



Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря  
Сікорського” Факультет інформатики та обчислювальної  
техніки  
Кафедра інформаційних систем та технологій

### **Лабораторна робота №2**

із дисципліни *«Технології розроблення програмного забезпечення»*

**Тема: «Використання основних команд Git»**

Виконала:

Студентка групи ІА-24

Сіденко Д.Д.

Перевірив:

Мякий М.Ю.

**Мета:** У межах даної лабораторної роботи передбачено освоїти використання локальної системи контролю версій Git.

### **Теоретичні відомості:**

Система контролю версій Git — це інструмент, який дозволяє розробникам відслідковувати зміни в коді, спільно працювати над проектами та зберігати історію змін. Git дозволяє створювати резервні копії, повертатися до попередніх версій коду, а також зливати зміни від різних розробників.

Основні команди Git:

`git init`: Ініціалізує новий Git-репозиторій.

`git clone [url]`: Копіює існуючий репозиторій з віддаленого сервера на локальний комп'ютер.

`git add [file]`: Додає зміни вказаного файлу до індексу (staging area).

`git commit -m "message"`: Зберігає знімок змін з коментарем.

`git status`: Показує стан репозиторію, які файли змінені, додані або готові до коміту.

`git push [origin] [branch]`: Відправляє локальні зміни на віддалений репозиторій.

`git pull [origin] [branch]`: Оновлює локальний репозиторій, завантажуючи зміни з віддаленого репозиторію.

`git branch`: Показує список всіх гілок в репозиторії.

`git checkout [branch]`: Переключається на вказану гілку.

`git merge [branch]`: Об'єднує зміни з вказаної гілки в поточну.

`git cherry-pick <hash>`: Дозволяє взяти один або кілька комітів з однієї гілки та застосувати їх до поточної гілки. Це корисно, коли потрібно перенести конкретні зміни без злиття всіх змін з однієї гілки.

`git switch`: Використовується для зміни активної гілки. Це зручніша альтернатива команді `git checkout`, яка в Git виконує кілька різних функцій.

### **Хід роботи:**

**Крок 1:** Для початку роботи потрібно проініціалізувати новий репозиторій Git.

```
D:\unik\3_1\trpz\lab1-zvit>git init
Initialized empty Git repository in D:/unik/3_1/trpz/lab1-zvit/.git/
```

**Крок 2:** Створити три гілки за допомогою різних команд з власною назвою

та одну базову “master”. Під час виконання даного кроку було допущено певну помилку, через що гілка master стала частиною гілки br1. Для уникнення даної прикрої помилки варто було створити файл для коміту до створення гілки br1.

Для створення гілок було використано наступні команди

**git branch new-branch-name** — створює нову гілку, не перемикаючи на неї;

**git checkout -b new-branch-name** — створює нову гілку і перемикає на неї (старіший спосіб);

**git switch -c new-branch-name** — створює нову гілку і перемикає на неї (сучасний спосіб).

```
D:\unik\3_1\trpz\lab1-zvit>git commit -m "init commit"
On branch master

Initial commit

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

D:\unik\3_1\trpz\lab1-zvit>git branch br1
fatal: not a valid object name: 'master'

D:\unik\3_1\trpz\lab1-zvit>git switch -c br1
Switched to a new branch 'br1'

D:\unik\3_1\trpz\lab1-zvit>git branch

D:\unik\3_1\trpz\lab1-zvit>git log
fatal: your current branch 'br1' does not have any commits yet

D:\unik\3_1\trpz\lab1-zvit>echo "cm1 to br1" >> text.txt

D:\unik\3_1\trpz\lab1-zvit>git add .

D:\unik\3_1\trpz\lab1-zvit>git commit -m "cm1 to br1"
[br1 (root-commit) a41c6d6] cm1 to br1
 1 file changed, 1 insertion(+)
 create mode 100644 text.txt

D:\unik\3_1\trpz\lab1-zvit>git branch
* br1

D:\unik\3_1\trpz\lab1-zvit>git branch master

D:\unik\3_1\trpz\lab1-zvit>git branch
* br1
  master
```

```
D:\unik\3_1\trpz\lab1-zvit>git switch master
Switched to branch 'master'

D:\unik\3_1\trpz\lab1-zvit>git checkout -b br2
Switched to a new branch 'br2'

D:\unik\3_1\trpz\lab1-zvit>git switch master
Switched to branch 'master'

D:\unik\3_1\trpz\lab1-zvit>git checkout -b br3
Switched to a new branch 'br3'

D:\unik\3_1\trpz\lab1-zvit>git switch master
Switched to branch 'master'

D:\unik\3_1\trpz\lab1-zvit>git branch
  br1
  br2
  br3
* master
```

**Крок 3:** Створити відповідну кількість комітів у кожній гілці.

За допомогою команд “echo "cmN to brN" >> text.txt”, “git add .”, “git commit

-m “cmN to brN” я досягла поставленої мети.

```
D:\unik\3_1\trpz\lab1-zvit>git log --all --oneline
9a7543e (HEAD -> br3) cm3 to br3
3dd20d9 cm2 to br3
df09b19 cm1 to br3
6160b9a (br2) cm2 to br2
56092a4 cm1 to br2
a41c6d6 (master, br1) cm1 to br1
```

**Крок 4:** Створити коміт у гілці “master” та оновити гілки 1,2,3 до версії мастер за допомогою 3-х різних команд.

Для цього було використано:

*git merge* — створює новий коміт із об'єднанням змін;

*git rebase* — переносить коміти однієї гілки поверх іншої, створюючи лінійну історію;

*git cherry-pick <commit-hash>* — переносить конкретний коміт в поточну гілку.

При вирішенні конфліктів зберігався повний текст з обох гілок.

```
D:\unik\3_1\trpz\lab1-zvit>git switch br1
Switched to branch 'br1'

D:\unik\3_1\trpz\lab1-zvit>git merge master
Updating a41c6d6..7192419
Fast-forward
 text.txt | 1 +
 1 file changed, 1 insertion(+)

D:\unik\3_1\trpz\lab1-zvit>git switch br2
Switched to branch 'br2'

D:\unik\3_1\trpz\lab1-zvit>git rebase master
Auto-merging text.txt
CONFLICT (content): Merge conflict in text.txt
error: could not apply 56092a4... cm1 to br2
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicted files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
Could not apply 56092a4... cm1 to br2

D:\unik\3_1\trpz\lab1-zvit>git add .

D:\unik\3_1\trpz\lab1-zvit>git rebase --continue
hint: Waiting for your editor to close the file... unix2dos: converting file D:/unik/3_1/trpz/lab1-zvit/.git/COMMIT_EDITMSG
to DOS format...
dos2unix: converting file D:/unik/3_1/trpz/lab1-zvit/.git/COMMIT_EDITMSG to Unix format...
[detached HEAD 39a095c] cm1 to br2, master to br2
 1 file changed, 1 insertion(+)
Successfully rebased and updated refs/heads/br2.

D:\unik\3_1\trpz\lab1-zvit>git switch br3
Switched to branch 'br3'

D:\unik\3_1\trpz\lab1-zvit>git cherry-pick 7192419cfa2f9d1341f7cd8743f28a15bc72390f
Auto-merging text.txt
CONFLICT (content): Merge conflict in text.txt
error: could not apply 7192419... cm1 to master
hint: After resolving the conflicts, mark them with
hint: "git add/rm <paths>", then run
hint: "git cherry-pick --continue".
hint: You can instead skip this commit with "git cherry-pick --skip".
hint: To abort and get back to the state before "git cherry-pick",
hint: run "git cherry-pick --abort".

D:\unik\3_1\trpz\lab1-zvit>git add .

D:\unik\3_1\trpz\lab1-zvit>git cherry-pick --continue
hint: Waiting for your editor to close the file... unix2dos: converting file D:/unik/3_1/trpz/lab1-zvit/.git/COMMIT_EDITMSG to DOS format...
dos2unix: converting file D:/unik/3_1/trpz/lab1-zvit/.git/COMMIT_EDITMSG to Unix format...
[br3 b28cf29] cm1 to master, master to br3
Date: Fri Sep 27 12:48:32 2024 +0300
 1 file changed, 3 insertions(+), 2 deletions(-)
```

Я отримала такий результат:

```
D:\unik\3_1\trpz\lab1-zvit>git log --all --oneline --graph
* b28cf29 (HEAD -> br3) cm1 to master, master to br3
* 9a7543e cm3 to br3
* 3dd20d9 cm2 to br3
* df09b19 cm1 to br3
| * 76bd432 (br2) cm2 to br2
| * 39a095c cm1 to br2, master to br2
| * 7192419 (master, br1) cm1 to master
|/
* a41c6d6 cm1 to br1
```

**Висновок:** у даній лабораторній роботі я спробувала на практиці основні команди для локального контролю версій Git.