

Міністерство освіти і науки, молоді і спорту України
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій

Кафедра ІОСУ

Лабораторна робота №6.1
З дисципліни “Основи комп'ютерних наук”
На тему: “Основи Роботи з GIT та GitHub”

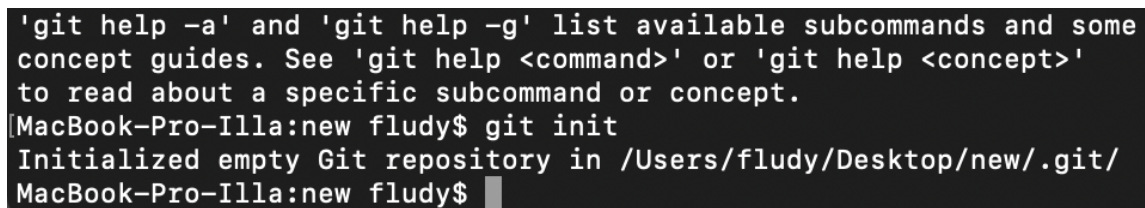
Виконав
студент групи КНШІ-11
Корецький Ілля

Тернопіль 2023

Мета: Ознайомлення студентів з основами системи контролю версій GIT та платформи GitHub. Розробка навичок управління версіями коду та організації спільної роботи над проектом.

Хід роботи

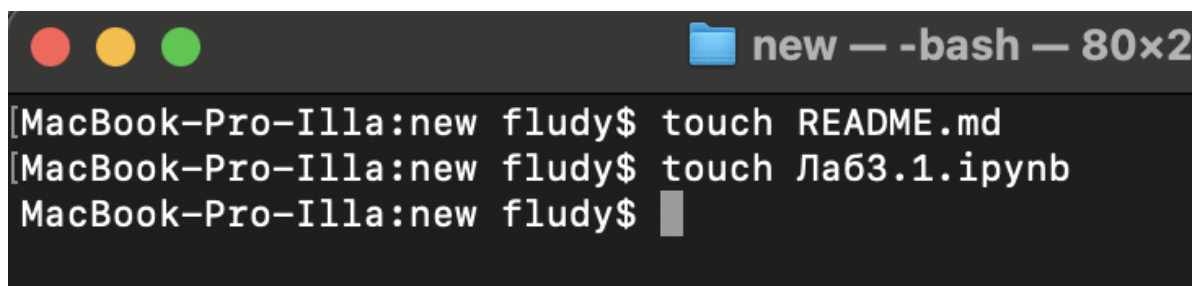
1. Створюю нову папку для репозиторію та ініціалізую її як GIT репозиторій (рисунок 1).



```
'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
[MacBook-Pro-Illa:new fludy$ git init
Initialized empty Git repository in /Users/fludy/Desktop/new/.git/
MacBook-Pro-Illa:new fludy$
```

Рисунок 1 – Вікно консолі Git Bash з введеними командами.

2. Додаю кілька файлів до репозиторію (README.md, Лаб3.1 ірυνb та звіт по ЛБ6) (рисунок 2).



```
new — -bash — 80x2
[MacBook-Pro-Illa:new fludy$ touch README.md
[MacBook-Pro-Illa:new fludy$ touch Лаб3.1.ірυνb
MacBook-Pro-Illa:new fludy$
```

Рисунок 2 – Вікно Git Bash з введеними командами для створення файлів.

3. Роблю зміни в файлі README.md (рисунок 3).

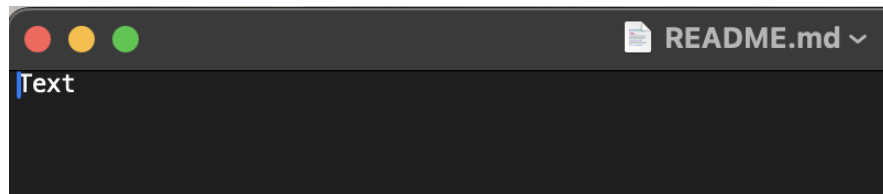


Рисунок 3 – Вікно файла README.md .

4. Використовую команди git add, git commit для збереження змін (рисунок 4).

```
MacBook-Pro-Illa:~ fludy$ git add README.md
MacBook-Pro-Illa:~ fludy$ git add Ла63.1.ipynb
MacBook-Pro-Illa:~ fludy$ git commit -m 'Внесенно зміни в файл README'
[master (root-commit) 22bb059] Внесенно зміни в файл README
  Committer: Ілля <fludy@MacBook-Pro-Illa.local>
  Your name and email address were configured automatically based
  on your username and hostname. Please check that they are accurate.
  You can suppress this message by setting them explicitly. Run the
  following command and follow the instructions in your editor to edit
  your configuration file:

    git config --global --edit

  After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

  2 files changed, 1 insertion(+)
  create mode 100644 README.md
  create mode 100644 Ла63.1.ipynb
```

Рисунок 4 – Вікно Git Bash з введеними командами для збереження змін.

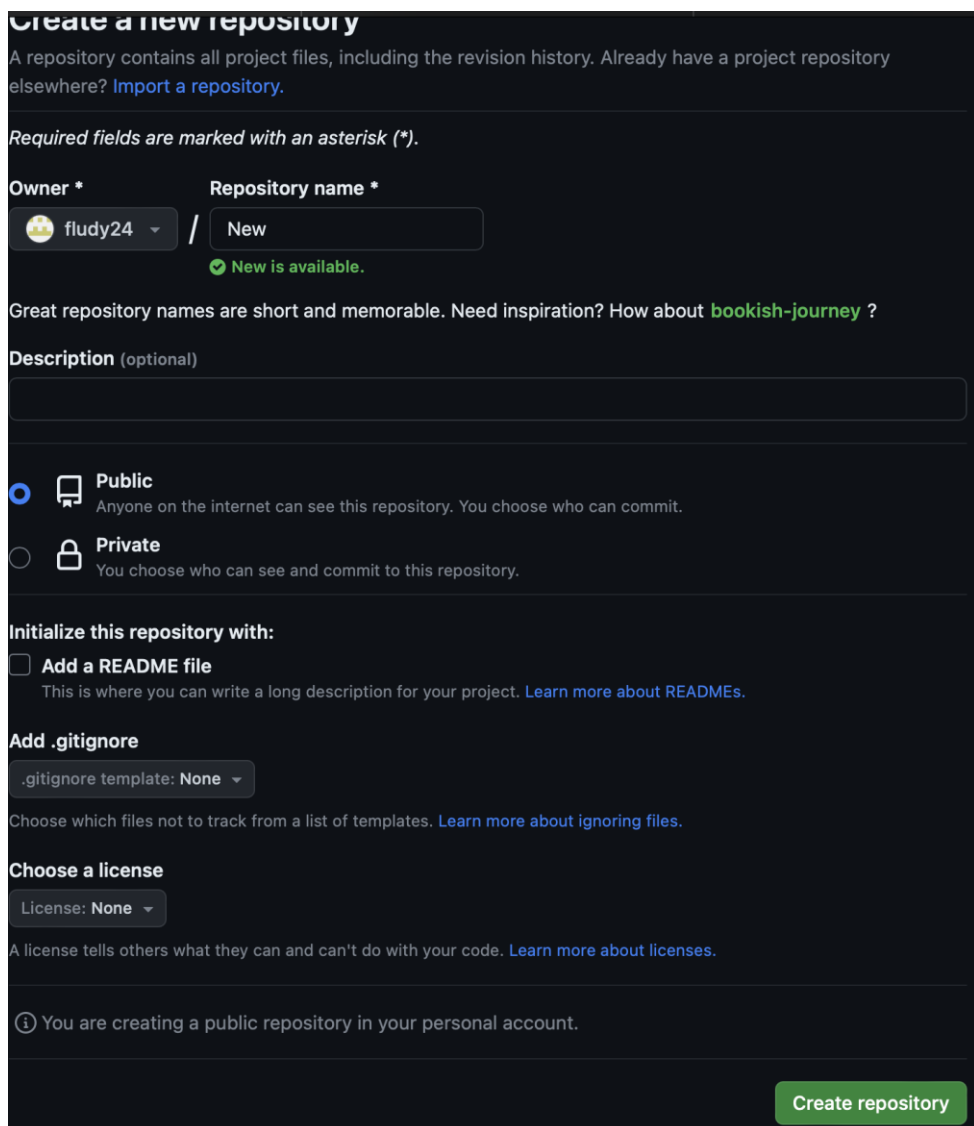
5. Переглядаю історію комітів за допомогою git log (рисунок 5).

```
MacBook-Pro-Illa:~ fludy$ git log
commit 22bb059f6bc4f0d13ab3a73e489115fcee301ed0 (HEAD -> master)
Author: Ілля <fludy@MacBook-Pro-Illa.local>
Date:   Mon Dec 11 17:21:43 2023 +0200

    Внесенно зміни в файл README
MacBook-Pro-Illa:~ fludy$
```

Рисунок 5 – Вікно Git Bash з введеною командою git log.

6. Створюю новий репозиторій на GitHub (рисунок 6).




Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).


Owner * **Repository name ***


 fludy24 /

✔ New is available.

Great repository names are short and memorable. Need inspiration? How about [bookish-journey](#) ?

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore


.gitignore template: **None**

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: **None**

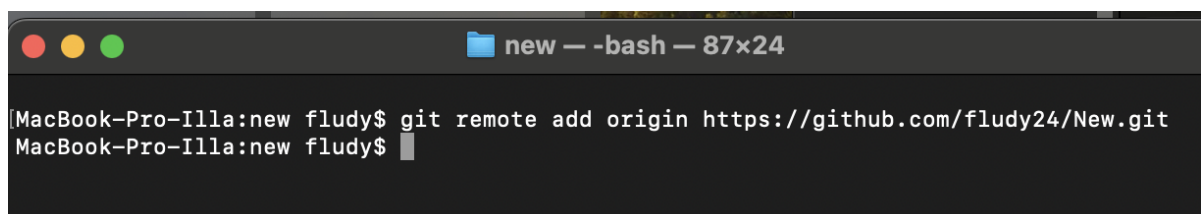
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.

[Create repository](#)

Рисунок 6 – Вікно створення репозиторію на сайті GitHub

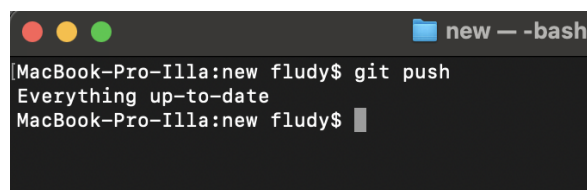
7. Зв'язую локальний репозиторій з віддаленим на GitHub за допомогою `git remote add origin [https://github.com/Finsar/Git.git]` (рисунок 7).



```
MacBook-Pro-Illa:new fludy$ git remote add origin https://github.com/fludy24/New.git
MacBook-Pro-Illa:new fludy$
```

Рисунок 7 – Вікно Git Bash з введеною командою `git remote add origin [URL]`.

8. Ввожу `git push` для відправлення локальних змін на GitHub (рисунок 8).



```
MacBook-Pro-Illa:new fludy$ git push
Everything up-to-date
MacBook-Pro-Illa:new fludy$
```

Рисунок 8 – Вікно Git Bash з введеною командою `git push`.

9. Вношу зміни в файл на GitHub і виконую `git pull`, щоб синхронізувати зміни з локальним репозиторієм (рисунок 9-11).

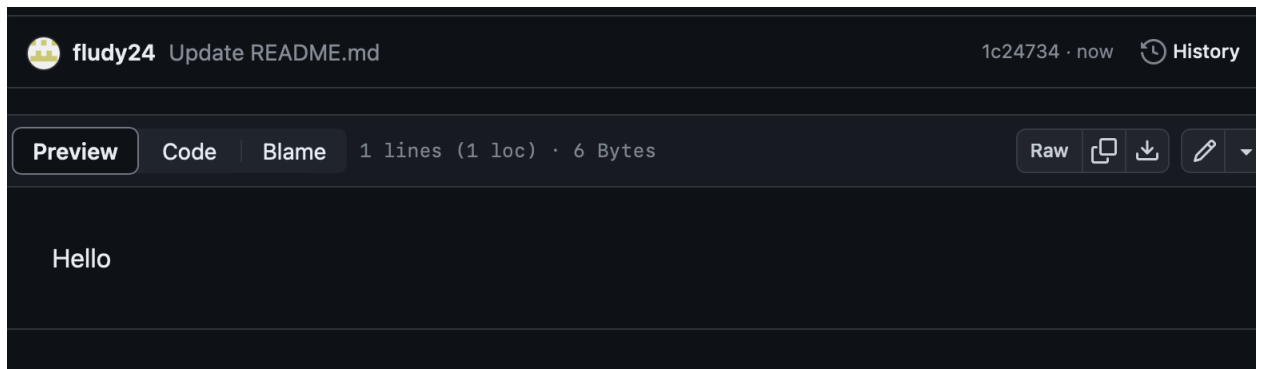


Рисунок 9 – Вікно GitHub з внесеними змінами в файлі.

```
MacBook-Pro-Illa:new fludy$ git pull
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
Unpacking objects: 100% (3/3), done.
From https://github.com/fludy24/New
   22bb059..1c24734  master    -> origin/master
Updating 22bb059..1c24734
Fast-forward
 README.md | 2 + -
 1 file changed, 1 insertion(+), 1 deletion(-)
MacBook-Pro-Illa:new fludy$
```

Рисунок 10 – Вікно Git Bash з введенною командою git pull.

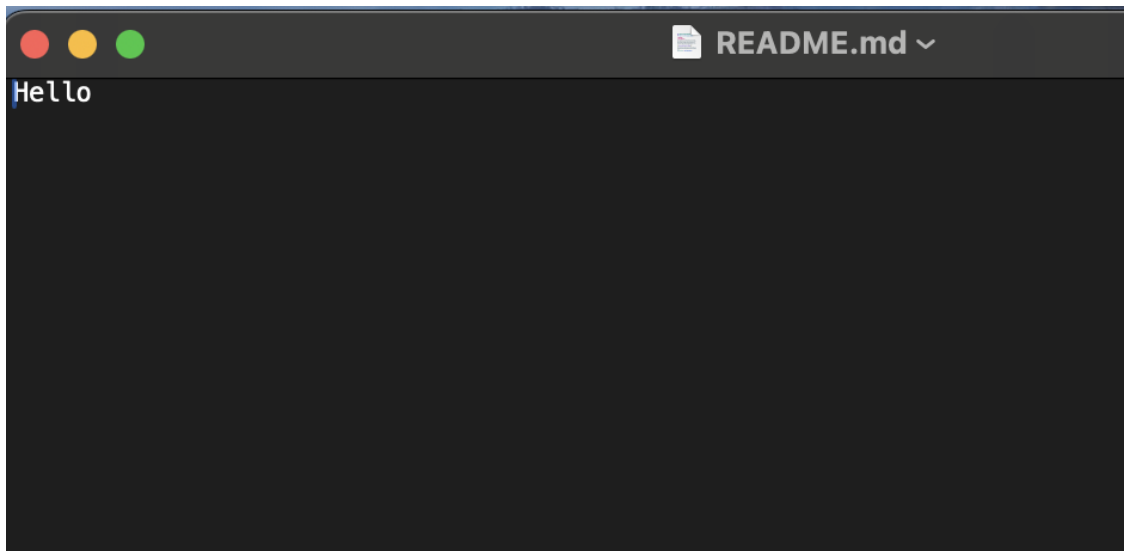


Рисунок 11 – Вікно файла який був змінений.

10. Створюю нову гілку за допомогою `git branch [LR6.1]` та переключаюсь на неї за допомогою `git checkout [LR6.1]` (рисунок 12).

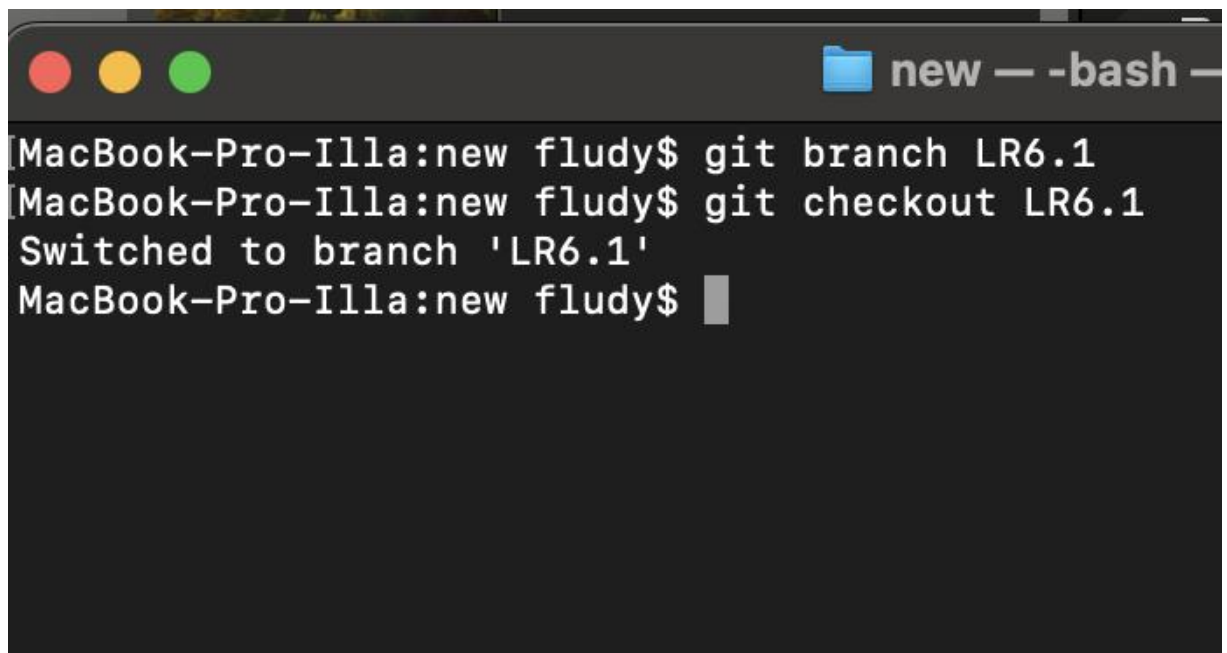
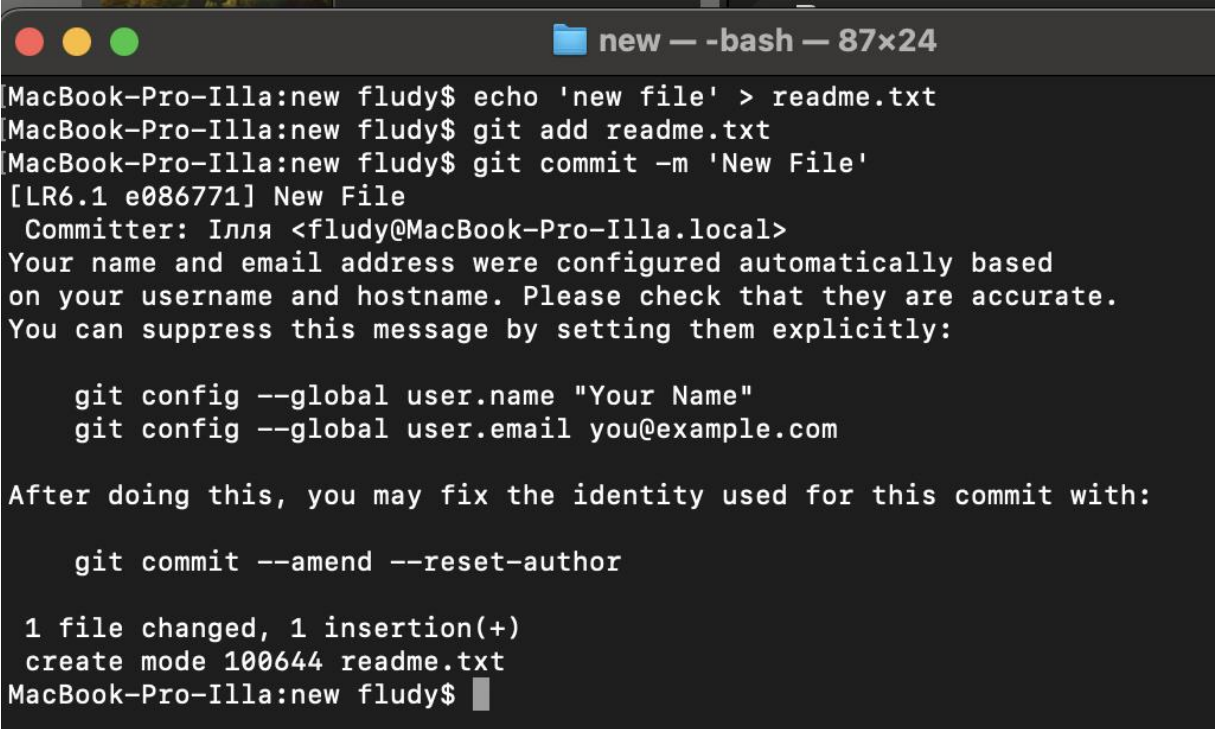


Рисунок 12 – Вікно Git Bash з введеними командами `git branch` і `git checkout`.

11. Вношу зміни в гілці та роблю коміт (рисунок 13).



```
MacBook-Pro-Illa:~$ new -bash — 87x24
MacBook-Pro-Illa:~$ echo 'new file' > readme.txt
MacBook-Pro-Illa:~$ git add readme.txt
MacBook-Pro-Illa:~$ git commit -m 'New File'
[LR6.1 e086771] New File
  Committer: Ілля <fludy@MacBook-Pro-Illa.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 1 insertion(+)
create mode 100644 readme.txt
MacBook-Pro-Illa:~$
```

Рисунок 13 – Вікно з внесеними змінами в гілці та добавленими комітом.

12. Відправляю зміни з гілки на GitHub за допомогою git push (рисунок 14).


```
MacBook-Pro-Illa:new fludy$ git push --set-upstream origin LR6.1
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 343 bytes | 343.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'LR6.1' on GitHub by visiting:
remote:   https://github.com/fludy24/New/pull/new/LR6.1
remote:
To https://github.com/fludy24/New.git
 * [new branch]      LR6.1 -> LR6.1
Branch 'LR6.1' set up to track remote branch 'LR6.1' from 'origin'.
MacBook-Pro-Illa:new fludy$
```

Рисунок 14 – Вікно Git Bash з введеною командою git push.

13. На GitHub створюю Pull Request для об'єднання гілки з основною гілкою. Опишіть зміни та виконайте Merge Pull Request. (рисунок 15-17).

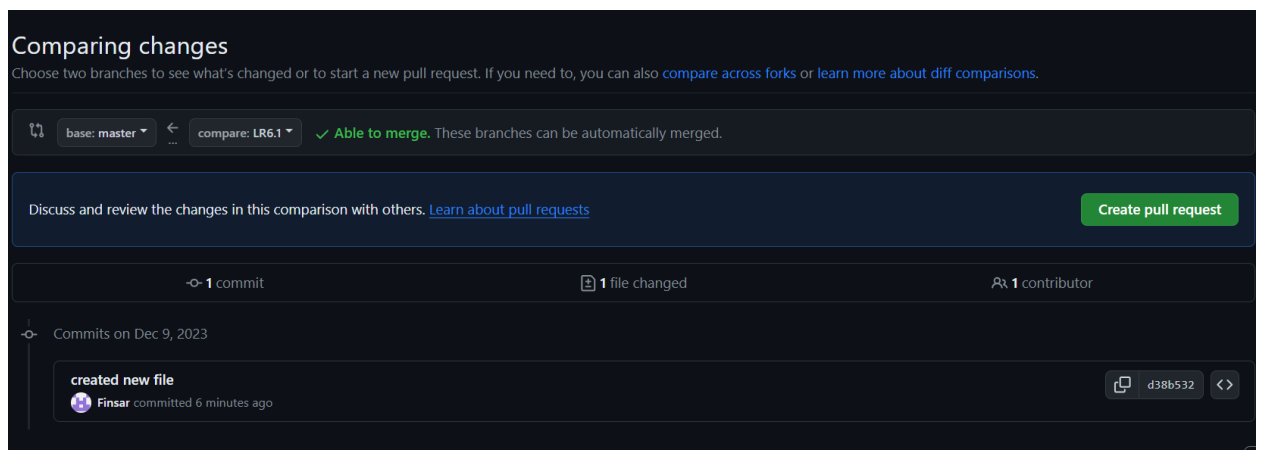


Рисунок 15 – Вікно створення Pull Request на GitHub.

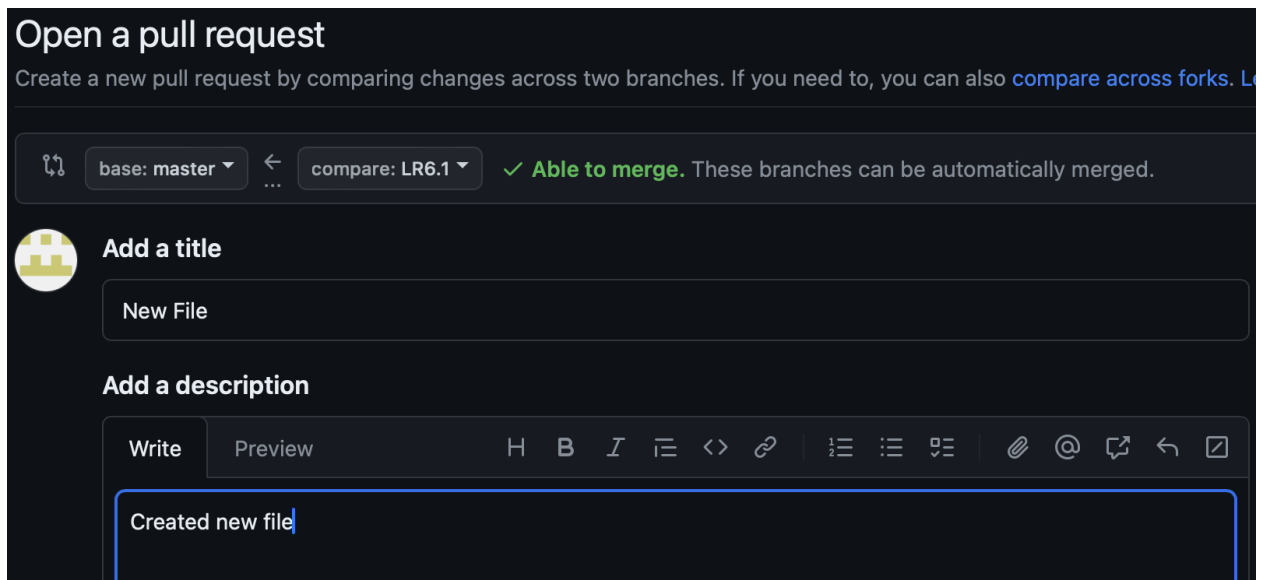


Рисунок 16 – Вікно опису змін GitHub.

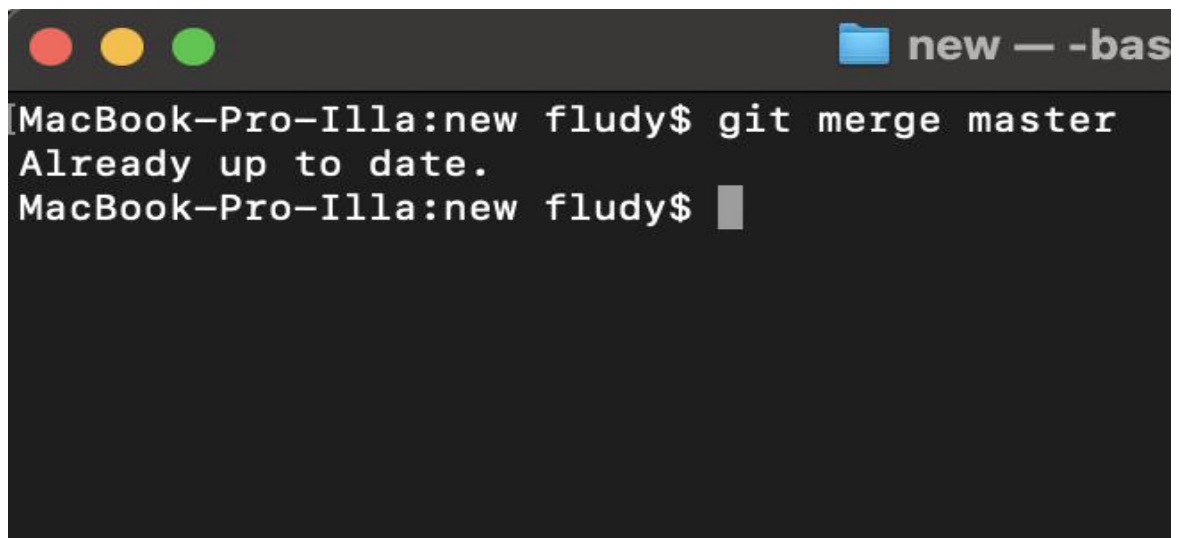


Рисунок 17 – Вікно Git Bash з введеною командою git merge.

14. Прохожу тренажер «Вступ», «Їдемо далі» та «Переміщуємо роботу туди-сюди» (рисунок 18).

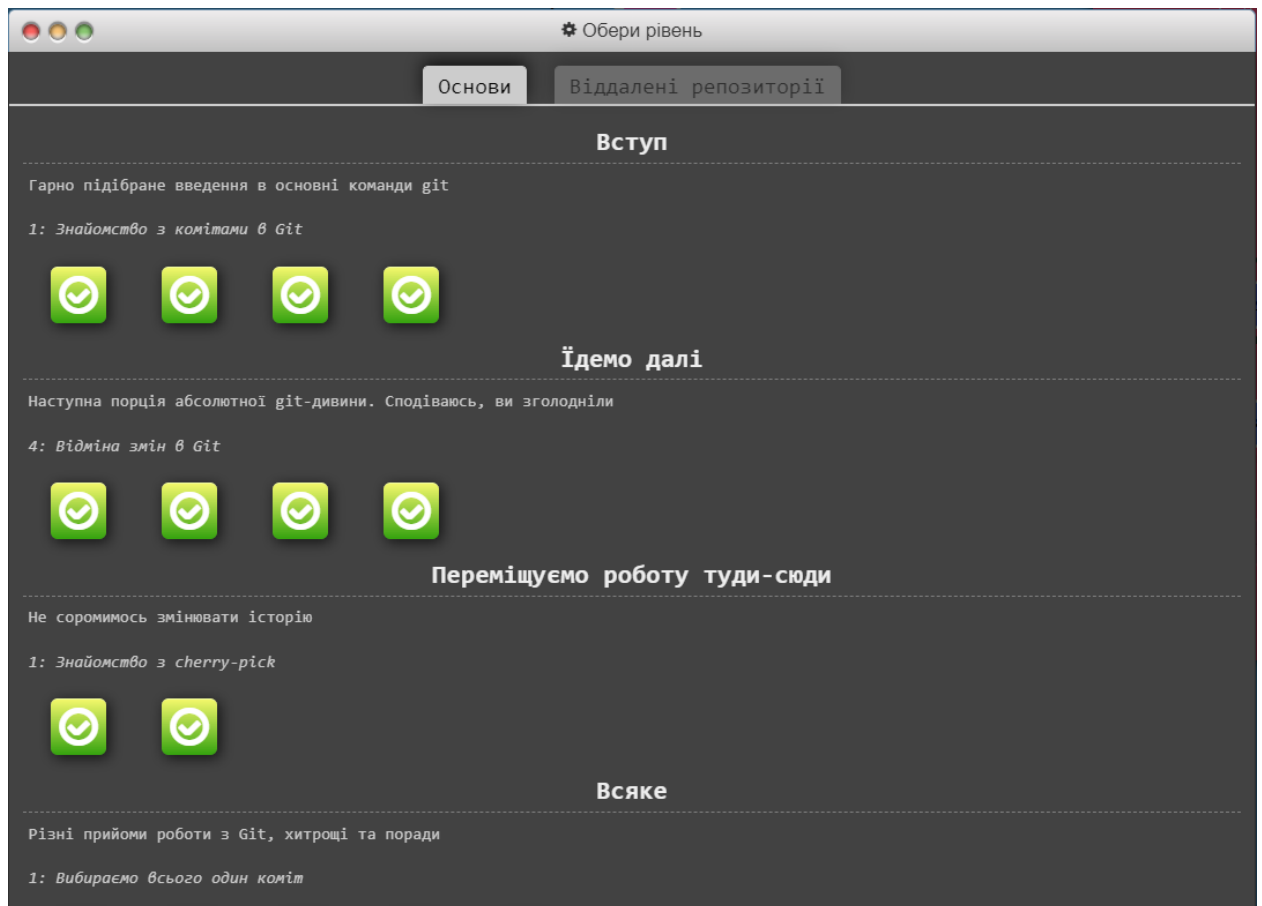


Рисунок 18 – Вікно пройденого тренажера.

Висновок:

Я навчився основам роботи з GIT та GitHub, включаючи управління версіями, роботу з гілками, створення Pull Request та співпрацю в командах та пройшов тренажер.