Отчёт по лабораторной работе №1

Система контроля версий Git

Агеева Лада НПИбд-01-19

Содержание

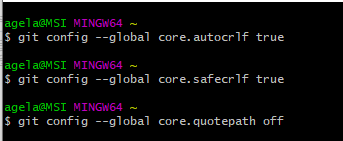
# Цель работы

Целью данной работы является изучение и освоение системы контроля версий Git

# Выполнение лабораторной работы

## Подготовка

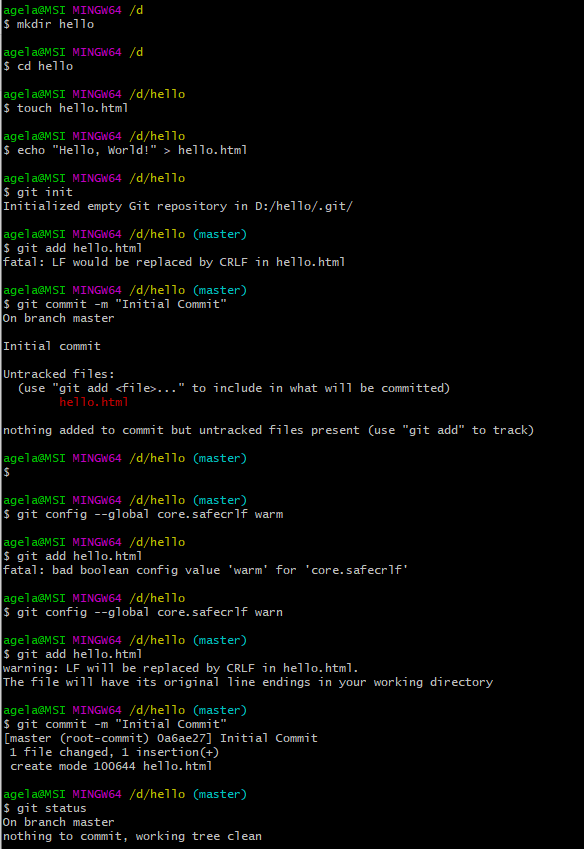
Изменение параметров установки окончаний строк и Установка отображения unicode



Подготовка

## Создаем локальный проект

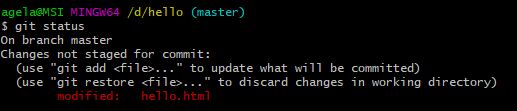
Создаем страницу «Hello, World» Созданем репозиторий Добавляем файл в репозиторий Проверяем состояние репозитория



Создание проекта

## Внесение изменений

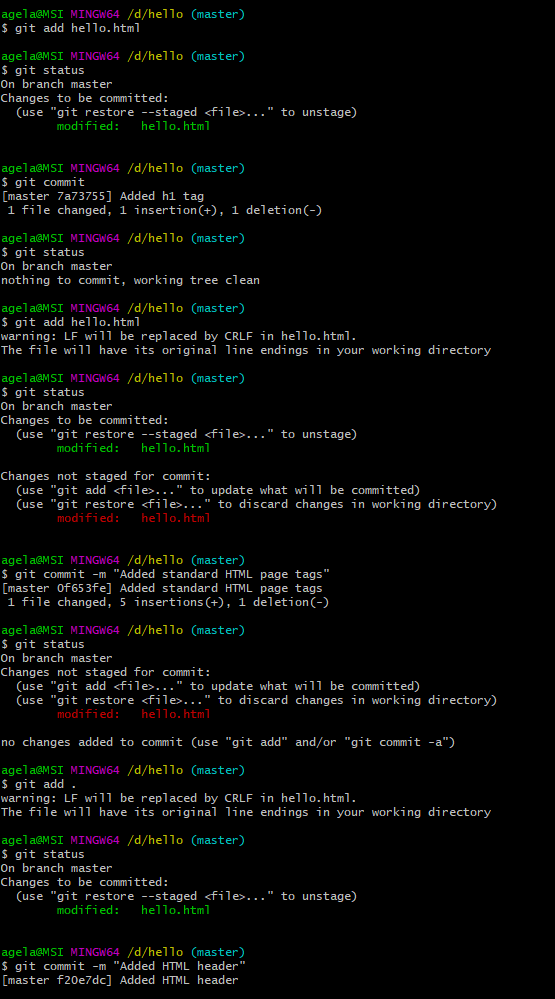
Изменяем страницу «Hello, World»



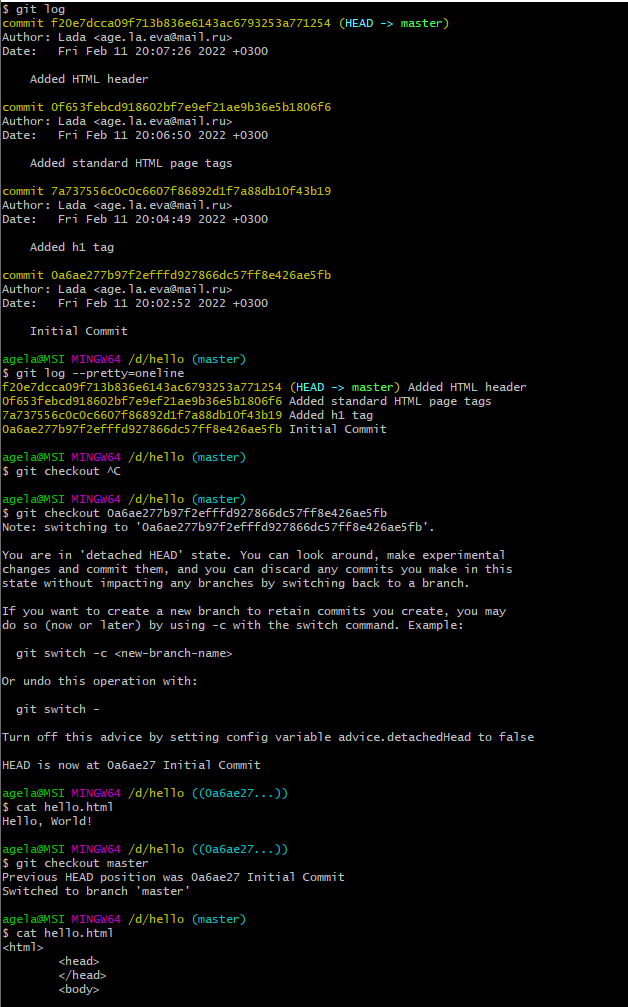
Внесение изменений

## Индексация изменений

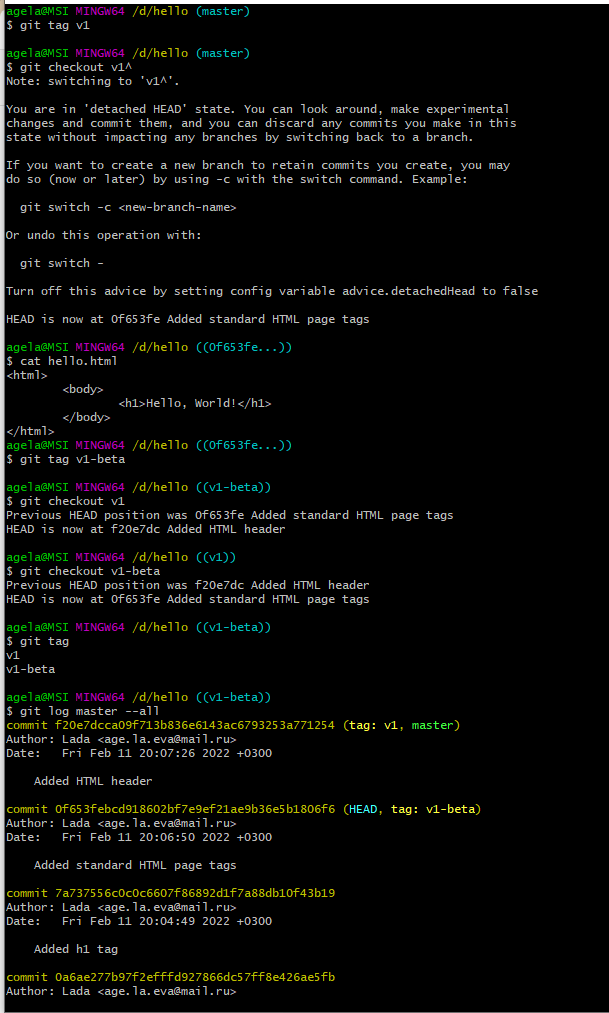
Коммитим изменения Добавляем стандартные теги страницы Проверяем список произведенных изменений Получяем старые версии Создаем теги версий Переключаемся по имени тега Просмотриваем теги с помощью команды tag



Индексация изменений 1



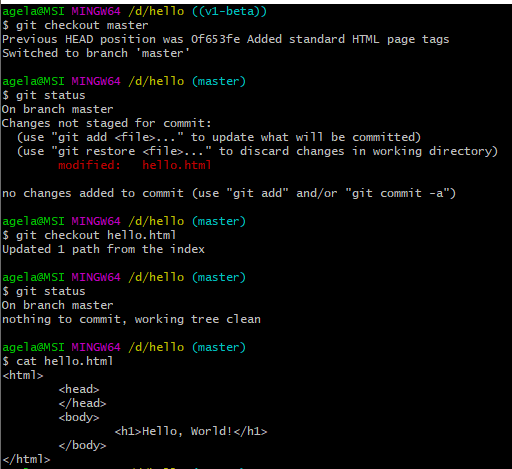
Индексация изменений 2



Индексация изменений 3

## Отмена локальных изменений (до индексации)

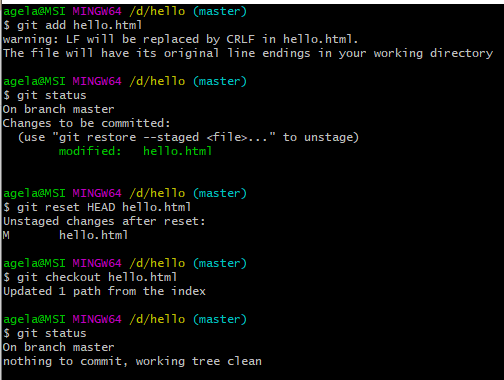
Переключаемся на ветку master Изменяем hello.html Проверяем состояние Отменяем изменения в рабочем каталоге



Отмена локальных изменений

## Отмена проиндексированных изменений (перед коммитом)

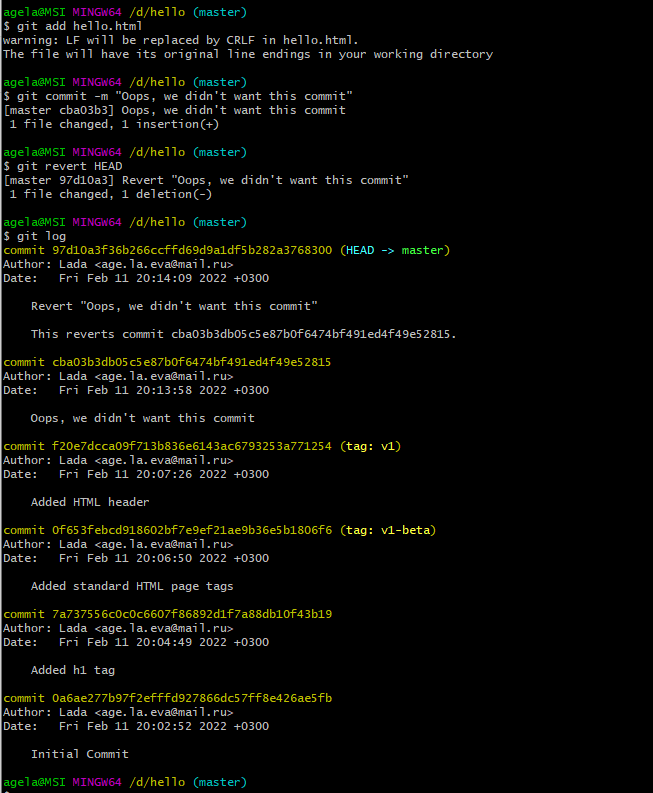
Изменяем файл и проиндексируем изменения Проверяем состояние Выполняем сброс буферной зоны Переключяемся на версию коммита



Отмена проиндексированных изменений

## Отмена коммитов

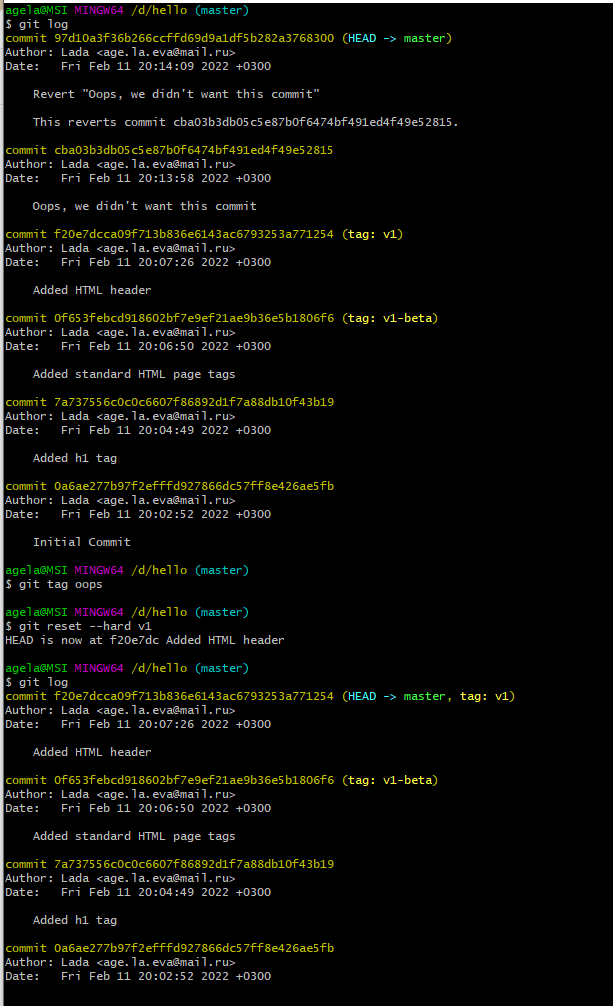
Изменяем файл и сделаем коммит Сделаем коммит с новыми изменениями, отменяющими предыдущие Проверим лог



Отмена коммитов

## Удаление коммиттов из ветки

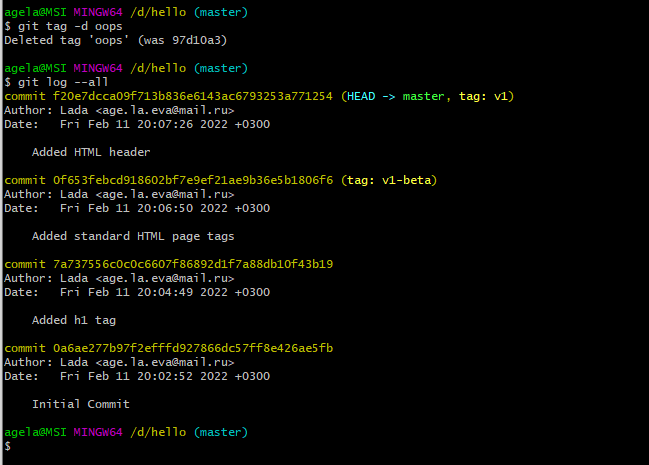
Проверяем нашу историю Отмечаем эту ветку Сбрасываем коммиты к предшествующим коммиту Oops



Удаление коммиттов из ветки

## Удаление тега oops

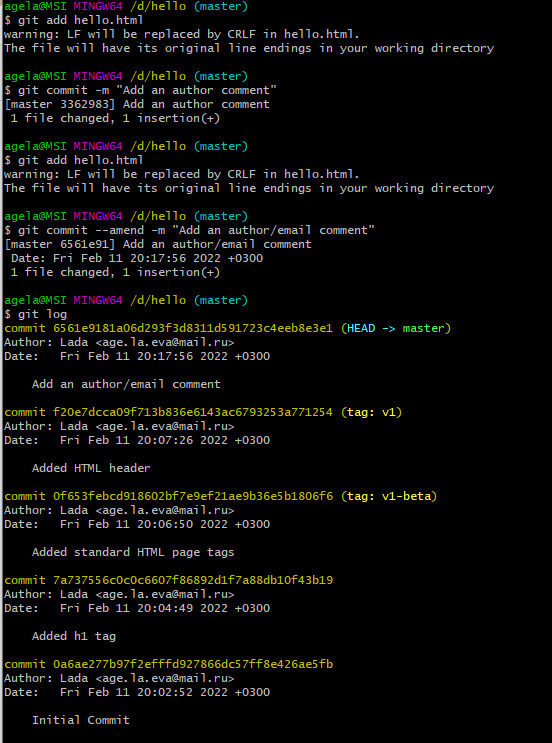
Удаляем тег oops



Удаление тега oops

## Внесение изменений в коммиты

Изменяем страницу, а затем сделаем коммит Добавляем email Изменяем предыдущий коммит Просматриваем историю



Внесение изменений в коммиты

## Перемещение файлов

Перемещаем файл hello.html в каталог lib Делаем коммит этого перемещения



Перемещение файлов

## Добавление index.html

Добавляем файл index.html в наш репозиторий

## Git внутри: Каталог .git

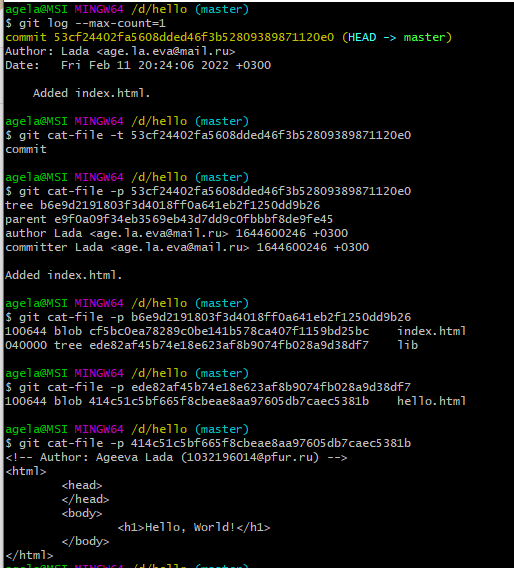
Углубимся в базу данных объектов



Git внутри: Каталог .git

## Работа непосредственно с объектами git

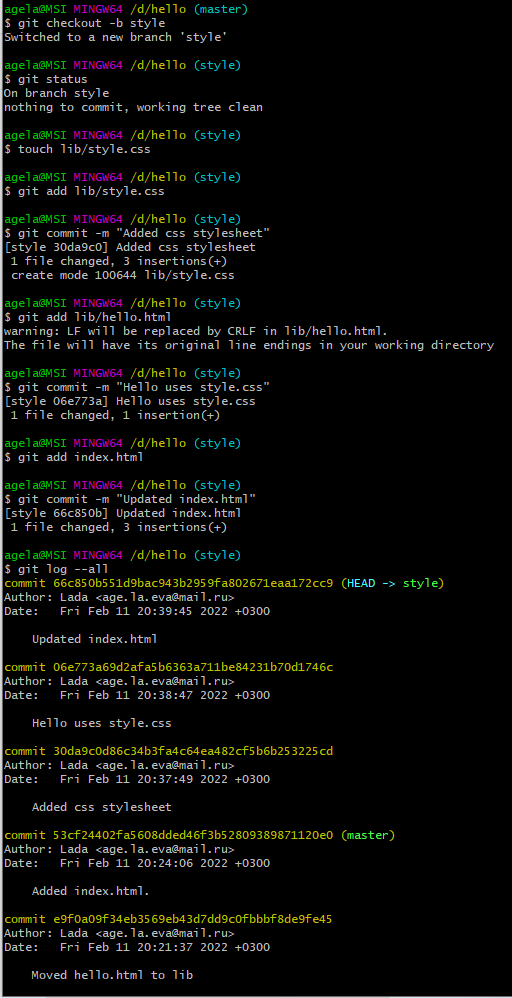
Найдем последний коммит Выводем последний коммит с помощью SHA1 хэша Выводем дерево каталогов Выводем каталог lib Выводем файл hello.html



Работа непосредственно с объектами git

## Создание ветки

Создаем ветку Добавляем файл стилей style.css Изменяем основную страницу Изменяем index.html



Создание ветки

## Навигация по веткам

Переключимся на ветку master Вернемся к ветке style



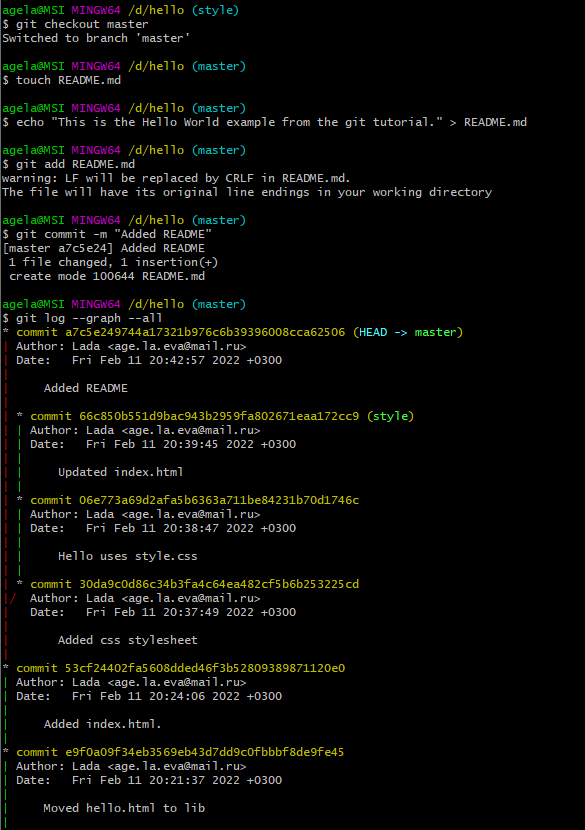
Навигация по веткам

## Изменения в ветке master

Создаем файл README в ветке master

## Сделайте коммит изменений README.md в ветку master.

Просмотрим отличающиеся ветки



Изменения в ветке master

## Слияние

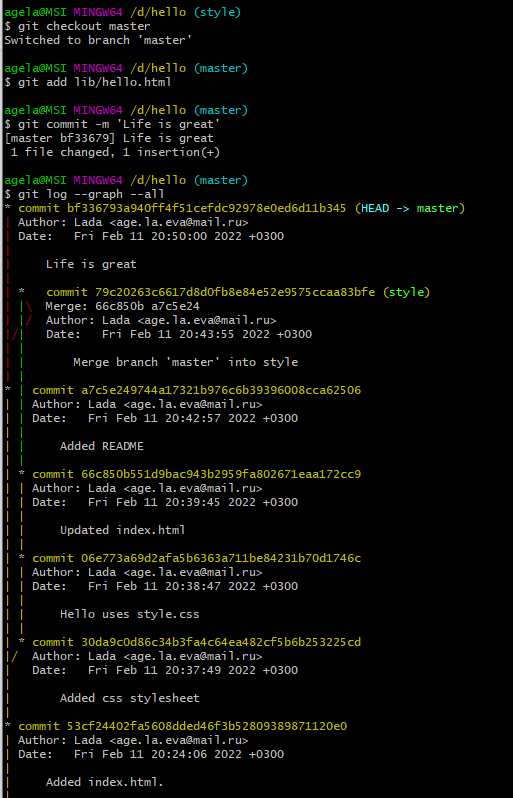
Выполним слияние веток



Слияние

## Создание конфликта

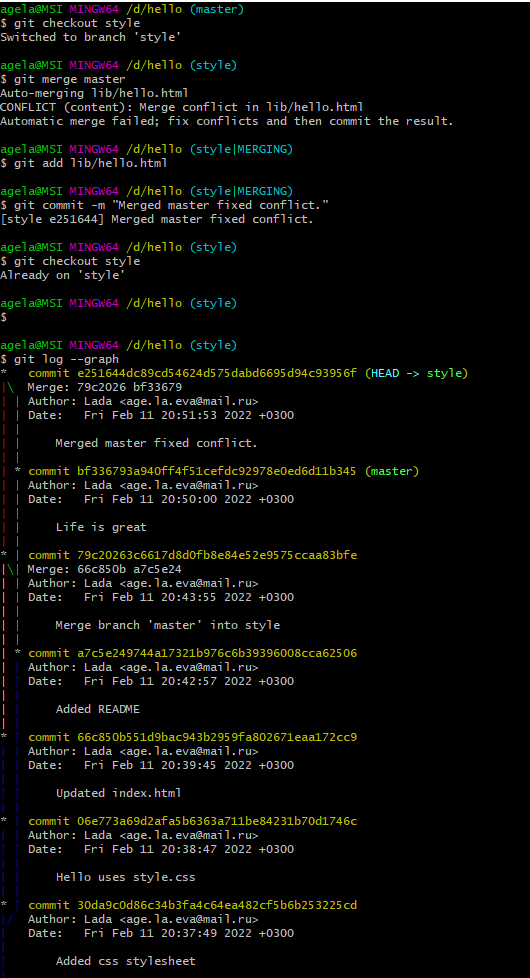
Вернемся в master и создадим конфликт Выполним Просмотр веток



Создание конфликта

## Разрешение конфликтов

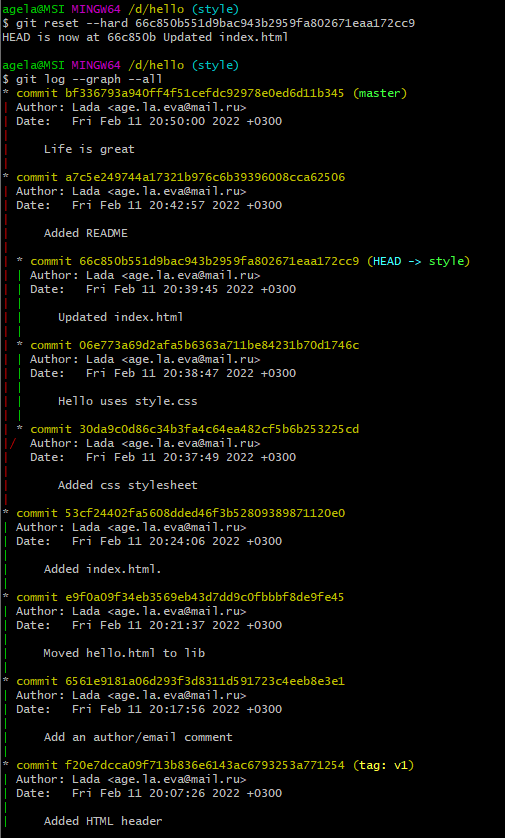
Возвращаемся к ветке style и попытаемся объединить ее с новой веткой master Решаем конфликт Сделаем коммит решения конфликта



Разрешение конфликтов

## Сброс ветки style

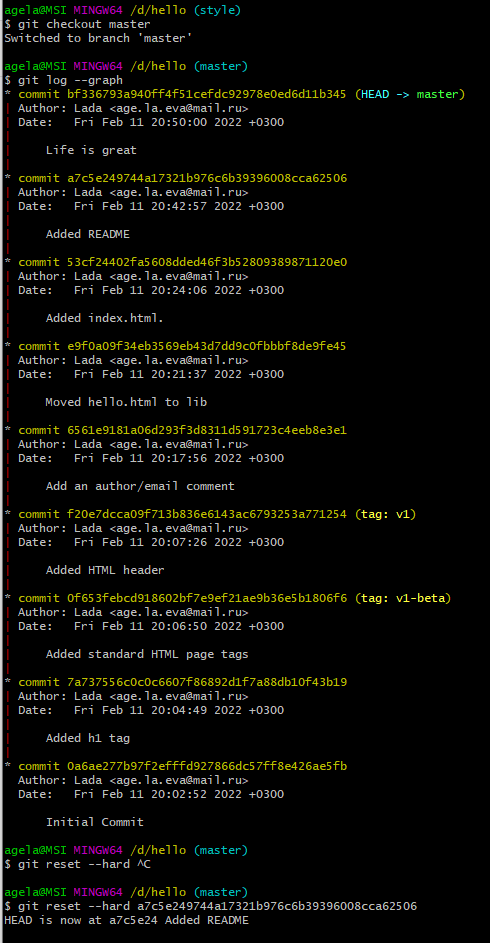
Вернемся на ветке style к точке перед тем, как слили ее с веткой master Сбросим ветку style к коммиту перед слиянием с master



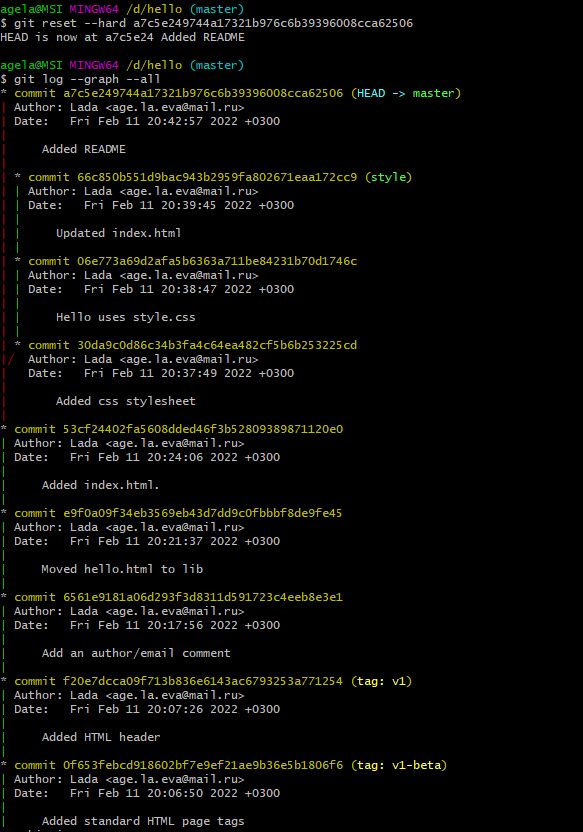
Сброс ветки style

## Сброс ветки master

Сбрасываем ветку master



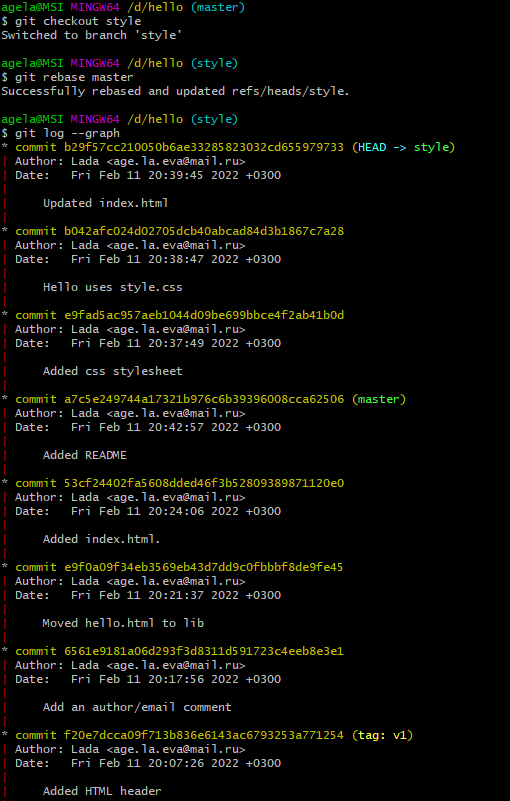
Сброс ветки master 1



Сброс ветки master 2

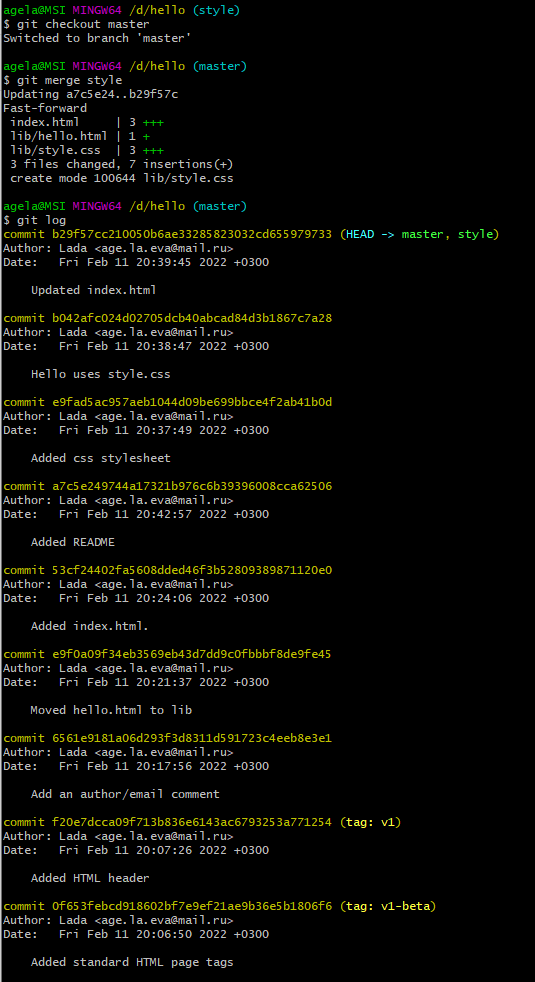
## Перебазирование

Используем команду rebase вместо команды merge



Перебазирование

## Слияние в ветку master



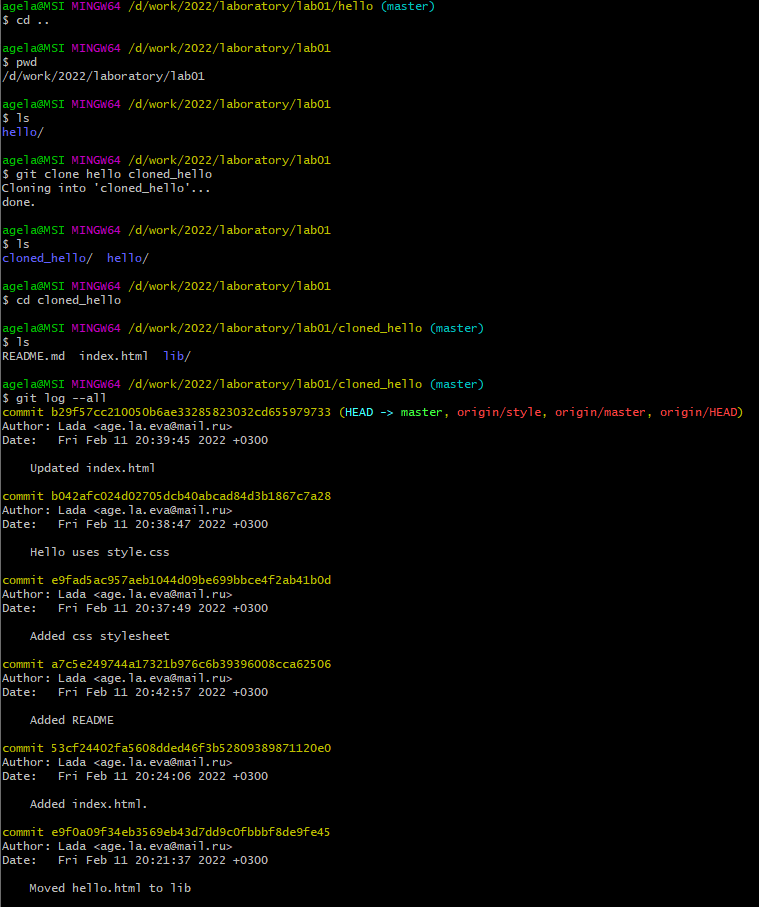
Слияние в ветку master

## Клонирование репозиториев

Перейдем в рабочий каталог Создим клон репозитория hello

## Просмотр клонированного репозитория

Взглянем на клонированный репозиторий Просмотрим историю репозитория



Клонирование репозиториев и Просмотр клонированного репозитория

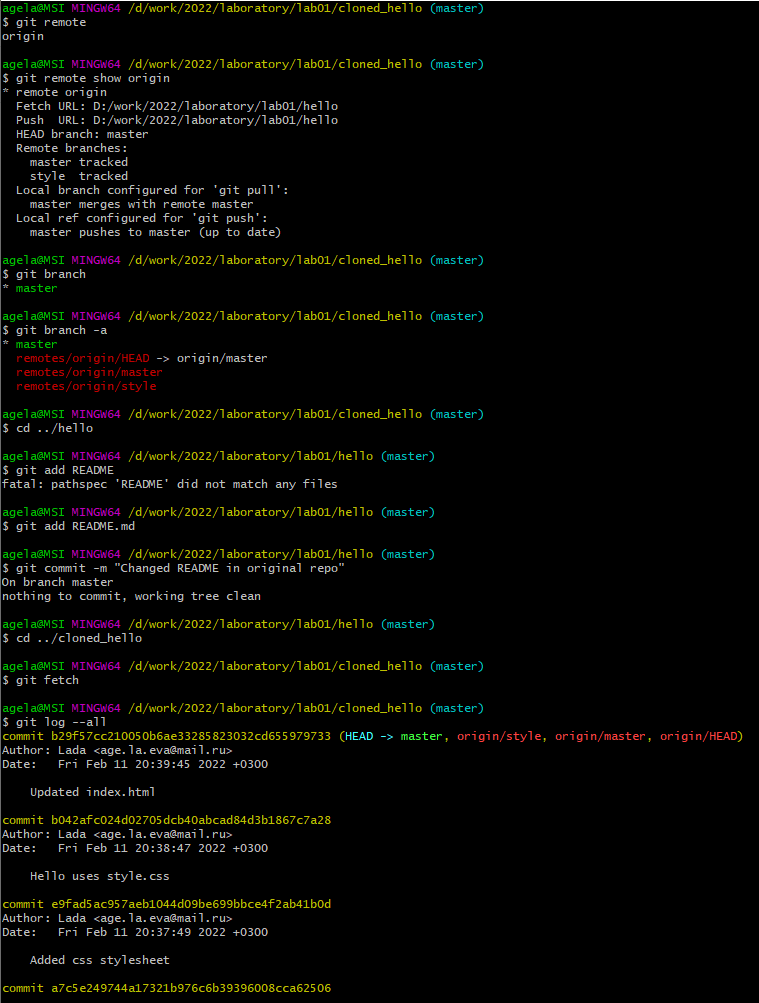
## Что такое origin?

## Удаленные ветки

Выведем список удаленных веток

## Изменение оригинального репозитория

Внесем изменения в оригинальный репозиторий hello Извлечем изменения



origin, Удаленные ветки, Изменение оригинального репозитория

## Слияние извлеченных изменений Сольем извлеченные изменения в локальную ветку master Проверим файл README.md

## Добавление ветки наблюдения

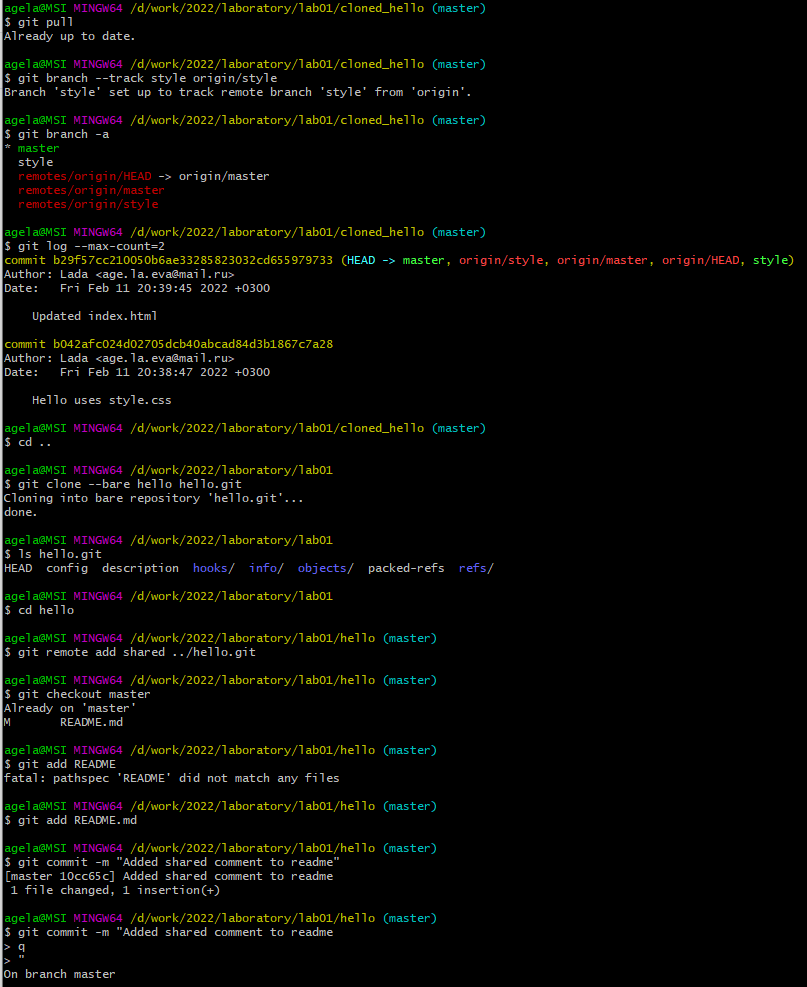
Добавим локальную ветку, которая отслеживает удаленную ветку

## Чистые репозитории

Создадим чистый репозиторий

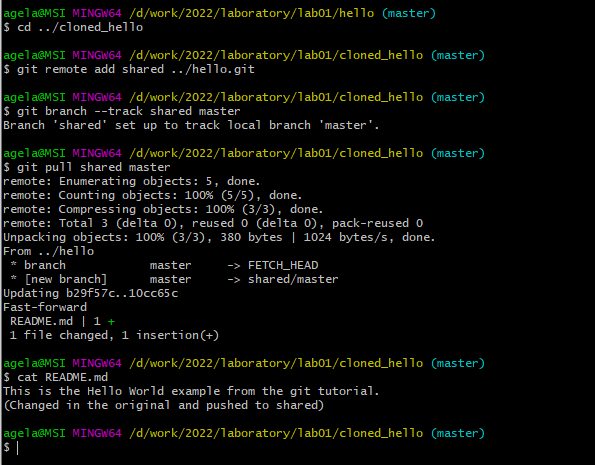
## Добавление удаленного репозитория

## Отправка изменений



Слияние извлеченных изменений, Добавление ветки наблюдения, Добавление чистого репозитория, Отправка изменений

## Извлечение общих изменений



Извлечение общих изменений

# Вывод

В ходе выполнения работы я приобрела практические навыки работы с системой контроля версий git и создала свой репозиторий

# Список литературы

1. [Основы Git](https://habr.com/ru/post/583478/)
2. [Руководство по оформлению Markdown файлов](https://gist.github.com/Jekins/2bf2d0638163f1294637)