**НПУ імені М.П.Драгоманова**

**Факультет математики, інформатики та фізики**

***Кафедра комп’ютерної та програмної інженерії***

**Лабораторна робота 2**

з курсу

***«Робота з даними на платформі .NET»***

Студент: Колодій Д.А.

Група: 41 ІПЗз

Факультет: Факультет математики, інформатики та фізики

Викладач: Січкаренко В. О.

**Київ 2024**

**ЛАБОРАТОРНА РОБОТА No2**

**Тема: Основні операції в системі контролю версій Git.**

**Мета: Ознайомитись та навчитись використовувати основні команди системи контролю версій Git.**

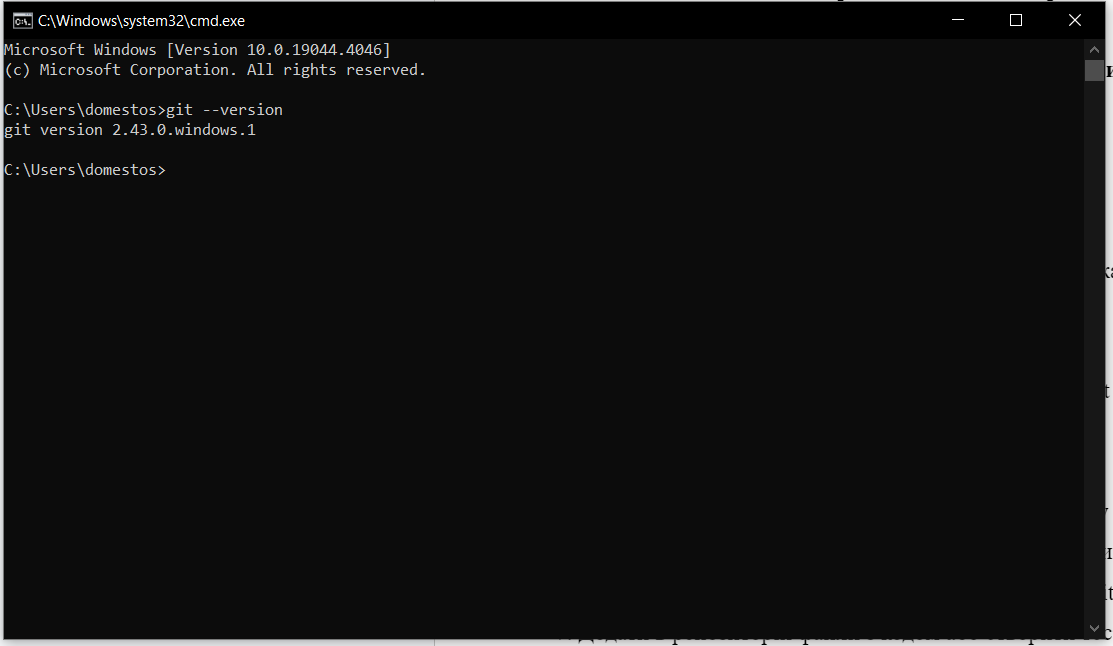
**ХІД РОБОТИ**

1. Ознайомитися з теоретичною частиною.

2. Створити обліковий запис в GitHub за посиланням вказаному в джерелі.

[Daniagrom2 (github.com)](https://github.com/Daniagrom2)

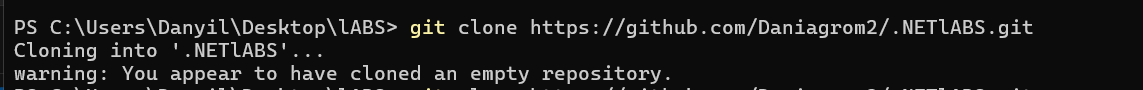
3. Встановити систему контролю версій Git (бажано Git CMD або Git Bash).



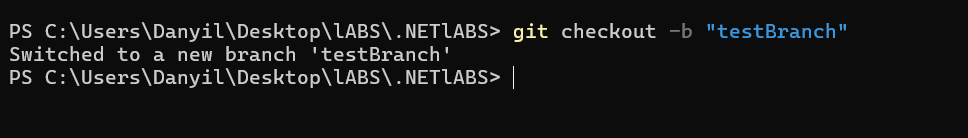
4. Створити відкритий (public) репозиторій на хостингу GitHub;

[Daniagrom2/.NETlABS (github.com)](https://github.com/Daniagrom2/.NETlABS)

5. Клонувати на локальний комп’ютер (використовуючи команду git clone);

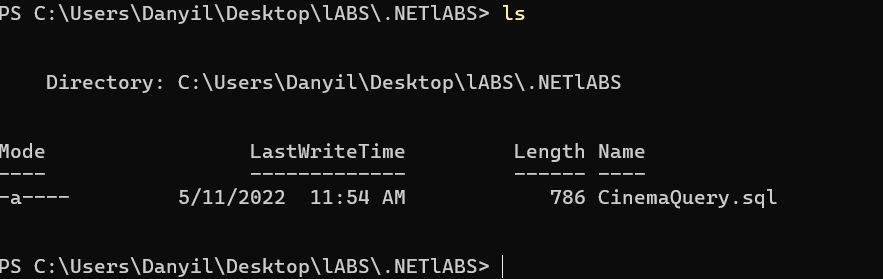


6. Створити в локальному репозиторії гілку (команда git checkout -b);

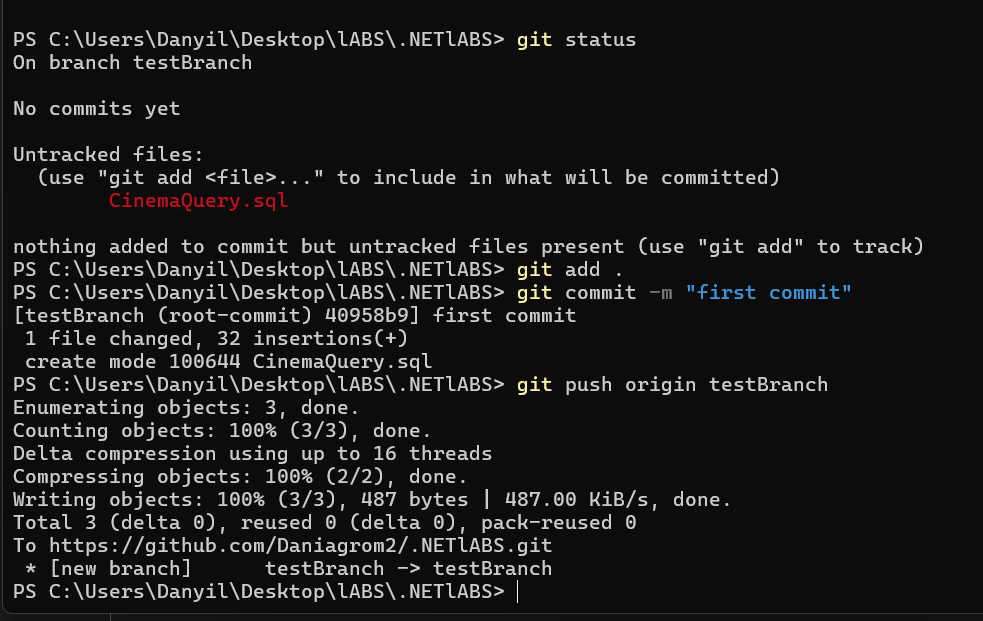


7. Додати в репозиторій файли з кодом або створити тестове рішення (можна використовувати

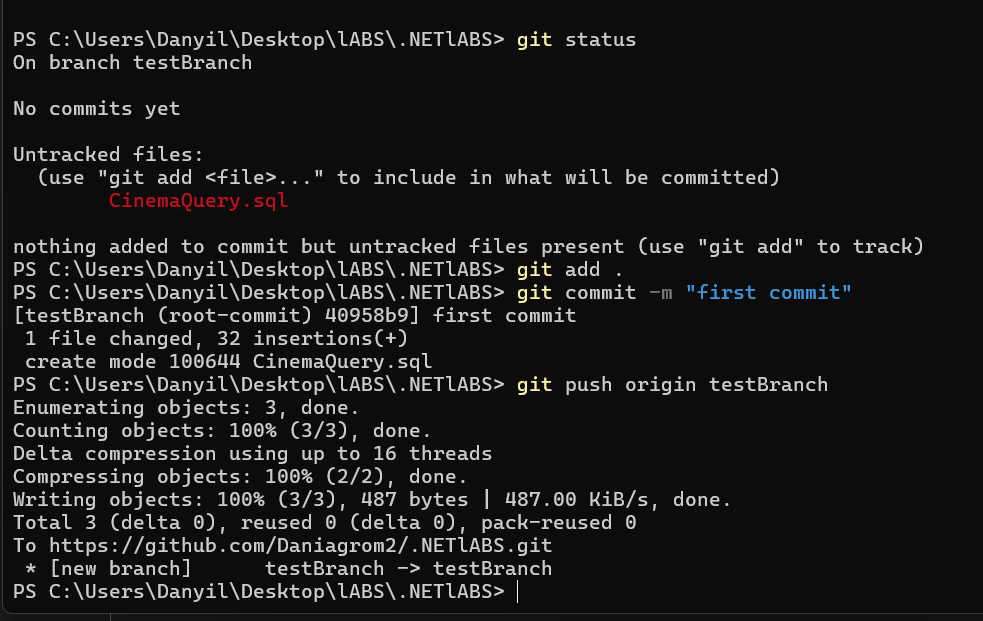
будь-яку IDE);



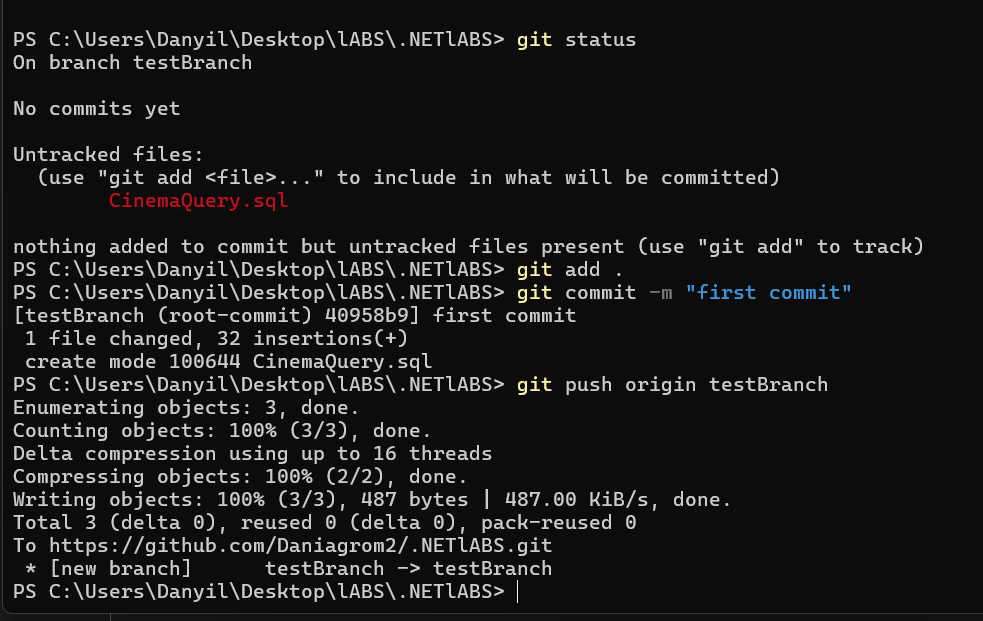
8. Індексувати файли або рішення в локальному репозиторії (команда git add);



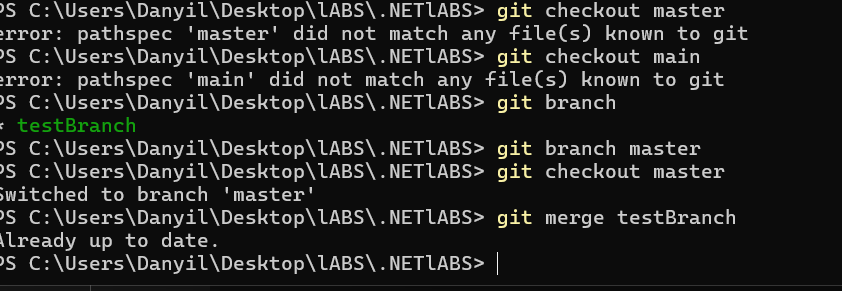
9. Зафіксувати зміни (команда git commit);



10. Надіслати запит на зміни у віддалений репозиторій (команда git push);



11. Поєднати код з запиту на зміни з головною гілкою (merge pull request);



12. Надіслати викладачу посилання на створений репозиторій в GitHub.

[Daniagrom2/.NETlABS (github.com)](https://github.com/Daniagrom2/.NETlABS)

**Контрольні запитання**

**1. Що таке системи контролю версій та їх призначення**

Системи контролю версій - це інструменти для відстеження та керування змінами в програмному коді та інших файлових структурах, що дозволяють ефективно співпрацювати над проектами.

**2. Призначення хостингів проектів**

Хостинги проектів - це платформи для зберігання та розгортання програмного коду, де можна зберігати та ділитися проектами з іншими користувачами через Інтернет.

**3. Що таке версія?**

Версія - це конкретний стан або варіант програмного забезпечення або файлу на певний момент часу.

**4. Що таке репозиторій та як поділяються за місцем розташування?**

Репозиторій - це місце для зберігання версій програмного коду та інших файлів. Вони поділяються на локальні (знаходяться на машині користувача) та віддалені (розташовані на серверах).

**5. Різниця між Git, GitHub, GitLab?**

Git - це система контролю версій; GitHub та GitLab - це хостинги для зберігання репозиторіїв Git та спільної роботи над ними.

**6. Яким чином можна створити репозиторій?**

Репозиторій можна створити за допомогою команди git init для локального репозиторію або через інтерфейс веб-сервісу для віддаленого репозиторію.

**7. Для чого призначені гілки?**

Гілки використовуються для роботи над різними аспектами проекту паралельно, щоб не впливати один на одного.

**8. Стратегії створення та ведення гілок**

Стратегії гілок включають розвиток, виправлення помилок, фічі та інші, залежно від потреб проекту.

**9. Основні команди управління гілками**

Основні команди для управління гілками: git branch, git checkout, git merge.

**10. Виявлення файлів для індексації**

Файли для індексації визначаються за допомогою команди git add.

**11. Фіксація змін**

Фіксація змін (або коміт) здійснюється за допомогою команди git commit.

**12. Запит на зміни у віддалений репозиторій**

Запит на зміни у віддалений репозиторій відправляється за допомогою команди git push.