



Мета роботи

Ознайомитися з системою контролю версій **Git** та платформою **GitHub**, навчитися:

- створювати локальний репозиторій;
 - виконувати базові команди Git;
 - працювати з комітами та історією змін;
 - відновлювати файл з попереднього коміту;
 - публікувати проєкт на GitHub.
-



Обладнання та програмне забезпечення

- ПК з ОС Windows / Linux / macOS
 - Встановлений **Git**
 - Текстовий редактор або **VS Code**
 - Обліковий запис на **GitHub**
-



Завдання до виконання



Завдання 1. Створення локального репозиторію

1. Створити папку з назвою lab_git.
 2. Перейти в цю папку та ініціалізувати Git-репозиторій:
 3. git init
 4. Перевірити стан репозиторію командою:
 5. git status
-



Завдання 2. Перший коміт

1. Створити файл main.txt.
 2. Додати в нього будь-який текст.
 3. Додати файл до staging-зони:
 4. git add main.txt
 5. Зробити перший коміт:
 6. git commit -m "Перший коміт"
-



Завдання 3. Зміни та історія

1. Змінити вміст файлу main.txt.
 2. Виконати ще один коміт.
 3. Переглянути історію комітів:
 4. git log --oneline
-



Завдання 4. Відновлення файлу з попереднього коміту

1. Знайти хеш (номер) першого коміту.
2. Повернути файл main.txt до стану з цього коміту:
3. git restore --source <commit_id> main.txt

4. Зберегти відновлену версію файлу новим комітом.

Завдання 5. Робота з GitHub

1. Створити новий репозиторій на GitHub.
 2. Підключити віддалений репозиторій:
 3. `git remote add origin https://github.com/username/lab_git.git`
 4. Відправити зміни на GitHub:
 5. `git push -u origin main`
-

Звіт повинен містити

- тему та мету роботи;
 - перелік виконаних команд;
 - скріншоти результатів виконання;
 - посилання на GitHub-репозиторій;
 - короткі висновки.
-

Критерії оцінювання

- правильність виконання команд Git;
 - наявність комітів та історії змін;
 - коректне відновлення файлу з коміту;
 - успішна публікація на GitHub.
-

Контрольні запитання

1. Що таке Git і для чого він використовується?
2. Що таке коміт?
3. Для чого використовується команда `git add`?
4. Як повернути файл до попередньої версії?
5. Чим відрізняється Git від GitHub?