

ASDT : EXERCISE

Done by : Mariam Chajia 78241

CONTENT

Create a new remote repository.

Push all commits from Exercise 6a/6b to the remote repository.

Play with the git remote, git fetch, git push, git pull, git branch

Execute fast-forward merge.

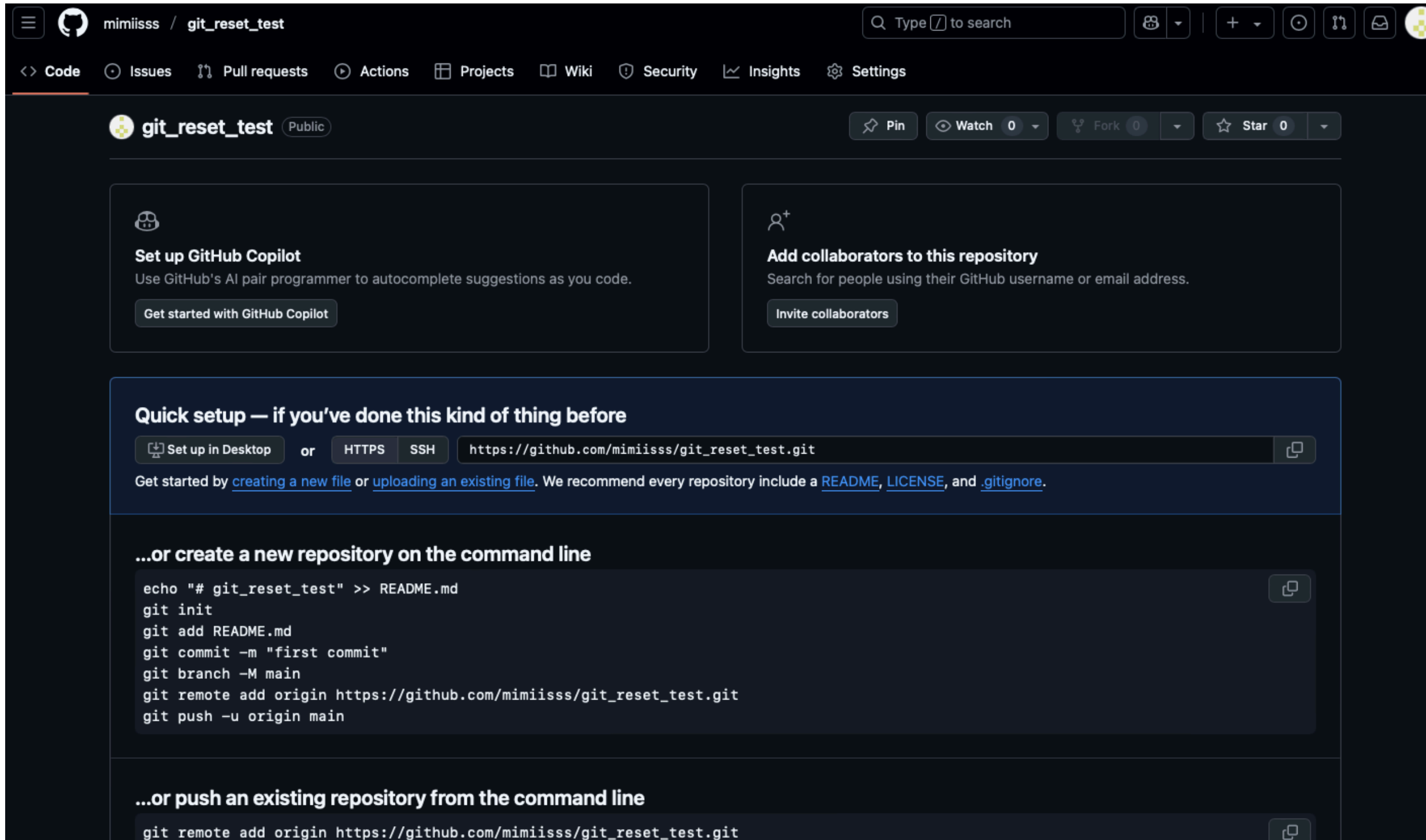
Execute a 3-way merge.

Resolve conflict.

Add to the remote repository and to the Tasks in MS Teams a report showing results of the

Invite me to the repository (username: MansSlowiczek)

1 Create a new remote repository.



The screenshot shows the GitHub interface for a new repository named 'git_reset_test' by user 'mimiisss'. The repository is public. The page offers several options for setting up the repository:

- Set up GitHub Copilot:** Use GitHub's AI pair programmer to autocomplete suggestions as you code. A button 'Get started with GitHub Copilot' is available.
- Add collaborators to this repository:** Search for people using their GitHub username or email address. A button 'Invite collaborators' is available.
- Quick setup — if you've done this kind of thing before:** This section provides a 'Set up in Desktop' button, an 'or' separator, and buttons for 'HTTPS' and 'SSH'. The SSH URL is `https://github.com/mimiisss/git_reset_test.git`. Below this, it says 'Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).'
- ...or create a new repository on the command line:** This section provides a code block with the following commands:

```
echo "# git_reset_test" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/mimiisss/git_reset_test.git
git push -u origin main
```
- ...or push an existing repository from the command line:** This section provides a code block with the following command:

```
git remote add origin https://github.com/mimiisss/git_reset_test.git
```

2 Push all commits from Exercise 6a/6b to the remote repository.

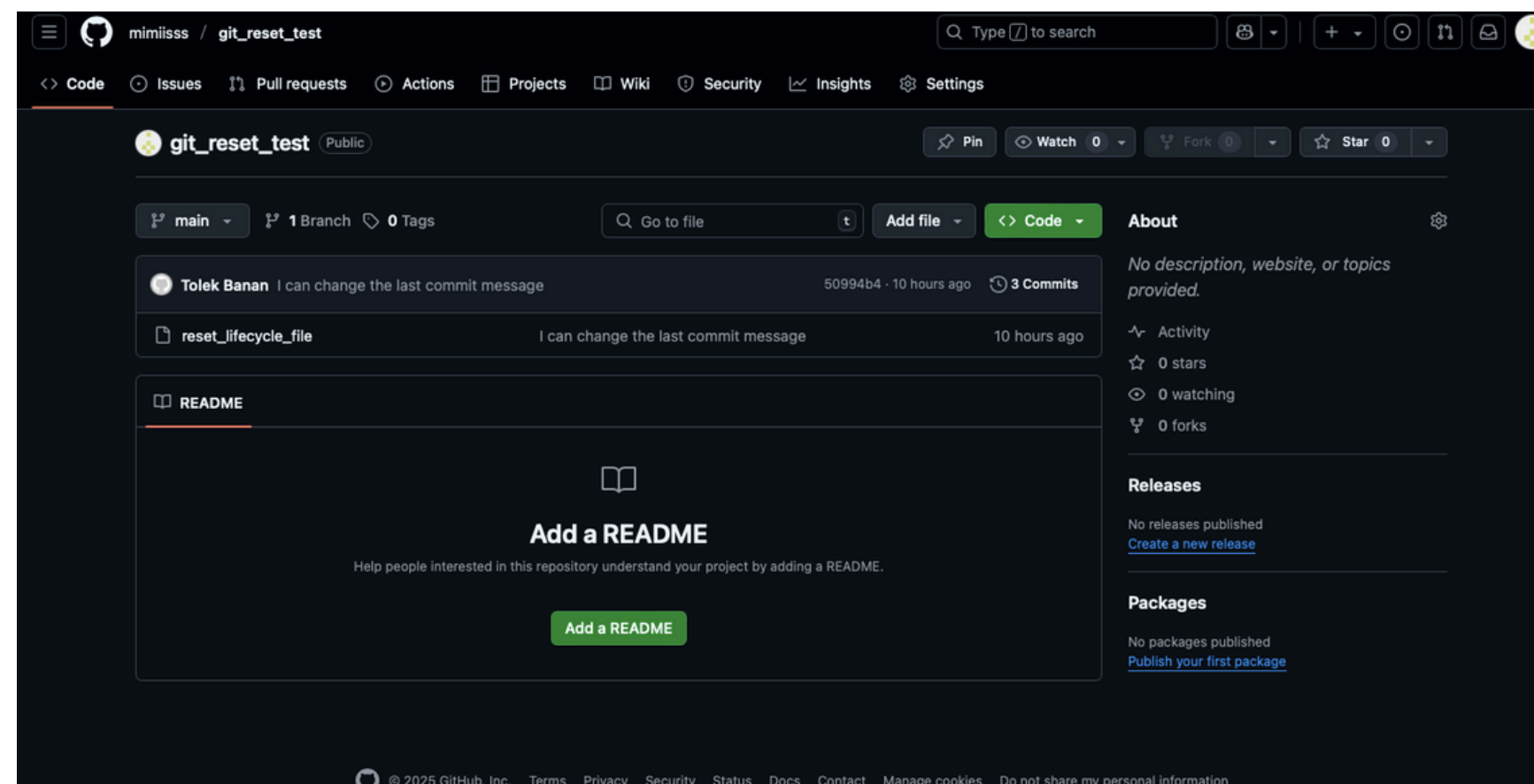
```
[nouveaucompte@macbook-air-de-mimii git_reset_test % cd ~/git_reset_test

[nouveaucompte@macbook-air-de-mimii git_reset_test % git remote add origin https://github.com/mimiisss/git_reset_test.git

[nouveaucompte@macbook-air-de-mimii git_reset_test % git branch -M main

[nouveaucompte@macbook-air-de-mimii git_reset_test % git push -u origin main

Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (7/7), 619 bytes | 206.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/mimiisss/git_reset_test.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
nouveaucompte@macbook-air-de-mimii git_reset_test %
```



3 Play with the git remote, git fetch, git push, git pull, git branch

```
[nouveaucompte@macbook-air-de-mimii git_reset_test % git remote -v
origin  https://github.com/mimiisss/git_reset_test.git (fetch)
origin  https://github.com/mimiisss/git_reset_test.git (push)
[nouveaucompte@macbook-air-de-mimii git_reset_test % git fetch
[nouveaucompte@macbook-air-de-mimii git_reset_test % git pull

Already up to date.
[nouveaucompte@macbook-air-de-mimii git_reset_test % git branch
* main
```

4 Execute fast-forward merge.

```
nouveaucmpte@macbook-air-de-mimii git_reset_test % git checkout -b feature-ff
echo "fast forward content" > ff.txt
git add ff.txt
git commit -m "Add ff.txt for fast-forward demo"
```

```
Switched to a new branch 'feature-ff'
[feature-ff 38d5227] Add ff.txt for fast-forward demo
 1 file changed, 1 insertion(+)
 create mode 100644 ff.txt
```

```
nouveaucmpte@macbook-air-de-mimii git_reset_test % git checkout main
git merge feature-ff
```

```
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
Updating 50994b4..38d5227
Fast-forward
 ff.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 ff.txt
```

```
nouveaucmpte@macbook-air-de-mimii git_reset_test %
```

5 Execute a 3-way merge.

```
nouveaucompte@macbook-air-de-mimii git_reset_test % git checkout -b feature-merge

Switched to a new branch 'feature-merge'
nouveaucompte@macbook-air-de-mimii git_reset_test % echo "Change from feature-merge" >> reset_lifecycle_file
git commit -am "Change in feature-merge"

[feature-merge b9dbcd1] Change in feature-merge
 1 file changed, 1 insertion(+)
nouveaucompte@macbook-air-de-mimii git_reset_test % git checkout main
echo "Change from main" >> reset_lifecycle_file
git commit -am "Change in main"

Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
[main b0124bf] Change in main
 1 file changed, 1 insertion(+)
nouveaucompte@macbook-air-de-mimii git_reset_test % git merge feature-merge

Auto-merging reset_lifecycle_file
CONFLICT (content): Merge conflict in reset_lifecycle_file
Automatic merge failed; fix conflicts and then commit the result.
nouveaucompte@macbook-air-de-mimii git_reset_test %
```

6 Resolve conflict.

```
[nouveaucompte@macbook-air-de-mimii git_reset_test % nano reset_lifecycle_file

[nouveaucompte@macbook-air-de-mimii git_reset_test %
nouveaucompte@macbook-air-de-mimii git_reset_test % git add reset_lifecycle_file
git commit -m "Resolve merge conflict between main and feature-merge"

[main e8c15e0] Resolve merge conflict between main and feature-merge
[nouveaucompte@macbook-air-de-mimii git_reset_test % git push

Counting objects: 12, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (8/8), done.
Writing objects: 100% (12/12), 1.13 KiB | 386.00 KiB/s, done.
Total 12 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/mimiisss/git_reset_test.git
   50994b4..e8c15e0  main -> main
nouveaucompte@macbook-air-de-mimii git_reset_test %
```


7 Add to the remote repository and to the Tasks in MS Teams a report showing results of the Exercise 7.

```
nouveaucpte@macbook-air-de-mimii git_reset_test % git remote add origin https://github.com/mimiisss/git_reset_test.git
git branch -M main
git push -u origin main

fatal: remote origin already exists.
Branch 'main' set up to track remote branch 'main' from 'origin'.
Everything up-to-date
nouveaucpte@macbook-air-de-mimii git_reset_test % git add reset_lifecycle_file
git commit -m "Resolve merge conflict between main and feature-merge"
git push

On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  new_file

nothing added to commit but untracked files present
Everything up-to-date
```