**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3**

**Тема:** Знакомство с системой *GIT*.

**Цель:** Целью работы является выполнение основных видов работ по управлению версиями ПО с использованием системы контроля версий *GIT*.

**6 Вариант:** Использовать в работе 3 разных файла, и количество версий каждого файла не должно быть меньше 3.

**Ход работы**

1. Изучили теоретическую часть.
2. Установили GIT (рисунок 1).

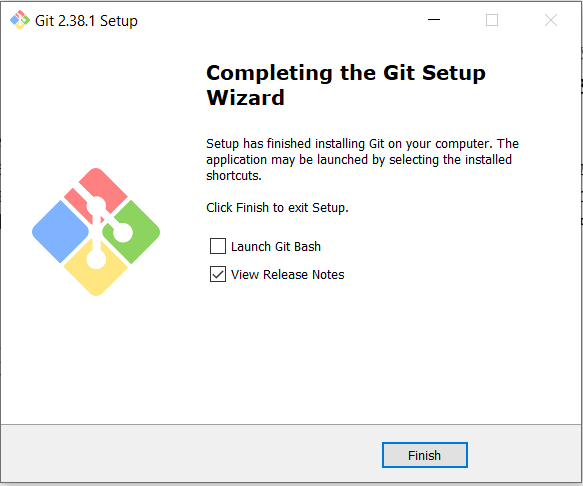


Рисунок 1 – Завершение установки

1. Задали имя и емайл. Выбрали каталог, в котором будут хранится данные об изменениях версий проекта (рисунок 2).

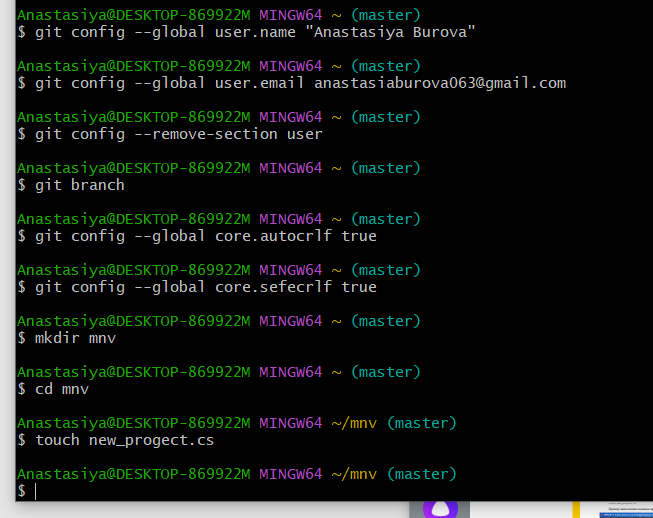


Рисунок 2 – Работа в *Git*

1. Создали 3 файла разных форматов (рисунок 3).

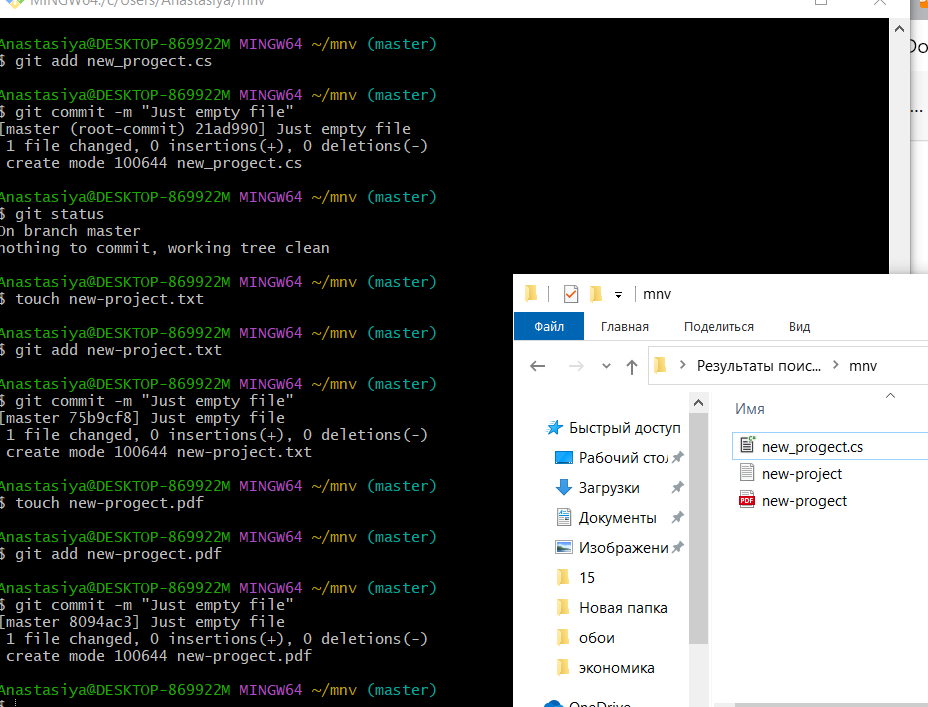


Рисунок 3 – Новые файлы

1. Посмотрели историю проекта (рисунок 4).

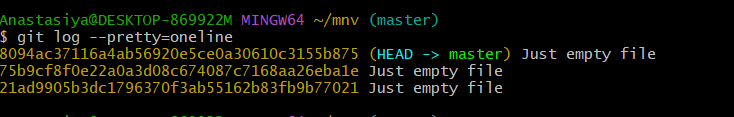


Рисунок 4 – История

1. Создали еще пару файлов. Посмотрели слияние веток (рисунок 5).

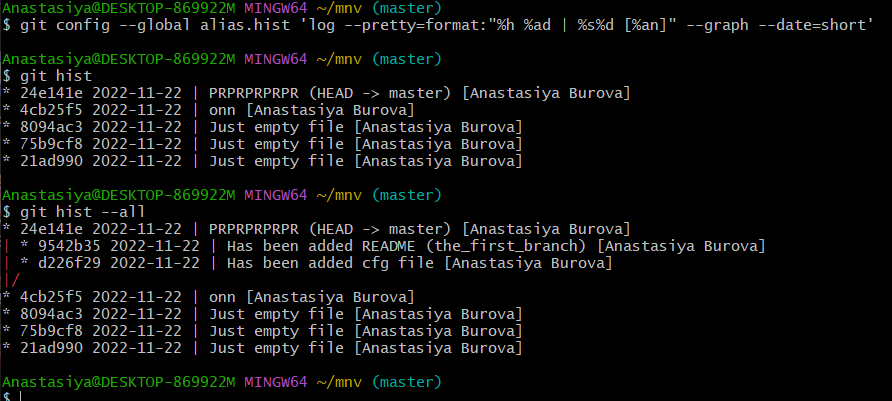


Рисунок 5 - Слияние веток

1. Создание чистого репозитория (рисунок 6).

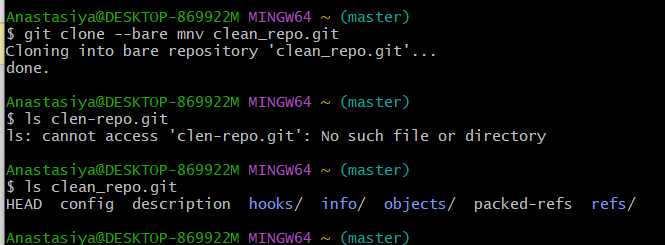


Рисунок 6 – Чистый репозиторий

**Ответы на контрольные вопросы**

1. Что представляет собой система GIT?

Это ПО, позволяющее отслеживать изменения в документах, при необходимости производить их откат, определять, кто и когда внес исправления и т.п.

2. Перечислите возможности систем управления версиями.

Система управления версиями позволяет хранить несколько версий одного и того же документа, при необходимости возвращаться к более ранним версиям, определять, кто и когда сделал то или иное изменение, и многое другое.

3. Что такое коммит и когда он выполняется?

Коммит представляет из себя слепок внесенного изменения, имеющий собственный кэш. Вызывается он командой git commit –m ”Текст сделанного изменения”

4. Как посмотреть историю коммитов?

Git hist

git log

5. Как создать пустой GIT репозиторий?

Зайти в папку нашего будущего проекта, нажать правую кнопку мыши и нажать Git Bash Here, затем ввести команду в консоль git init, у нас создастся пустой репозиторий, который мы будем заполнять нашим проектом

6. Какую архитектуру имеет система контроля версий GIT?

-Хранение данных

-Отслеживание изменений содержимого (хранение истории изменений, включающей метаданные слияния)

-Распространение данных и истории изменений между разработчиками

7. Как создать новую ветку в GIT и соединить ветки?

Новая ветка создаётся командой

Git checkout –b <название ветки> -создание

Git merge <название ветки> -слияние

8. Поясните понятие «staging area».

Файл, хранящийся в каталоге GIT, который содержит информацию о том, что должно войти в следующий коммит

9. Как сравнить версии файла в GIT?

Git diff --cached