**Міністерство освіти і науки України**

**Національний університет «Запорізька політехніка»**

кафедра програмних засобів

**ЗВІТ**

з лабораторної роботи №1

з дисципліни "Верифікація цифрових систем"

на тему:

"ОСНОВИ GIT"

Варіант №7

Виконав:

студент групи КНТ-227 Ю. Є. Горіченко

Прийняв:

к.т.н., доцент Т. І. Каплієнко

2019

**Тема:** основи GIT.

**Мета:** вивчити основні команди по роботі з локальним репозиторієм GIT.

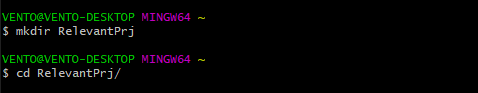
**Завдання на лабораторну роботу:**

1. Ознайомитися з теоретичними відомостями щодо основних команд GIT.
2. Виконати свій варіант завдання, аналогічно нульовому варіанту.

**Виконання роботи (варіант 3)**

Ініціалізувати папку з майбутнім проектом в git. Створити файл .gitignore з описом ігнорованих файлів (\*. [oa], \* ~). Користувач вводить свої дані, створює папку проекту, створює в ній файл із завданнями. Виконує комміт для фіксації внесених змін. Створює нову гілку, відкриває файл і виконує завдання, виконує комміт. Повертається в гілку master. Переглядає історію коммітів використовуючи одностроковий формат. Вносить зміни у файл, виконує комміт. Переглядає історію, використовуючи різні формати контролю відображення записів.

1. Створюємо каталог з майбутнім проектом та переходимо у нього:



1. Створюємо файл «.gitignore»:



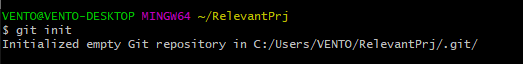
1. Відкриваємо файл «.gitignore»:



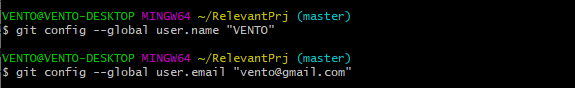
1. Описуємо ігноровані файли у файлі «.gitignore» та зберігаємо файл:



1. Створюємо репозиторій для проекту:

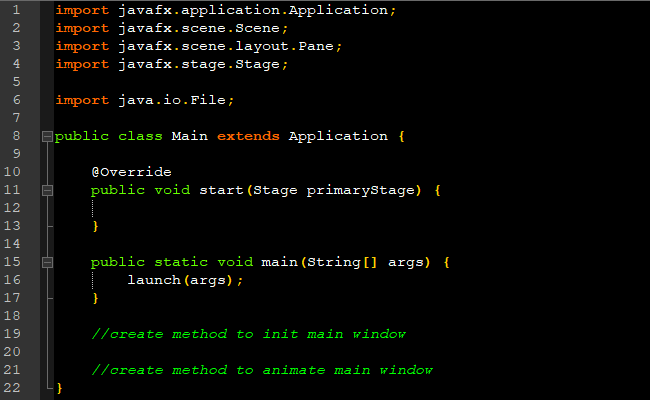


1. Користувач вводить свої дані (ім’я та email):



1. Користувач створює файл із завданнями:

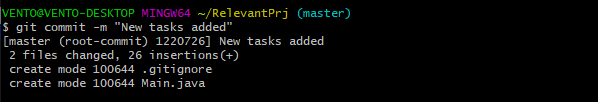




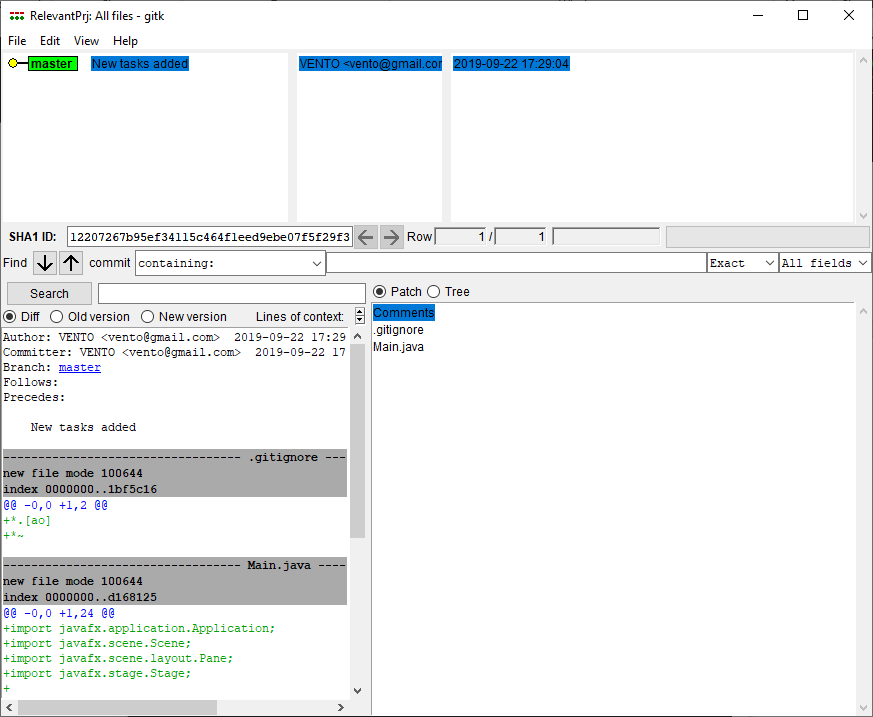
1. Користувач додає усі файли до репозиторію:



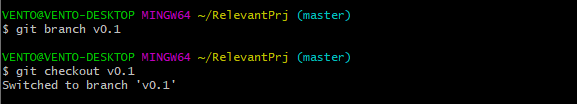
1. Користувач виконує коміт:



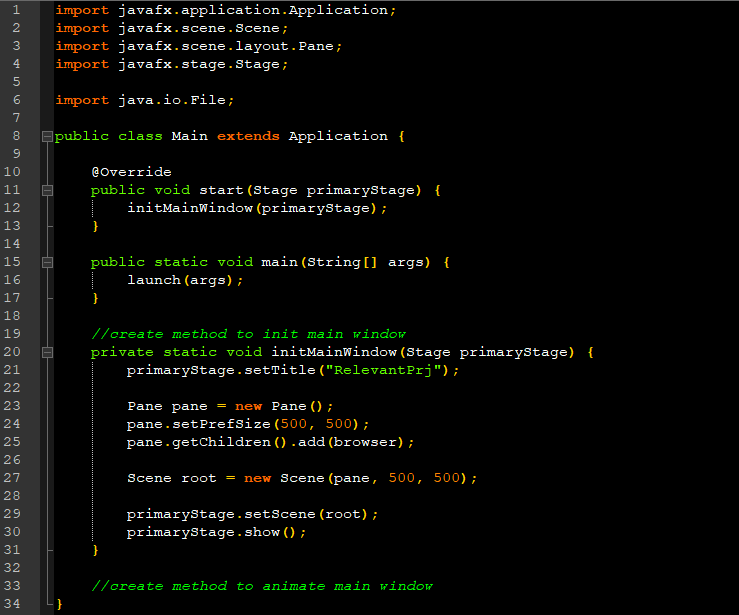
При виконанні команди «gitk» отримуємо:



1. Користувач створює нову гілку та переходить у неї:



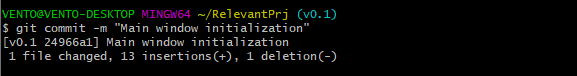
1. Користувач виконує перше завдання:



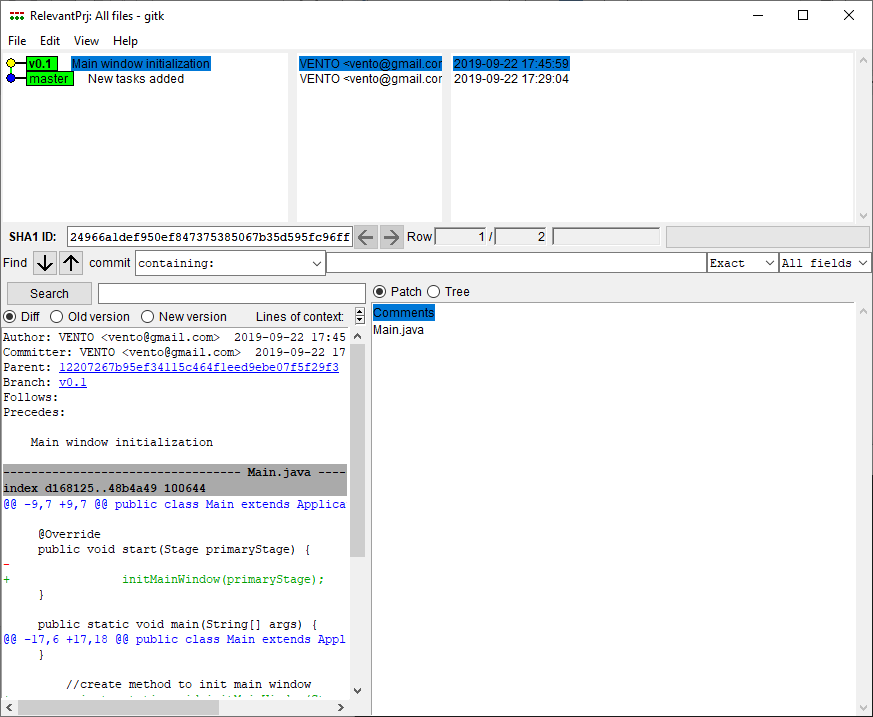
1. Користувач додає усі файли до репозиторію:



1. Користувач виконує коміт:



При виконанні команди «gitk» отримуємо:



1. Користувач повертається у гілку майстра:



1. Користувач переглядає історію коммітів використовуючи одностроковий формат:



Також, можна завдати свій власний формат та виводити тільки потрібні дані:

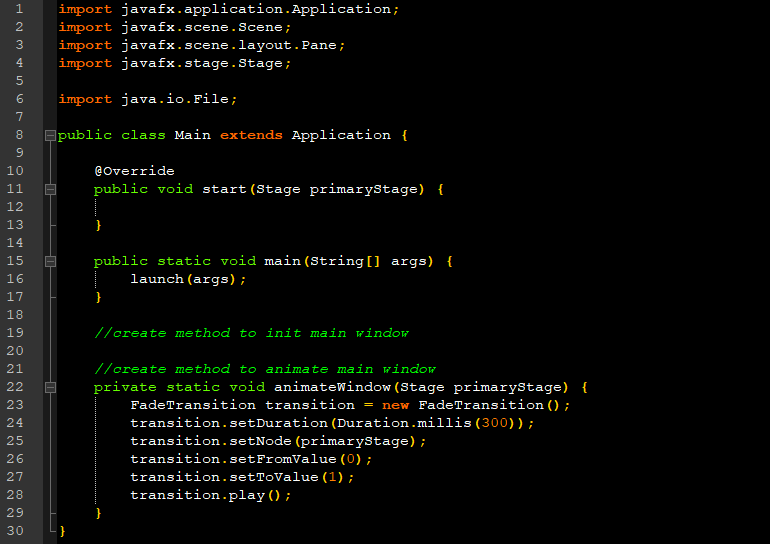


Список всіх параметрів формату наведено у таблиці 1.1.

Таблиця 1.1 – Параметри формату для команди «git log»

|  |  |
| --- | --- |
| **Параметр** | **Опис даних, що виводяться** |
| %H | Хеш коміта |
| %h | Скорочений хеш коміта |
| %T | Хеш дерева |
| %t | Скорочений хеш дерева |
| %P | Хеші батьківських комітів |
| %p | Скорочені хеші батьківських комітів |
| %an | Ім'я автора |
| %ae | Електронна пошта автора |
| %ad | Дата автора |
| %ar | Дата автора, відносна |
| %cn | Ім'я комітера |
| %ce | Електронна пошта комітера |
| %cd | Дата комітера |
| %cr | Дата комітера, відносна |
| %s | Коментар |

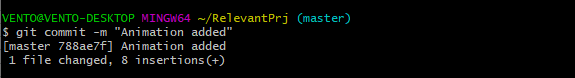
1. Користувач вносить зміни до файлу:



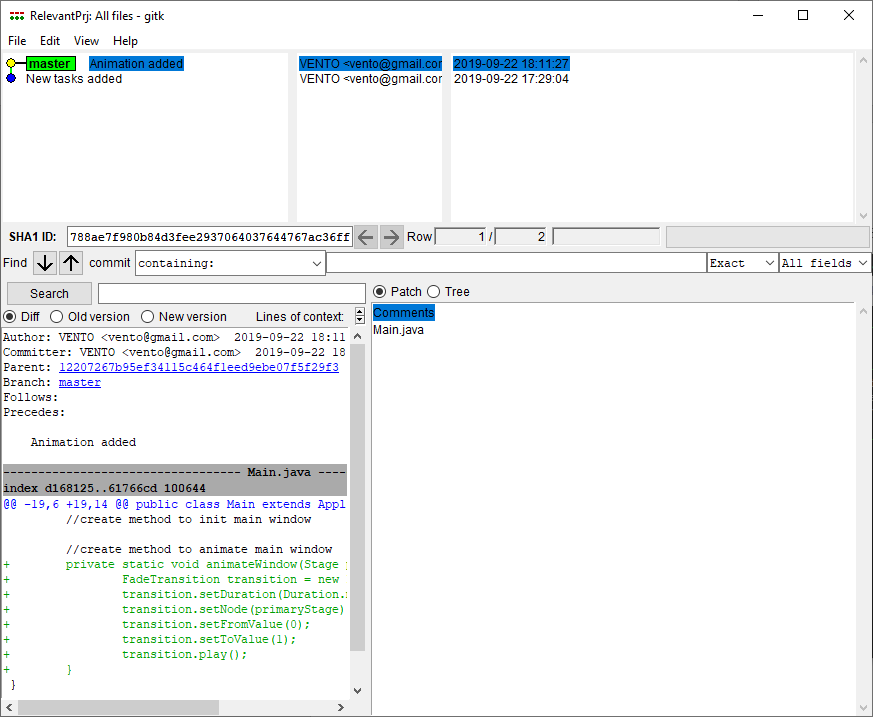
1. Користувач додає усі файли до репозиторію:



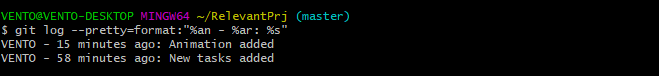
1. Користувач виконує коміт:



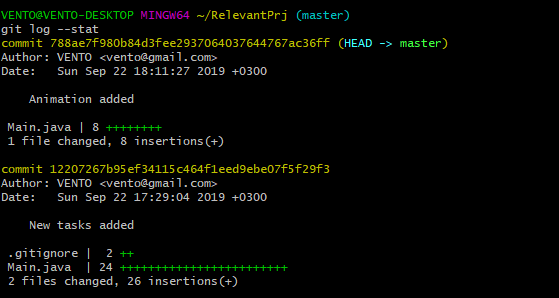
При виконанні команди «gitk» отримуємо:



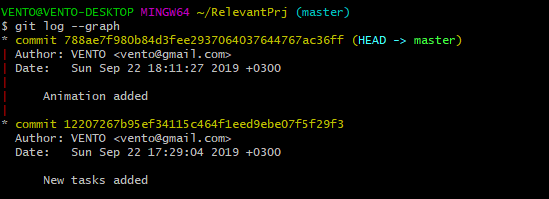
1. Користувач переглядає історію, використовуючи різні формати контролю відображення записів:
   1. Використовуючи параметр «--pretty=format:"…"» для створення власного формату:



* 1. Використовуючи параметр «--stat» для виведення під кожним комітом список змінених файлів, кількість змінених файлів, а також кількість доданих та видалених рядків у цих файлах:

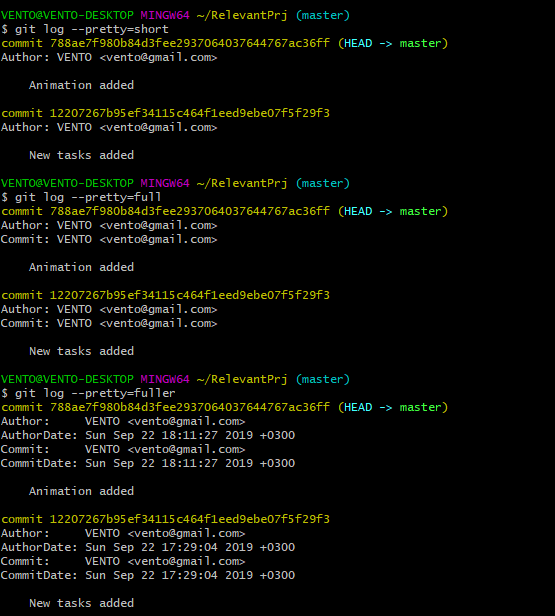


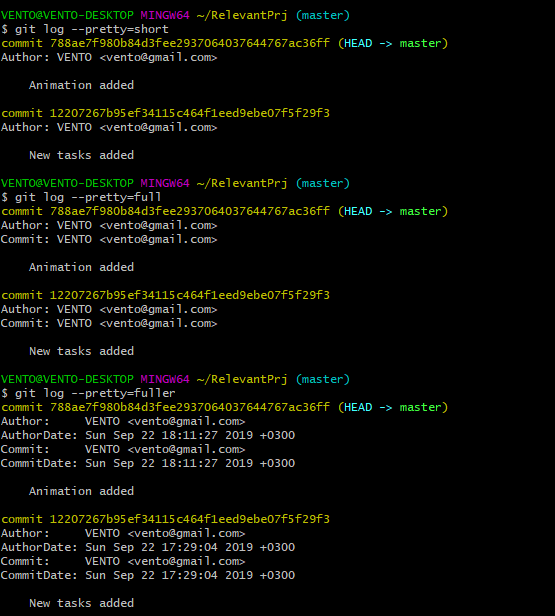
* 1. Використовуючи параметр «--graph» для виведення ASCII-графу історії розгалужень і злиттів поруч з висновком логу.

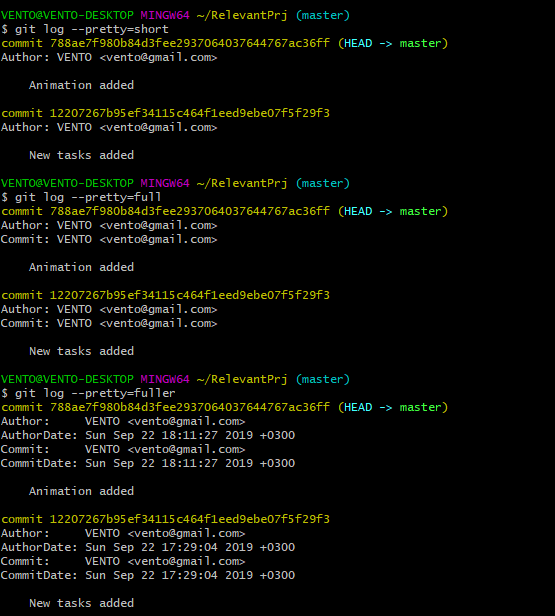


* 1. Використовуючи параметри «--pretty=short», «--pretty=full» та

«--pretty=fuller», які практично не змінюють формат виведення, але дозволяють виводити менше або більше деталей відповідно.







**Висновок:** в ході виконання лабораторної роботи вивчив основні команди по роботі з локальним репозиторієм GIT, а саме команди для створення репозиторію, внесення власних даних, додавання, переміщення та видалення файлів з репозиторію, створення гілок, переміщення між ними та їх об’єднання, створення коміту, а також переглядати історію комітів, використовуючи різний формат виведення.