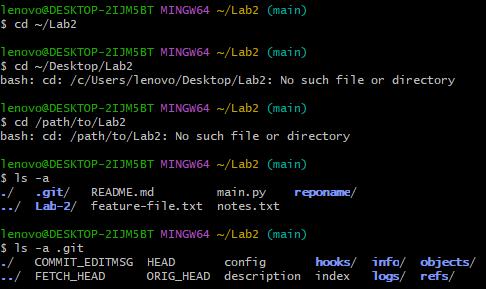
1. Run:



1. Verify config:

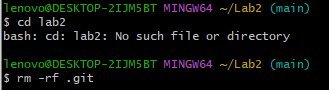


## 🔹 **Task 4: Explore the .git Folder**



## **🔹 Task 5: Local Repository Management**

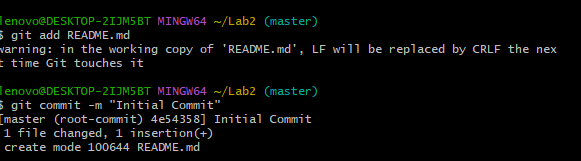
1. Delete .git folder:



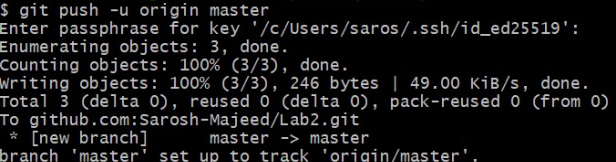
1. Re-initialize:



1. Create README.md:

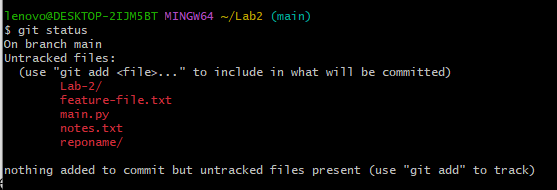


1. Connect to GitHub:

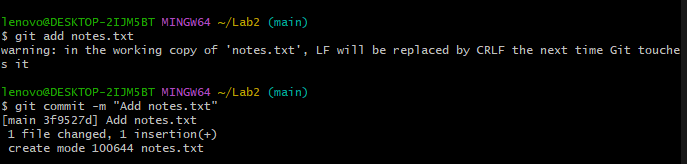


## **🔹 Task 6: File Status & Staging**

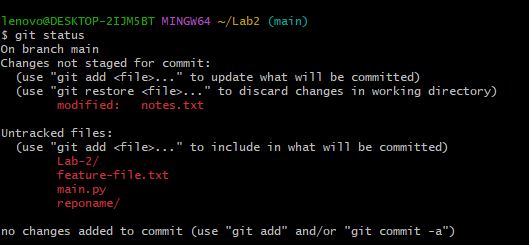
1. Create file:

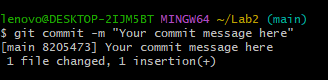


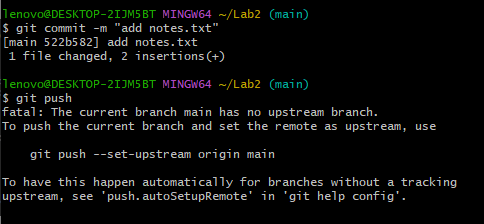
1. Stage + commit:



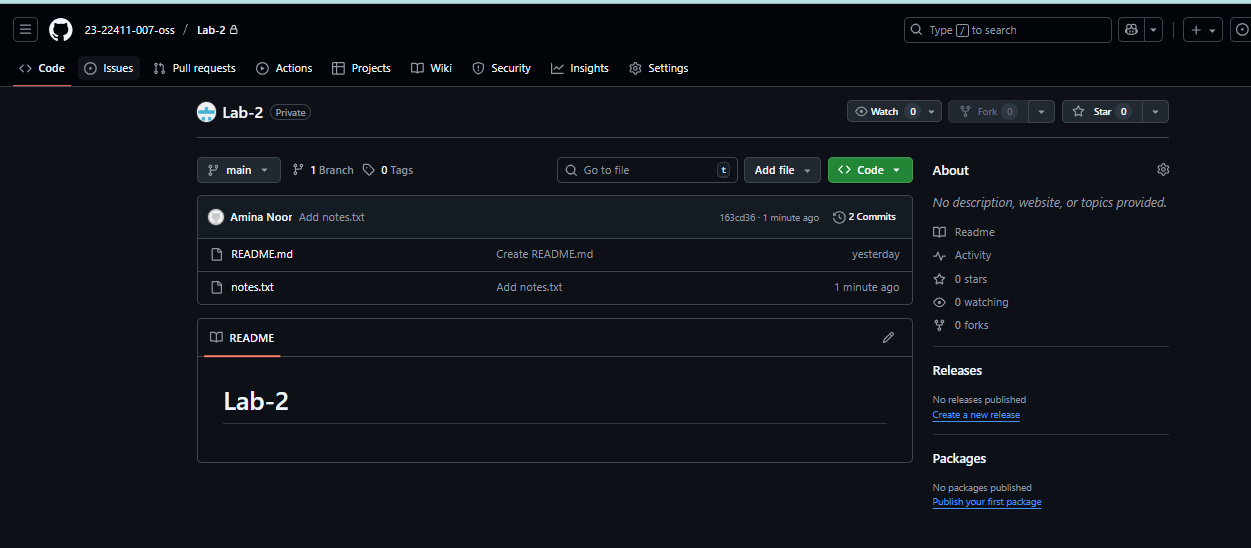
1. Edit notes.txt → repeat status, add, commit.



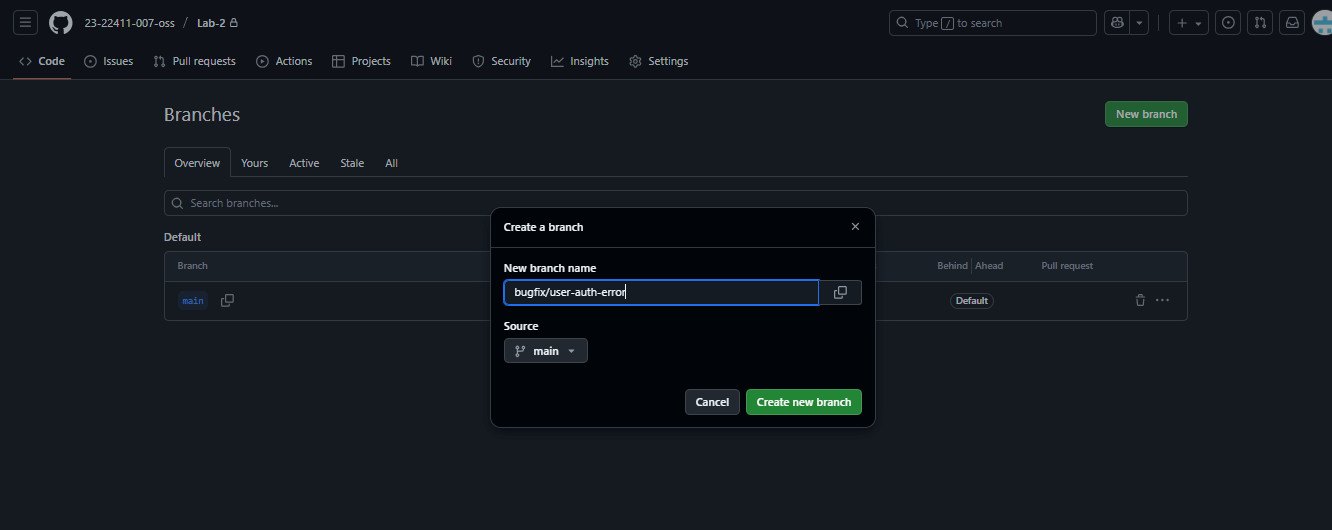


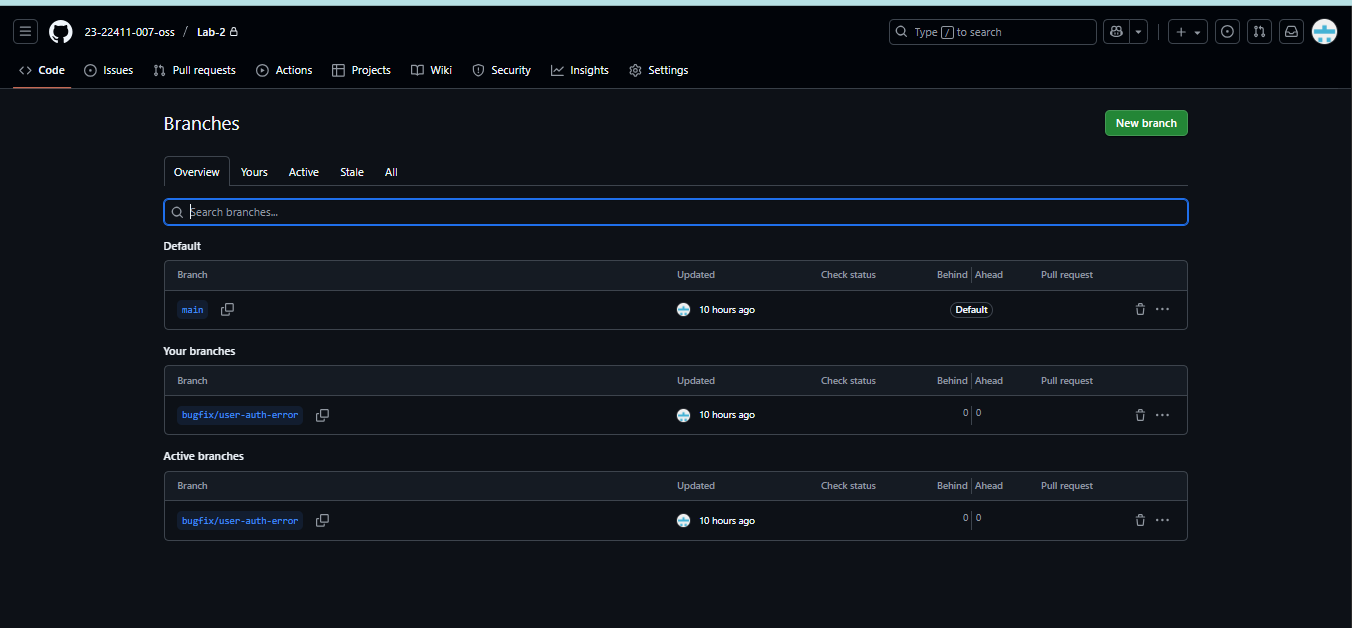


## **🔹 Task 7: Branch Creation (GitHub GUI)**

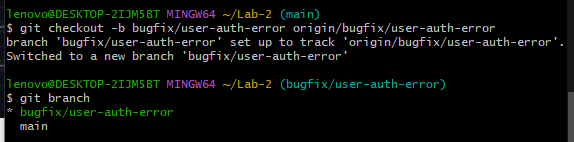


1. On GitHub → go to repo → **Branch: main → create bugfix/user-auth-error**.



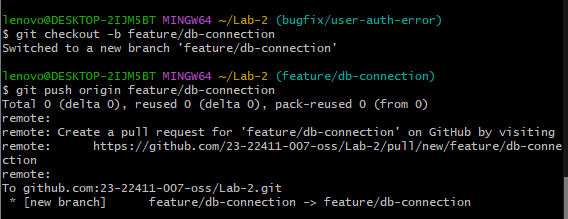


1. Pull to local:



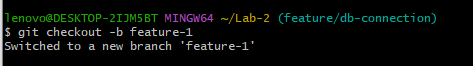
## **🔹 Task 8: Branch Creation (Git Bash)**

1. Run:

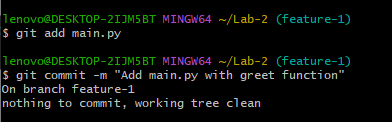


## **🔹 Task 9: Branching & Merging**

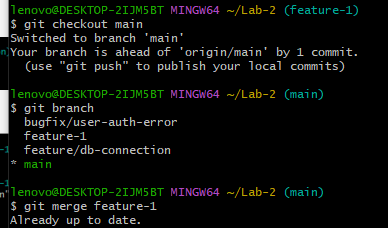
1. Create feature branch:



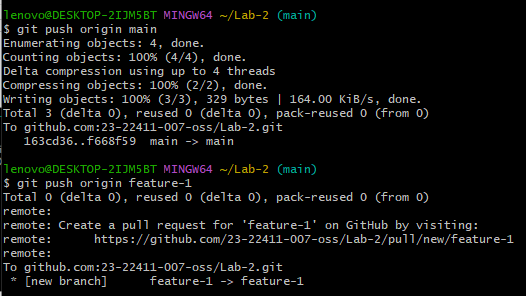
1. Modify **main.py**, then:

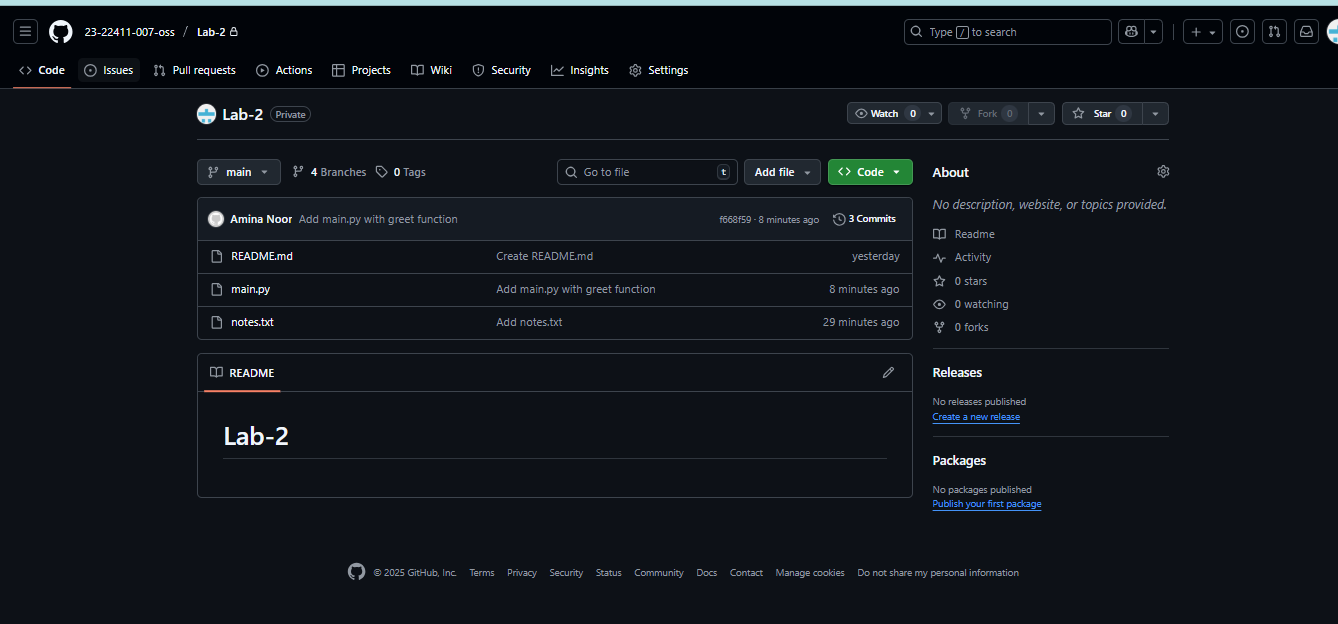


1. Switch & merge:



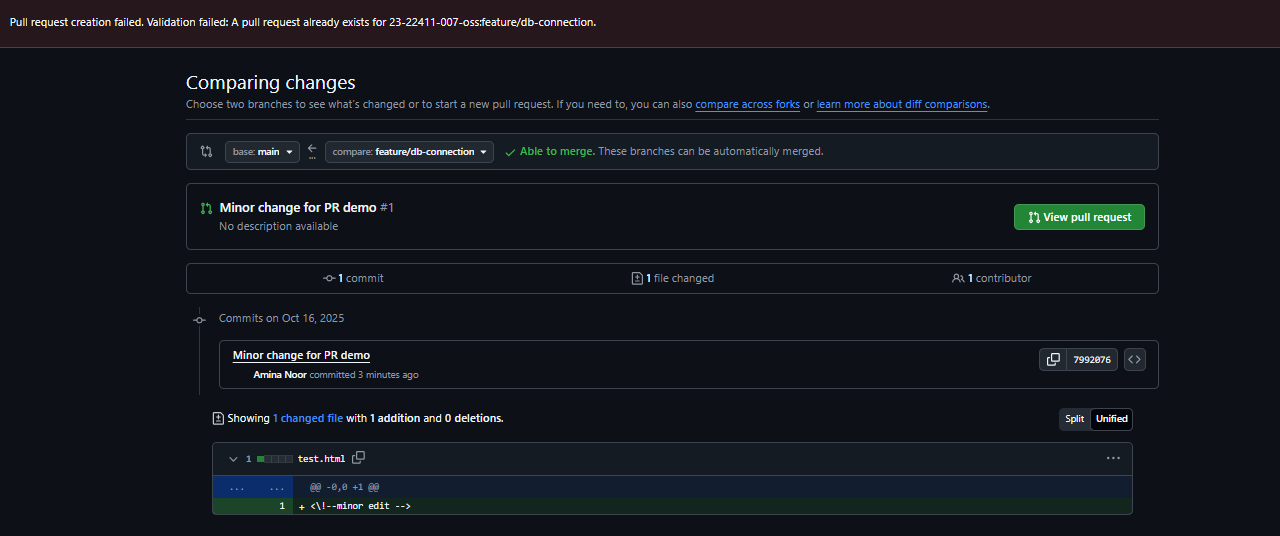
1. Push:



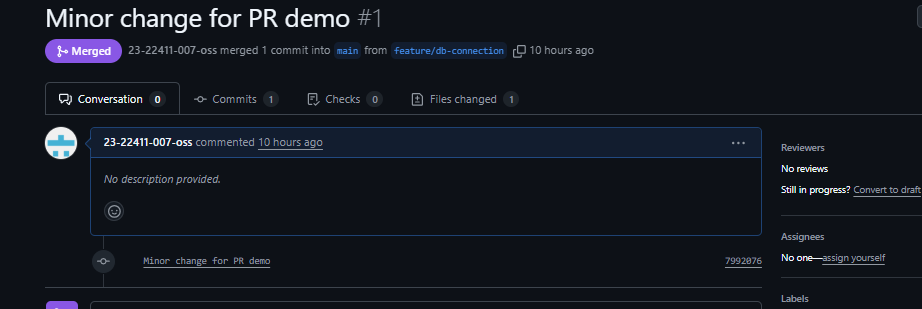


## **🔹 Task 10: Pull Request & Review (GitHub GUI)**

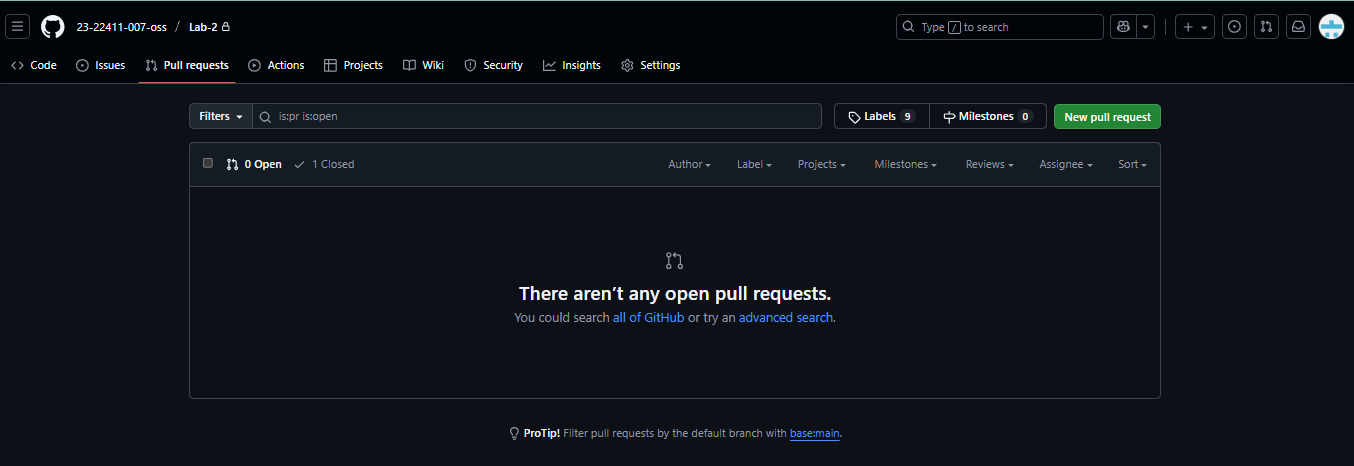
1. On GitHub → create PR **feature/db-connection → main**.



1. Review & merge PR.

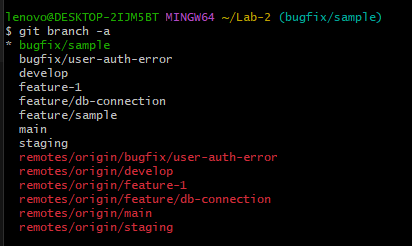


1. Delete branch in GUI.

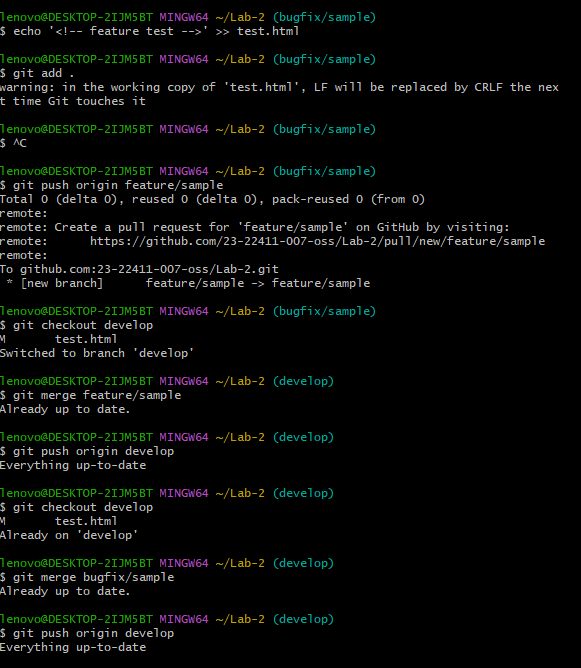


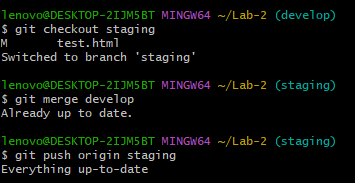
## **🔹 Task 11: Branch Strategy (Develop/Staging)**

## **Step 1 — Create the branches**

****

**Step 2 — Simulate development and merge to develop**





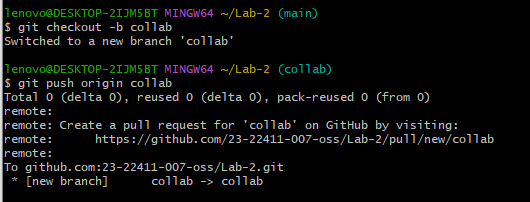
**Step 4 — Final merge into main (production)**

## 

**Bonus Task: Simulated Team Collaboration**

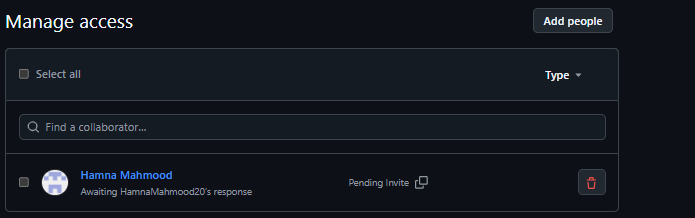
1. Create a branch called **collab**.
2. Add a **collaborator** to your GitHub repo.
3. Both users update a shared file (notes.txt).
4. Push commits to the same branch.
5. Merge the branch into **main**.
6. Take screenshots for submission.

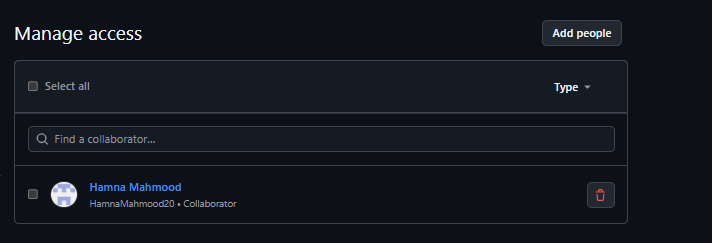
Step 1 — Create the collab branch.



## **Step 2 — Add a collaborator**

## 

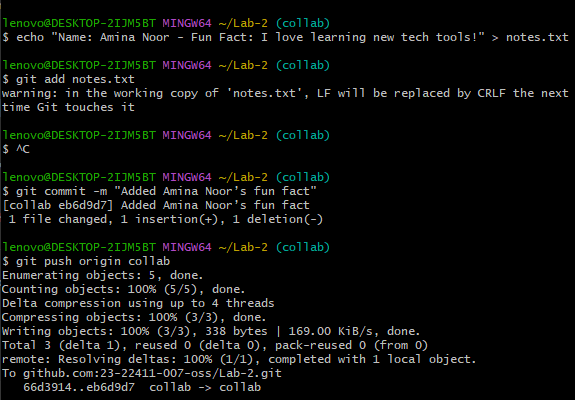




**Step 3 — Pull latest changes**

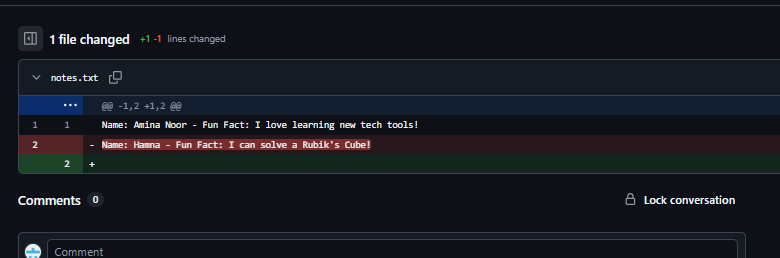
## 

**Step 4 — Edit notes.txt**



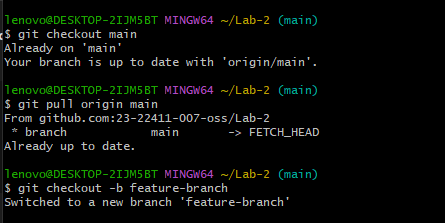
**Step 5 — Collaborator repeats**

**Step 6 — Merge collab → main**

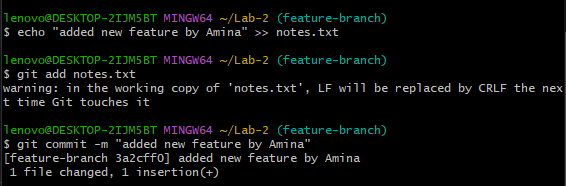


## **🔹 Task 12: Code Review Workflow**

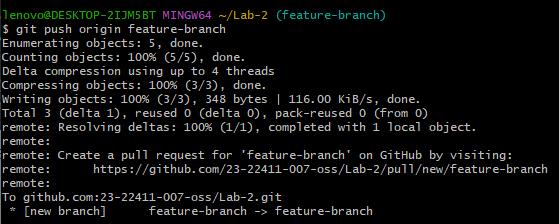
1. **Step 1: Create and Switch to a Feature Branch**



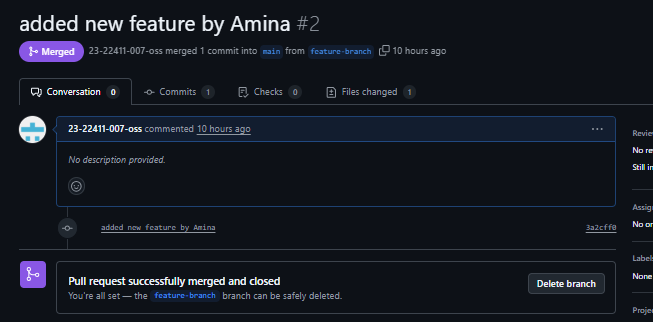
1. **Step 2: Make and Commit a Change**



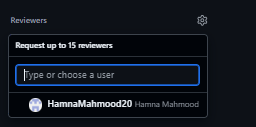
1. Step 3: Push Feature Branch to GitHub



1. **Step 4: Create a Pull Request**

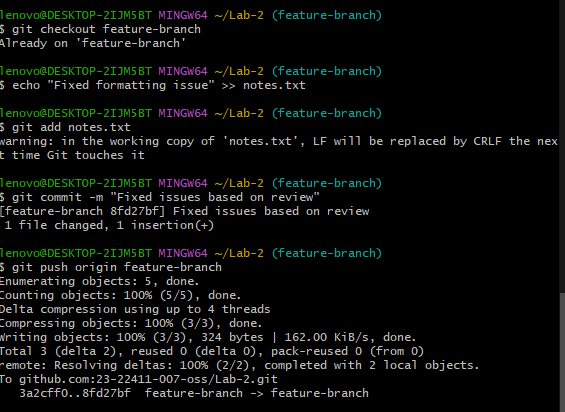


1. **Step 5: Assign a Reviewer**

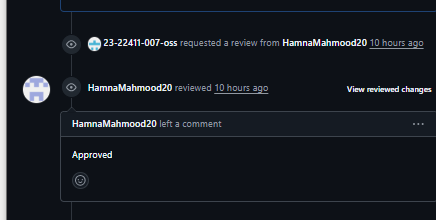




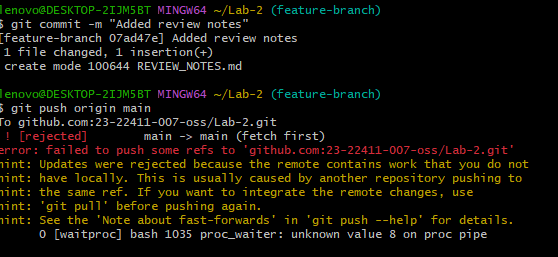
**Step 6: Author Fixes the Issues**



**Step 7: Reviewer Approves and Merge**

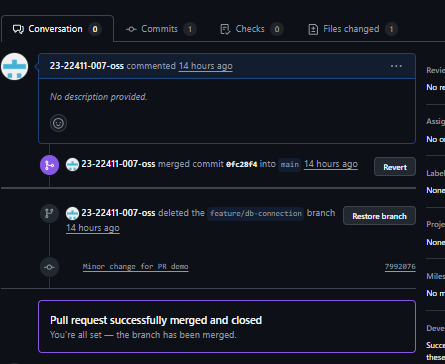


**Step 8: Branch Handling After Merge**

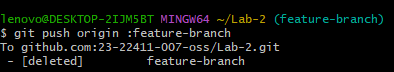


## **🔹 Task 13: Branch Cleanup**

1. Delete remote branch (GitHub UI).



1. Or via command:



1. Cleanup locally:

