****

Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота №1  
**Технології розроблення програмного забезпечення**

Виконала

студентка групи ІА–22:

Микитенко Ірина

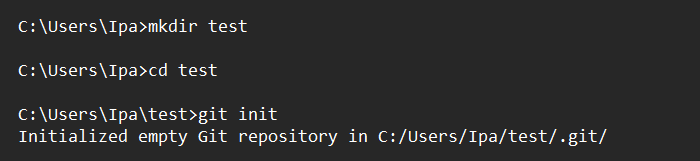
Київ 2024

**Системи контролю версій. Git.**

**Мета роботи:** Ознайомитися з основними командами Git, вивчити механізми створення репозиторіїв, комітів, роботи з гілками, злиття та вирішення конфліктів.

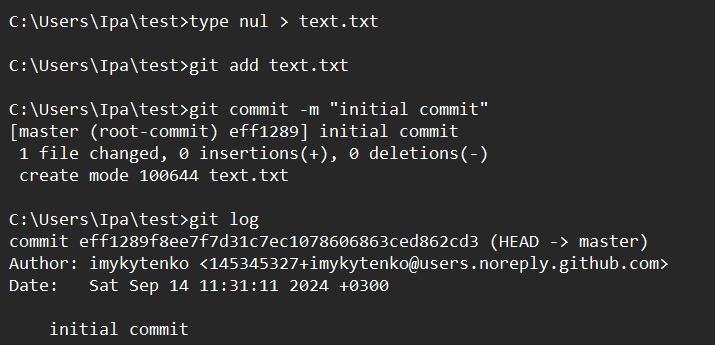
**Хід роботи**

Крок 1. Створення нової директорії та ініціалізація репозиторію:



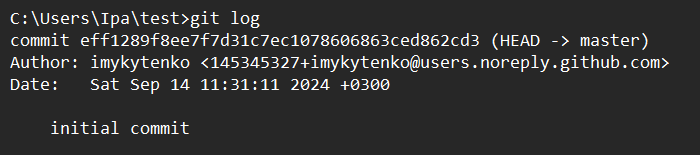
Команда mkdir створює нову директорію, а git init ініціалізує новий Git-репозиторій.

Крок 2. Створення нового файлу, додавання його на stage та створення коміту:

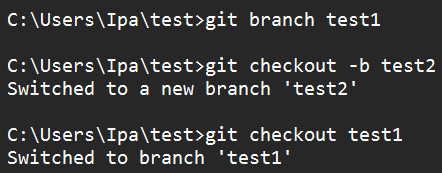


Команда type nul > text.txt створює порожній файл. git add додає його на stage, а git commit фіксує зміни з повідомленням "initial commit".

Крок 3. Перегляд журналу комітів:

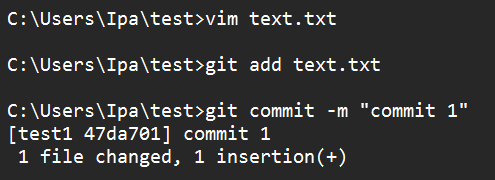


Крок 4. Створення нових гілок та перемикання між ними:



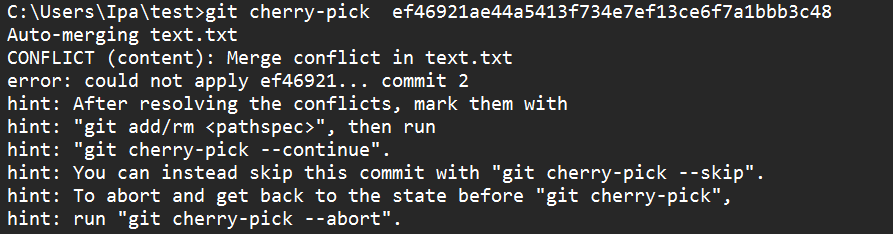
git branch створює нову гілку, а git checkout -b створює гілку та перемикається на неї одночасно.

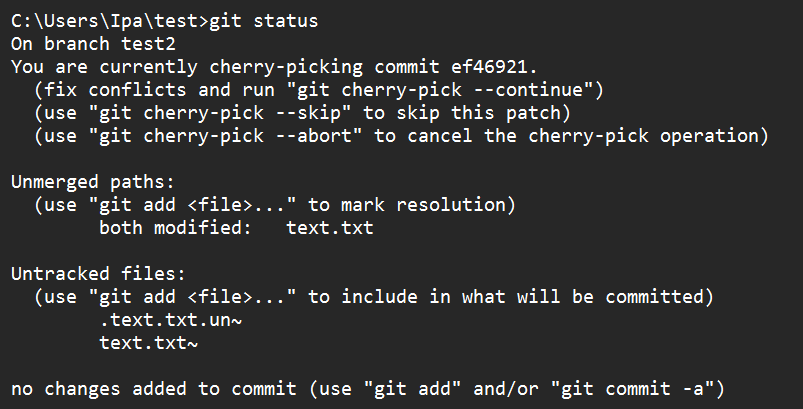
Крок 5. Редагування файлу та комітування змін:

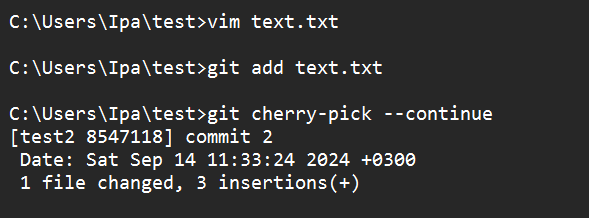


Після редагування файлу через редактор vim, команди git add та git commit фіксують нові зміни. Відповідно так ми створюємо по декілька комітів в різних гілках.

Крок 6. Виконання команди cherry-pick та вирішення конфлікту:

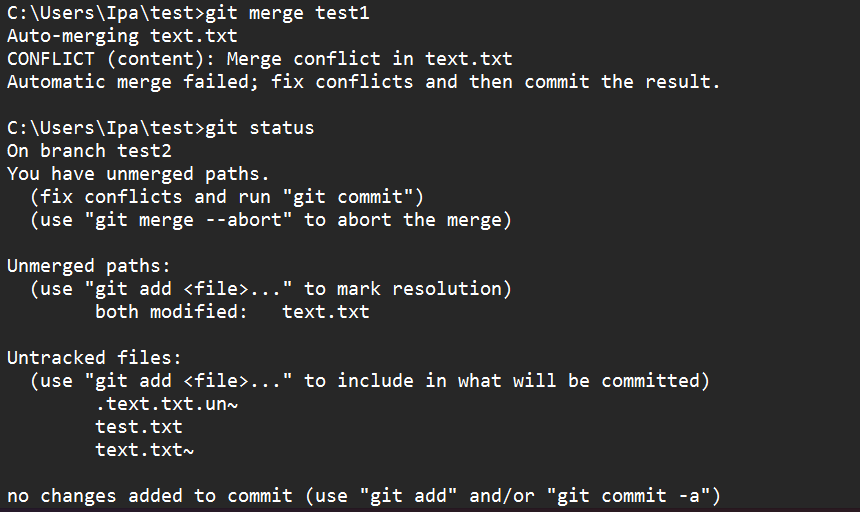


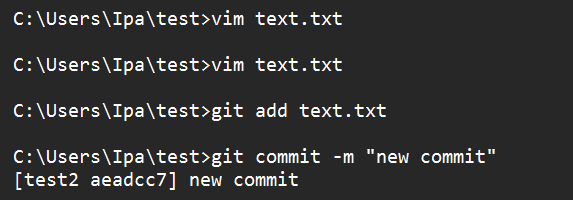




Команда git cherry-pick дозволяє перенести конкретний коміт (в даному випадку, коміт з хешем ef46921) з іншої гілки в поточну гілку (test2). Під час застосування цього коміту Git намагається злити зміни, але в процесі виникає конфлікт у файлі text.txt. Під час автоматичного злиття у файлі text.txt виникає конфлікт, оскільки однакові частини файлу були змінені по-різному в обох гілках. Git повідомляє про конфлікт і зупиняє процес cherry-pick до його вирішення. Відкриваємо файл text.txt в текстовому редакторі, щоб вручну вирішити конфлікт. У файлі будуть показані конфліктуючі зміни між гілками, позначені рядками. Вручну обираємо потрібні частини коду або об'єднуємо їх, а потім зберігаємо файл. Команда продовжує процес cherry-pick після того, як конфлікт був вирішений і файл text.txt був доданий в індекс. Як результат, коміт ef46921 успішно додано до гілки test2, і зміни з цього коміту відображаються в історії гілки.

Крок 7. Остаточне злиття гілок:





Ми почали процес злиття змін з гілки test1 в поточну гілку test2. Під час автоматичного злиття Git виявив конфлікт у файлі text.txt, оскільки в обох гілках були внесені різні зміни до однієї й тієї ж частини файлу. git status показав, що файл text.txt має конфлікт, і процес злиття не завершений. Нам потрібно було вручну вирішити конфлікт. Ми відкрили файл text.txt у текстовому редакторі для вирішення конфлікту. Після того як ми вирішили конфлікт, додали файл на stage за допомогою команди git add, щоб підготувати його до коміту. Нарешті, ми зафіксували злиття з повідомленням "new commit", завершуючи процес об'єднання змін з гілки test1 в test2.

**Висновок:** У ході виконання лабораторної роботи ми ознайомилися з основними командами Git, що дозволяють працювати з репозиторіями, гілками та змінами у файлах. Ми створювали нові гілки, перемикалися між ними, фіксували зміни (коміти), а також виконали злиття гілок. Окремо розглянули ситуації виникнення конфліктів під час злиття, вирішували їх вручну та успішно завершували процес об'єднання.