

Лабораторна робота №3. Клонування репозиторію засобами Git for Windows

Мета:Отримати навички створення локальної робочої копії репозиторію за допомогою Git for Windows

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

Хід роботи:

1. Перевірити глобальні налаштування Git for Windows
2. У випадку невідповідності вашим даним налаштувати глобальну конфігурацію Git for Windows
3. Клонувати створений раніше репозиторій як локальну робочу копію
4. Завантажити раніше створений репозиторій з Github в архіві. Дослідити зміст клонованих репозиторіїв, розархівованих репозиторіїв зі змістом на сервісі Github

```
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-first-repo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-first-repo (main)
$ ls -a
./ ../ .git/ README.md
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-first-repo (main)
$ cd .git
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-first-repo/.git (GIT_DIR!)
$ ls
HEAD config description hooks/ index info/ logs/ objects/ packed-refs refs/
```

Клонований репозиторій (через git clone):

Містить всі файли проєкту (код, README, конфігураційні файли)

Є прихована папка .git, яка зберігає історію комітів, гілки та налаштування репозиторію

Дає змогу працювати з Git локально: коміти, створення гілок, повернення до старих версій

Розархівований репозиторій (ZIP):

Містить тільки файли проєкту, без історії Git

Немає папки .git, тому неможливо відновити попередні версії або працювати з гілками

Веб-інтерфейс GitHub:

Показує всі файли та README

					Лабораторна робота № 3			
Ізм.	Лист	№ докум.	Підп.	Дата				
Розроб.		Литвиненко			Тема: Отримати навички створення локальної робочої копії репозиторію за допомогою Git for Windows	Літера	Лист	Листів
Перевір.						У	1	3
						Група 451		
Н. контр.								
Затв.								

Дозволяє переглядати коміти та гілки, але не дозволяє редагувати локально без клонування репозиторію

5. Зробити висновок про відмінності у змісті клонованого репозиторію, завантаженого та даних, що відображає веб-інтерфейс

Головна відмінність полягає в тому, що клонований репозиторій зберігає повну історію змін і гілки завдяки наявності папки `.git`. Завантажений архів містить лише файли проєкту без історії Git. Веб-інтерфейс показує файли та коміти, але не дає можливості працювати з репозиторієм локально без його клонування.

6. Повторити пункти 3 - 5 пункти для сервісів Bitbucket та Gitlab з урахуванням змін глобальних налаштувань, якщо це потрібно

```
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git
$ cd my-bitbucket-repo

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-bitbucket-repo (main)
$ ls
README.md

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-bitbucket-repo (main)
$ ls -a
./ ../ .git/ .gitignore README.md

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git
$ cd my-gitlab-repo

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-gitlab-repo (master)
$ ls
CONTRIBUTING.md LICENSE Program.cs README.md dotnetcore.csproj

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-gitlab-repo (master)
$ ls -a
./ ../ .git/ .gitignore .gitlab-ci.yml .gitpod.yml CONTRIBUTING.md LICENSE Program.cs README.md dotnetcore.csproj
```

7. Дослідити зміст каталогу `.git` за методом описаним в посиланні [2]

```
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-gitlab-repo (master)
$ cd .git

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-gitlab-repo/.git (GIT_DIR!)
$ ls
HEAD config description hooks/ index info/ logs/ objects/ packed-refs refs/

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git
$ cd my-bitbucket-repo

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-bitbucket-repo (main)
$ cd .git

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git/my-bitbucket-repo/.git (GIT_DIR!)
$ ls
HEAD config description hooks/ index info/ logs/ objects/ packed-refs refs/
```

Каталог `.git` містить повну історію змін проєкту, інформацію про гілки та теги, налаштування репозиторію та автоматизаційні скрипти. Зробити висновок про зміст вашого каталогу `.git` та додати цю інформацію у звіт

8. З урахуванням навичок, отриманих під час виконання лабораторної роботи №2, виконати клонування репозиторію з сервісів Bitbucket та Gitlab з використанням SSH-тунелю

```

user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git (main)
$ git clone https://dmytrotlab-admin@bitbucket.org/dmytrotlab/my-bitbucket-repo.git
Cloning into 'my-bitbucket-repo'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (4/4), done.
user@WIN-E8GF7QGQ20K MINGW64 ~/dir_for_git (main)
$ git clone git@gitlab.com:my-gitlab-group6050940/my-gitlab-repo.git
Cloning into 'my-gitlab-repo'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 13 (delta 0), reused 0 (delta 0), pack-reused 10 (from 1)
Receiving objects: 100% (13/13), 6.94 KiB | 1014.00 KiB/s, done.

```

9. Для кожного етапу роботи зробити знімки екрану або скопіювати текст консолі та додати їх у звіт з описом кожного скіншота
10. Дати відповіді на контрольні запитання
11. Зберегти звіт у форматі PDF

Контрольні питання:

1. Яка команда відповідає за клонування віддаленого репозиторію у локальну робочу копію?
Команда, що відповідає за клонування віддаленого репозиторію у локальну робочу копію, це `git clone <адреса репозиторію>`. Вона створює локальну копію проекту з усіма файлами, історією комітів та гілками.
2. Що потрібно зробити, для завантаження репозиторію, якщо на комп'ютері відсутній Git для сервісу GitHub?
Якщо на комп'ютері відсутній Git для сервісу GitHub, потрібно встановити Git із сайту git-scm.com, після чого виконати команду `git clone` або завантажити репозиторій у форматі ZIP через кнопку Download ZIP на сторінці проекту.
3. Що потрібно зробити, для завантаження репозиторію, якщо на комп'ютері відсутній Git для сервісу Bitbucket?
Для завантаження репозиторію з Bitbucket, якщо Git не встановлено, також слід інсталиувати Git і скористатися командою `git clone` або кнопкою Download repository → ZIP.
4. Що таке GitHub Desktop, яке його призначення?
GitHub Desktop це графічна програма, яка дозволяє працювати з Git та репозиторіями GitHub без використання терміналу. Вона спрощує виконання основних дій, таких як клонування, коміти, пуші та перевірка змін.

					<i>Лабораторна робота № 3</i>			
Ізм.	Лист	№ докум.	Підп.	Дата				
Розроб.		Литвиненко			<i>Тема: Отримати навички створення локальної робочої копії репозиторію за допомогою Git for Windows</i>	Літера	Лист	Листів
Перевір.						У	1	3
Н. контр.						<i>Група 451</i>		
Затв.								

5. Які візуальні оболонки Git ви знаєте?

Серед найпопулярніших візуальних оболонок Git є GitHub Desktop, Sourcetree, GitKraken, TortoiseGit, SmartGit та Visual Studio Code, який має вбудовану підтримку Git.

					Лабораторна робота № 3	Лист
Ізм.	Лист	№ докум.	Підп.	Дата		2