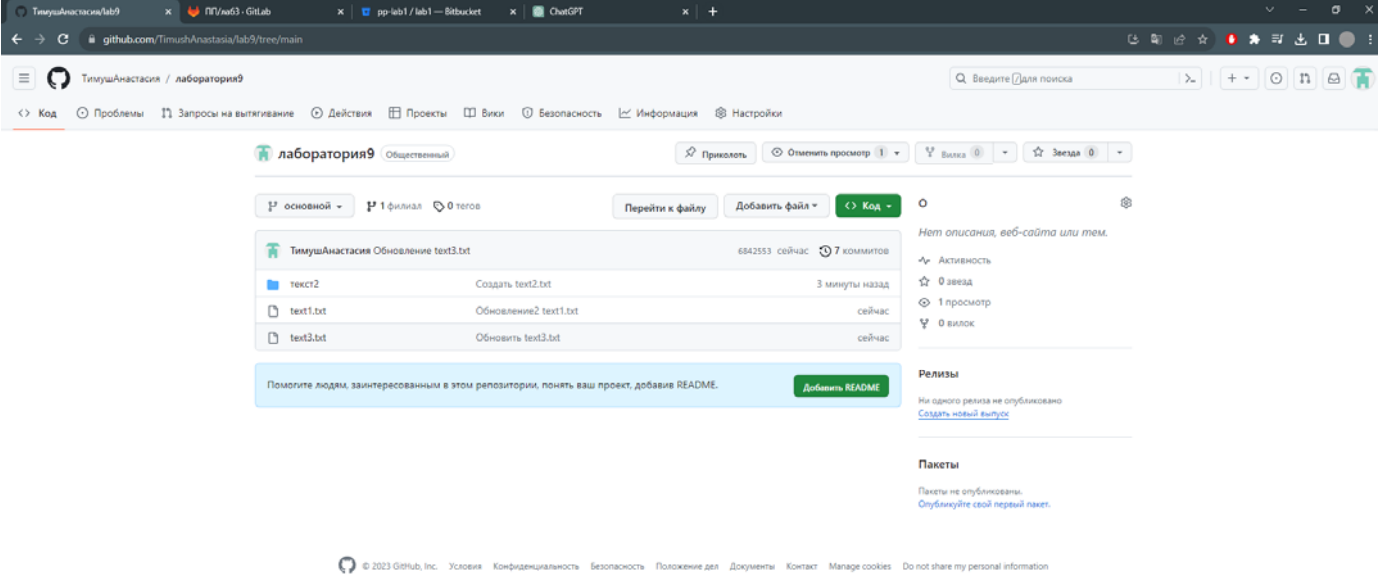


Лабораторна робота № 9  
з дисципліни «Проектний практикум»

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

Створюється новий репозиторій.



Клонувати створений раніше репозиторій (репозиторій повинен мати декілька ревізій для всіх своїх файлів) як локальну робочу копію

					Лабораторна робота № 9							
Зм	Лист	№ докум	Підпис	Дата	Робота з історією змін та отримання старих версій.							
Розробив												
Перевірив												
Оцінка												
Затв					Група451							
					Літ.		Лист.		Листів.			

```
MINGW64:/c:/Users/timus/lab9/lab9
$ mkdir lab9

timus@DESKTOP-Q5DTFL8 MINGW64 ~
$ cd lab9

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9
$ git init
Initialized empty Git repository in C:/Users/timus/lab9/.git/

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9 (master)
$ git clone https://github.com/TimushAnastasia/lab9
Cloning into 'lab9'...
remote: Enumerating objects: 22, done.
remote: Counting objects: 100% (22/22), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 22 (delta 7), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (22/22), 4.50 KiB | 2.25 MiB/s, done.
Resolving deltas: 100% (7/7), done.

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9 (master)
$ cd lab9

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 (main)
$
```

Відображається список змін.

```
MINGW64:/c:/Users/timus/lab9/lab9
Date: Mon Dec 11 17:40:18 2023 +0200

    Update text3.txt

commit d27225ee3d1455993029304d0aace2b73cd9
Author: TimushAnastasia <13395928@TimushAnastasiausers.noreply.github.com>
Date: Mon Dec 11 17:39:10 2023 +0200

    Update2 text1.txt

commit 1bc3d7f65cf4daa81d0e33c1d077c7368d96
Author: TimushAnastasia <13395928@TimushAnastasiausers.noreply.github.com>
Date: Mon Dec 11 17:39:29 2023 +0200

    Update text3.txt

commit daeb08770ac62b37e9b3eb3343f6bc37cf5act4
Author: TimushAnastasia <13395928@TimushAnastasiausers.noreply.github.com>
Date: Mon Dec 11 17:39:01 2023 +0200

    Update text1.txt

commit b184e83f57adfeab8df9b4e25274b1810558caf6
Author: TimushAnastasia <13395928@TimushAnastasiausers.noreply.github.com>
Date: Mon Dec 11 17:38:30 2023 +0200

    Create text3.txt

commit 33af1009a8525c8c1e1f9406b6adfeab483349
Author: TimushAnastasia <13395928@TimushAnastasiausers.noreply.github.com>
Date: Mon Dec 11 17:37:23 2023 +0200

    Create text2.txt

commit 11f73f704f91aba2686641d4afda239b5ad81
Author: TimushAnastasia <13395928@TimushAnastasiausers.noreply.github.com>
Date: Mon Dec 11 17:36:23 2023 +0200

    Create text1.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 (main)
```

Завантажити одну з попередніх версій проекту

					Лабораторна робота № 9	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

```
MINGW64:/c/Users/timus/lab9/lab9

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 (main)
$ git checkout 19c3d7bf65cf4d4a4910a431ce1d077c7268bd96
Note: switching to '19c3d7bf65cf4d4a4910a431ce1d077c7268bd96'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 19c3d7b Update text3.txt
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 ((19c3d7b...))
$
```

Повернутися до останньої версії проекту  
Скасувати останній коміт

```
MINGW64:/c/Users/timus/lab9/lab9

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 19c3d7b Update text3.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 ((19c3d7b...))
$ git checkout main
Previous HEAD position was 19c3d7b Update text3.txt
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 (main)
$ git rest --soft HEAD~1
git: 'rest' is not a git command. See 'git --help'.

The most similar commands are
    restore
    reset

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 (main)
$ git reset --soft HEAD~1

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab9/lab9 (main)
$
```

Переглянути історію комітів, та порівняти її з попередньою

					Лабораторна робота № 9	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

```
MINGW64 ~/Users/Timush/lab9/lab9
$ git checkout 19c3d7b6cf404a4910a411ce1d07c7268bd96
Note: switching to '19c3d7b6cf404a4910a411ce1d07c7268bd96'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

git switch -c <new-branch-name>

Or undo this operation with:

git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 19c3d7b Update text1.txt

$ git checkout main
Previous HEAD position was 19c3d7b Update text1.txt
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

$ git rest --soft HEAD~1
git: 'rest' is not a git command. See 'git --help'.

The most similar commands are
  restore
  reset

$ git reset --soft HEAD~1
$ git log
commit 427321cc1d453913093944084c6a1b31b3c [HEAD]
Author: TimushAnastasia <13395928@timushanastasiausers.github.com>
Date: Mon Dec 11 17:39:50 2023 +0200

    Update1 text1.txt

commit 19c3d7b6cf404a4910a411ce1d07c7268bd96
Author: TimushAnastasia <13395928@timushanastasiausers.github.com>
Date: Mon Dec 11 17:39:29 2023 +0200

    Update text1.txt

commit d481047700c62b3787b5613418f8c37c75ac164
Author: TimushAnastasia <13395928@timushanastasiausers.github.com>
Date: Mon Dec 11 17:39:09 2023 +0200

    Update text1.txt

commit b184e8f57adfeab8d09be25274b1810516caf6
Author: TimushAnastasia <13395928@timushanastasiausers.github.com>
Date: Mon Dec 11 17:38:30 2023 +0200

    Create text1.txt

$
```

Контрольні питання:

1.Скільки існує основних способів вибору ревізій?

- 1.Ідентифікатор коміту
- 2.Змінна HEAD
- 3.Назва гілки
- 4.Відносна адреса коміту
- 5.Ім'я тегу

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

SHA-1 генерує хеш у вигляді 40-значного шістнадцяткового числа (160 біт).

3.За допомогою якої команди можливо продивитися історію коммітів?

git log

4.Перелічіть відомі вам опції команди git log

- 1.--oneline
- 2.-n <кількість> або --max-count=<кількість>
- 3.--author=<ім'я>
- 4.--grep=<шаблон>
- 5.--all

5.Опишіть опції команди git log

- 1.--oneline: Виводить кожен коміт у скороченому форматі на одному рядку.
- 2.-n <кількість> або --max-count=<кількість>: Обмежує кількість виведених комітів.
- 3.--author=<ім'я>: Виводить коміти, здійснені вказаним автором.
- 4.--grep=<шаблон>: Виводить коміти, які містять вказаний шаблон в повідомленні.

					Лабораторна робота № 9	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

5.--all: Виводить історію всіх гілок, а не тільки поточної.

					<i>Лабораторна робота № 9</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		