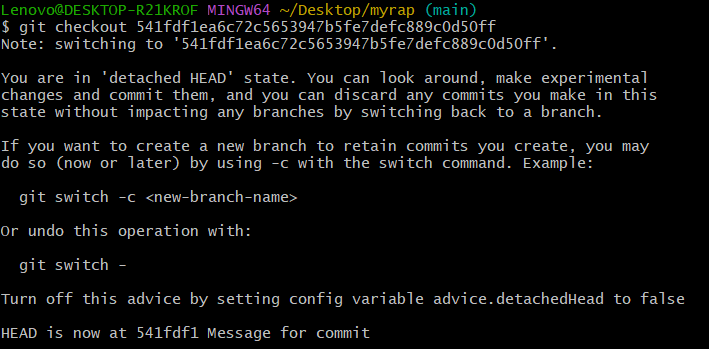
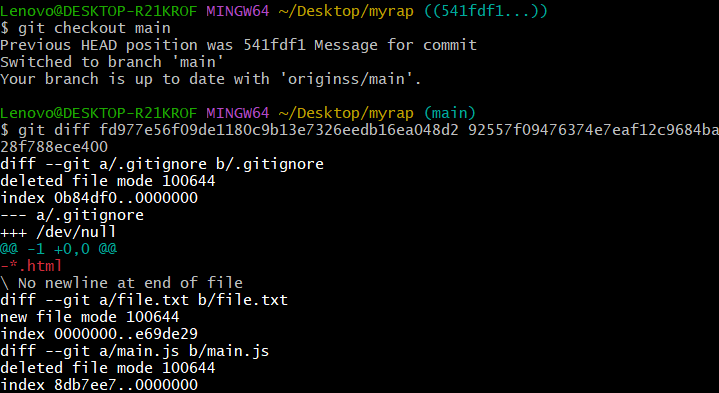
* **Тема:** Робота з історією змін та отримання старих версій
* **Мета:** Отримати навички переміщення по історії змін у СКВ Git
* **Хід роботи:**

1. Перевірити глобальні налаштування Git for Windows
2. У випадку невідповідності вашим даним налаштувати глобальну конфігурацію Git for Windows
3. Клонувати створений раніше репозиторій (репозиторій повинен мати декілька ревізій для всіх своїх файлів) як локальну робочу копію
4. За допомогою консольних інструментів переглянути історію комітів проекту.  
   
5. Завантажити одну з попередніх версій проекту



1. Порівняти зміни у завантаженій попередній версії з останньою. Зробити висновки щодо відмінності між поточною версією і старою



1. Повернутися до останньої версії проекту
2. Скасувати останній коміт



1. Переглянути історію комітів, та порівняти її з попередньою
2. Для кожного етапу роботи зробити знімки екрану та додати їх у звіт з описом кожного скіншота
3. Дати відповіді на контрольні запитання
4. Зберегти звіт у форматі PDF
5. Завантажити збережений PDF у репозиторій для лабораторних робіт

* **Контрольні питання:**

1. Скільки існує основних способів вибору ревізій?

Основні способи вибору ревізій:

Ідентифікатор комміту (SHA-1 хеш).

Гілка.

Тег.

Відносний посилальний знак (наприклад, HEAD~2 - два комміти назад від поточного положення).

1. Скільки символів зазвичай достатньо для збереження унікальності значень в проекті?

Символи для збереження унікальності значень в проекті:

Довжина хешу SHA-1 у Git складає 40 символів. Таким чином, хеші коммітів зазвичай використовуються для унікальності.

1. За допомогою якої команди можливо продивитися історію коммітів?

Команда для перегляду історії коммітів:

git log

1. Перелічіть відомі вам опції команди git log

Опції команди git log:

-p: Показує зміни у вигляді патчів.

--stat: Показує статистику змін.

--graph: Показує графічне представлення гілок та злиттів.

--oneline: Скорочена версія виводу з одним рядком на комміт.

--author: Фільтрує комміти за автором.

--since, --until: Фільтрує комміти за діапазоном дат.

--grep: Фільтрує комміти за рядком у повідомленні комміту.

1. Опишіть опції команди git log

Опції команди git log (розширений опис):

--all: Показує історію всіх гілок.

--decorate: Показує теги та гілки у виводі.

--graph: Відображає граф гілок та злиттів.

--oneline: Показує кожен комміт у стислій однорядковій формі.

--p: Виводить зміни у вигляді патчу.

--stat: Показує статистику змін для кожного комміту.

--author: Фільтрує комміти за автором.

--since, --until: Фільтрує комміти за діапазоном дат.

--grep: Фільтрує комміти за вказаним рядком у повідомленні комміту.

--follow: Виводить історію файлу, якщо він перейменований.