Лабораторна робота №1 **Git. Створення репозиторія і перший комміт**

*Мета роботи:* ознайомитися з основними командами Git, робота з сервісом GitHub.

*Знати:* основні команди Git.

*Уміти:* налаштовувати такористуватися основними можливостями Git*;* створювати репозиторій.

*Обладнання:* технічне забезпечення: персональний комп’ютер, програмне забезпечення: текстовий редактор Notepad++, Visual Studio Code, GitBash, браузер.

***Самостійна підготовка до виконання лабораторної роботи***

1. Записати в зошит тему, мету, обладнання, практичні завдання свого варіанту(при наявності).
2. За літературою до лабораторної роботи вивчити необхідний теоретичний матеріал та письмово відповісти на запитання для самостійної підготовки.
3. За інструкцією до лабораторної роботи ознайомитись з порядком виконання роботи.

***Короткі теоретичні відомості***

Git – це розподілена система керування версіями файлів, яка була створена Лінуксом Торвальдсом у 2005-му році для керування розробкою ядра Linux. До проектів, які використовують дану систему також належать: Android, Drupal, Cairo, GNU Core Utilities, Mesa, Wine, Chromium, jQuery, PHP, NASM, MediaWiki, osCommerce, проекти Twitter та Yahoo, а також деякі дистрибутиви Linux.

Система розроблялась як набір програм командної стрічки з параметрами, спеціально розроблених для використання у сценаріях, що дозволило створювати на її базі спеціалізовані системи контролю версій.

Сховище Git являє собою каталог файлової системи, в якому знаходяться:

1. Файли налаштувань сховища.
2. Файли журналів, які зберігають зберігають операції, здійснені над сховищем.
3. Індекс – описує розташування файлів.
4. Сховище, яке власне і зберігає файли.

Файли в Git можуть знаходитись в одному із трьох станів: зафіксованому, зміненому і підготовленому. Зафіксований файл той, який вже збережений у локальні базі. До змінених файлів відносяться ті, які змінились, але ще не були зафіксовані. Підготовлені файли – це змінені файли, що помічені для включення у наступну фіксацію (commit).

Тому в проектах, що використовують Git є три частини: каталог Git (Git repository), робочий каталог (working directory) і область підготовлених файлів (staging area) (рис. 1).



Рис. 1. Частини Git.

Git repository – це місце, де Git зберігає метадані і базу даних об'єктів вашого проекту. Це найбільш важлива частина Git, саме вона копіюється, коли ви отримуєте (clone) сховище з іншого комп'ютера. За замовчуванням сховище зберігається в підкаталозі .git кореневого каталогу локальної копії дерева файлів. Структура сховища не відповідає реальні структурі файлового дерева.

Working directory – це отримана (clone) зі сховища копія певної версії проекту.

Staging area – це файл, який містить інформацію про те, що повинно ввійти в наступну фіксацію (commit).

Будь-яке дерево файлів можна перетворити у сховище git виконавши команду *git init.*

*Тааблиця 1*

Основні команди Git *ййййййййййййййййййййййй*

|  |  |
| --- | --- |
|  |  |
| git push | внесе зміни у віддалений репозиторій |
| git status | покаже стан проекту |
| git reset | перейде до попередньої комітом |
| git revert | відмінить зміни, які виконані окремим попереднім комітом |
| git rm -r | проіндексує зміни |
| git commit -m «примітка» | виконає локально коміт |
| git add | внесення файлів у індекс |
| git log | виведе інформацію про коміти |
| git show | покаже зміни внесені певним комітом |
| git diff | створить копію віддаленого репозиторію |
| git pull | створить компію віддаленого репозиторію |
| git branch | перечислить гілки проекту |
| git checkout master | переключить в гілку master |
| git clone | створює копію віддаленого сховища |

***Порядок виконання роботи***

1. Вивчіть основні конструкції та поняття, які розглянуті на лекції.
2. Створіть папку з файлом html. Використовуючи основні команди Git:

* ініціалізуйте git у даній папці;
* змініть стандартні конфігурації імені та електронної адреси на власні;
* перевірте стан проекту;
* внесіть файли у індекс;
* створіть перший комміт та перевірте стан проекту;
* внесіть деякі зміни у файл html (н-д, додайте рядок з тегом <h1>) та перевірте стан проекту;
* внесіть файли у індекс та виведіть інформацію про комміти;
* видаліть останній комміт з історії коммітів;

1. Створіть папку з файлами власного проекту (н-д, вашого сайту). Використовуючи можливості Git`а та сервіса GitHub додайте проект у власний репозиторій, створіть декілька коммітів.

***Питання для самостійної роботи***

1. Розкрийте поняття «система керування версіями». Які існують системи керування версіями?

2. Опишіть основні переваги та недоліки Git’а порівняно з іншими системами керування версіями.

3. Перерахуйте основні команди Git.

***Література:***

1. <https://githowto.com/uk>
2. <https://git-scm.com/book/>

***Звіт до роботи***

1. Оформити лабораторну роботу в зошиті. Записати тему, мету, знати, уміти, обладнання.
2. Письмово відповісти на запитання для самостійної підготовки.
3. Виконати завдання згідно свого варіанту.
4. Зробити висновок.

***Індивідуальне завдання***

1. Розробити проект, використовуючи можливості Git та GitHub (галуження, злиття гілок та мітки).

2. Розробіть колективний проект вашого сайту використовуючи основні можливості Git’а та GitHub (галуження, злиття гілок та мітки).