

Лабораторна робота № 5

Тема: Внесення змін до репозиторію.

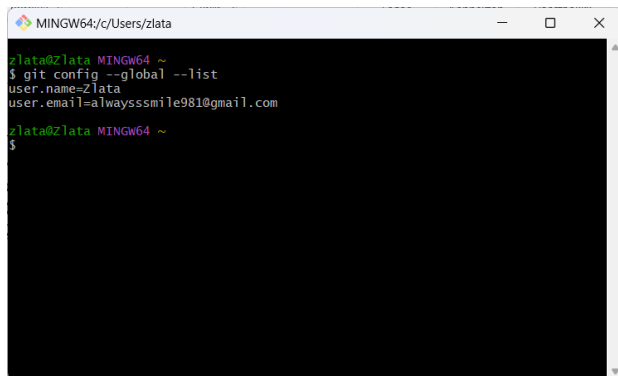
Мета: Отримати навички внесення змін до локальної робочої копії та репозиторію на Github, навчитися працювати з файлом .gitignore.

Обладнання та ПЗ: Персональний комп'ютер. Текстовий редактор Sublime Text 3 або Notepad++. Web-браузер Chrome, Firefox, Opera, Internet Explorer, Git for Windows

Хід роботи та завдання

Пункт 1.

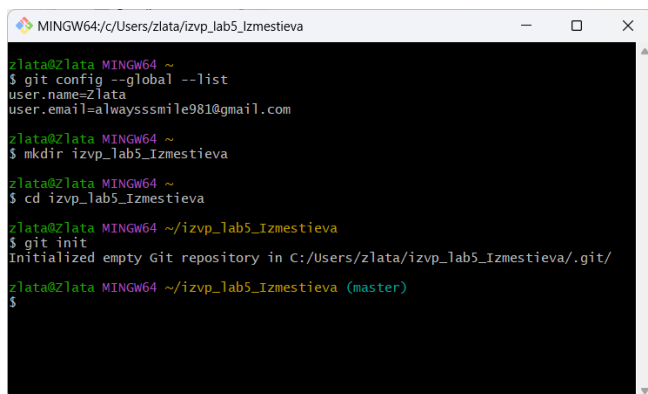
Перевірка глобальних налаштувань Git for Windows.



```
MINGW64~/Users/zlata
zlata@Zlata MINGW64 ~
$ git config --global --list
user.name=Zlata
user.email=alwaysssmile981@gmail.com
zlata@Zlata MINGW64 ~
$
```

Пункт 2-3.

Створено порожній каталог для нового репозиторію та за допомогою команди git init ініціалізовано репозиторій.



```
MINGW64~/Users/zlata/izvp_lab5_Izmestieva
zlata@Zlata MINGW64 ~
$ git config --global --list
user.name=Zlata
user.email=alwaysssmile981@gmail.com
zlata@Zlata MINGW64 ~
$ mkdir izvp_lab5_Izmestieva
zlata@Zlata MINGW64 ~
$ cd izvp_lab5_Izmestieva
zlata@Zlata MINGW64 ~/izvp_lab5_Izmestieva
$ git init
Initialized empty Git repository in C:/Users/zlata/izvp_lab5_Izmestieva/.git/
zlata@Zlata MINGW64 ~/izvp_lab5_Izmestieva (master)
$
```

					Лабораторна робота № 5			
Зм	Лист	№ докум	Підпис	Дата	Внесення змін до репозиторію	Літ.	Лист.	Листів.
Розробив	Ізмestьєва					У	3	5
Перевірів	Левицткий					Група 451		
Оцінка								
Затв								

Пункт 4-7.

Репозиторій проініціалізовано правильно, тому в середині каталогу з'явиться підкаталог .git. У каталозі створено нові файли, перевірити статус репозиторію за допомогою команди `git status`, додано до індексу нові файли та зафіксувати зміни в репозиторії.

```
MINGW64~/c/Users/zlata/izvp_lab5_izmestieva
zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

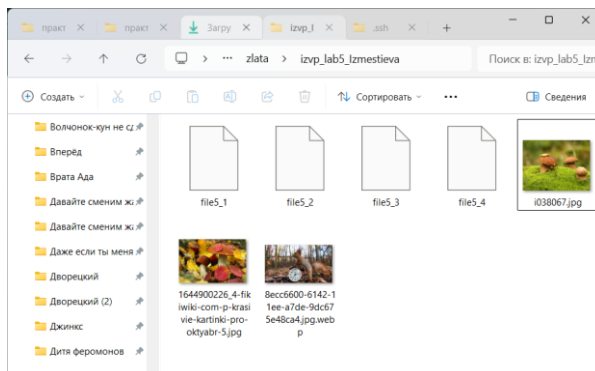
zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ touch file5_1 file5_2 file5_3 file5_4

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git add file5_1 file5_2 file5_3 file5_4

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git commit -m "Зміни зафіксовано"
[master (root-commit) 941611d] зміни зафіксовано
4 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file5_1
create mode 100644 file5_2
create mode 100644 file5_3
create mode 100644 file5_4
```

Пункт 8.

Додано нові файли (фото) за допомогою файлового менеджера, а також перевірено статус репозиторію.



```
MINGW64~/c/Users/zlata/izvp_lab5_izmestieva
$ git add file5_1 file5_2 file5_3 file5_4

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git commit -m "Зміни зафіксовано"
[master (root-commit) 941611d] зміни зафіксовано
4 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file5_1
create mode 100644 file5_2
create mode 100644 file5_3
create mode 100644 file5_4

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git status
On branch master

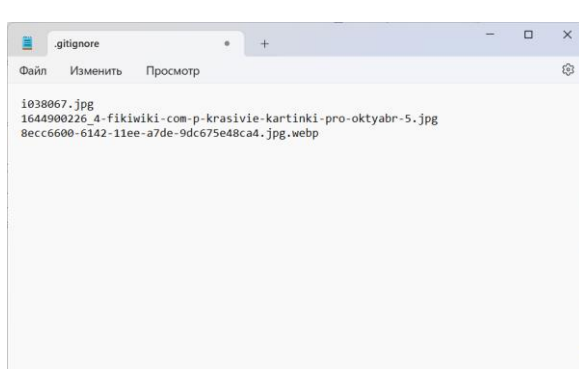
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    1644900226_4-fikiwiki-com-p-krasivie-kartinki-pro-oktyabr-5.jpg
    8ecc6600-6142-11ee-a7de-9dc675e48ca4.jpg.webp
    i038067.jpg

nothing added to commit but untracked files present (use "git add" to track)

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$
```

Пункт 9.

Створено файл .gitignore, у який записано всі файли, які треба ігнорувати. Після цього перевірено статус репозиторію.



```
MINGW64~/c/Users/zlata/izvp_lab5_izmestieva
(use "git add <file>..." to include in what will be committed)
.gitignore

nothing added to commit but untracked files present (use "git add" to track)

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git add .gitignore

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git status
On branch master

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .gitignore

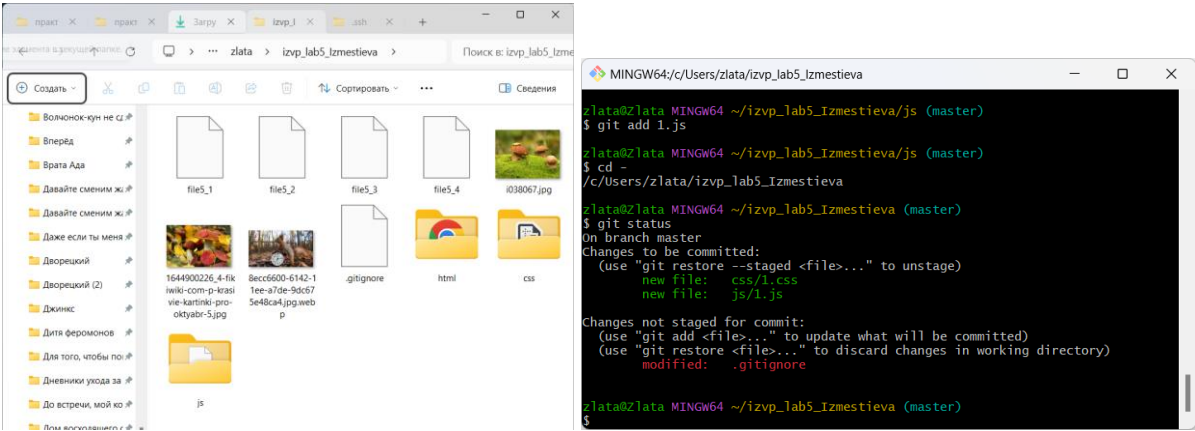
zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$ git commit -m "Dodano"
[master ba975f7] Dodano
1 file changed, 3 insertions(+)
create mode 100644 .gitignore

zlata@Zlata MINGW64 ~/izvp_lab5_izmestieva (master)
$
```

					Лабораторна робота № 5	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

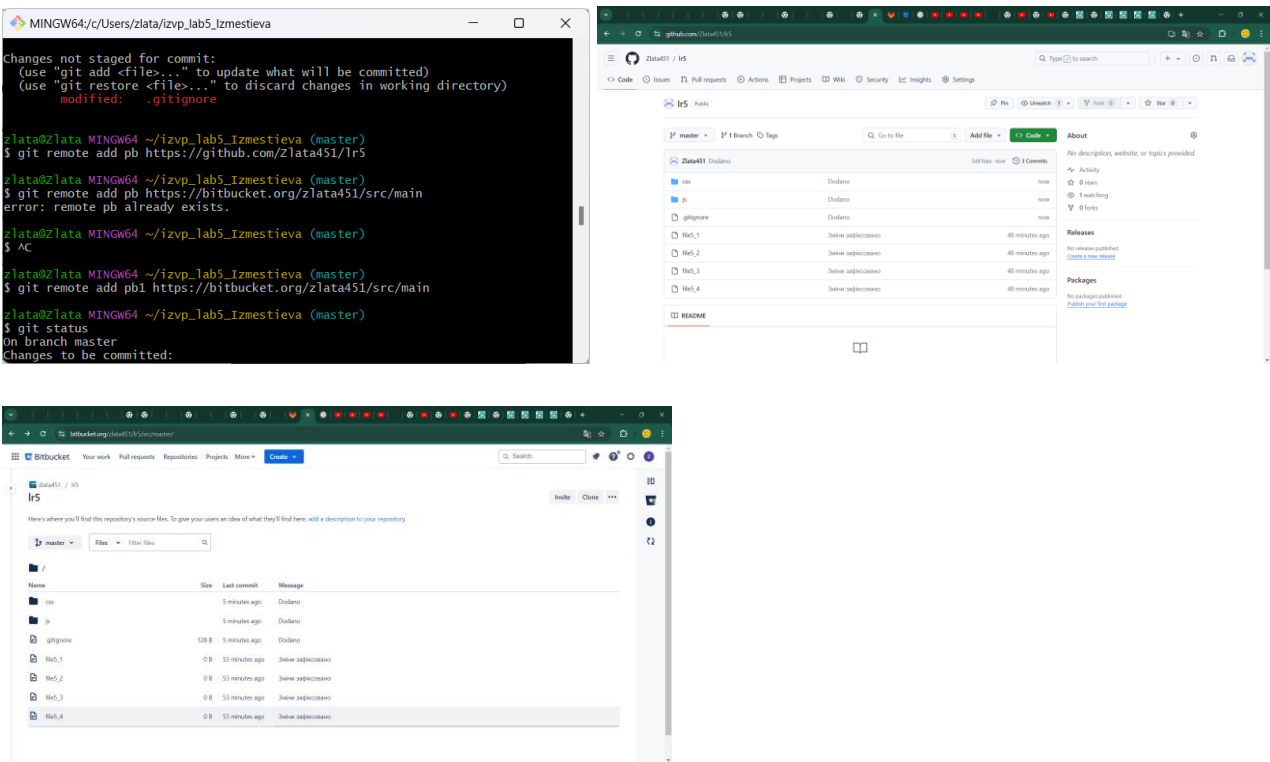
Пункт 10-13.

Додано кілька підкаталогів, де розміщено файли з розширенням 1.css, 1.js та 1.html відповідно. Файл 1.html додано до переліку ігнорування, інші нові файли додано до індексу, а також перевірено статус репозиторію.



Пункт 14-18.

За допомогою команди git remote під'єднано віддалені репозиторії заздалегіть створені на GitHub та Bitbucket. Після чого вивантажено файли локального репозиторію на Github та Bitbucket. Завантаження пройшло коректно та файли з переліку ігнорування не вивантажилися у віддалений репозиторій.



Контрольні питання

1. Як створити файл `.gitignore` в своєму проєкті?

Файл `.gitignore` можна створити в кореневій директорії проєкту, використовуючи текстовий редактор або команду `touch` у консолі.

2. Чому корисно використовувати `.gitignore` при роботі з репозиторієм Git?

Файл `.gitignore` дозволяє уникнути додавання до репозиторію непотрібних файлів, таких як тимчасові файли, налаштування середовища або згенеровані артефакти, що зберігає чистоту репозиторію і полегшує його обслуговування.

3. Де може бути розміщений файл `.gitignore`?

Файл `.gitignore` може бути розміщений у кореневій папці проєкту. Також можна створити `.gitignore` в підкаталогах для специфічного ігнорування файлів у цих папках.

4. Які типи файлів можна вказати в файлі `.gitignore` для їх ігнорування Git?

У файлі `.gitignore` можна вказати конкретні файли, розширення файлів (наприклад, `*.log`, `*.tmp`), папки (наприклад, `build/`) та шаблони.

5. Яким чином ігнорувати всі файли `*.php` окрім `index.php`?

У файлі `.gitignore` можна додати такі рядки:

```
*.php
!index.php
```

Це означає, що всі `.php` файли будуть ігноруватися, крім `index.php`.

6. Які дії виконує шаблон `**`?

Шаблон `**` у файлі `.gitignore` використовується для ігнорування файлів у всіх підкаталогах. Наприклад, `**/*.log` буде ігнорувати всі `.log` файли у проєкті, незалежно від того, в якій папці вони знаходяться.

7. Які дії виконує шаблон `«!»` ?

Шаблон `!` у файлі `.gitignore` використовується для виключення файлів з ігнорування. Наприклад, якщо вказати `*.log` та `!important.log`, всі `.log` файли будуть ігноруватися, крім `important.log`.

8. Що таке файл `.gitignore` і яка роль його відіграє в системі контролю версій Git?

					Лабораторна робота № 5	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

Файл **.gitignore** – це текстовий файл, який вказує Git, які файли або каталоги потрібно ігнорувати. Це дозволяє уникнути додавання до репозиторію непотрібних або конфіденційних файлів, зберігаючи чистоту та організованість проекту.

9. Як додати коментарі до файлу **.gitignore** для пояснення включених правил?

Коментарі в **.gitignore** починаються з символу **#**. Наприклад:

```
# Ігноруємо всі лог-файли  
*.log
```

10. Якщо виникає конфлікт між локальним та глобальним файлами **.gitignore**, як його вирішити?

Якщо у локальному репозиторії є свій **.gitignore**, він має пріоритет над глобальним. Щоб вирішити конфлікт, потрібно перевірити налаштування **.gitignore** в обох файлах і, за потреби, об'єднати їх. У глобальному **.gitignore** можна ігнорувати файли, які не стосуються конкретного проекту, а в локальному – специфічні для цього репозиторію.

Висновок: У ході лабораторної роботи я отримала навички внесення змін до локальної робочої копії та репозиторію на Github, навчитися працювати з файлом **.gitignore**.

					<i>Лабораторна робота № 5</i>	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		