

з дисципліни «Проектний практикум»

Тема: Створення та видалення тегів.

Мета роботи: Отримати навички додавання тегів до ревізій проекту в СКВ Git.

Хід роботи:

Клонувати створений раніше репозиторій як локальну робочу копію

За допомогою консольного інструменту `git tag` додати теги для поточної ревізії

Зробити зміни в локальній робочій копії, зафіксувати зміни та додати новий тег для нової ревізії

Перевірити наявні теги за допомогою інструменту `git tag`

```

MINGW64~/c/Users/Bwaa/lab8
Вика@DESKTOP-NOBHIAE MINGW64 ~ (master)
$ mkdir lab8
Вика@DESKTOP-NOBHIAE MINGW64 ~ (master)
$ cd lab8
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git clone https://github.com/Timush-Viktoriya/lab8
Cloning into 'lab8'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ dir indit
dir: cannot access 'indit': No such file or directory
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git indit
git: 'indit' is not a git command. See 'git --help'.

The most similar command is
    init
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git init
Initialized empty Git repository in C:/Users/Вика/lab8/.git/
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git tag
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git tag -a v1.4 -m "version 1.4"
fatal: Failed to resolve 'HEAD' as a valid ref.
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git show-ref HEAD
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (master)
$ git symbolic-ref HEAD refs/heads/lab8
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (lab8)
$ git show-ref HEAD
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (lab8)
$ git commit -m "commit"
On branch lab8

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    lab8/

nothing added to commit but untracked files present (use "git add" to track)
Вика@DESKTOP-NOBHIAE MINGW64 ~/lab8 (lab8)

```

Перемкнутися за допомогою теґа на попередню раніше створений коміт

[Повернутися на останню версію репозиторію](#)

Вивантажити зафіксовані зміни на Github

Вивантажити створені теги на Github

					Лабораторна робота № 8						
Зм	Лист	№ докум	Підпис	Дата							
Розробив					Створення та видалення тегів.			Літ.	Лист.	Листів.	
Перевірів											
Оцінка											
Затв											
						Група451					

```
MINGW64/c/Users/Bma/lab8
Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git add .
warning: adding embedded git repository: lab8
hint: You've added another git repository inside your current repository.
hint: Clones of the outer repository will not contain the contents of
hint: the embedded repository and will not know how to obtain it.
hint: If you meant to add a submodule, use:
hint:     git submodule add <url> lab8
hint: If you added this path by mistake, you can remove it from the
hint: index with:
hint:     git rm --cached lab8
hint: See "git help submodule" for more information.
Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git status
On branch lab8

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   lab8

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag -a v1.4 -m "my version 1.4"
fatal: Failed to resolve 'HEAD' as a valid ref.

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git branch labb
fatal: not a valid object name: 'lab8'

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git branch lab8
fatal: not a valid object name: 'lab8'

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git checkout lab8
Updated 0 paths from the index

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag -a v1.4 -m "my version 1.4"
fatal: Failed to resolve 'HEAD' as a valid ref.

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag -a 1.4 -m "my message"
fatal: Failed to resolve 'HEAD' as a valid ref.

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git show-ref HEAD
```

```
MINGW64/c/Users/Bma/lab8
Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag v1.4
fatal: Failed to resolve 'HEAD' as a valid ref.

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git status
On branch lab8

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   lab8

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git tag 1.4
fatal: Failed to resolve 'HEAD' as a valid ref.

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git show-ref HEAD

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git symbolic-ref HEAD refs/heads/lab8

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ cat .git/HEAD
ref: refs/heads/lab8

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git update-ref -d HEAD

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git symbolic-ref HEAD refs/heads/lab8

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git log
fatal: your current branch 'lab8' does not have any commits yet

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git show-ref

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git checkout -b lab8
Switched to a new branch 'lab8'

Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ git status
On branch lab8

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   lab8
```

					Лабораторна робота № 8	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

```
MINGW64:/c:/Users/Bma/lab8
$ git tag
v1.4
$ git tag v1.5
$ git tag v1.5.2
$ git tag v1.5.2
$ git tag
v1.4
v1.5
v1.5.2
$ git push origin v1.4
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
$ git push origin --tags
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
$ git remote -v
$ git remote add origin https://github.com/Timush-Viktoriya/lab8
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/Timush-Viktoriya/lab8
  Push URL: https://github.com/Timush-Viktoriya/lab8
  HEAD branch: main
  Remote branch:
    main new (next fetch will store in remotes/origin)
$ git push origin --tags
Enumerating objects: 2, done.
Counting objects: 100% (2/2), done.
Writing objects: 100% (2/2), 195 bytes | 97.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Timush-Viktoriya/lab8
 * [new tag]          v1.4 -> v1.4
 * [new tag]          v1.5 -> v1.5
 * [new tag]          v1.5.2 -> v1.5.2
$
```

Перевірити наявність нових файлів на Github через web-інтерфейс

Стара версія

 **lab8** Public

Pin Unwatch 1

main 1 branch 0 tags

Go to file Add file Code

Нова версія

 **lab8** Public

Pin Unwatch 1

main 1 branch 3 tags

Go to file Add file Code

					Лабораторна робота № 8	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

```
MINGW64:/c/Users/Вика/dir_for_git
Вика@DESKTOP-NOBH1AE MINGW64 ~/lab8 (lab8)
$ cd ..

Вика@DESKTOP-NOBH1AE MINGW64 ~ (master)
$ cd dir_for_git

Вика@DESKTOP-NOBH1AE MINGW64 ~/dir_for_git (master)
$ git remote add origin https://github.com/Timush-Viktoriya/lab8
error: remote origin already exists.

Вика@DESKTOP-NOBH1AE MINGW64 ~/dir_for_git (master)
$ git clone https://github.com/Timush-Viktoriya/lab8
Cloning into 'lab8'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 5 (delta 0), reused 2 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.

Вика@DESKTOP-NOBH1AE MINGW64 ~/dir_for_git (master)
$ git tag

Вика@DESKTOP-NOBH1AE MINGW64 ~/dir_for_git (master)
$
```

Контрольні питання:

1. За допомогою якої команди можливо переглянути список наявних тегів в Git?
git tag

2. Які два основних типи тегів використовує Git?

Легковагові теги (Lightweight Tags): наприклад git tag v1.0

Анотовані теги (Annotated Tags): наприклад git tag -a v1.0 -m "Release version 1.0"

3. Розкажіть про кожен тип тегів Git

Легковагові теги:

Створюються за допомогою команди git tag без додаткових параметрів.

Є простими покажчиками на конкретний коміт у історії проекту.

Не містять додаткової інформації, такої як ім'я автора, електронна пошта чи дата створення.

Зазвичай використовуються для простого позначення конкретних комітів, наприклад, для визначення версій програмного продукту.

Анотовані теги:

Створюються за допомогою опції -a або --annotate команди git tag, або через git tag -am.

Містять додаткову інформацію, таку як ім'я автора, електронна пошта, дата створення тегу та повідомлення.

					Лабораторна робота № 8	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

Є повноцінними об'єктами Git і можуть містити коментарі, подібні до коментарів комітів.

Зазвичай використовуються для визначення конкретних версій програмного продукту та можуть містити більше деталей про версію.

4. Чи можливо позначати вже пройдені комміти?

Щоб позначити вже існуючий коміт тегом, використовуйте команду `git tag` разом з хешем коміту або його частковим ідентифікатором. Наприклад: `git tag v1.0 1a2b3c4`

5. Для видалення тега в локальному репозиторії необхідно виконати команду?

Для видалення тега в локальному репозиторії використовуйте команду `git tag -d` разом із ім'ям тегу.

					<i>Лабораторна робота № 8</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		