Министерство науки и высшего образования РФ

федеральное государственное бюджетное образовательное

учреждение высшего образования

«Алтайский государственный технический университет им. И.И. Ползунова»

Факультет информационных технологий Кафедра информационных систем в экономике

Отчёт защищён с оценкой

Преподаватель Убметов. С. В

(подпись) (ФИО)

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2024г.

Отчет по лабораторной работе №1

«Работа с системой контроля версий Git»

По дисциплине

«Алгоритмизация и программирование»

ЛР 09.03.03

Студент группы ПИЭ-33 Коршиков И. Д

Преподаватель ассистент, к.т.н Умбетов С. В

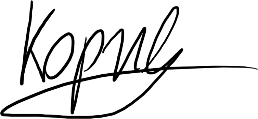
Барнаул 2024

## Цель работы:

Ознакомиться с системой контроля версий GIT и GitHub Desktop.

## Задание:

1. Скачать GIT (консольная версия)
2. Создрать локальный репозиторий
3. Сделать не менее трёх коммитов
4. Создать новую ветку в этом репозитории
5. Создать не менее 2 коммитов в этой ветке
6. Произвести слияние веток
7. Вывести статус коммитов
8. Скачать GitHub Desktop
9. Повторить пункты 2-7 для GitHub Desktop и отправить на удалённый репозиторий



**Задание принял:** Коршиков И.Д

# Ход работы

После скачивания и установке GIT инициализирем репозиторий (git init), добавляем все файлы в каталоге (git add \*), выполняем коммит (git commit). Результаты выполнения этих команд на рисунке 1.

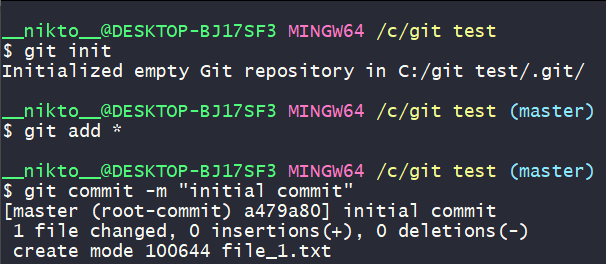


Рисунок 1 – Инициализация репозитория

Создаём 2 коммита. Ход выполнения и результат на рисунке 2.

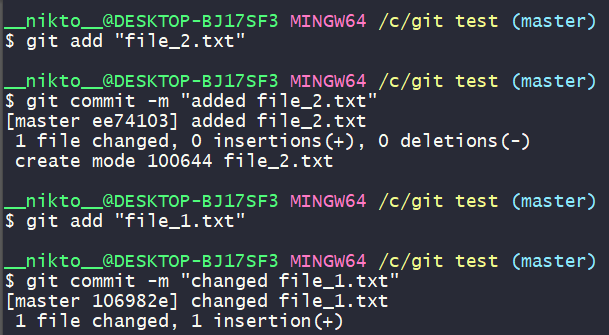


Рисунок 2 – Создание коммитов

На рисунке 3 показано создание новой ветки (git branch) и переключение на неё (git checkout).

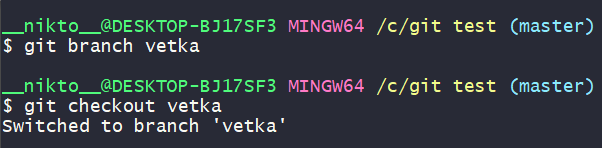


Рисунок 3 – Создание новой ветки

На рисунке 4 показано добавление двух коммитов в новую ветку

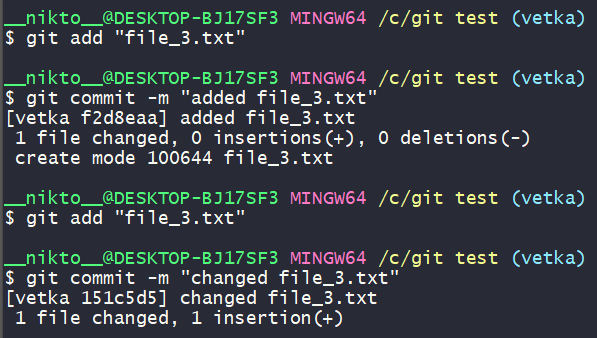


Рисунок 4 – Добавление коммитов в новую ветку

Переключаемся на главную ветку (master), выполняем слияние веток (git merge) и выводим статус коммитов. Этот процесс показан на рисунке 5

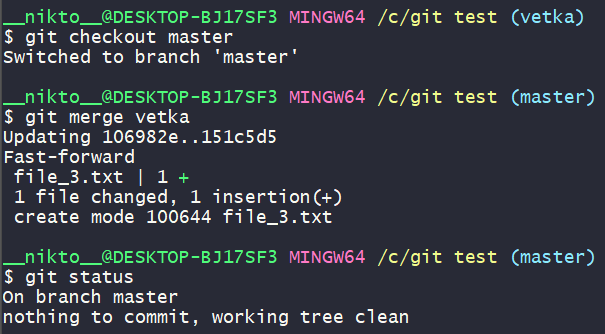


Рисунок 5 – Слияние веток

Используем git log –graph, чтобы вывести древо коммитов. Результат показан на рисунке 6

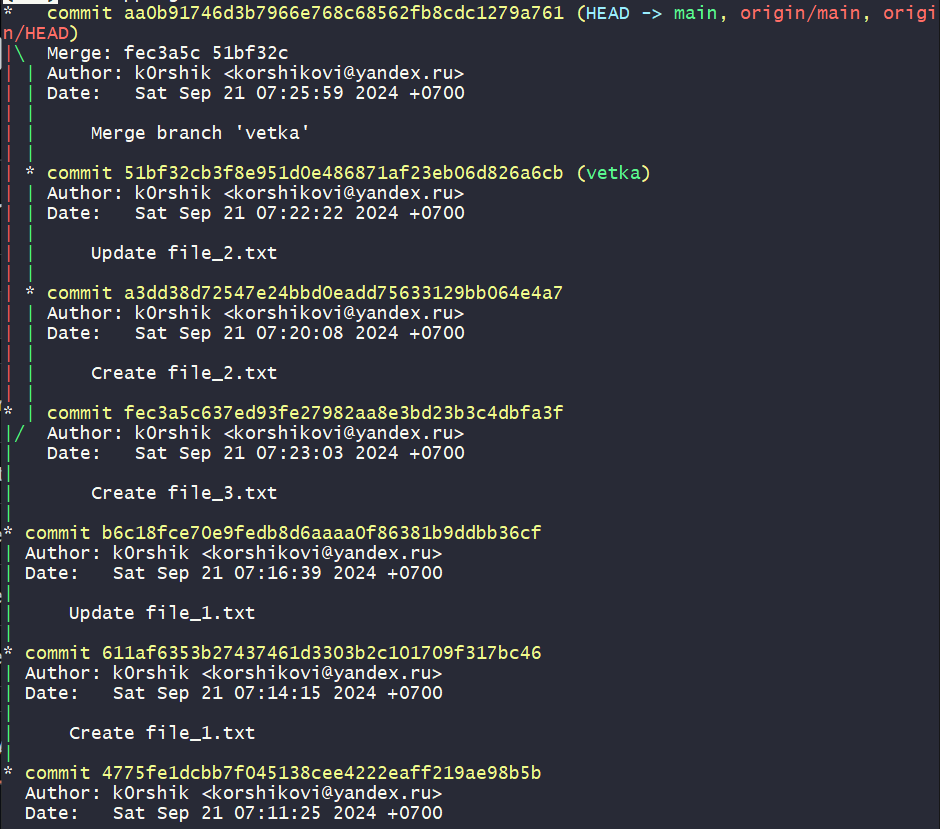


Рисунок 6 – Древо коммитов

После скачивания и установки GitHub Desktop, создаём локальный репозиторий с помощью кнопки показанной на рисунке 7.

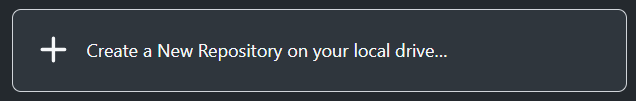


Рисунок 7 – Кнопка создания локального репозитория в GitHub Desktop

Создаём коммиты в GitHub Desktop. На рисунке 8 показана область програмы GitHub Desktop для создания комитов.

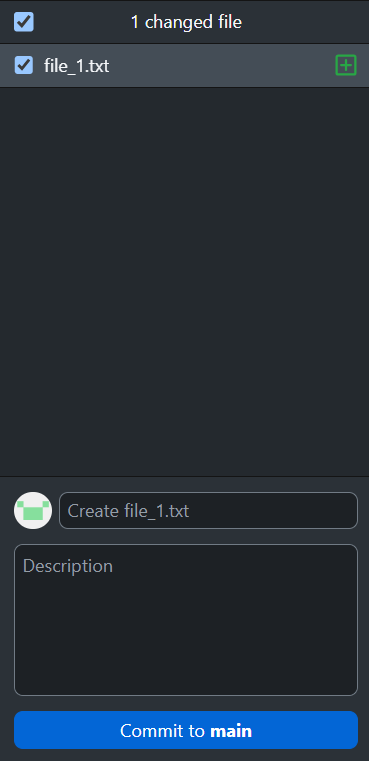


Рисунок 8 – Создание коммита в GitHub Desktop

Создаём новую ветку в GitHub Desktop. На рисунке 9 показано окно, которое всплыает при создании новой ветки в этой программе

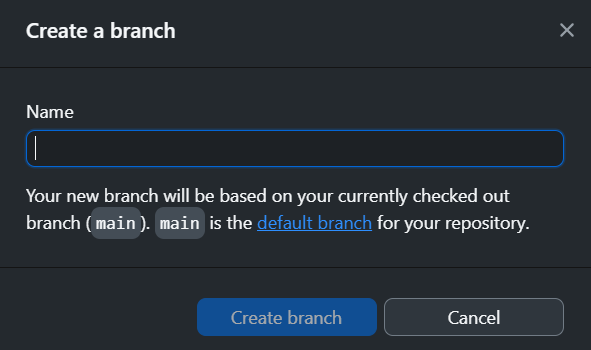


Рисунок 9 – Создание новой ветки в GitHub Desktop

На рисунке 10 показано слияние веток в GitHub Desktop.

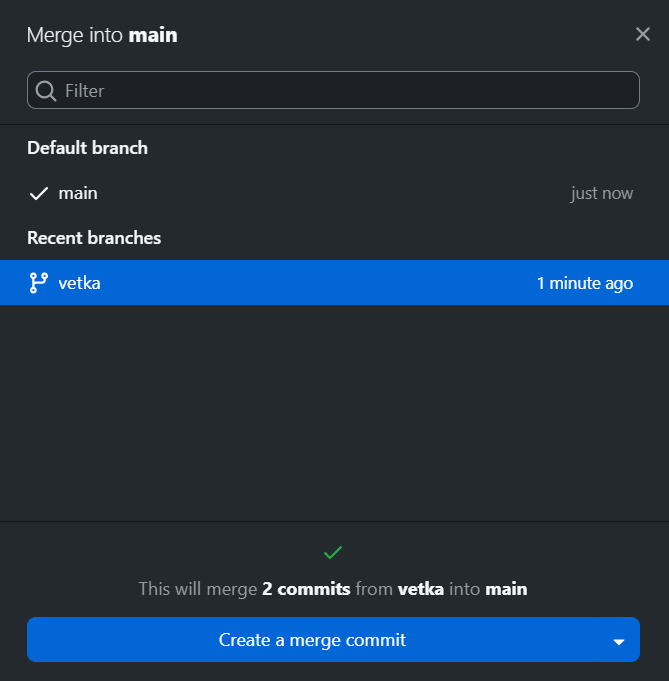


Рисунок 10 – Слияние веток в GitHub Desktop

На рисунке 11 можно увидеть историю коммитов в GitHub Desktop

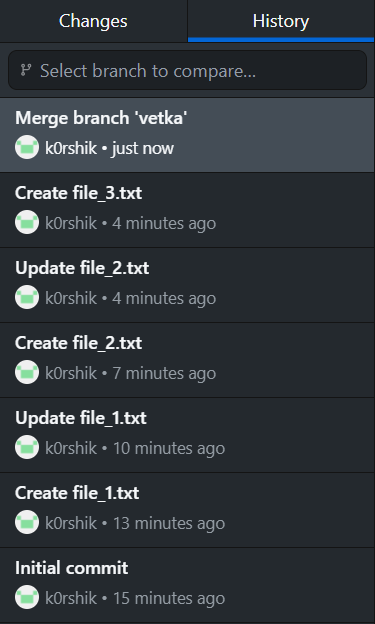


Рисунок 11 – История коммитов

На рисунке 12 показано древо комитов из GitHub



Рисунок 12 – Древо коммитов GitHub

# Вывод

В ходе данной работы я вспомнил некоторые базовые понятия и команды системы контроля версий Git: git init, git add, git commit, git branch, git merge, git status, git log, а также ознакомился с GitHub Desktop.

Возникали сложности со слиянием веток и выводом древа коммитов. На древе не отображалась вторая ветка. Это было связано с тем, что коммиты в ветках производились последовательно.

Так же были сложности с созданием удалённого репозитория. При отправке локального репозитория на удалённый отображался один комит. Поэтому пришлось одельно отправлять каждый комит.

## Ссылка на репозиторий:

<https://github.com/k0rshik/korshikov_pie_33_0.git>