

Лабораторна робота № 5
з дисципліни «Проектний практикум»

Тема: Внесення змін до репозиторію.

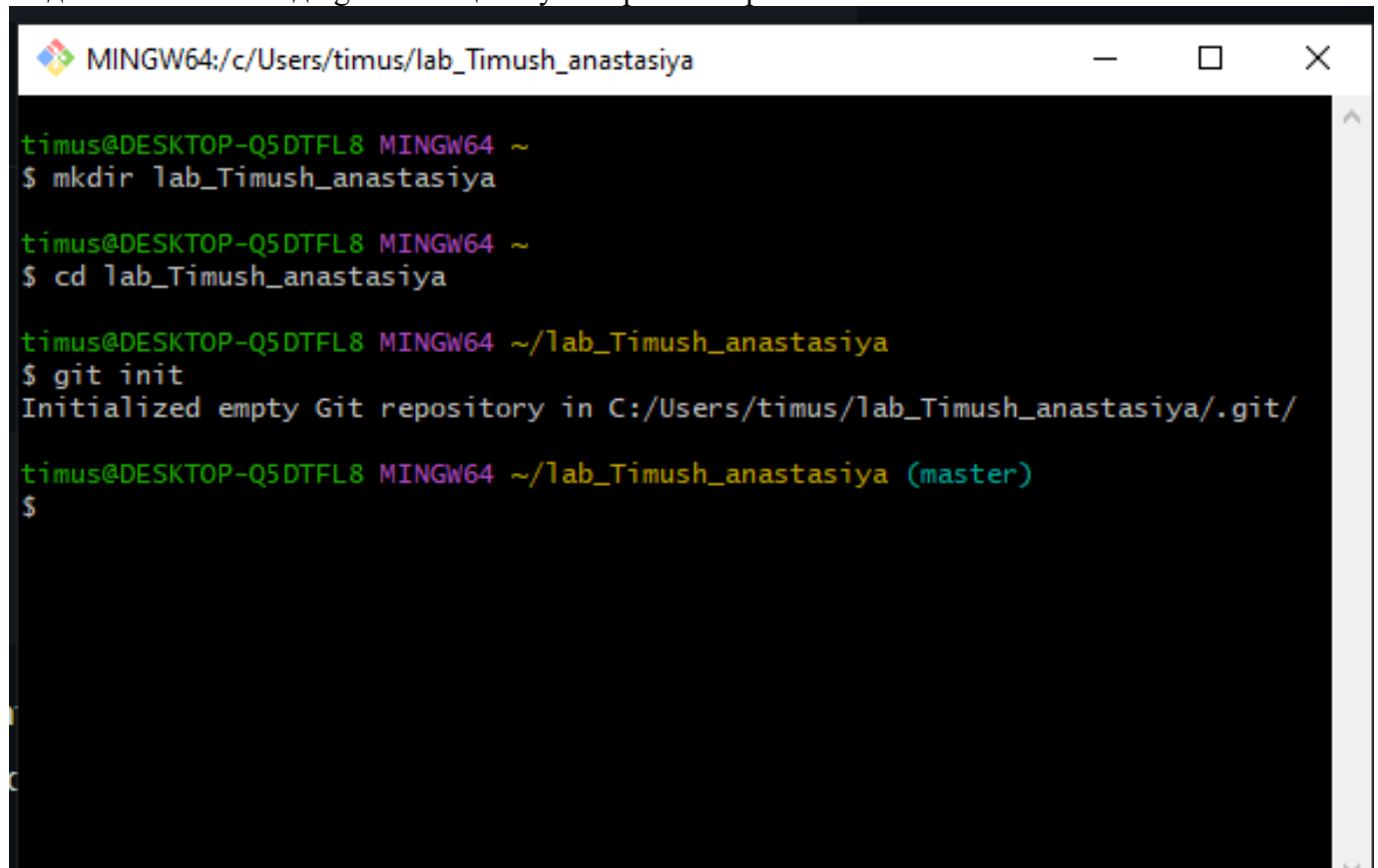
Мета роботи: Отримати навички внесення змін до локальної робочої копії та репозиторію на Github, навчитися працювати з файлом .gitignore.

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

Хід роботи:

Створити порожній каталог для нового репозиторію

За допомогою команди git init ініціалізувати репозиторій



```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

timus@DESKTOP-Q5DTFL8 MINGW64 ~
$ mkdir lab_Timush_anastasiya

timus@DESKTOP-Q5DTFL8 MINGW64 ~
$ cd lab_Timush_anastasiya

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya
$ git init
Initialized empty Git repository in C:/Users/timus/lab_Timush_anastasiya/.git/

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$
```

Додати до індексу нові файли

					<i>Лабораторна робота № 5</i>		
Зм	Лист	№ докум	Підпис	Дата	Внесення змін до репозиторію.		
Розробив							
Перевірив							
Оцінка							
Затв							
					Літ.	Лист.	Листів.
					Група451		

```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git status
On branch master

No commits yet

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

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        text1.txt.txt
        text2.txt.txt

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

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$
```

Зафіксувати зміни в репозиторії

```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        text1.txt.txt
        text2.txt.txt

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

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git commit -m "new file"
[master (root-commit) 3a9f178] new file
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 text1.txt
 create mode 100644 text2.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$
```

Помістити в каталог файли через інтерфейс операційної системи або за допомогою файлового менеджера, та за допомогою консолі перевірити статус репозиторію

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

```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

Untracked files:
(use "git add <file>..." to include in what will be committed)
text1.txt.txt
text2.txt.txt

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

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git commit -m "new file"
[master (root-commit) 3a9f178] new file
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 text1.txt
create mode 100644 text2.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git status
On branch master
nothing to commit, working tree clean

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$
```

Додати нові файли до переліку ігнорування

```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git commit -m "new file"
[master (root-commit) 3a9f178] new file
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 text1.txt
create mode 100644 text2.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git status
On branch master
nothing to commit, working tree clean

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git commit -m "ignor"
[master 648a13e] ignor
1 file changed, 1 insertion(+)
create mode 100644 .gitignore.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ |
```

Додати кілька підкаталогів, розмістити там файли з розширенням *.css, *.js та *.html

Додати файли *.html до переліку ігнорування

Додати до індексу нові файли

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

```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

git commit -m "new file2"
master 1771d18] new file2
10 files changed, 0 insertions(+), 0 deletions(-)
rename .gitignore.txt => .gitignore (100%)
create mode 100644 katalog1/test1.css
create mode 100644 katalog1/test2.css
create mode 100644 katalog1/test3.css
create mode 100644 katalog2/test4.js
create mode 100644 katalog2/test5.js
create mode 100644 katalog2/test6.js
create mode 100644 katalog3/test7.html
create mode 100644 katalog3/test8.html
create mode 100644 katalog3/test9.html

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
git commit -m "ignor2"
master 605e5c1] ignor2
1 file changed, 2 insertions(+), 1 deletion(-)

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
```

Перевірити статус репозиторію

За допомогою команди git remote під'єднати віддалений репозиторій заздалегіть створений на GitHub

```
MINGW64:/c/Users/timus/lab_Timush_anastasiya

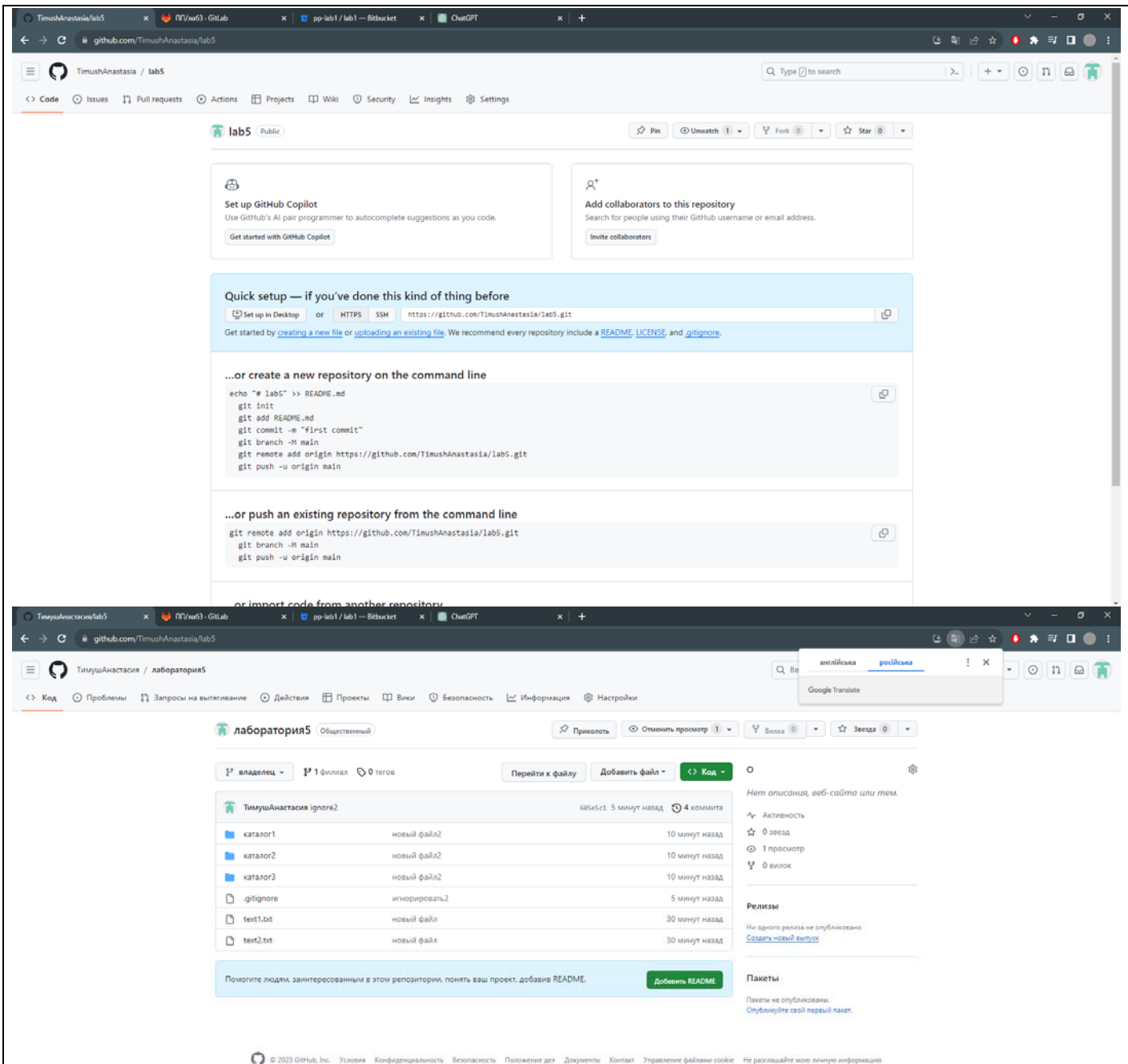
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git remote add origin https://github.com/TimushAnastasia/lab5.git

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/TimushAnastasia/lab5.git'

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$ git push -u origin master
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 8 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (14/14), 1.18 KiB | 606.00 KiB/s, done.
Total 14 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/TimushAnastasia/lab5.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab_Timush_anastasiya (master)
$
```

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



Контрольні питання:

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

1. Створити файл:

`touch .gitignore;`

2. Редагувати файл;

3. Зберегти та закрити файл;

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

4. Додати та зафіксувати файл у Git:

```
git add .gitignore
```

```
git commit -m "Додано файл .gitignore".
```

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

1. Виключення тимчасових та згенерованих файлів;
2. Безпека та конфіденційність;
3. Легша робота з репозиторієм;
4. Зменшення обсягу репозиторію;
5. Покращення продуктивності.

3. Де може бути розміщений файл .gitignore?

Файл .gitignore може бути розміщений в кореневому каталозі Git-репозиторію.

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

```
*.log      # Всі файли з розширенням .log
*.tmp      # Всі файли з розширенням .tmp
/temp/     # Весь каталог temp/
/build/    # Весь каталог build/
secret.txt # Конкретний файл secret.txt
database.yml # Конкретний файл database.yml
debug*/    # Всі каталоги, які починаються з "debug"
*.bak      # Всі файли з розширенням .bak
!important.txt # Виключити відстежування конкретного файлу
/logs/*    # Всі файли в каталозі logs/
```

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

файл .gitignore може виглядати приблизно так:

```
plaintext
Copy code
*.php      # Ігнорувати всі файли з розширенням .php
!index.php # Однак, не ігнорувати файл index.php
```

6. Які дії виконує шаблон **?

шаблон **/ ігнорує будь-які підкаталоги.

7. Які дії виконує шаблон !?

У шаблоні .gitignore, коли ви використовуєте !, це слугує як зворотній шаблон або виняток. Використовуючи !, ви можете заборонити ігнорування файлів чи каталогів, які вже попадають під попередні шаблони ігнорування.

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

Файл .gitignore - це текстовий файл у кореневому каталозі Git-репозиторію, який містить правила для системи контролю версій Git щодо того, які файли та каталоги слід ігнорувати при відстежуванні змін.

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

Роль файлу .gitignore в системі контролю версій Git – це ігнорування тимчасових та згенерованих файлів.

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

В файлі .gitignore, коментарі додаються шляхом використання символу #. Все, що слідує за символом #, вважається коментарем і ігнорується системою Git.

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

Ось кілька кроків, які ви можете виконати:

1. Перегляньте конфлікт:

Відкрийте конфліктний файл .gitignore у текстовому редакторі та подивіться, які правила конфліктують.

2. Вирішіть конфлікт:

Відредагуйте файл .gitignore, вирішивши конфліктні частини.

3. Запустіть злиття:

Після вирішення конфлікту запустіть злиття (merge) або здійсніть коміт зі змінами у файлі .gitignore.

4. Перевірте стан:

Перевірте стан репозиторію за допомогою команди git status та впевніться, що конфлікти вирішені і ви готові до продовження роботи.

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