

Лабораторна робота № 13
з дисципліни «Проектний практикум»

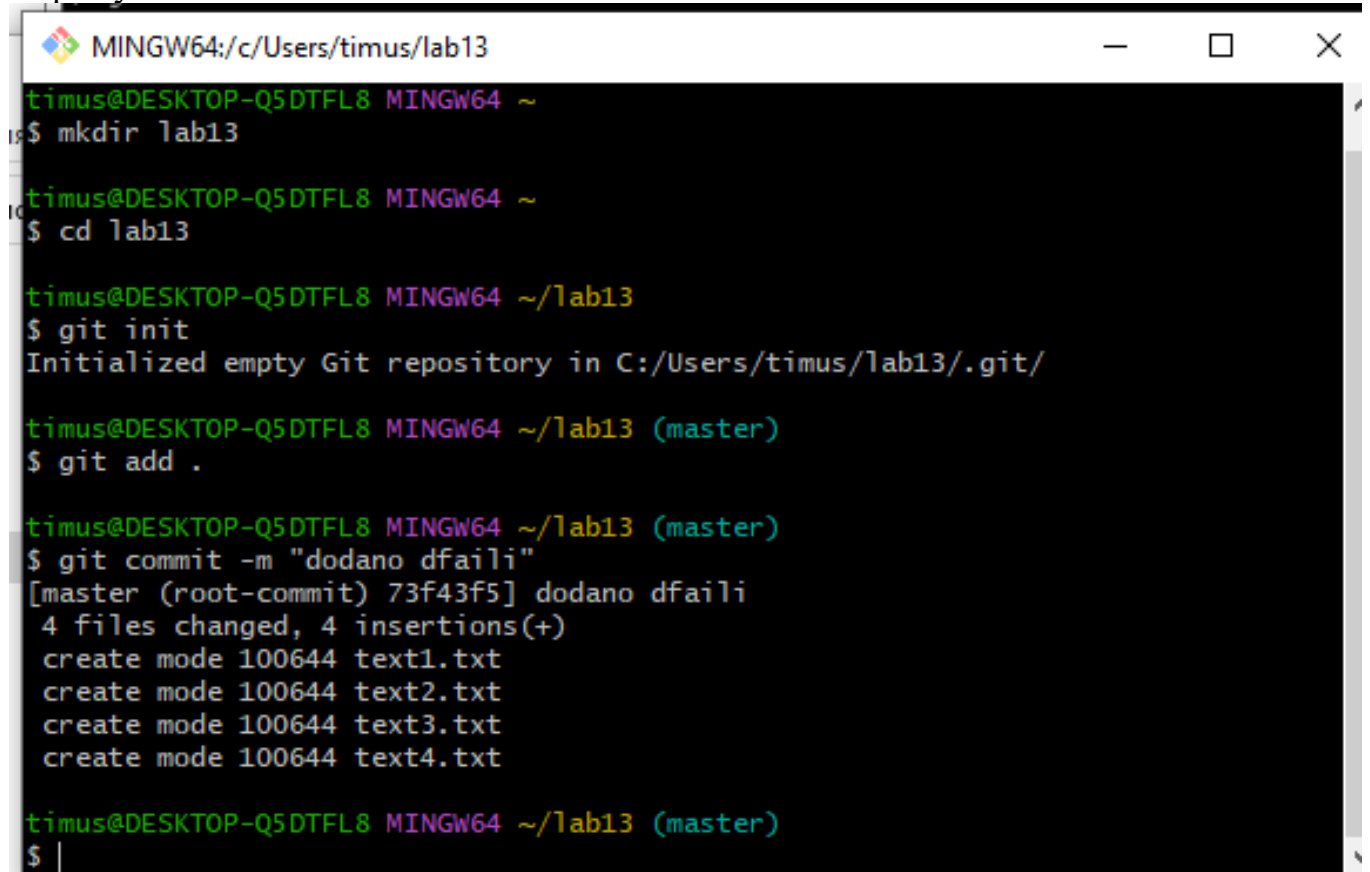
Тема: Штучне створення конфлікту та його вирішення

Мета роботи: Отримати навички вирішення конфліктів у розгалужених проектах Git

Хід роботи:

Для свого репозиторію створити файли та заповнити їх текстовим вмістом, в них в подальшому будуть створені конфлікти

Зафіксувати зміни.



```
MINGW64:/c:/Users/timus/lab13
timus@DESKTOP-Q5DTFL8 MINGW64 ~
$ mkdir lab13

timus@DESKTOP-Q5DTFL8 MINGW64 ~
$ cd lab13

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13
$ git init
Initialized empty Git repository in C:/Users/timus/lab13/.git/

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git commit -m "dodano dfaili"
[master (root-commit) 73f43f5] dodano dfaili
4 files changed, 4 insertions(+)
create mode 100644 text1.txt
create mode 100644 text2.txt
create mode 100644 text3.txt
create mode 100644 text4.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ |
```

Створити нову гілку. Назву гілки сформуєте наступним чином: branchYourLastName1

					<i>Лабораторна робота № 13</i>		
Зм	Лист	№ докум	Підпис	Дата	Штучне створення конфлікту та його вирішення.		
Розробив							
Перевірив							
Оцінка							
Затв							
					Літ.	Лист.	Листів.
					Група451		

```
MINGW64:/c/Users/timus/lab13
$ cd lab13

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13
$ git init
Initialized empty Git repository in C:/Users/timus/lab13/.git/

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git commit -m "dodano dfaili"
[master (root-commit) 73f43f5] dodano dfaili
 4 files changed, 4 insertions(+)
 create mode 100644 text1.txt
 create mode 100644 text2.txt
 create mode 100644 text3.txt
 create mode 100644 text4.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git checkout -b branchTimushA1
Switched to a new branch 'branchTimushA1'

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (branchTimushA1)
$
```

Далі створимо нову гілку, яка буде використовуватися як конфліктуюча при злитті.

```
MINGW64:/c/Users/timus/lab13

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git commit -m "dodano dfaili"
[master (root-commit) 73f43f5] dodano dfaili
 4 files changed, 4 insertions(+)
 create mode 100644 text1.txt
 create mode 100644 text2.txt
 create mode 100644 text3.txt
 create mode 100644 text4.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git checkout -b branchTimushA1
Switched to a new branch 'branchTimushA1'

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (branchTimushA1)
$ git add .

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (branchTimushA1)
$ git commit -m "new text"
[branchTimushA1 860bcb0] new text
 1 file changed, 1 insertion(+)
 create mode 100644 text5.txt

timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (branchTimushA1)
$
```

Ця послідовність команд виконує перемикання на гілку master, додає вміст файл text5.txt і робить комміт.

					Лабораторна робота № 13	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

```
MINGW64:/c/Users/timus/lab13
$ echo "new text in file" > text5.txt
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git commit -am "text5.txt conflict"
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        text5.txt

nothing added to commit but untracked files present (use "git add" to track)
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git add .
warning: in the working copy of 'text5.txt', LF will be replaced by CRLF the next time Git touches it
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git commit -am "conflict"
[master 43eb98e] conflict
1 file changed, 1 insertion(+)
create mode 100644 text5.txt
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$
```

Виник конфлікт

```
MINGW64:/c/Users/timus/lab13
  (use "git add <file>..." to include in what will be committed)
        text5.txt

nothing added to commit but untracked files present (use "git add" to track)
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git add .
warning: in the working copy of 'text5.txt', LF will be replaced by CRLF the next time Git touches it
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git commit -am "conflict"
[master 43eb98e] conflict
1 file changed, 1 insertion(+)
create mode 100644 text5.txt
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master)
$ git merge branchTimushA1
Auto-merging text5.txt
CONFLICT (add/add): Merge conflict in text5.txt
Automatic merge failed; fix conflicts and then commit the result.
timus@DESKTOP-Q5DTFL8 MINGW64 ~/lab13 (master|MERGING)
$
```

Контрольні питання:

1.Що таке конфлікт?

					Лабораторна робота № 13	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

Конфлікт у Git або GitHub виникає, коли дві або більше гілок (branches) або версії файлів мають зміни в одному й тому ж місці, і система не може автоматично визначити, яку версію слід обрати. Це може трапитися, наприклад, коли два різних розробники вносять зміни в один і той же рядок коду або коли ви намагаєтеся злити гілку з основною гілкою, яка вже змінилася.

2.Як можливо штучно створити конфлікт?

- 1.Створіть нову гілку;
- 2.Внесіть зміни в одному файлі;
- 3.Зробіть зміни в основній гілці;
- 4.Злите зміни з нової гілки в основну.

3.Як можливо вирішити конфлікт?

- 1.Виріште конфлікт:

Git позначить конфлікт у файлах і потрібно вручну вирішити конфлікт.

Редагуйте конфліктні файли, видаливши непотрібні рядки або обираючи необхідні зміни.

- 2.Збережіть зміни:

Після вирішення конфлікту збережіть зміни

4.Яку команду потрібно використати, щоб порівняти результат злиття з тим, що було у вашій гілці до злиття, або іншими словами побачити, що привнесло дане злиття?

git diff. Ось приклад використання:

git diff ваша_гілка..гілка_після_злиття

5.Який синтаксис потрібно використати щоб отримати повний список усіх унікальних комітів, що є у будь-якій з конфліктуючих гілок потрібно?

Для отримання повного списку усіх унікальних комітів, які є у будь-якій з конфліктуючих гілок, ви можете скористатися командою git log. Далі, ви можете скористатися командою sort та uniq для вилучення дублікатів.

					Лабораторна робота № 13	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		