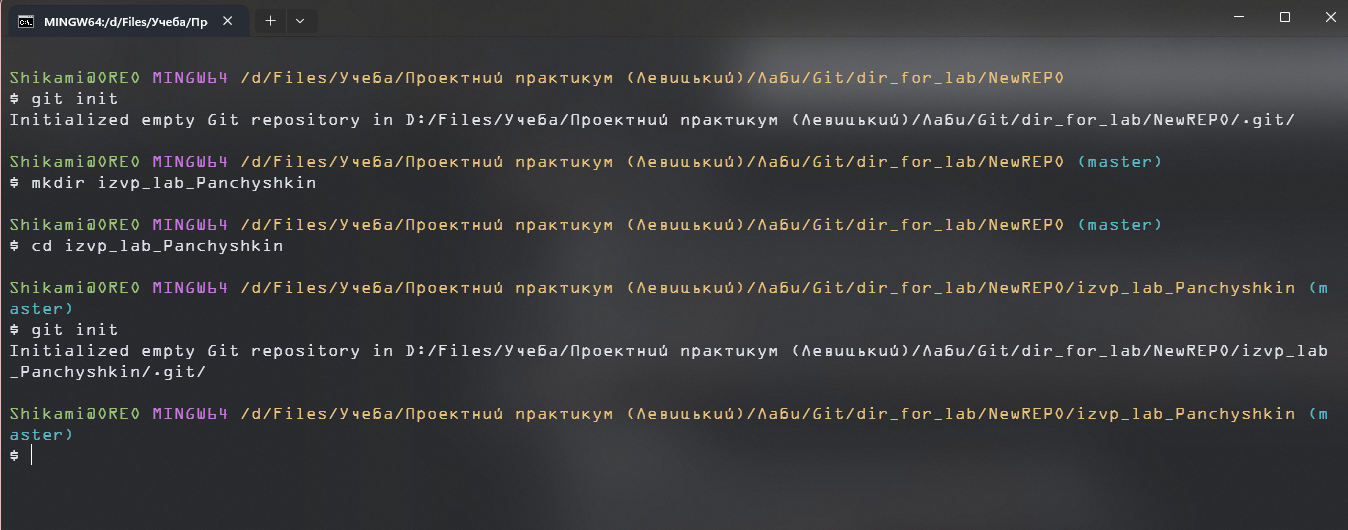
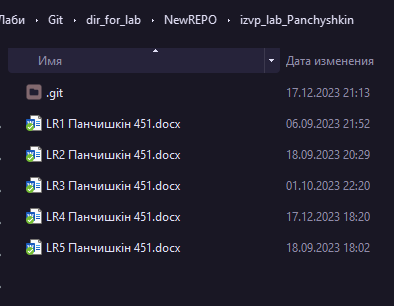
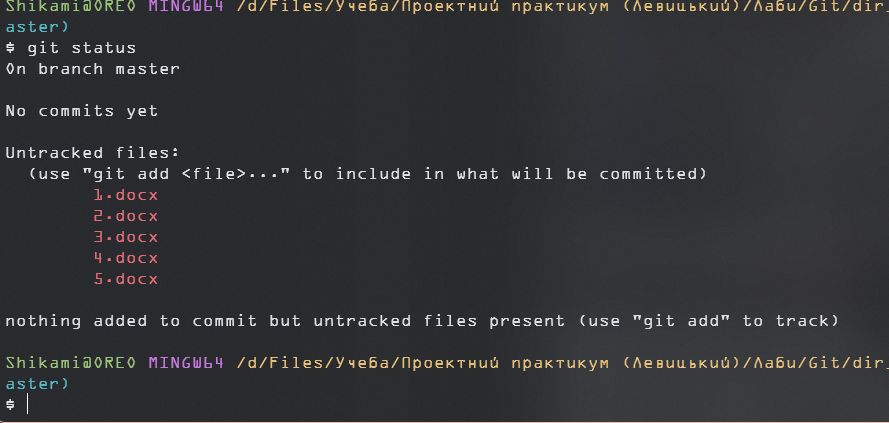
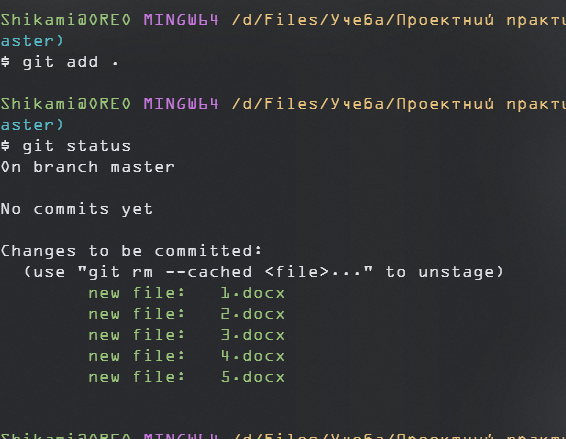
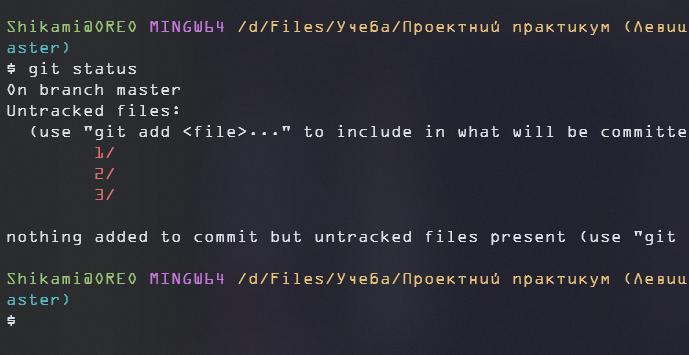
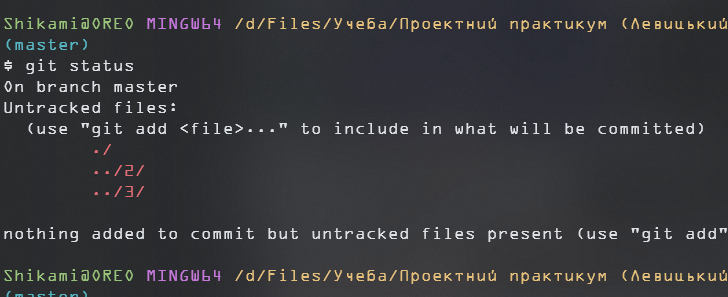
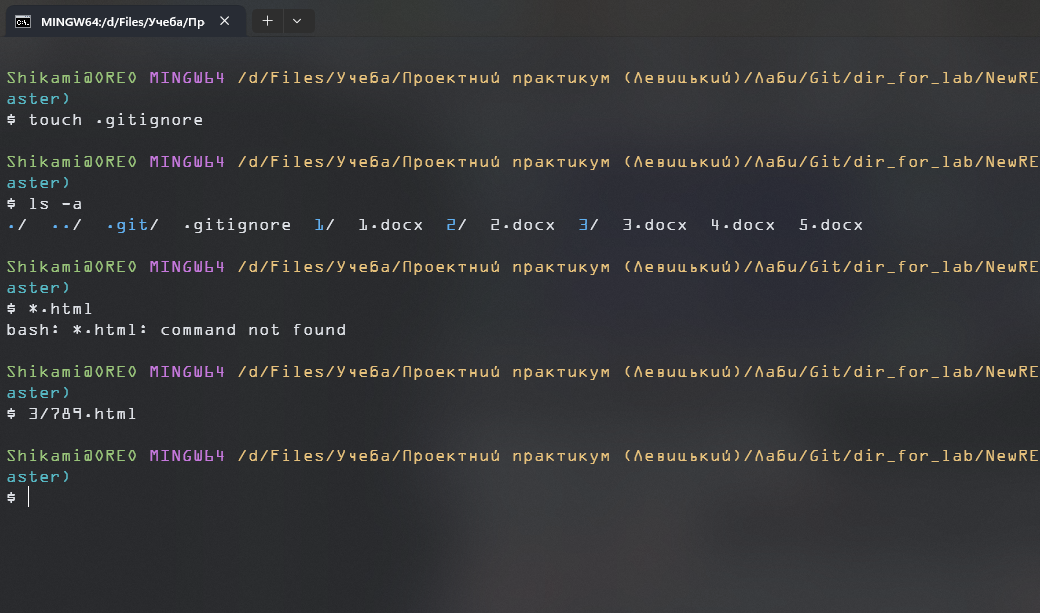
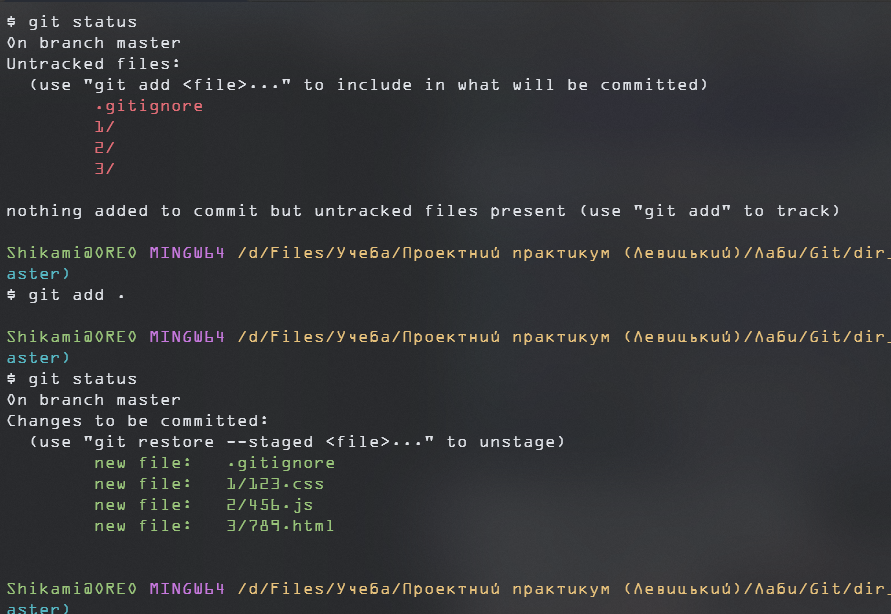
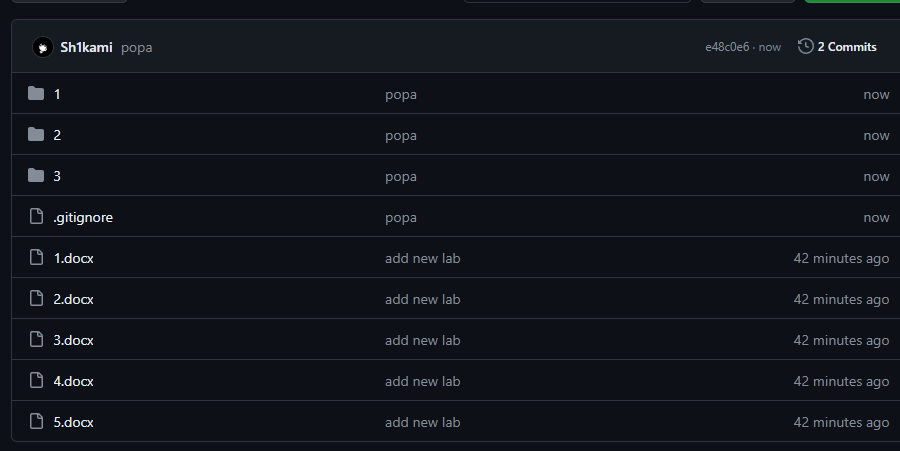
* **Тема:** Внесення змін до репозиторію
* **Мета:** Отримати навички внесення змін до локальної робочої копії та репозиторію на Github, навчитися працювати з файлом .gitignore
* **Обладнання:** Персональний комп'ютер. Текстовий редактор Sublime Text 3 або Notepad++. Web-браузер Chrome, Firefox, Opera, Internet Explorer, Git for Windows
* **Хід роботи:**

1. Перевірити глобальні налаштування Git for Windows
2. Створити порожній каталог для нового репозиторію
3. За допомогою команди git init ініціалізувати репозиторій  
   $ mkdir izvp\_lab\_my\_lastname

$ cd izvp\_lab\_my\_lastname

$ git init

1. Якщо репозиторій проініціалізовано правильно, в середені каталогу з'явиться підкаталог .git  
    
2. Помістити в каталог файли, та за допомогою консолі перевірити статус репозиторію за допомогою команди git status  
   
3. Додати до індексу нові файли
4. Зафіксувати зміни в репозиторії  
   
5. Помістити в каталог файли через інтерфейс операційної системи або за допомогою файлового менеджера, та за допомогою консолі перевірити статус репозиторію
6. Додати нові файли до переліку ігнорування
7. Додати кілька підкаталогів, розмістити там файли з розширенням \*.css, \*.js та \*.html  
    
8. Додати файли \*.html до переліку ігнорування  
   
9. Додати до індексу нові файли
10. Перевірити статус репозиторію  
    
11. За допомогою команди git remote під'єднати віддалений репозиторій заздалегіть створений на GitHub  
    $ git remote add pb https://github.com/your\_name/your\_repo   
    
12. Підєднати ще один віддалений репозиторій на Bitbucket
13. Вивантажити файли локального репозиторію на Github та Bitbucket
14. Впевнитися за допомогою web-інтерфейсу, що завантаження пройшло корректно
15. Впевнитися за допомогою web-інтерфейсу, що Файли з переліку ігнорування не вивантажилися у віддалений репозиторій
16. Для кожного етапу роботи зробити знімки екрану або скопіювати текст консолі та додати їх у звіт з описом кожного скіншота
17. Дати відповіді на контрольні запитання
18. Зберегти звіт у форматі PDF

* **Контрольні питання:**

1. Як створити файл .gitignore в своєму проекті?  
   Щоб створити файл .gitignore в своєму проекті, вам потрібно створити новий текстовий файл з назвою .gitignore в кореневому каталозі вашого проекту. Ви можете використовувати будь-який текстовий редактор для цього. Вміст файлу .gitignore буде визначати, які файли або шляхи повинні бути ігноровані Git.
2. Чому корисно використовувати .gitignore при роботі з репозиторієм Git?  
   Використання .gitignore є корисним при роботі з репозиторієм Git, оскільки він дозволяє Git ігнорувати та не відслідковувати певні файли або каталоги в вашому проекті. Це може бути корисним, наприклад, для ігнорування файлів, які генеруються в процесі збірки, або для ігнорування файлів, які не повинні бути включені в коміти.
3. Де може бути розміщений файл .gitignore?  
   Файл .gitignore може бути розміщений в кореневому каталозі вашого проекту або в будь-якому підкаталозі. Ви також можете встановити глобальний файл .gitignore, використовуючи команду git config --global core.excludesfile ~/.gitignore\_global
4. Які типи файлів можна вказати в файлі .gitignore для їх ігнорування Git?  
   Ви можете вказати будь-який тип файлу в файлі .gitignore для його ігнорування Git. Наприклад, ви можете вказати конкретні розширення файлів, імена файлів або шаблони шляхів, які повинні бути ігноровані.
5. Яким чином ігнорувати всі файли \*.php окрім index.php?  
   Щоб ігнорувати всі файли \*.php, окрім index.php, ви можете додати наступні рядки до вашого файлу .gitignore:  
   \*.php  
   !index.php   
   Перший рядок ігнорує всі файли .php, а другий рядок виключає index.php з ігнорування.
6. Які дії виконує шаблон \*\*?  
   Шаблон \*\* в .gitignore використовується для відповідності каталогам на будь-якому рівні репозиторію
7. Які дії виконує шаблон !?  
   Шаблон ! в .gitignore використовується для відміни дії шаблону. Якщо файл відповідає певному шаблону, але також відповідає шаблону відміни, вказаному після нього, такий файл не буде ігнорований
8. Що таке файл .gitignore і яка роль його відіграє в системі контролю версій Git?  
   Файл .gitignore - це спеціальний файл, який використовується системою контролю версій Git для вказівки, які файли або шляхи повинні бути ігноровані. Він дозволяє Git ігнорувати та не відслідковувати певні файли або каталоги в вашому проекті.
9. Як додати коментарі до файлу .gitignore для пояснення включених правил?  
   Щоб додати коментарі до файлу .gitignore для пояснення включених правил, ви можете просто додати рядок, що починається з символу #. Git буде ігнорувати будь-які рядки в .gitignore, що починаються з #, вважаючи їх коментарями
10. Якщо виникає конфлікт між локальним та глобальним файлами .gitignore, як його вирішити?

Якщо виникає конфлікт між локальним та глобальним файлами .gitignore, ви повинні вирішити, які правила вам потрібні для вашого проекту. Якщо ви хочете, щоб певні файли були ігноровані лише в конкретному проекті, ви повинні додати ці правила до локального файлу .gitignore цього проекту. Якщо ви хочете, щоб ці правила застосовувались до всіх ваших проектів, ви повинні додати ці правила до вашого глобального файлу .gitignore.