

# CLOUD COMPUTING

## Lab 03

**Name:** Abiha Nadeem

**Roll no:** 2023-BSE-001

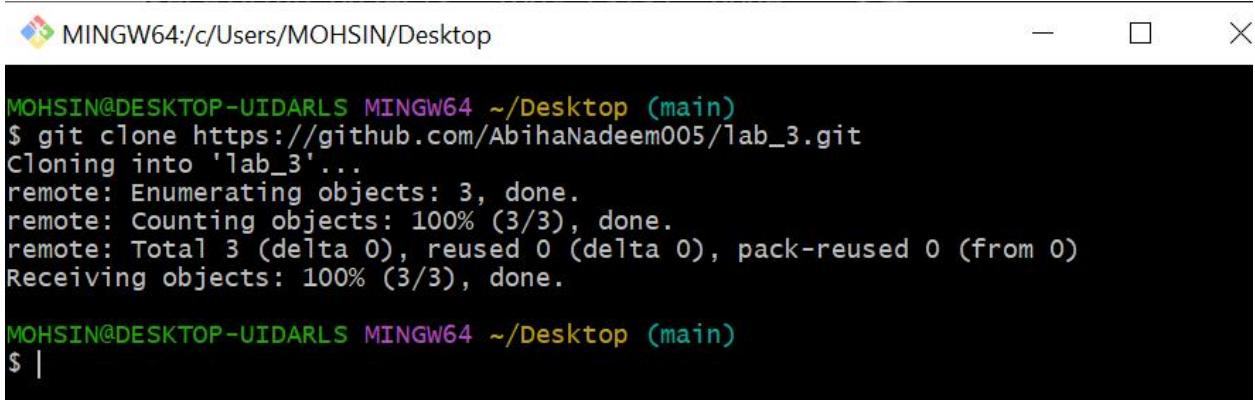
**Submitted to:** Engr. Muhammad Shoaib

## Task 1 – Handling Local and Remote Commit Conflicts (Pull vs Pull --rebase)

This task demonstrates what happens when your local repository and remote repository both have new commits — and how git pull and git pull --rebase behave differently.

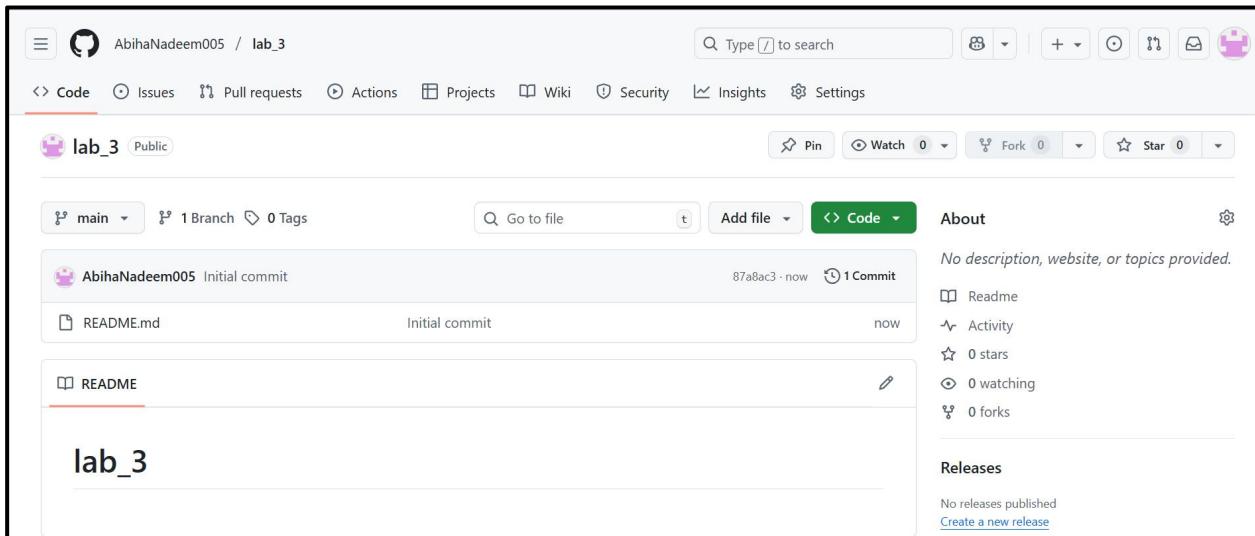
### Steps

1. Open your repository on **GitHub** and edit the README.md file directly in the browser.



```
MINGW64:/c/Users/MOHSIN/Desktop
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop (main)
$ git clone https://github.com/AbihaNadeem005/lab_3.git
Cloning into 'lab_3'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop (main)
$ |
```



The screenshot shows a GitHub repository page for 'lab\_3'. The repository is public and has one branch ('main') and one commit ('Initial commit' by 'AbihaNadeem005'). The README.md file contains the text 'Initial commit'. On the right side of the page, there is an 'About' section with the message 'No description, website, or topics provided.' Below it, there are sections for 'Readme', 'Activity', 'Stars', 'Watching', and 'Forks'. There are also sections for 'Releases' and 'Create a new release'.

- Add a new line such as:
- This line was added remotely from GitHub.
- Commit your change on GitHub (e.g., message: *Remote update to README*).
- Save a screenshot of your browser showing the change as `remote_edit.png`.

The screenshot shows a GitHub repository page for 'AbihaNadeem005 / lab\_3'. The 'Code' tab is selected. A pull request from 'AbihaNadeem005' titled 'Remote update to README' has been merged. The commit message is 'This line was added remotely from GitHub.' The code editor shows the line 'lab\_3'.

2. On your **local machine**, open the same repository folder.

- Edit the same README.md file locally, for example:
- This line was added locally.
- Stage and commit your change:
- `git add README.md`

`git commit -m "Local update to README"`

- Save a screenshot of your terminal as `local_commit.png`.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop (main)
$ cd lab_3

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ ls
README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ nano README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

The screenshot shows a terminal window with two main sections. The top section is a terminal window titled "MINGW64:/c/Users/MOHSIN/Desktop/lab\_3". It displays the output of a "git commit" command:

```
GNU nano 8.6 README.md Modified
# Tab_3
This line was added locally.

^G Help      ^O Write Out ^F Where Is  ^K Cut      ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste    ^J Justify  ^/ Go To Line
```

The bottom section is a terminal window showing the command history:

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Local update to README"
[main dc6013f] Local update to README
 1 file changed, 3 insertions(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

3. Try to push:

```
git push origin main
```

You'll see an error message similar to:

```
! [rejected] main -> main (fetch first)
error: failed to push some refs to 'github.com:username/repo.git'
```

hint: Updates were rejected because the remote contains work that you do not have locally.

- o Save a screenshot of this error as push\_error.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push origin main
To https://github.com/AbihaNadeem005/lab_3.git
 ! [rejected]      main -> main (fetch first)
error: failed to push some refs to 'https://github.com/AbihaNadeem005/lab_3.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

4. To fix this, pull the latest changes from remote:

git pull --no-rebase origin main

- Git will **merge** your local and remote commits, creating a *merge commit*.
- Save a screenshot as merge\_commit.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull --no-rebase origin main
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 949 bytes | 41.00 KiB/s, done.
From https://github.com/AbihaNadeem005/lab_3
 * branch            main      -> FETCH_HEAD
   87a8ac3..3c06bc8  main      -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|MERGING)
$
```

MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6 README.md

```
# Tab_3
<<<<< HEAD
This line was added locally.

This line was added remotely from GitHub.
>>>>> 3c06bc89345c976bc3c557f6b8eae299b2256386
```

[ Read 7 Lines (converted from DOS format) ]

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location  
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6 README.md Modified

```
# Tab_3
This line was added locally.
This line was added remotely from GitHub.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|Merging)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|Merging)
$ git commit -m "Resolved README.md merge conflict"
[main 2397193] Resolved README.md merge conflict

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull --no-rebase origin main
From https://github.com/AbihaNadeem005/lab_3
 * branch            main      -> FETCH_HEAD
Already up to date.
```

5. Push again:

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

- o Save a screenshot as push\_after\_merge.png.

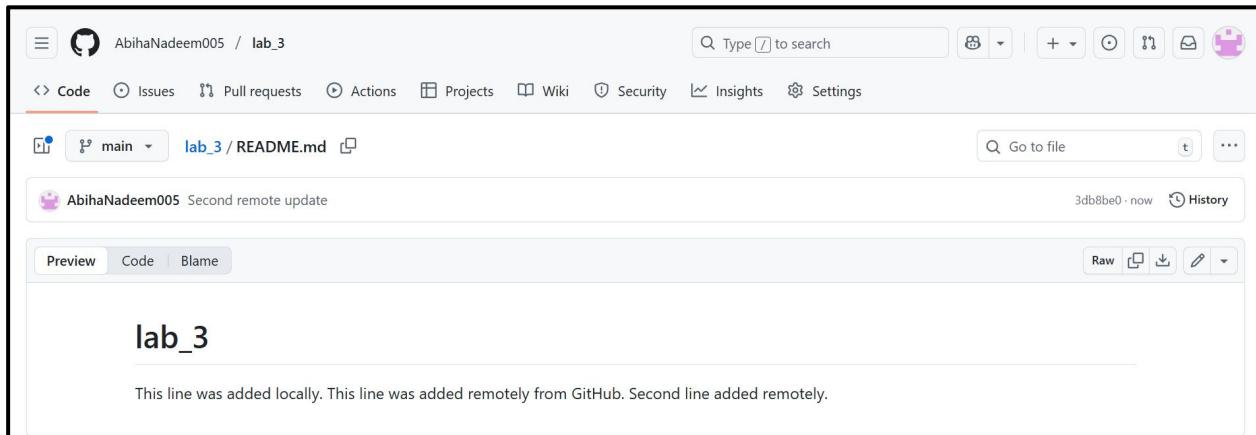
```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push -u origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (6/6), 628 bytes | 314.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
  3c06bc8..2397193 main -> main
branch 'main' set up to track 'origin/main'.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

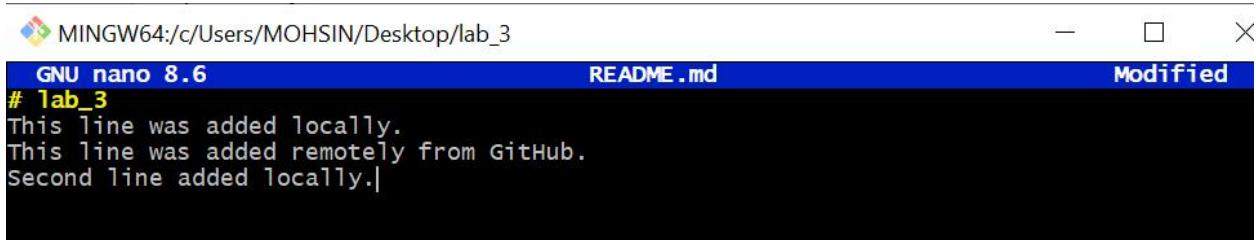
6. Now repeat the same process, but this time use **rebase instead of merge**:

- Make another remote edit on GitHub in README.md.



The screenshot shows a GitHub repository interface for the 'lab\_3' branch. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation is a search bar and a file navigation bar showing 'main' and 'lab\_3 / README.md'. A commit history is displayed, with the most recent commit by 'AbihaNadeem005' titled 'Second remote update' and timestamped '3db8be0 · now'. The commit message contains the text 'This line was added locally. This line was added remotely from GitHub. Second line added remotely.' Below the commit message are buttons for Preview, Code, and Blame, and a Raw link. The commit is shown in a light gray box with a dark gray border.

- Then make another local edit and commit it.



The screenshot shows a terminal window with a black background and white text. The title bar indicates 'MINGW64:/c/Users/MOHSIN/Desktop/lab\_3'. The window title is 'GNU nano 8.6' and the file name is 'README.md'. The status bar at the bottom right shows 'Modified'. The content of the file is '# lab\_3' followed by three lines of text: 'This line was added locally.', 'This line was added remotely from GitHub.', and 'Second line added locally.|'. The cursor is positioned at the end of the third line. The terminal window has standard window controls (minimize, maximize, close) at the top right.

```

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Second local update"
[main a098676] Second local update
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

- Instead of using git pull, run:

git pull --rebase origin main

- Git will **replay your local commits on top** of the pulled commits, keeping history linear.
- Save a screenshot as rebase\_pull.png.

```

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull --rebase origin main
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 948 bytes | 67.00 KiB/s, done.
From https://github.com/AbihaNadeem005/lab_3
 * branch            main      -> FETCH_HEAD
   2397193..3db8be0  main      -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: could not apply a098676... Second local update
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicted_files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
hint: Disable this message with "git config set advice.mergeConflict false"
Could not apply a098676... # Second local update

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$
```

```

MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6 README.md
# Tab_3
This line was added locally.
This line was added remotely from GitHub.
<<<<< HEAD
Second line added remotely.
_____
Second line added locally.
>>>>> a098676 (Second local update)
```

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6          README.md          Modified
# Tab_3
This Line was added locally.
This line was added remotely from GitHub.
Second line added remotely.
Second line added locally.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$ git commit -m "Resolved README.md merge conflict"
[detached HEAD 2a8ff3b] Resolved README.md merge conflict
 1 file changed, 2 insertions(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$
```

- Push again:

```
git push -u origin main
    ■ Save a screenshot as push_after_rebase.png.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$ git rebase --continue
Successfully rebased and updated refs/heads/main.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push -u origin main --force
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 323 bytes | 323.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/AbihaNadeem005/lab_3.git
  3db8be0..2a8ff3b  main -> main
branch 'main' set up to track 'origin/main'.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

## Task 2 – Creating and Resolving Merge Conflicts Manually

This task will help you understand how Git handles file conflicts when two people modify the same line of code differently.

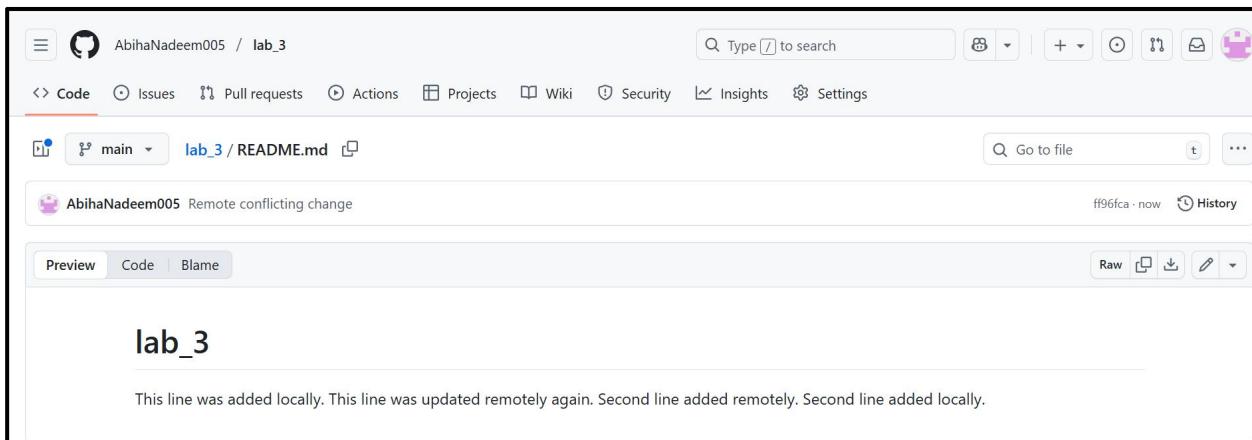
### Steps

1. On **GitHub (remote)**, open your README.md file and change an existing line.  
For example, if it currently says:
2. This line was added remotely from GitHub.

Change it to:

This line was updated remotely again.

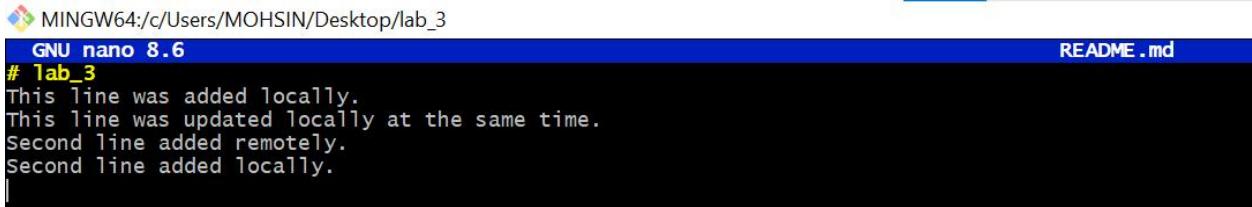
- o Commit your change with the message: *Remote conflicting change*
- o Save a screenshot as remote\_conflict\_edit.png.



The screenshot shows a GitHub repository interface. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main content area shows a file named 'lab\_3 / README.md'. A commit by 'AbihaNadeem005' titled 'Remote conflicting change' is displayed. The commit message is: 'This line was added locally. This line was updated remotely again. Second line added remotely. Second line added locally.' Below the commit, there are tabs for Preview, Code, and Blame, and a Raw file link.

3. On your **local machine**, edit the **same line** in the same file but make a **different change**:
4. This line was updated locally at the same time.

- o Save a screenshot of your edit in VS Code as local\_conflict\_edit.png.



The screenshot shows a terminal window running on a MINGW64 system. The title bar indicates the path 'MINGW64:/c/Users/MOHSIN/Desktop/lab\_3'. The window title is 'GNU nano 8.6' and the file name is 'README.md'. The content of the file is:  
# lab\_3  
This Line was added locally.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
A cursor is visible at the end of the last line.

5. Stage and commit your local change:
6. git add README.md

```
git commit -m "Local conflicting change"
```

- Save a screenshot as local\_conflict\_commit.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Local conflicting change"
[main 98387d8] Local conflicting change
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

7. Try to push:

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

- The push will be **rejected**, since the remote version has conflicting changes.
- Save a screenshot of this error as conflict\_push\_error.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push -u origin main
To https://github.com/AbihaNadeem005/lab_3.git
 ! [rejected]          main -> main (fetch first)
error: failed to push some refs to 'https://github.com/AbihaNadeem005/lab_3.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

8. Pull with rebase to bring in remote changes:

```
git pull --rebase origin main
```

- Git will **pause** the rebase and notify you about a conflict.
- Save a screenshot of the conflict message as conflict\_message.png.

```

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull --rebase origin main
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 979 bytes | 65.00 KiB/s, done.
From https://github.com/AbihaNadeem005/lab_3
 * branch            main      -> FETCH_HEAD
   2a8ff3b..ff96fca  main      -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: could not apply 98387d8... Local conflicting change
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicted_files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
hint: Disable this message with "git config set advice.mergeConflict false"
Could not apply 98387d8... # Local conflicting change

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$ |

```

9. Open the README.md file in your editor — you'll see conflict markers like this:

10. <<<<< HEAD

11. This line was updated locally at the same time.

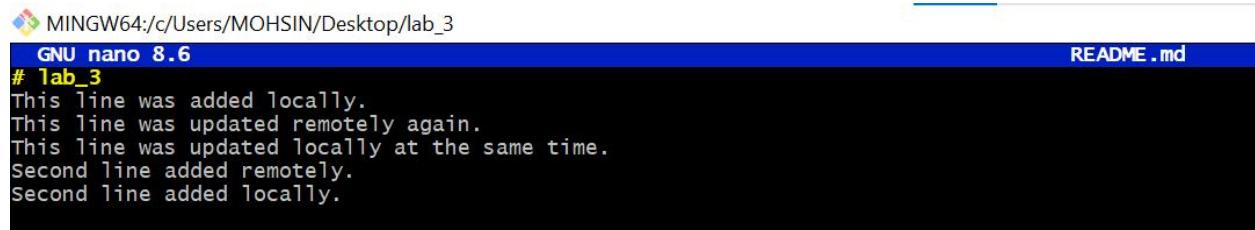
12. =====

13. This line was updated remotely again.

14. >>>>> origin/main

- Manually edit the file to keep the correct line (for example, merge both ideas).
- Save a screenshot of the resolved file as resolved\_readme.png.

---



```

MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6                                         README.md
# lab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.

```

15. After fixing the file, mark the conflict as resolved and continue the rebase:

16. git add README.md

git rebase --continue

- Save a screenshot of this step as rebase\_continue.png.

```
[detached HEAD 06b1230] Local conflicting change  
 1 file changed, 1 insertion(+)  
Successfully rebased and updated refs/heads/main.  
  
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)  
$
```

17. Finally, push your changes:

- ```
git push -u origin main  
    ○ Save a screenshot as push_after_resolve.png.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)  
$ git push -u origin main  
Enumerating objects: 5, done.  
Counting objects: 100% (5/5), done.  
Delta compression using up to 4 threads  
Compressing objects: 100% (2/2), done.  
Writing objects: 100% (3/3), 363 bytes | 363.00 KiB/s, done.  
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
To https://github.com/AbihaNadeem005/lab_3.git  
  ff96fc..06b1230 main -> main  
branch 'main' set up to track 'origin/main'.  
  
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)  
$ |
```

---

### Task 3 – Managing Ignored Files with .gitignore and Removing Tracked Files

In this task, you will learn how to use a .gitignore file to exclude files or directories from version control and how to remove them from Git's tracking while keeping them locally.

#### Steps

1. Create a new folder named textfiles inside your repository:

```
mkdir textfiles
```

2. Inside the textfiles folder, create three text files:
  3. echo "File A content" > textfiles/a.txt
  4. echo "File B content" > textfiles/b.txt

```
echo "File C content" > textfiles/c.txt
```

5. Add and commit the new directory:

6. git add .

7. git commit -m "Added textfiles directory with three files"

git push origin main

- o Save a screenshot as push\_textfiles.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ mkdir textfiles

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ echo "File A content" > textfiles/a.txt
echo "File B content" > textfiles/b.txt
echo "File C content" > textfiles/c.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .
warning: in the working copy of 'textfiles/a.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'textfiles/b.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'textfiles/c.txt', LF will be replaced by CRLF the next time Git touches it

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Added textfiles directory with three files"
[main b01f99d] Added textfiles directory with three files
 3 files changed, 3 insertions(+)
 create mode 100644 textfiles/a.txt
 create mode 100644 textfiles/b.txt
 create mode 100644 textfiles/c.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (6/6), 466 bytes | 233.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
 06b1230..b01f99d main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

8. Now, create a .gitignore file in the root of your repository:

echo "textfiles/\*" > .gitignore

9. Add and commit the .gitignore file:

10. git add .gitignore

11. git commit -m "Added .gitignore to ignore textfiles directory"

git push origin main

- o Save a screenshot as gitignore\_push.png.

```

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ echo "textfiles/*" > .gitignore

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Added .gitignore to ignore textfiles directory"
[main b527826] Added .gitignore to ignore textfiles directory
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore

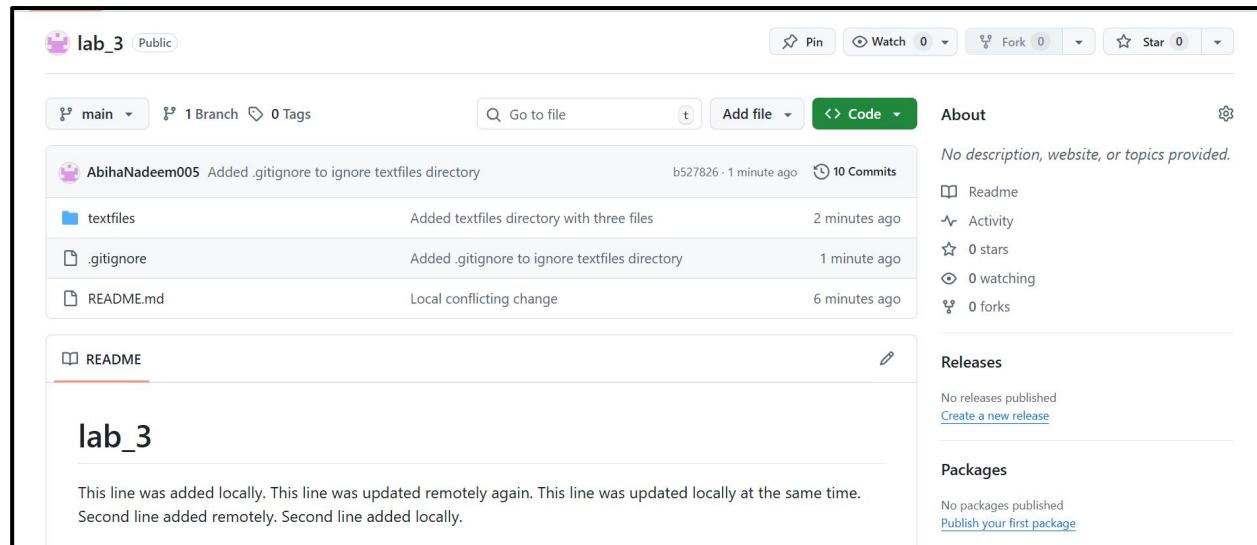
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 356 bytes | 356.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
 b01f99d..b527826 main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |

```

12. Go to **GitHub** and check your repository — notice that the textfiles directory is **still visible** on the remote.

- Save a screenshot as repo\_still\_has\_textfiles.png.



13. To remove already tracked files from Git (but not from your local system), run:

```
git rm -r --cached textfiles
```

14. Commit and push again:

```
git add .
```

```
git commit -m "Removed tracked textfiles directory"
```

```
git push origin main
```

- Save a screenshot as rm\_cached\_push.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git rm -r --cached textfiles
rm 'textfiles/a.txt'
rm 'textfiles/b.txt'
rm 'textfiles/c.txt'

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Removed tracked textfiles directory"
[main e855f60] Removed tracked textfiles directory
 3 files changed, 3 deletions(-)
 delete mode 100644 textfiles/a.txt
 delete mode 100644 textfiles/b.txt
 delete mode 100644 textfiles/c.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 245 bytes | 245.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/AbihaNadeem005/lab_3.git
  b527826..e855f60  main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

17. Check your GitHub repository again — the textfiles folder should now be **deleted remotely**.

- Save a screenshot as repo\_textfiles\_removed.png.

The screenshot shows a GitHub repository page for 'lab\_3'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation is a header with the repository name 'lab\_3' and a public status. A search bar and various repository stats (Pin, Watch, Fork, Star) are also present. The main content area shows a commit history with one commit from 'AbihaNadeem005' that removed a tracked textfiles directory. Below the commit history is a 'README' file containing the text 'lab\_3'. The right sidebar provides repository statistics: no description, website, or topics provided; 11 commits; 0 stars; 0 forks; 0 releases published; and a link to 'Create a new release'. It also has sections for Readme, Activity, and Packages.

## Task 4 – Create Temporary Changes and Use git stash

This task demonstrates how git stash allows you to temporarily save uncommitted changes so that you can safely switch branches without losing work.

### Steps

1. Create a feature-branch.

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

2. Modify any file (for example, README.md) by adding a few test lines.

The terminal window shows a nano editor session in a MINGW64 environment. The current file is 'README.md'. The content of the file includes several test lines: 'This line was added locally.', 'This line was updated remotely again.', 'This line was updated locally at the same time.', 'Second line added remotely.', 'Second line added locally.', and 'This is a test line for feature-branch.'.

3. git add README.md

```
git commit -m "Changes the README.md file"
```

- Save a screenshot as modified\_readme.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git checkout -b feature-branch
Switched to a new branch 'feature-branch'

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ nano README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git commit -m "Changes the README.md file"
[feature-branch c98b662] Changes the README.md file
 1 file changed, 1 insertion(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$
```

4. Modify any file (for example, README.md) by adding a few new lines.

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
```

```
GNU nano 8.6
# lab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is a test line for feature-branch.
Testing stash command.
```

5. Without committing or stashing the changes, try to switch to another branch:

```
git checkout main
```

You'll see an error message similar to:

error: Your local changes to the following files would be overwritten by checkout:

README.md

Please commit your changes or stash them before you switch branches.

- Save a screenshot as checkout\_error.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git checkout main
error: Your local changes to the following files would be overwritten by checkout:
  README.md
Please commit your changes or stash them before you switch branches.
Aborting

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ |
```

6. To fix this, **stash your changes**:

git stash

- Save a screenshot as `stash_command.png`.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git stash
Saved working directory and index state WIP on feature-branch: c98b662 Changes the README.md file

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ |
```

7. Try switching branches again — it should now work:

git checkout main

- Save a screenshot as `branch_switched.png`.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

8. Return to your previous branch (for example, feature-branch):

git checkout feature-branch

- Save a screenshot as `back_to_feature.png`.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git checkout feature-branch
Switched to branch 'feature-branch'

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ |
```

9. Check your working directory status:

git status

It should show:

nothing to commit, working tree clean

- Save a screenshot as status\_clean.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git status
On branch feature-branch
nothing to commit, working tree clean

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ |
```

10. Now restore your stashed changes:

git stash pop

You'll see a message indicating that changes were reapplied:

On branch feature-branch

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

- Save a screenshot as stash\_pop.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git stash pop
On branch feature-branch
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (94cb699a1b8e4538f3fccf572706d3de7984986a)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$
```

11. Confirm that your previous edits are back in the file.

 MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6  
# 1ab\_3  
This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is a test line for feature-branch.  
Testing stash command.

---

### Task 5 – Checkout a Specific Commit Using git log

1. View commit history:

```
git log --oneline
```

- Save a screenshot as log\_before\_checkout.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git log --oneline
c98b662 (HEAD -> feature-branch) Changes the README.md file
e855f60 (origin/main, origin/HEAD, main) Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fca Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$
```

2. Copy any previous commit hash (e.g., a1b2c3d).

3. Checkout that commit:

```
git checkout <commit-hash>
```

- Save a screenshot as detached\_head.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (feature-branch)
$ git checkout 87a8ac3
Note: switching to '87a8ac3'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 ((87a8ac3...))
$
```

4. To return to your main branch:

git checkout main

- o Save a screenshot as back\_to\_main.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 ((87a8ac3...))
$ git checkout main
Previous HEAD position was 87a8ac3 Initial commit
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

---

## Task 6 – Resetting Commits (Soft vs Hard Reset) (With Verification Steps)

This task demonstrates the difference between a soft and hard reset in Git. In this version, you will **verify the presence of changes in files and confirm the existence of commits after each reset.**

### Steps

1. Add a new line in any file and commit it:
2. # Edit any file (e.g., README.md), add a line

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6   README.md
# Tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.|
```

3. git add .

git commit -m "Added test line"

- o Save a screenshot as first\_commit.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Added test line"
[main 4e57070] Added test line
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

4. Add another change and commit again:

5. # Edit the file again, add a different line

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6   README.md   Modified
# Tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is the second test line.
```

6. git add .

git commit -m "Second test commit"

- Save a screenshot as second\_commit.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Second test commit"
[main e0d248f] Second test commit
 1 file changed, 2 insertions(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

#### 7. View the history before reset:

git log --oneline

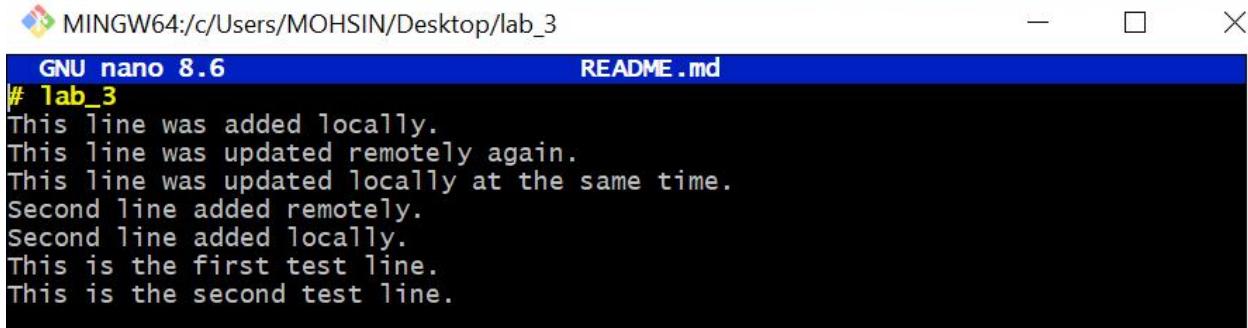
- Save a screenshot as log\_before\_reset.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git log --oneline
e0d248f (HEAD -> main) Second test commit
4e57070 Added test line
e855f60 (origin/main, origin/HEAD) Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fc当地 Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

#### 8. Check file contents for both changes:

- Open the edited file (e.g., README.md) and visually confirm BOTH added lines exist.
- Save a screenshot as file\_before\_reset.png.



MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6 README.md

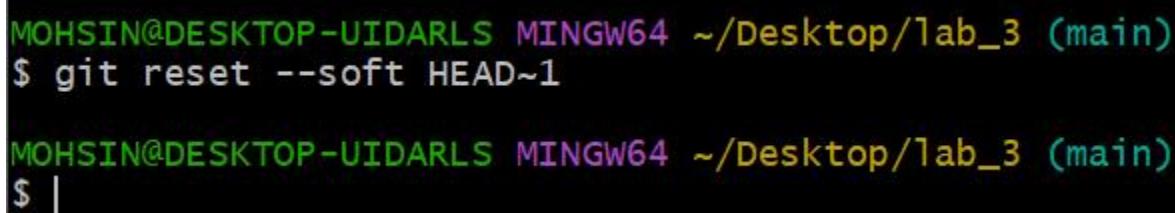
# 1ab\_3

This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is the first test line.  
This is the second test line.

9. Perform a soft reset (keeps changes in working directory):

git reset --soft HEAD~1

- o Save a screenshot as soft\_reset.png.



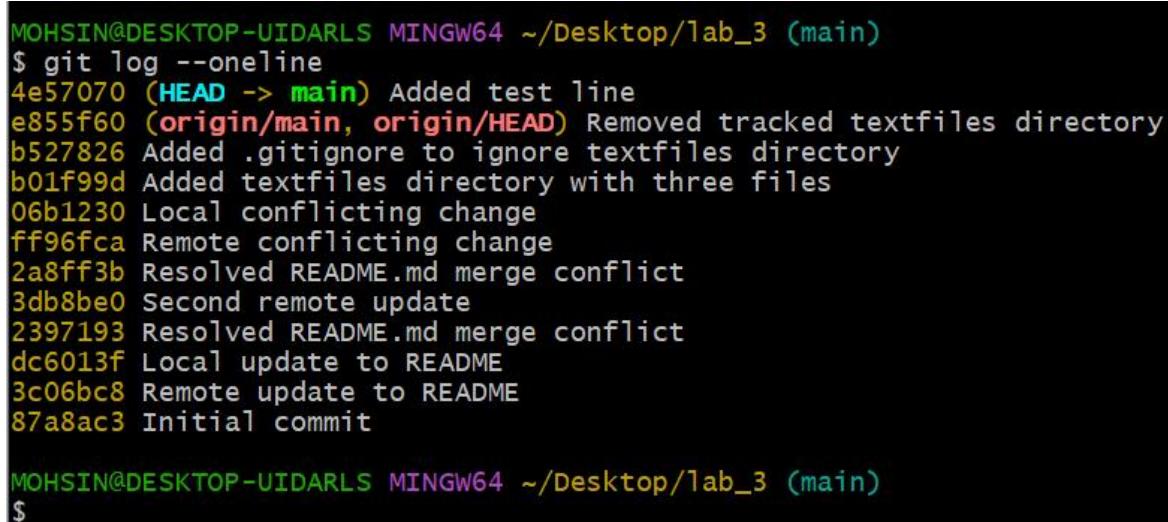
```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git reset --soft HEAD~1

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

10. Check commit history after soft reset:

git log --oneline

- o Save a screenshot as log\_after\_soft\_reset.png.



```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git log --oneline
4e57070 (HEAD -> main) Added test line
e855f60 (origin/main, origin/HEAD) Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fc当地 Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

11. Verify changes in the file:

- Open the file again and confirm both edits are still present (nothing lost).
- Save a screenshot as file\_after\_soft\_reset.png.

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6          README.md
# Lab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is the second test line.
```

## 12. Check git status:

git status

- You should see your changes are staged (ready to commit again).

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

## 13. Perform commit

git commit -m "Second Test commit"

- Save a screenshot as file\_after\_hard\_reset.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Second Test commit"
[main 1715980] Second Test commit
 1 file changed, 2 insertions(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

**14. Perform a hard reset (discards changes completely):**

```
git reset --hard HEAD~1
```

- Save a screenshot as hard\_reset.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git reset --hard HEAD~1
HEAD is now at 4e57070 Added test line

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

**11. Check commit history after hard reset:**

```
git log --oneline
```

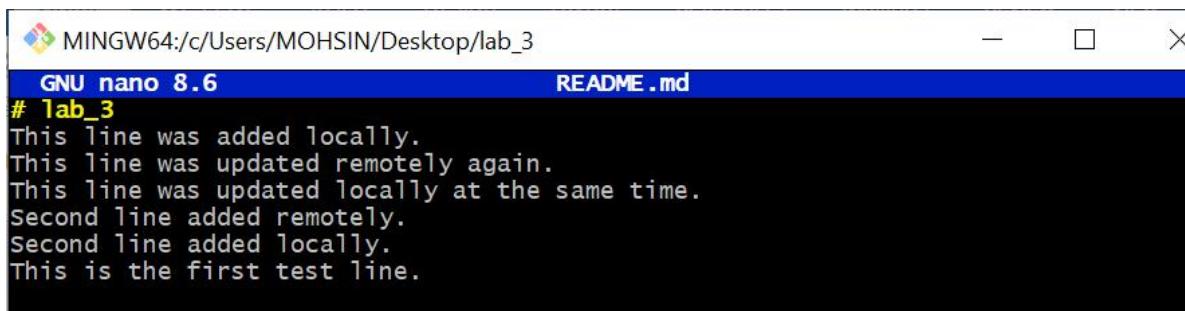
- Save a screenshot as log\_after\_hard\_reset.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git log --oneline
4e57070 (HEAD -> main) Added test line
e855f60 (origin/main, origin/HEAD) Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fca Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

**12. Verify changes in the file:**

- Open the file and confirm the **second edit is gone** (reverted to previous commit).
- Save a screenshot as file\_after\_hard\_reset.png.



The screenshot shows a terminal window with the title "MINGW64:/c/Users/MOHSIN/Desktop/lab\_3". The window contains a "GNU nano 8.6" text editor displaying the "README.md" file. The file content is as follows:

```
# Tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
```

### 13. Check git status:

git status

- Should report "nothing to commit, working tree clean".

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

---

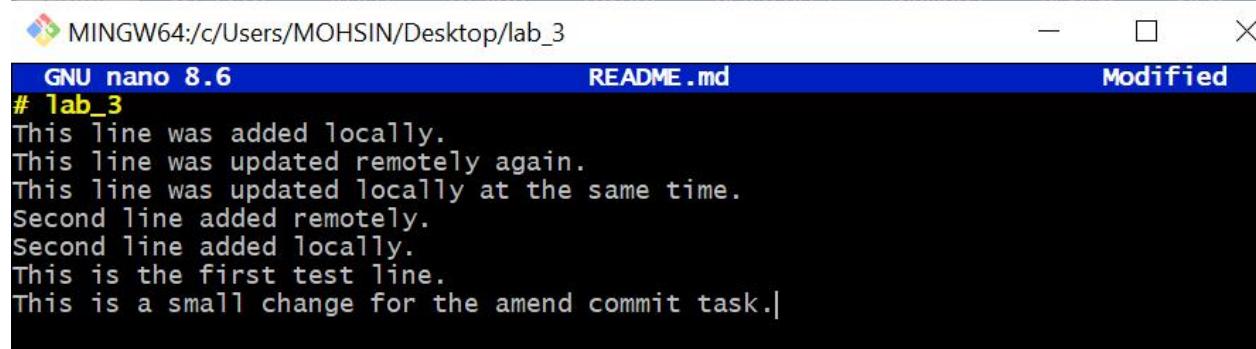
### Task 7 – Amending the Last Commit

1. Make a small change in any file.
2. Stage it and commit:

git add .

git commit -m "Fix log message"

- Save a screenshot as first\_amend\_commit.png.



MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

| GNU nano 8.6 | README.md                                                                                                                                                                                                                                                                  | Modified |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
| # Tab_3      | This line was added locally.<br>This line was updated remotely again.<br>This line was updated locally at the same time.<br>Second line added remotely.<br>Second line added locally.<br>This is the first test line.<br>This is a small change for the amend commit task. |          |

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Fix log message"
[main d8ee8e1] Fix log message
 1 file changed, 1 insertion(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

3. Realize you forgot to change another file. Update it now.

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6          README.md      Modified
# tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is a small change for the amend commit task.
This is an additional change for the amended commit.
```

- #### 4. Add it and amend the last commit:

```
git add .
```

```
git commit --amend
```

- Save a screenshot as amend\_commit.png

```
Fix log message
```

```
[main b6571ad] Fix log message
Date: Tue Nov 4 09:17:47 2025 +0500
1 file changed, 2 insertions(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

---

### Task 8 – Reverting a Commit (Safe Undo on Remote Branch)

1. Make a change and commit it:

```
echo "temporary text" >> temp.txt
git add .
git commit -m "Added temporary text file"
git push origin main
    ○ Save a screenshot as commit_temp_file.png.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ echo "temporary text" >> temp.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .
warning: in the working copy of 'temp.txt', LF will be replaced by CRLF the next time Git touches it

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Added temporary text file"
[main 1974a07] Added temporary text file
 1 file changed, 1 insertion(+)
 create mode 100644 temp.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 949 bytes | 474.00 KiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/AbihaNadeem005/lab_3.git
 e855f60..1974a07 main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

2. Now revert that commit safely (do not delete it):

```
git log --oneline
```

```
git revert <commit-hash>
```

- o Save a screenshot as revert\_commit.png.

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
Revert "Added temporary text file"

This reverts commit 1974a07bbc68f26576c4b5af2c50161ace2a1aff.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch main
# Your branch is up to date with 'origin/main'.
#
# Changes to be committed:
#       deleted:    temp.txt
#
```

```
.git/COMMIT_EDITMSG [unix] (12:17 04/11/2025)  
"~/Desktop/lab_3/.git/COMMIT_EDITMSG" [unix] 13L, 355B
```

```
[main b166faf] Revert "Added temporary text file"
 1 file changed, 1 deletion(-)
 delete mode 100644 temp.txt
```

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab\_3 (main)  
\$

3. Push the revert commit:

git push origin main

- Save a screenshot as revert\_push.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 266 bytes | 266.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/AbihaNadeem005/lab_3.git
  1974a07..b166faf  main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

---

### Task 9 – Force Push (With Caution)

⚠ Use this only in your own branch (never on main or develop).

1. Create a new branch:

git checkout -b test-force

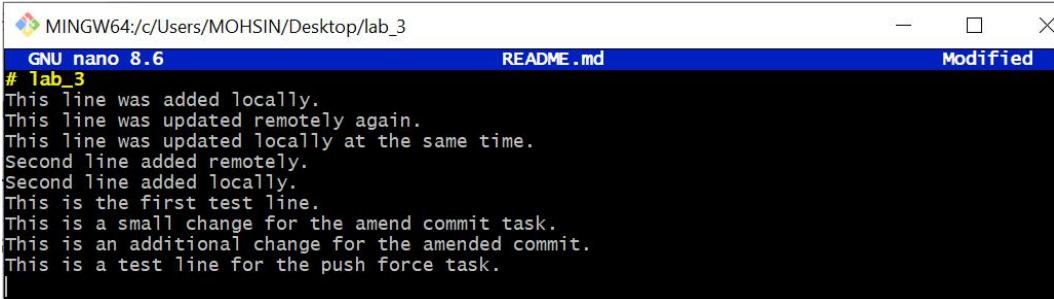
- Save a screenshot as new\_branch.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git checkout -b test-force
Switched to a new branch 'test-force'

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ |
```

2. Make and commit a small change.

- Save a screenshot as force\_commit.png.



```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6          README.md      Modified
# Tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is a small change for the amend commit task.
This is an additional change for the amended commit.
This is a test line for the push force task.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ git add .

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ git commit -m "Add test line for force push"
[test-force ee310ed] Add test line for force push
 1 file changed, 2 insertions(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ |
```

### 3. Push it:

```
git push origin test-force
```

- Save a screenshot as push\_force\_branch.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ git push origin test-force
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 366 bytes | 366.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'test-force' on GitHub by visiting:
remote:     https://github.com/AbihaNadeem005/lab_3/pull/new/test-force
remote:
To https://github.com/AbihaNadeem005/lab_3.git
 * [new branch]      test-force -> test-force

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$
```

### 4. Perform a hard reset:

```
git reset --hard HEAD~1
```

- Save a screenshot as hard\_reset\_force.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ git reset --hard HEAD~1
HEAD is now at b166faf Revert "Added temporary text file"

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$
```

5. Push again

git push origin test-force

- Rejected by remote repository
- Save a screenshot as normal\_push.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ git push origin test-force
To https://github.com/AbihaNadeem005/lab_3.git
 ! [rejected]      test-force -> test-force (non-fast-forward)
error: failed to push some refs to 'https://github.com/AbihaNadeem005/lab_3.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. If you want to integrate the remote changes,
hint: use 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ |
```

6. Now force-push to remote:

git push origin test-force --force

- Save a screenshot as force\_push.png.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$ git push origin test-force --force
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
 + ee310ed...b166faf test-force -> test-force (forced update)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (test-force)
$
```

The screenshot shows a GitHub repository page for 'lab\_3'. At the top, there's a yellow banner indicating 'test-force had recent pushes 2 minutes ago'. Below the banner, the repository details show 'main' branch, '2 Branches', and '0 Tags'. A search bar and file navigation buttons ('Go to file', 'Add file', 'Code') are present. The repository has 15 commits from 'AbihaNadeem005'. The commit history includes:

- Revert "Added temporary text file" (b166faf) - 30 minutes ago
- Added .gitignore to ignore textfiles directory (13 hours ago)
- Fix log message (40 minutes ago)

The 'README' file is shown with its content:

```
lab_3

This line was added locally. This line was updated remotely again. This line was updated locally at the same time. Second line added remotely. Second line added locally. This is the first test line. This is a small change for the amend commit task. This is an additional change for the amended commit.
```

On the right side, there are sections for 'About' (no description), 'Readme', 'Activity', 'Stars' (0 stars), 'Watching' (0 watching), 'Forks' (0 forks), 'Releases' (no releases published), and 'Packages' (no packages published).

Link: [https://github.com/AbihaNadeem005/lab\\_3](https://github.com/AbihaNadeem005/lab_3)

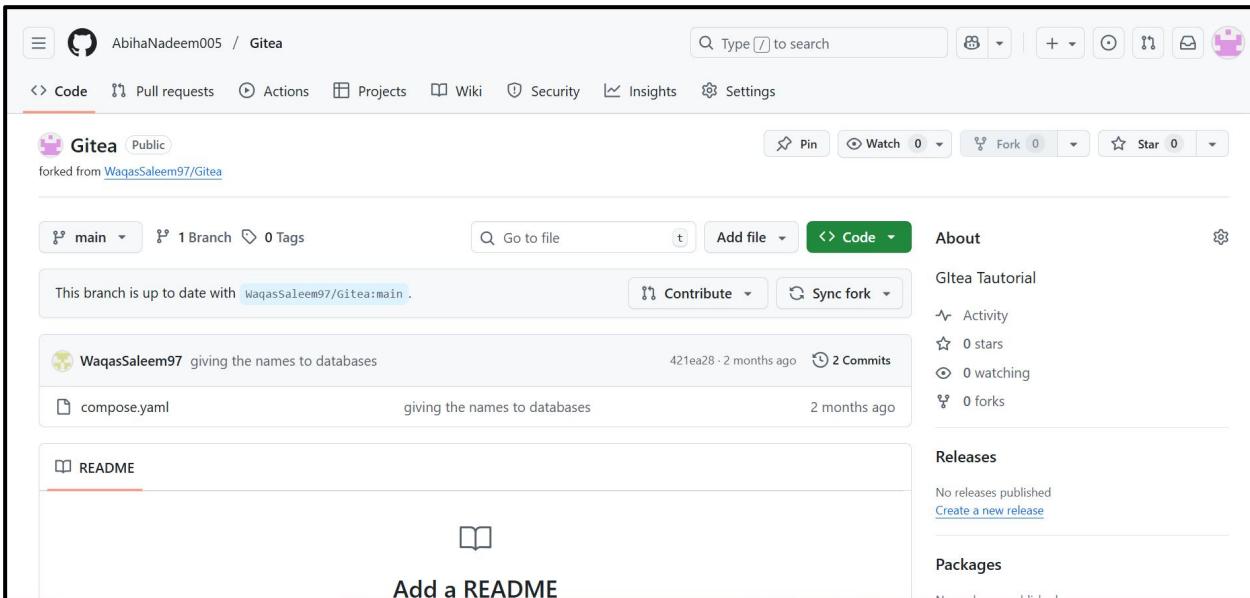
---

## Task 10 – Running Gitea in GitHub Codespaces via Docker Compose

This task will guide you through forking a repository, running a full web application in a GitHub Codespace, and interacting with Gitea (a self-hosted Git service) inside Codespaces.

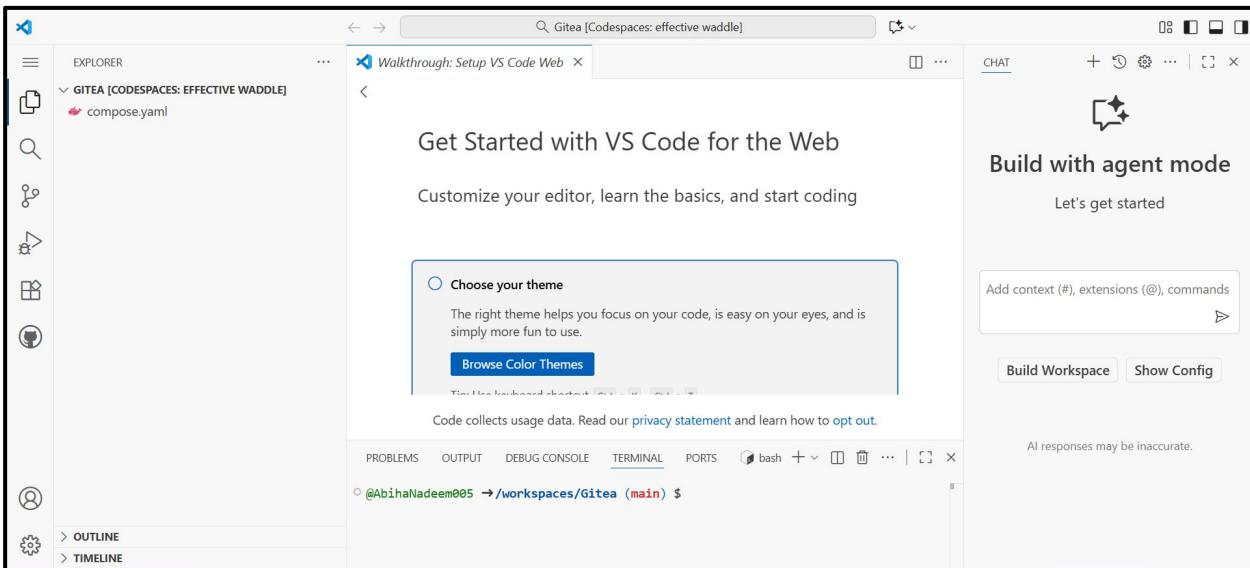
### Steps

1. Fork the Gitea Repository
  - Go to [Gitea](#).
  - Click the **Fork** button at the top right and fork it into your own GitHub account.
  - Save a screenshot of your forked repository page as `forked_gitea.png`.



## 2. Open the Forked Repo in GitHub Codespaces

- On your forked repo, click **Code > Codespaces > Create codespace on main**.
- Save a screenshot of the Codespace loading as `codespace_loading.png`.



## 3. Start Gitea with Docker Compose

- In the Codespace terminal, run:

`docker compose up -d`

- Wait for the containers to start.

- Save a screenshot of the terminal showing the containers running as docker\_up.png.

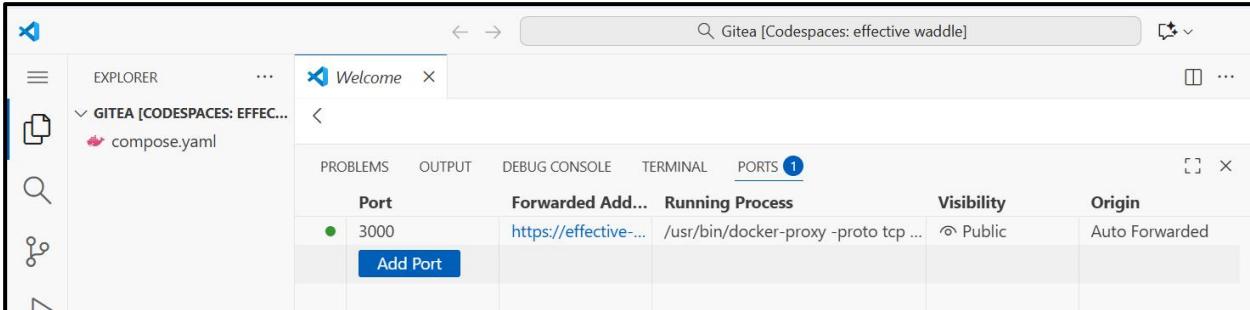
A screenshot of the Visual Studio Code for the Web interface. The title bar shows "Gitea [Codespaces: effective waddle]". The left sidebar has icons for Explorer, Search, and Outline. The main area shows a "Welcome" page with the heading "Get Started with VS Code for the Web" and sub-instruction "Customize your editor, learn the basics, and start coding". Below this is a terminal window showing the command "@AbihaNadeem@05 → /workspaces/Gitea (main) \$ docker compose up -d" followed by a list of 20 "Running 0/2" entries with times from 1.2s to 2.6s. The bottom status bar shows "Codespaces: effective waddle", "main", and "File Layout: US". On the right, there's a "CHAT" section with a message "Build with agent mode" and "Let's get started", and a "Build Workspace" button.

The screenshot shows the VS Code for the Web interface. The left sidebar has icons for Explorer, Search, and Outline. The Explorer view shows a folder named 'GITEA [CODESPACES: EFFECTIVE WADDLE]' containing a file named 'compose.yaml'. The main area is titled 'Welcome' with the sub-section 'Get Started with VS Code for the Web'. Below it, a message says 'Customize your editor, learn the basics, and start coding'. The bottom half of the screen is a terminal window. The terminal title bar says '@AbihaNadeem@05 → /workspaces/Gitea (main) \$ docker compose up -d'. The terminal output shows the following log entries:

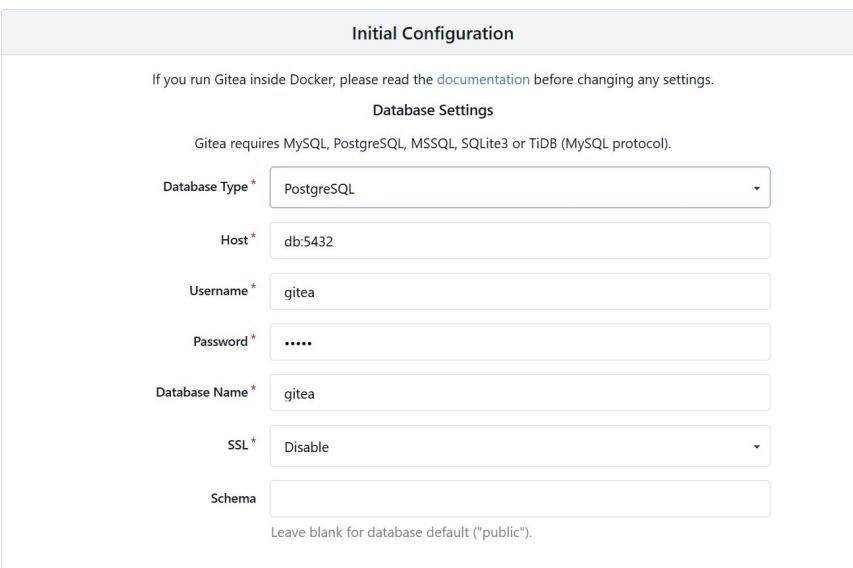
```
@AbihaNadeem@05 → /workspaces/Gitea (main) $ docker compose up -d
[+] Running 0/2
[+] Running 17/17
  ✓ db Pulled
  ✓ gitea Pulled
[+] Running 5/5
  ✓ Network webnet      Created   0.1s
  ✓ Volume "gitea"       Created   0.0s
  ✓ Volume "gitea_postgres" Created   0.0s
  ✓ Container gitea_db   Started   1.7s
  ✓ Container gitea      Started   1.7s
```

#### 4. Access Gitea Web Interface

- In Codespaces, forward **port 3000** (see Codespaces port forwarding UI).
  - Change the Visibility of **port 3000** from **private** to **public**.



- Click the forwarded port 3000 link to open Gitea in your browser tab.
- Save a screenshot of the Gitea install page as gitea\_install\_page.png.



Initial Configuration

If you run Gitea inside Docker, please read the [documentation](#) before changing any settings.

**Database Settings**

Gitea requires MySQL, PostgreSQL, MSSQL, SQLite3 or TiDB (MySQL protocol).

|                 |                                              |
|-----------------|----------------------------------------------|
| Database Type * | PostgreSQL                                   |
| Host *          | db:5432                                      |
| Username *      | gitea                                        |
| Password *      | .....                                        |
| Database Name * | gitea                                        |
| SSL *           | Disable                                      |
| Schema          | Leave blank for database default ("public"). |

## 5. Install Gitea

- Fill out the installation form, providing an **admin username** and **password** (e.g., admin / yourpassword).
- Save a screenshot of the completed setup (show the admin account setup) as admin\_setup.png.

Creating an administrator account is optional. The first registered user will automatically become an administrator.

Administrator Username

admin

Email Address

23-22411-001@se.fjwu.edu.pk

Password

.....

Confirm Password

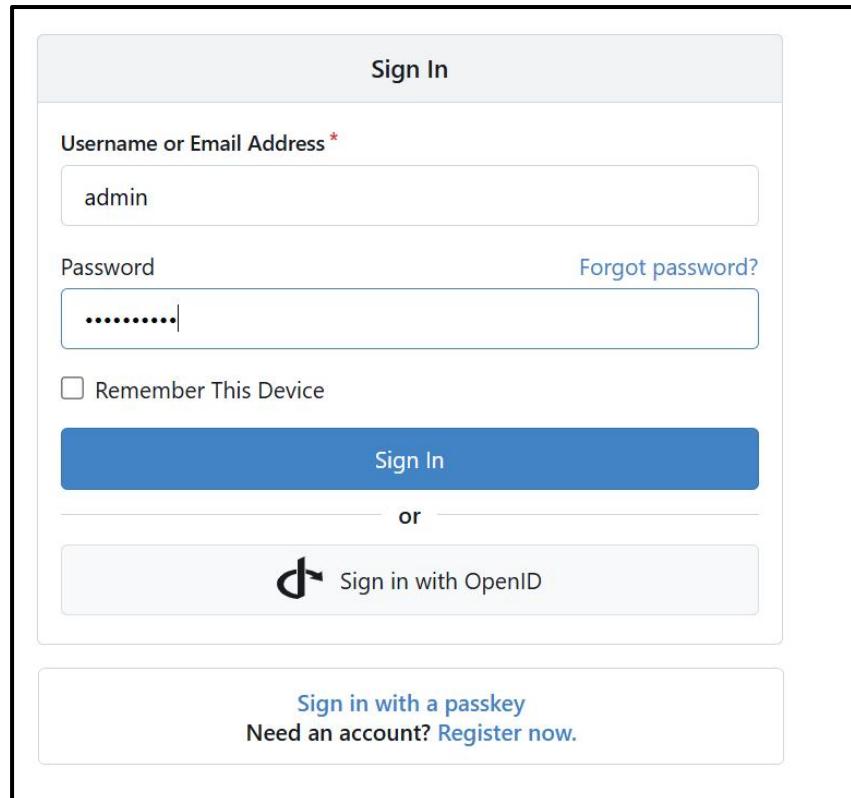
.....

These configuration options will be written into: `/data/gitea/conf/app.ini`

[Install Gitea](#)

## 6. Log In to Gitea

- Use your admin credentials to log in.
- Save a screenshot of the Gitea dashboard after login as `gitea_dashboard.png`.



The screenshot shows the Gitea Sign In interface. At the top, it says "Sign In". Below that is a field labeled "Username or Email Address \*". Inside the field, the text "admin" is entered. To the right of this field is a "Forgot password?" link. Below the username field is a "Password" field containing ".....". To the right of the password field is a "Remember This Device" checkbox, which is unchecked. Below these fields is a large blue "Sign In" button. Underneath the sign-in area, there is a horizontal line with the word "or" in the center. Below this line is a "Sign in with OpenID" button, featuring the OpenID logo (a stylized person icon) and the text "Sign in with OpenID". At the bottom of the page, there is a box containing the text "Sign in with a passkey" and "Need an account? [Register now.](#)".

## 7. Create a New Repository in Gitea

- Click **New Repository** or + > **New Repository** in Gitea.
- Fill out the repository details and create it.
- Save a screenshot of the new repository page as `gitea_new_repo.png`.

The screenshot shows the 'New Repository' creation interface. At the top, there's a header bar with links for Issues, Pull Requests, Milestones, and Explore. Below that is a 'New Repository' form. The 'Owner' field is set to 'admin'. The 'Repository Name' field contains 'first-repo'. Under 'Visibility', there's a checkbox for 'Make repository private' which is unchecked. A note says 'Only the owner or the organization members if they have rights, will be able to see it.' There's also a 'Description' field with the placeholder 'Enter short description (optional)' and a 'Template' field with the placeholder 'Select a template.'

The screenshot shows the main repository page for 'admin / first-repo'. The top navigation bar includes links for Issues, Pull Requests, Milestones, Explore, and a user icon. Below the navigation is a header with the repository name 'admin / first-repo' and icons for Code, Issues, Packages, Projects, and Wiki. To the right are buttons for Unwatch (1), Star (0), and Settings. A 'Quick Guide' section provides instructions for cloning the repository with options for New File, Upload File, HTTPS, and SSH, along with the URL `https://effective-waddle-97g9x5qvg9gpc79px-3000.app.github.dev/admin/first-repo.git`. Below this is a 'Creating a new repository on the command line' section containing a code block:

```
touch README.md
git init
git checkout -b main
git add README.md
git commit -m "first commit"
git remote add origin https://effective-waddle-97g9x5qvg9gpc79px-3000.app.github.dev/admin/first-repo.git
git push -u origin main
```

At the bottom, there's a 'Pushing an existing repository from the command line' section.

Link: <https://effective-waddle-97g9x5qvg9gpc79px-3000.app.github.dev/admin/first-repo>

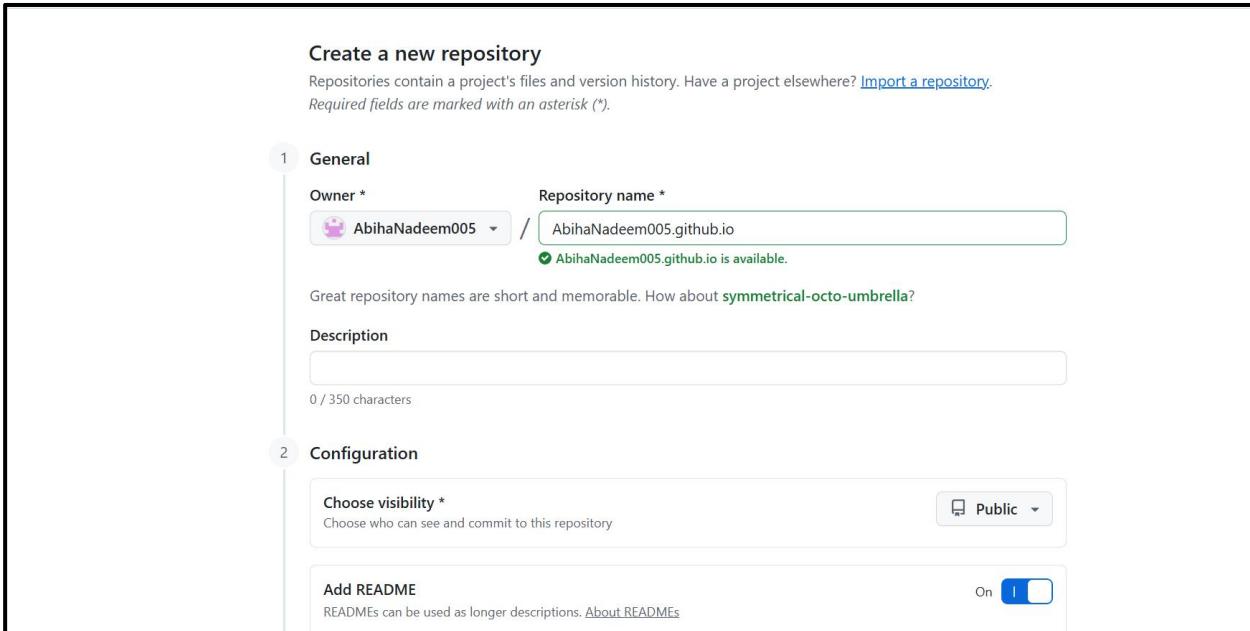
## Task 11 – Creating a GitHub Pages Portfolio Site

This task will guide you through creating a personal portfolio/static site using GitHub Pages.

### Steps

#### 1. Create a GitHub Pages Repository

- Go to GitHub and create a new repository named <your-username>.github.io (e.g., if your username is WaqasSaleem97, the repo should be WaqasSaleem97.github.io).
- Make sure the repository is **public**.
- Save a screenshot of the newly created repository as `github_pages_repo.png`.



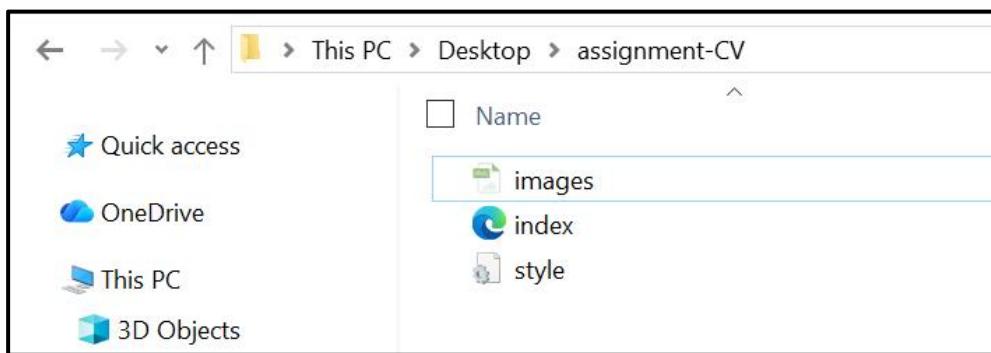
The screenshot shows the 'Create a new repository' page on GitHub. It's a two-step process:

- Step 1: General**
  - Owner \***: AbihaNadeem005
  - Repository name \***: AbihaNadeem005.github.io (highlighted in red)
  - A note says: "AbihaNadeem005.github.io is available."
  - Text below: "Great repository names are short and memorable. How about `symmetrical-octo-umbrella?`"
  - Description**: A large text input field with placeholder "0 / 350 characters".
- Step 2: Configuration**
  - Choose visibility \***: Public (highlighted in red)
  - Add README**: On (highlighted in red)

The screenshot shows a GitHub repository page for 'AbihaNadeem005.github.io'. The repository is public and contains one branch ('main') and one commit ('Initial commit' by AbihaNadeem005). The README file is present. The repository has 0 stars, 0 forks, and 0 releases.

## 2. Add Static Website Code

- Prepare a simple static website (e.g., your CV or portfolio) in HTML/CSS/JS by using the sample code provided in the repository [Iqra Bashir CV](#).
- Put your files in a local folder.
- Save a screenshot of your local files as local\_static\_site.png.



## 3. Push the Files to GitHub

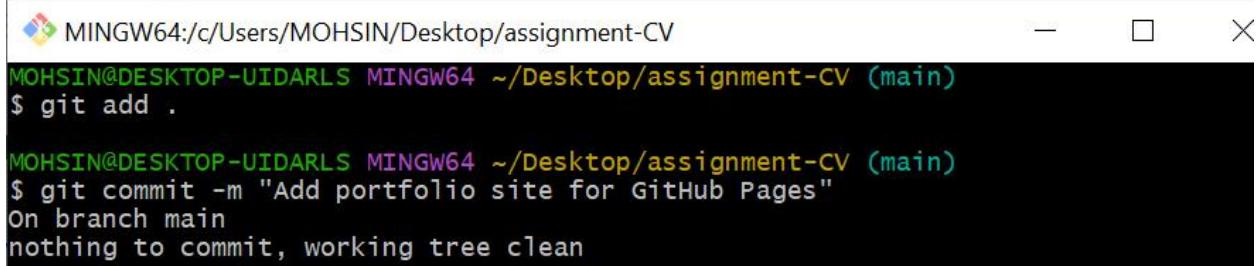
- Initialize git in your folder (if not already):
- `git init`

```
git remote add origin git@github.com:<your-username>/<your-username>.github.io.git
```

- Add and commit your files:
- `git add .`
- `git commit -m "Add portfolio site for GitHub Pages"`

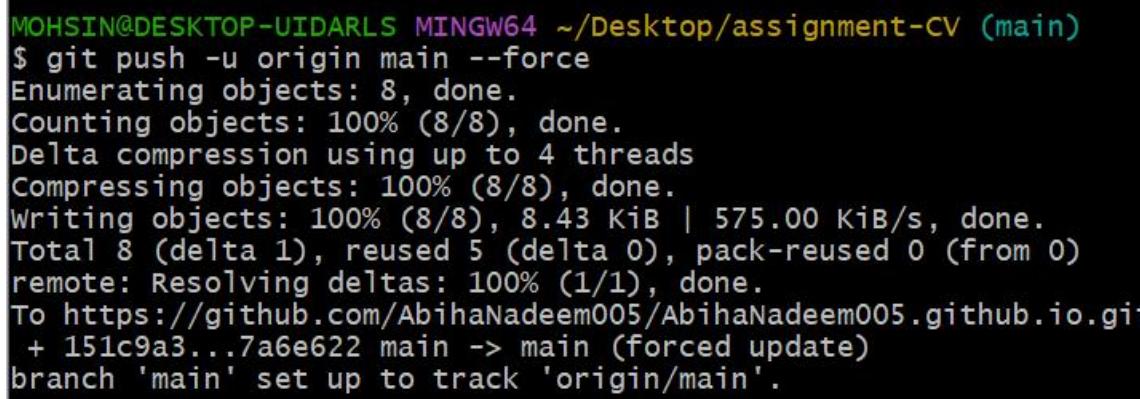
```
git push -u origin main
```

- Save a screenshot of your terminal showing a successful push as push\_static\_site.png.



```
MINGW64:/c/Users/MOHSIN/Desktop/assignment-CV
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/assignment-CV (main)
$ git add .

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/assignment-CV (main)
$ git commit -m "Add portfolio site for GitHub Pages"
On branch main
nothing to commit, working tree clean
```



```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/assignment-CV (main)
$ git push -u origin main --force
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 8.43 KiB | 575.00 KiB/s, done.
Total 8 (delta 1), reused 5 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/AbihaNadeem005/AbihaNadeem005.github.io.git
 + 151c9a3...7a6e622 main -> main (forced update)
branch 'main' set up to track 'origin/main'.
```

#### 4. Check GitHub Pages Settings

- Go to your repository on GitHub.
- Click **Settings > Pages**.
- Confirm that your site is published and see the link to your live site (e.g., <https://your-username.github.io>).
- Save a screenshot of the Pages settings as github\_pages\_settings.png.

The screenshot shows the GitHub Pages settings page for the repository 'AbihaNadeem005'. The main heading is 'GitHub Pages'. Below it, a message states: 'GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.' A status bar indicates: 'Your site is live at <https://abihanadeem005.github.io/>. Last deployed by  AbihaNadeem005 4 minutes ago.' There are two buttons: 'Visit site' and 'Unpublish site'. On the left, there's a sidebar with sections like 'General', 'Access', 'Collaborators', 'Moderation options', 'Code and automation' (with 'Branches', 'Tags', 'Rules', 'Actions', 'Models', 'Webhooks', 'Copilot', 'Environments'), and 'Preview'. The 'Build and deployment' section shows the site is built from the 'main' branch. A note says: 'Learn how to [add a Jekyll theme](#) to your site.'

## 5. Visit Your Live Site

- Open your GitHub Pages site in the browser.
- Save a screenshot of your live site as live\_site.png.

The screenshot shows a resume template on a light green background. It features a circular profile picture of a person wearing a blue hijab. The name 'Abiha Nadeem' is displayed in large blue font, followed by 'Software Engineering Student' in a smaller blue font. To the right, contact information is listed: a phone icon with '+923017399512', an envelope icon with 'abihanadeem2005@gmail.com', and a location pin icon with 'Peshawar Road, Rawalpindi'. Below the profile, details are provided: 'Nationality: Pakistani', 'Date of birth: 5-10-2005', 'Gender: Female', and 'Marital Status: Single'. A horizontal line separates this from the 'Objective' section, which contains a bold blue heading 'Objective' and a descriptive paragraph about the author's goals and qualities. Another horizontal line separates this from the 'Experience' section, which contains a bold blue heading 'Experience'.

Link: <https://github.com/AbihaNadeem005/AbihaNadeem005.github.io/settings/pages>

<https://abihanadeem005.github.io/>

## Exam Evaluation Questions

### 1. Local vs Remote Conflict Resolution

Steps:

1. On GitHub, edit a file (e.g., README.md) and commit the change.

- o Screenshot as Q1\_remote\_edit.png

The screenshot shows a GitHub repository interface. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation bar, the repository path is shown as AbihaNadeem005 / lab\_3. The main tab is 'Code' and the selected file is 'lab\_3 / README.md'. A commit card for 'Update README.md' by AbihaNadeem005 is displayed, showing the commit message and the date (a45d6d1 · now). The commit details show a single line of text: 'lab\_3'. Below the commit, the file content is shown in a code editor view. The content of the README.md file is:  
This line was added locally. This line was updated remotely again. This line was updated locally at the same time. Second line added remotely.  
Second line added locally. This is the first test line. This is a small change for the amend commit task. This is an additional change for the amended commit. This is Lab 3.

2. On your local machine, edit the same file differently (Avoid Conflict) and commit.

- o Screenshot as Q1\_local\_edit.png

The screenshot shows a terminal window titled 'MINGW64:/c/Users/MOHSIN/Desktop/lab\_3'. The window title bar also displays 'GNU nano 8.6' and the file name 'README.md'. The status bar at the bottom right indicates the file is 'Modified'. The terminal content shows the following text:  
# Tab\_3  
This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is the first test line.  
This is a small change for the amend commit task.  
This is an additional change for the amended commit.  
This is a local change.

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Local edit"
[main ee8aecf] Local edit
 1 file changed, 1 insertion(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

3. Try to push your local commit and observe the error.

- o Screenshot as Q1\_push\_error.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
To https://github.com/AbihaNadeem005/lab_3.git
 ! [rejected]      main -> main (fetch first)
error: failed to push some refs to 'https://github.com/AbihaNadeem005/lab_3.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

4. Resolve the conflict by running git pull (merge), then push.

- o Screenshot as Q1\_merge\_resolution.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 970 bytes | 51.00 KiB/s, done.
From https://github.com/AbihaNadeem005/lab_3
 b166faf..a45d6d1 main      -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
```

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6          README.md
# Tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is a small change for the amend commit task.
This is an additional change for the amended commit.
<<<<< HEAD
This is a local change.
_____
This is Lab 3.
>>>>> a45d6d117d0d641b55d88daaeb10233aaa084e7e
```

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6          README.md          Modified
# Tab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is a small change for the amend commit task.
This is an additional change for the amended commit.
This is a local change.
This is Lab 3.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|MERGING)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|MERGING)
$ git commit -m "Resolve merge conflict"
[main 75c3dae] Resolve merge conflict

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 638 bytes | 638.00 KiB/s, done.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/AbihaNadeem005/lab_3.git
  a45d6d1..75c3dae main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

A screenshot of a GitHub repository page for 'AbihaNadeem005 / lab\_3'. The 'Code' tab is selected. A pull request from 'AbihaNadeem005' titled 'Resolve merge conflict' is shown, last updated 75c3dae 1 minute ago. The commit message is 'Resolve merge conflict'. The README.md file contains the following content:

```
lab_3

This line was added locally. This line was updated remotely again. This line was updated locally at the same time. Second line added remotely.  
Second line added locally. This is the first test line. This is a small change for the amend commit task. This is an additional change for the  
amended commit. This is a local change. This is Lab 3.
```

5. Repeat with another remote/local change, but resolve using git pull --rebase, then push.

- Screenshot as Q1\_rebase\_resolution.png

A screenshot of a GitHub repository page for 'AbihaNadeem005 / lab\_3'. The 'Code' tab is selected. A pull request from 'AbihaNadeem005' titled 'Update README.md' is shown, last updated 9bd0305 now. The commit message is 'Update README.md'. The README.md file contains the following content:

```
lab_3

This line was added locally. This line was updated remotely again. This line was updated locally at the same time. Second line added remotely.  
Second line added locally. This is the first test line. This is a small change for the amend commit task. This is an additional change for the  
amended commit. This is a local change. This is Lab 3. Again this is a Lab 3.
```

A screenshot of a terminal window titled 'MINGW64:/c/Users/MOHSIN/Desktop/lab\_3'. The window shows the 'GNU nano 8.6' editor with the file 'README.md' open. The status bar indicates the file is 'Modified'. The content of the file is:

```
# Tab_3  
This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is the first test line.  
This is a small change for the amend commit task.  
This is an additional change for the amended commit.  
This is a local change.  
This is Lab 3.  
Again this is a local change.
```

```
MINGW64:/c/Users/MOHSIN/Desktop/lab_3
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ nano README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Local edit again"
[main 59ce0c5] Local edit again
 1 file changed, 1 insertion(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
To https://github.com/AbihaNadeem005/lab_3.git
 ! [rejected]          main -> main (fetch first)
error: failed to push some refs to 'https://github.com/AbihaNadeem005/lab_3.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull --rebase
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 981 bytes | 65.00 KiB/s, done.
From https://github.com/AbihaNadeem005/lab_3
 75c3dae..9bd0305 main      -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: could not apply 59ce0c5... Local edit again
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicted_files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
hint: Disable this message with "git config set advice.mergeConflict false"
Could not apply 59ce0c5... # Local edit again

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$ |
```

MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6 README.md

```
# 1ab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is a small change for the amend commit task.
This is an additional change for the amended commit.
This is a local change.
This is Lab 3.
<<<<< HEAD
Again this is a Lab 3.

Again this is a local change.
>>>>> 59ce0c5 (Local edit again)
```

```
[detached HEAD 67d6a25] Local edit again
 1 file changed, 2 insertions(+)
Successfully rebased and updated refs/heads/main.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 355 bytes | 355.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/AbihaNadeem005/lab_3.git
 9bd0305..67d6a25 main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

The screenshot shows a GitHub repository interface. At the top, there's a navigation bar with 'Code' selected. Below it, a header bar shows the repository path 'AbihaNadeem005 / lab\_3' and various project management tabs like 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. A search bar is at the top right. The main content area displays the 'lab\_3 / README.md' file. The file content is identical to the one shown in the terminal above. Below the file content, there's a preview section with tabs for 'Preview', 'Code', and 'Blame'. On the right side of the preview, there are download and edit buttons. At the bottom of the preview area, there's a note about the file being updated 1 minute ago.

```
lab_3

This line was added locally. This line was updated remotely again. This line was updated locally at the same time. Second line added remotely.
Second line added locally. This is the first test line. This is a small change for the amend commit task. This is an additional change for the amended commit. This is a local change. This is Lab 3. Again this is a Lab 3. Again this is a local change.
```

## 2. Manual Merge Conflict Handling

### Steps:

1. On GitHub, change a specific line in a file and commit.

- o Screenshot as Q2\_remote\_conflict\_edit.png

The screenshot shows a GitHub repository interface for a user named AbihaNadeem005. The repository is named 'lab\_3'. The user has updated the README.md file. The commit message is "Update README.md". The commit hash is 7841eca · now. The commit was made by AbihaNadeem005. The file content is as follows:

```
lab_3

This line was added locally. This line was updated remotely again. This line was updated locally at the same time. Second line added remotely.
Second line added locally. This is the first test line. This is a small change for the amend commit task. This is an additional change for the amended commit. This is a local change. This is Lab 3 edited on github. Again this is a Lab 3. Again this is a local change.
```

2. Locally, change the same line differently and commit.

- o Screenshot as Q2\_local\_conflict\_edit.png

The screenshot shows a terminal window titled "MINGW64:/c/Users/MOHSIN/Desktop/lab\_3". The window title bar also shows "GNU nano 8.6" and the file name "README.md". The status bar indicates the file is "Modified". The terminal content is as follows:

```
# 1ab_3
This line was added locally.
This line was updated remotely again.
This line was updated locally at the same time.
Second line added remotely.
Second line added locally.
This is the first test line.
This is a small change for the amend commit task.
This is an additional change for the amended commit.
This is a local change.
This is Lab 3 edited locally.
Again this is a Lab 3.
Again this is a local change.
```

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ nano README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Local conflicting change"
[main 4987282] Local conflicting change
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

3. Try to push your local change and observe the conflict error.

- Screenshot as Q2\_conflict\_push\_error.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
To https://github.com/AbihaNadeem005/lab_3.git
 ! [rejected]      main -> main (fetch first)
error: failed to push some refs to 'https://github.com/AbihaNadeem005/lab_3.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

4. Use git pull --rebase to fetch changes and trigger the conflict.

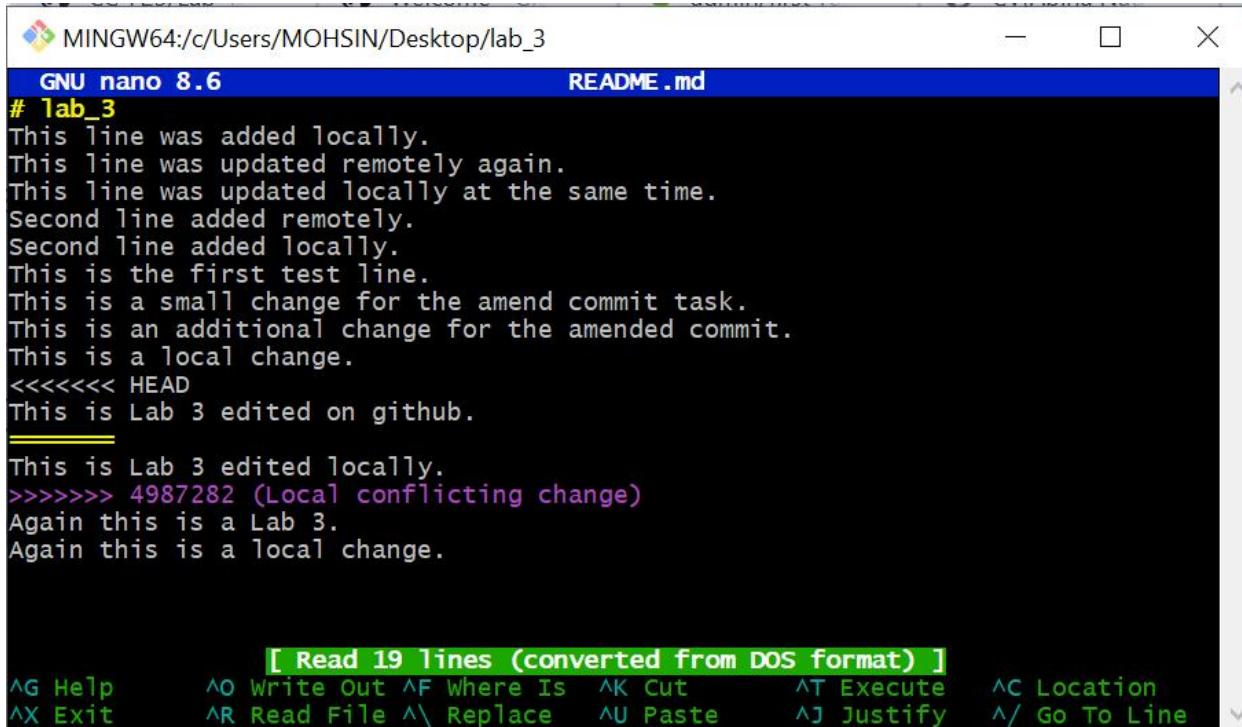
- Screenshot as Q2\_rebase\_conflict.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git pull --rebase
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 977 bytes | 31.00 KiB/s, done.
From https://github.com/AbihaNadeem005/lab_3
 67d6a25..7841eca main      -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: could not apply 4987282... Local conflicting change
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicted_files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
hint: Disable this message with "git config set advice.mergeConflict false"
Could not apply 4987282... # Local conflicting change

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main|REBASE 1/1)
$
```

5. Edit the conflicted file to resolve the conflict manually.

- o Screenshot as Q2\_resolved\_file.png



MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

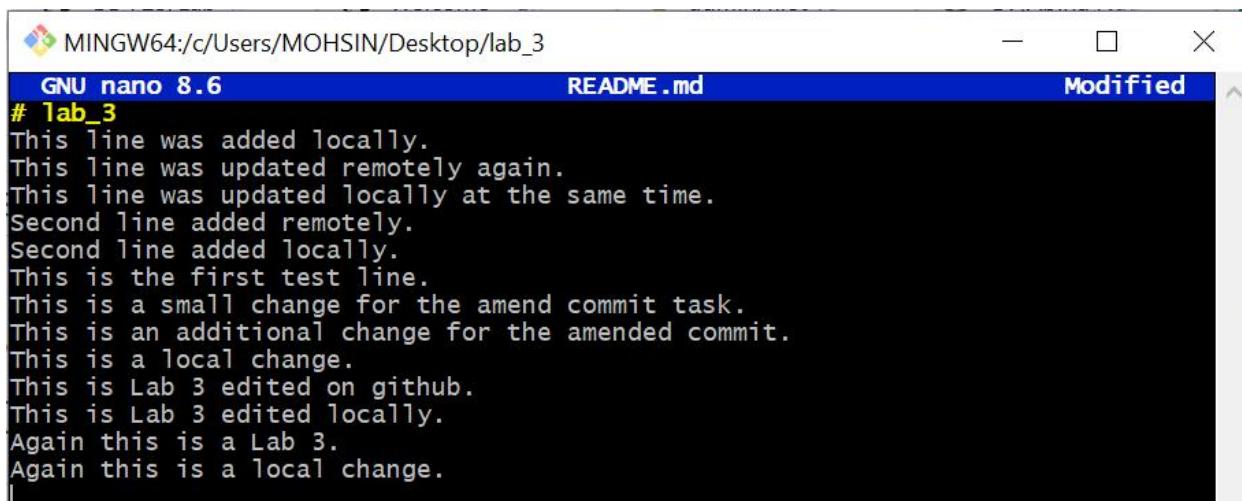
GNU nano 8.6 README.md

# Tab\_3

This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is the first test line.  
This is a small change for the amend commit task.  
This is an additional change for the amended commit.  
This is a local change.  
<<<<< HEAD  
This is Lab 3 edited on github.  
  
This is Lab 3 edited locally.  
>>>>> 4987282 (Local conflicting change)  
Again this is a Lab 3.  
Again this is a local change.

[ Read 19 Lines (converted from DOS format) ]

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location  
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line



MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6 README.md Modified

# Tab\_3

This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is the first test line.  
This is a small change for the amend commit task.  
This is an additional change for the amended commit.  
This is a local change.  
This is Lab 3 edited on github.  
This is Lab 3 edited locally.  
Again this is a Lab 3.  
Again this is a local change.

6. Mark the conflict resolved (git add <file>, git rebase --continue) and push.

- o Screenshot as Q2\_resolution\_complete.png

```
[detached HEAD 48d6b8c] Local conflicting change
 1 file changed, 1 insertion(+)
Successfully rebased and updated refs/heads/main.

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 358 bytes | 119.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/AbihaNadeem005/lab_3.git
 7841eca..48d6b8c main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

---

### 3. Managing Ignored and Tracked Files

#### Steps:

1. Create a new folder (e.g., DocFiles) and add several files inside.
  - o Screenshot as Q3\_folder\_created.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ mkdir DocFiles

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ echo "This is file 1" > DocFiles/doc1.txt
echo "This is file 2" > DocFiles/doc2.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ ls DocFiles
doc1.txt doc2.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

2. Commit and push the folder/files to GitHub.

- o Screenshot as Q3\_files\_pushed.png

```

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add DocFiles
warning: in the working copy of 'DocFiles/doc1.txt', LF will be replaced by CRLF
the next time Git touches it
warning: in the working copy of 'DocFiles/doc2.txt', LF will be replaced by CRLF
the next time Git touches it

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Add DocFiles folder with documents"
[main 7a16695] Add DocFiles folder with documents
 2 files changed, 2 insertions(+)
 create mode 100644 DocFiles/doc1.txt
 create mode 100644 DocFiles/doc2.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 448 bytes | 89.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
 48d6b8c..7a16695 main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |

```

The screenshot shows a GitHub repository named 'lab\_3' owned by 'AbihaNadeem005'. The repository is public. It contains 23 commits. The commits are:

- Add DocFiles folder with documents (7a16695, 1 minute ago)
- DocFiles (1 minute ago)
- .gitignore (yesterday)
- README.md (15 minutes ago)

The repository has 0 stars, 0 forks, and 0 releases. The README file is present.

3. Add the folder to your .gitignore file.

- Screenshot as Q3\_gitignore\_added.png

```

MINGW64:/c/Users/MOHSIN/Desktop/lab_3
GNU nano 8.6 .gitignore Modified
textfiles/*
DocFiles/

```

4. Commit and push the .gitignore update.

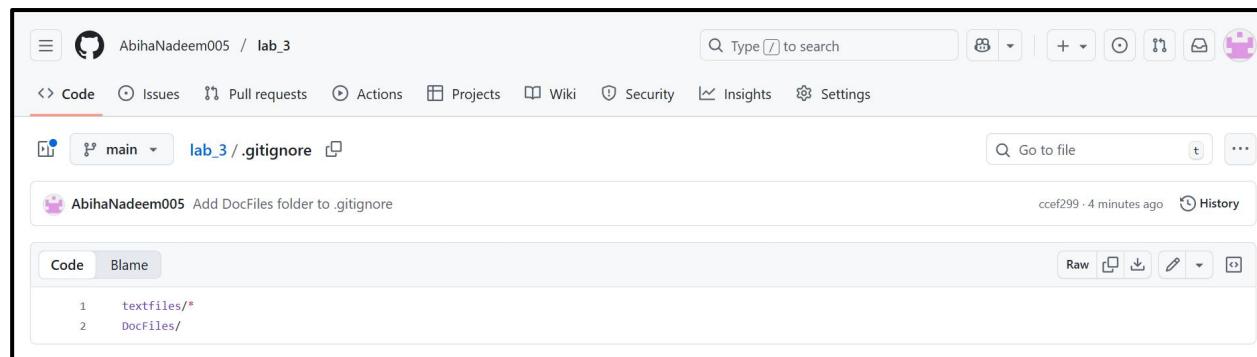
- o Screenshot as Q3\_gitignore\_pushed.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add .gitignore

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Add DocFiles folder to .gitignore"
[main ccef299] Add DocFiles folder to .gitignore
 1 file changed, 2 insertions(+)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 359 bytes | 89.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
 7a16695..cceff299 main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```



5. Remove the folder from tracking using git rm -r --cached <folder>.

- o Screenshot as Q3\_folder\_untracked.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git rm -r --cached DocFiles
rm 'DocFiles/doc1.txt'
rm 'DocFiles/doc2.txt'

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

6. Commit and push the change, then verify the folder is no longer tracked on GitHub.

- Screenshot as Q3\_folder\_removed\_github.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Remove DocFiles from tracking"
[main 4a1fa17] Remove DocFiles from tracking
 2 files changed, 2 deletions(-)
   delete mode 100644 DocFiles/doc1.txt
   delete mode 100644 DocFiles/doc2.txt

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 293 bytes | 146.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AbihaNadeem005/lab_3.git
  ccef299..4a1fa17 main -> main

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

The screenshot shows a GitHub repository named 'lab\_3' owned by 'AbihaNadeem005'. The repository is public. It has 2 branches (main) and 0 tags. The main branch has 25 commits. Recent commits include 'Remove DocFiles from tracking' (4a1fa17), '.gitignore' (4a1fa17), and 'README.md' (4a1fa17). The README file is visible on the right. The repository has 0 stars, 0 forks, and 0 releases.

## 4. Commit History Manipulation and Recovery

### Steps:

1. Make a change and commit it.
  - Screenshot as Q4\_first\_commit.png

MINGW64:/c/Users/MOHSIN/Desktop/lab\_3

GNU nano 8.6 README.md Modified

# Tab\_3

This line was added locally.  
This line was updated remotely again.  
This line was updated locally at the same time.  
Second line added remotely.  
Second line added locally.  
This is the first test line.  
This is a small change for the amend commit task.  
This is an additional change for the amended commit.  
This is a local change.  
This is Lab 3 edited on github.  
This is Lab 3 edited locally.  
Again this is a Lab 3.  
Again this is a local change.  
Change for the task 4|

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location  
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "First commit for task 4"
[main dbc32e3] First commit for task 4
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

2. Make another change and commit again.

- o Screenshot as Q4\_second\_commit.png

The screenshot shows a terminal window with two main sections. The top section is a terminal window titled "MINGW64:/c/Users/MOHSIN/Desktop/lab\_3". It displays the contents of a file named "README.md" which has been modified. The file contains several lines of text, many of which start with "# Tab\_3". The bottom section is a command-line interface where the user is performing Git operations. The commands shown are:

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git add README.md

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Second commit for task 4"
[main 763aac4] Second commit for task 4
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

3. View your commit history.

- o Screenshot as Q4\_commit\_history.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git log --oneline
763aac4 (HEAD -> main) Second commit for task 4
dbc32e3 First commit for task 4
4a1fa17 (origin/main, origin/HEAD) Remove DocFiles from tracking
cceef299 Add DocFiles folder to .gitignore
7a16695 Add DocFiles folder with documents
48d6b8c Local conflicting change
7841eca Update README.md
67d6a25 Local edit again
9bd0305 Update README.md
75c3dae Resolve merge conflict
ee8aecf Local edit
a45d6d1 Update README.md
b166faf (origin/test-force, test-force) Revert "Added temporary text file"
1974a07 Added temporary text file
3063a6d Fix log message
4e57070 Added test line
e855f60 Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fc当地 Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

4. Perform a soft reset (git reset --soft HEAD~1) and observe your file and history.
  - o Screenshot as Q4\_soft\_reset.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git reset --soft HEAD~1

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git log --oneline
dbc32e3 (HEAD -> main) First commit for task 4
4a1fa17 (origin/main, origin/HEAD) Remove DocFiles from tracking
cceef299 Add DocFiles folder to .gitignore
7a16695 Add DocFiles folder with documents
48d6b8c Local conflicting change
7841eca Update README.md
67d6a25 Local edit again
9bd0305 Update README.md
75c3dae Resolve merge conflict
ee8aecf Local edit
a45d6d1 Update README.md
b166faf (origin/test-force, test-force) Revert "Added temporary text file"
1974a07 Added temporary text file
3063a6d Fix log message
4e57070 Added test line
e855f60 Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fcfa Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ |
```

5. Make commit again.
  - o Screenshot as Q4\_third\_commit.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git commit -m "Third commit after soft reset"
[main 6a0709d] Third commit after soft reset
 1 file changed, 1 insertion(+), 1 deletion(-)

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

6. Perform a hard reset (git reset --hard HEAD~1) and observe the changes.
  - o Screenshot as Q4\_hard\_reset.png

```
MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git reset --hard HEAD~1
HEAD is now at dbc32e3 First commit for task 4

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$ git log --oneline
dbc32e3 (HEAD -> main) First commit for task 4
4a1fa17 (origin/main, origin/HEAD) Remove DocFiles from tracking
cceef299 Add DocFiles folder to .gitignore
7a16695 Add DocFiles folder with documents
48d6b8c Local conflicting change
7841eca Update README.md
67d6a25 Local edit again
9bd0305 Update README.md
75c3dae Resolve merge conflict
ee8aecf Local edit
a45d6d1 Update README.md
b166faf (origin/test-force, test-force) Revert "Added temporary text file"
1974a07 Added temporary text file
3063a6d Fix log message
4e57070 Added test line
e855f60 Removed tracked textfiles directory
b527826 Added .gitignore to ignore textfiles directory
b01f99d Added textfiles directory with three files
06b1230 Local conflicting change
ff96fc8 Remote conflicting change
2a8ff3b Resolved README.md merge conflict
3db8be0 Second remote update
2397193 Resolved README.md merge conflict
dc6013f Local update to README
3c06bc8 Remote update to README
87a8ac3 Initial commit

MOHSIN@DESKTOP-UIDARLS MINGW64 ~/Desktop/lab_3 (main)
$
```

\*\*\*\*\*

