**МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Программное обеспечение информационных технологий»

**Практика написания программного кода**

**ЛАБОРАТОРНЫЕ РАБОТЫ.**

Система контроля версий GIT

Выполнил:

**Шведов Н. В.**

Проверил:

**Мищенко И.И.**

Могилев 2025

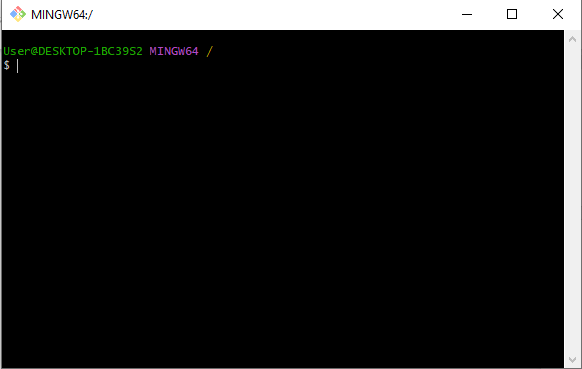
**ЛАБОРАТОРНЫЕ РАБОТЫ.**

Система контроля версий GIT

***Цель работы***: изучить систему контроля версий GIT.

**Задание 1.1**

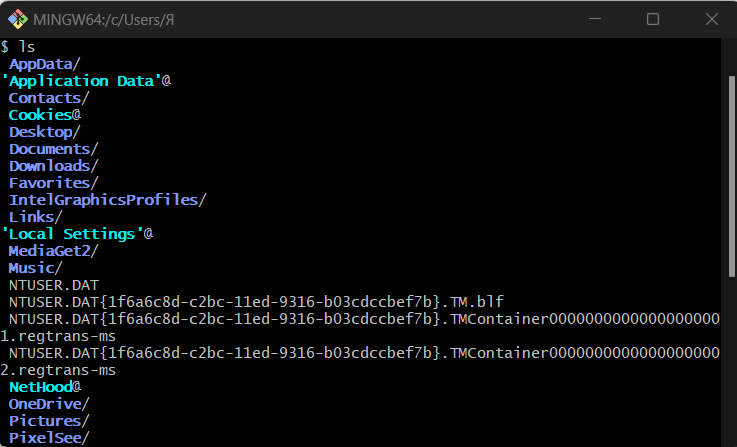
Установка GIT:



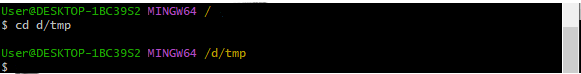
**Задание 1.2**

Создание репозитория.

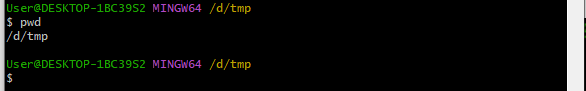
Команда ls для проверки успешной установки Ваsh:



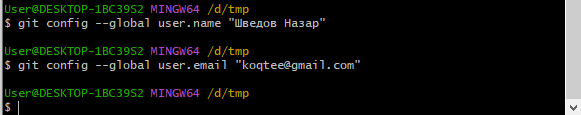
Создание папки и переход в нее:



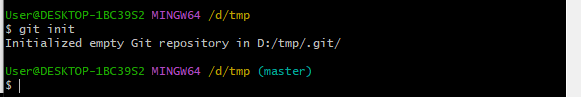
Проверка директории:



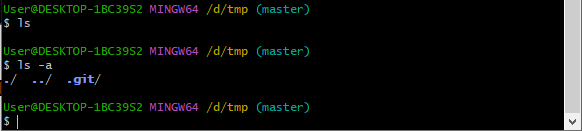
Настройки GIT:



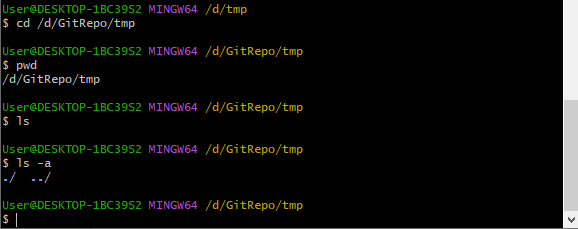
Инициализация репозитория:



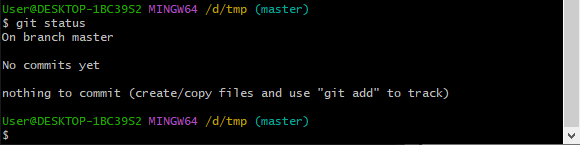
Ввод команды ls -a:



Перенос папки tmp в GitRepo и проверка на наличие скрытых файлов:



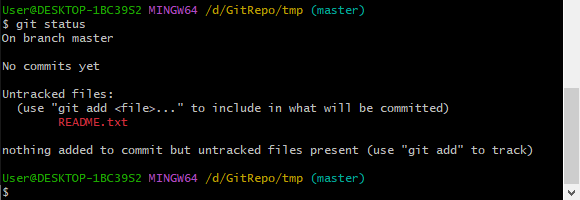
Команда git status:



**Задание 1.3**

Просмотр истории коммитов. Операции отмены.

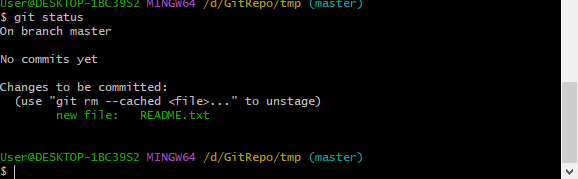
Создание файла README.txt в папке tmp:



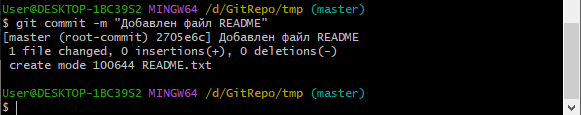
Команда git add:



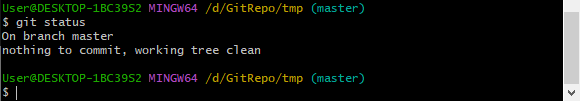
Проверка git status:



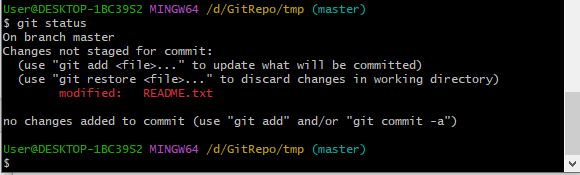
Создание коммита:



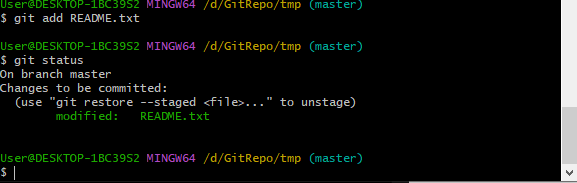
Проверка git status:



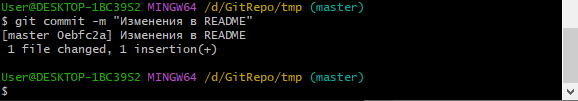
Открытие файла README.txt и сохранение изменений в нем:



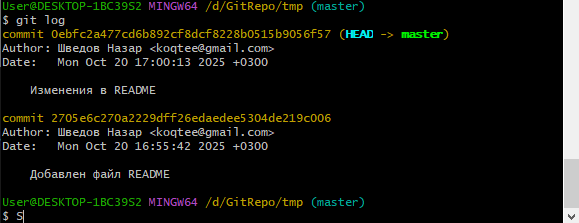
Подготовка изменения к фиксации и проверка текущего статуса:



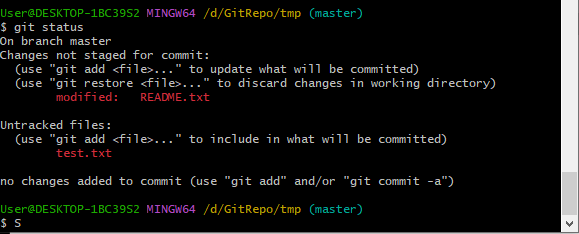
Создание коммита:



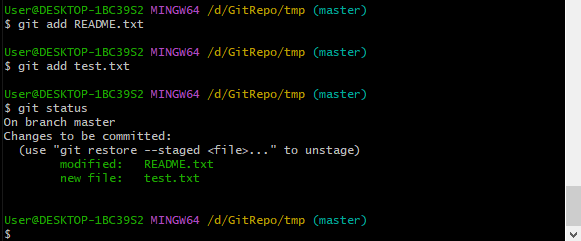
Просмотр истории коммитов:



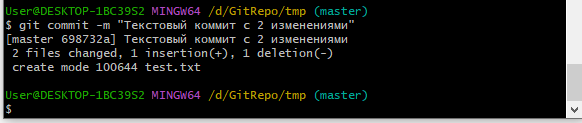
Создание нового файла и добавление изменений в README. Проверка статуса:



Индексируем оба файла:



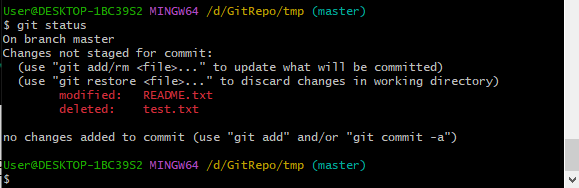
Фиксация двух изменений в одном коммите:



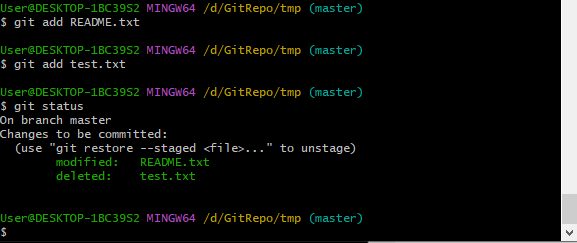
Проверка истории коммитов:



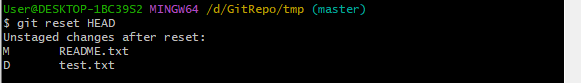
Проверка состояния после изменений в файлах:



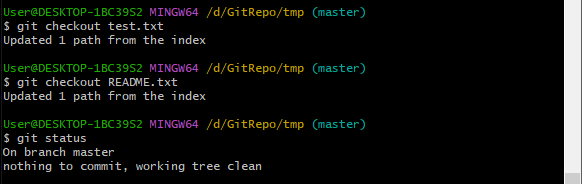
Индексация изменений и проверка статуса:



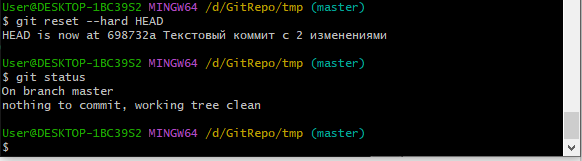
Откат к состоянию до индексации:



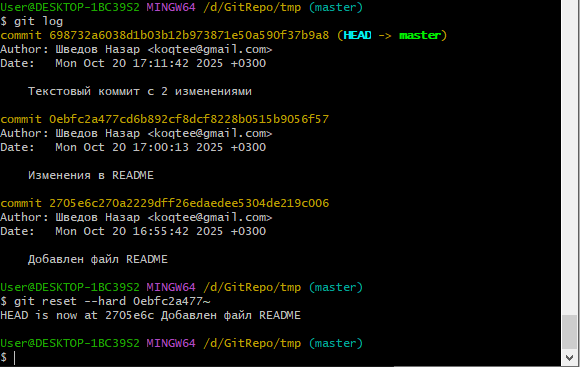
Откат к состоянию до изменений:



Команда git reset –hard HEAD:



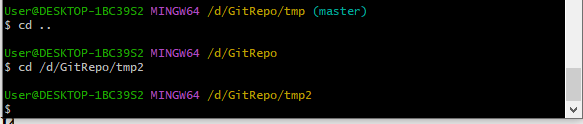
Откат к предпоследнему коммиту:



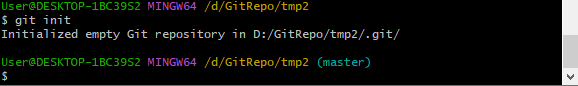
**Задание 1.4**

Работа с удаленными репозиториями.

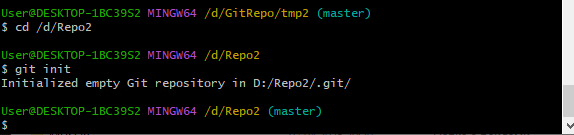
Создание нового пустого репозитория и переход в новую папку:



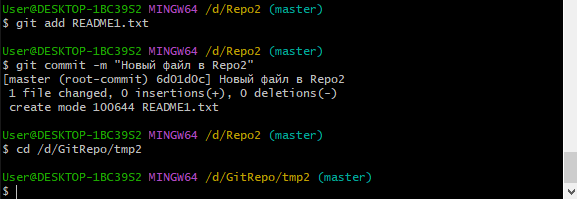
Инициализация нового репозитория:



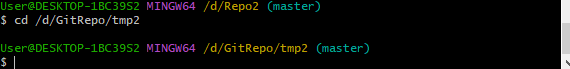
Создание второго репозитория:



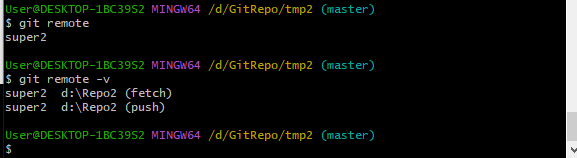
Создание файла README1.txt, создание коммита и переход в первый репозиторий:



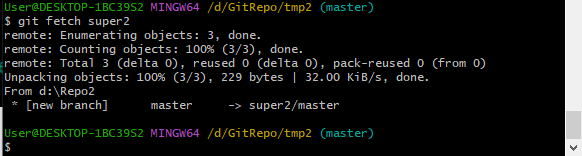
Команда git remote add:



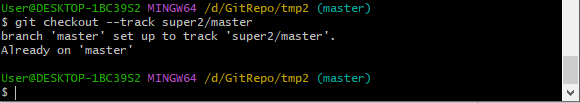
Список прикрепленных удаленных репозиториев:



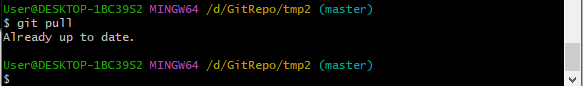
Команда git fetch super2:



Команда git checkout –track super2/master:



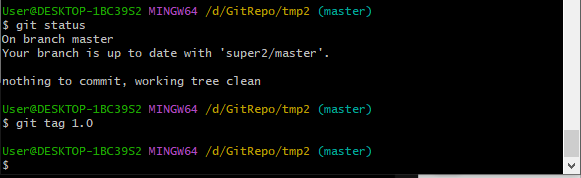
Команда git pull:



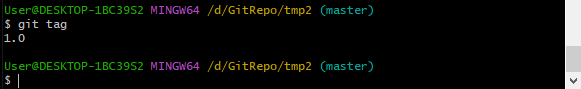
**Задание 1.5**

Работа с метками.

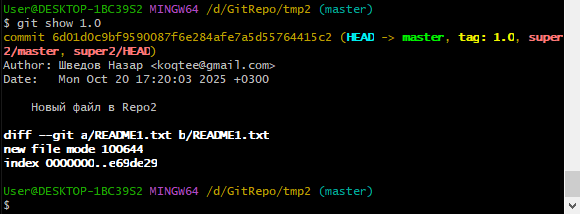
Установка метки на коммит:



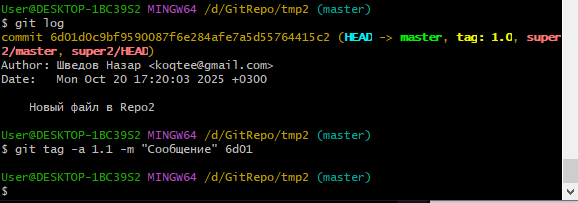
Проверка gittag:



Команда gitshow:

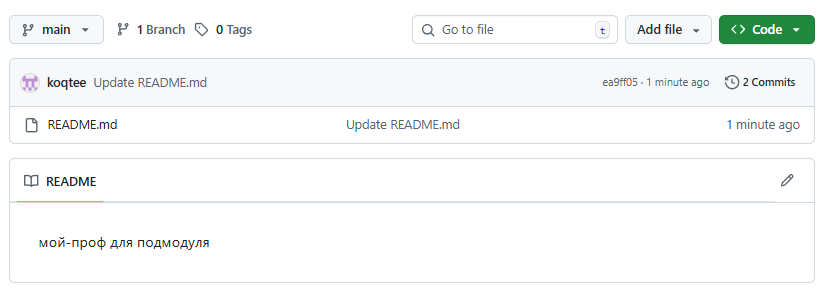


Создание тега для коммита:

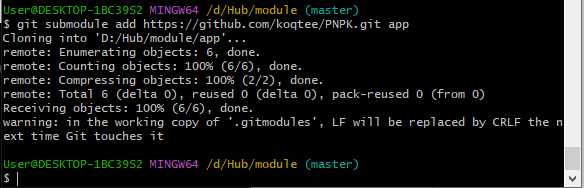


Субмодули.

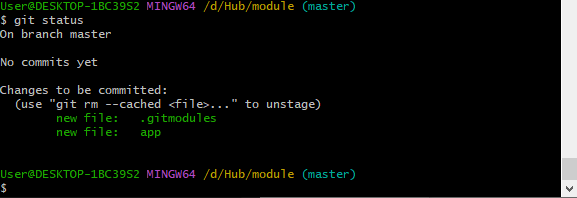
Создание второго репозитория на GitHub:



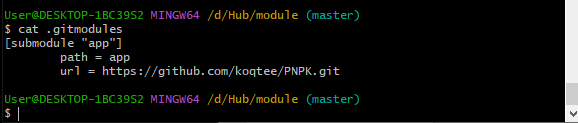
Включение ссылки на наш репозиторий в другой:



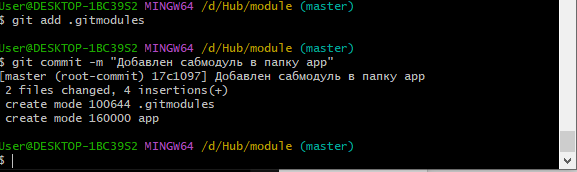
Проверка статуса:



Чтение gitmodules:



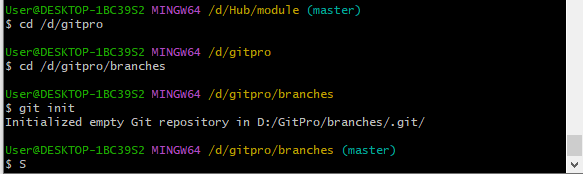
Индексация и коммит:



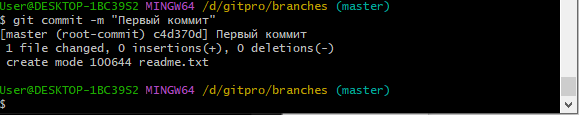
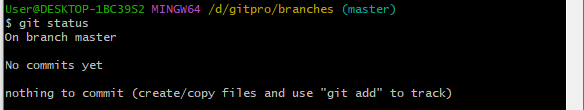
**Задание 1.6**

Работа с ветками.

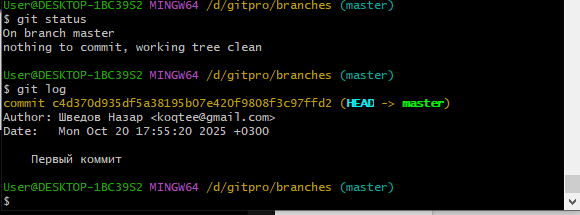
Создание нового пустого репозитория и его инциализация:



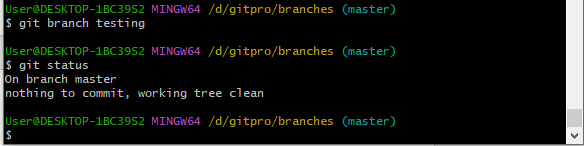
Проверка статуса и создание коммита:



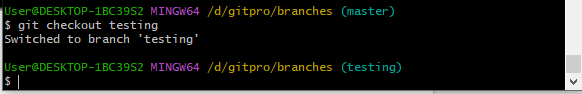
Проверка статуса и вывод списка коммитов:



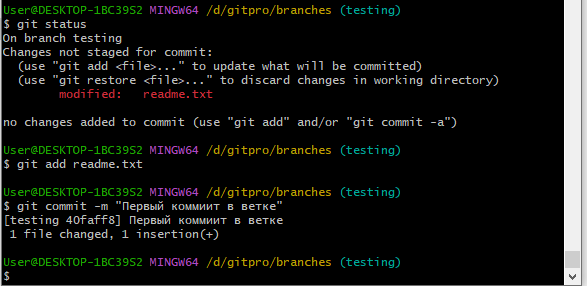
Создание новой ветки и проверка статуса:



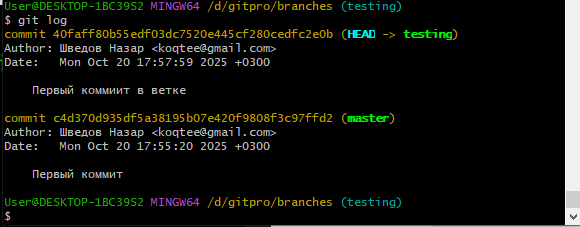
Переход на новую ветку:



Проверка статуса в ветке и создание коммита в ней:



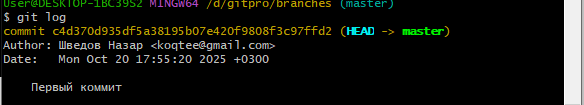
Вывод списка коммитов:



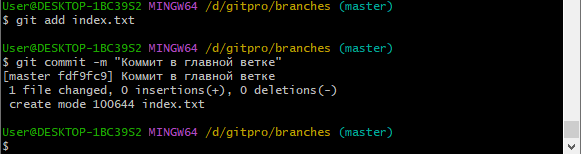
Переход в главную ветку:



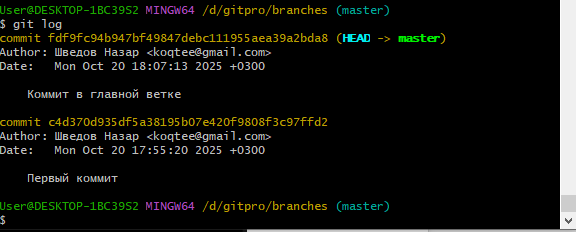
Вывод истории коммитов:



Создание нового файла и его коммит:

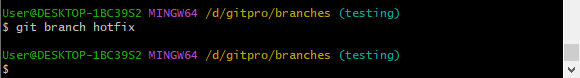


История коммитов в главной ветке и в ветке testing:

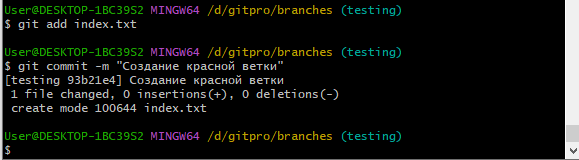




Создание ветки hotfix:



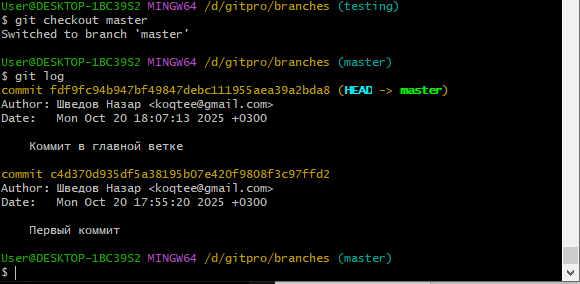
Добавили новый текст в файл index.txt и создали коммит



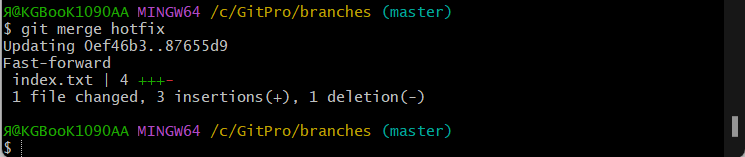
Список коммитов в ветке hotfix:



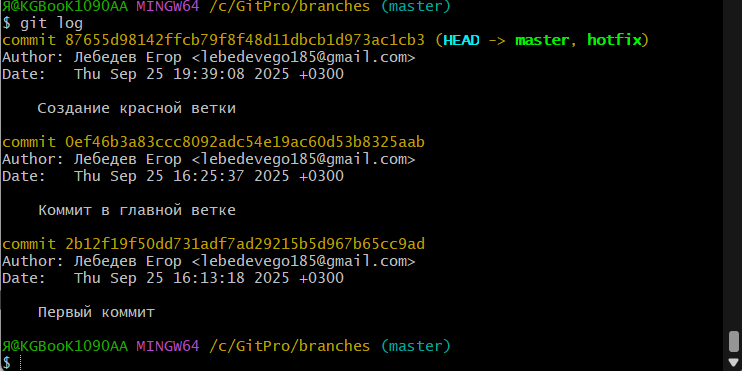
Список коммитов в ветке master:



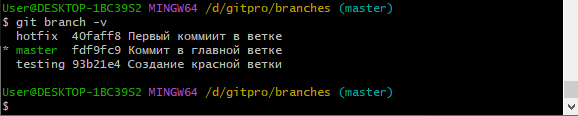
Слияние веток:



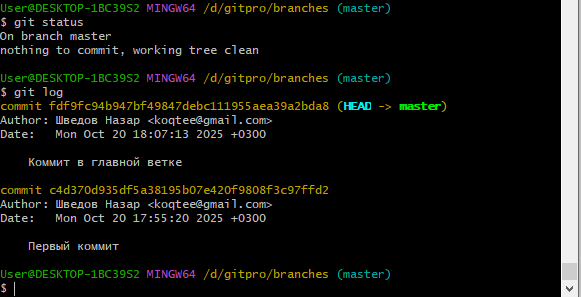
Вывод списка коммитов:



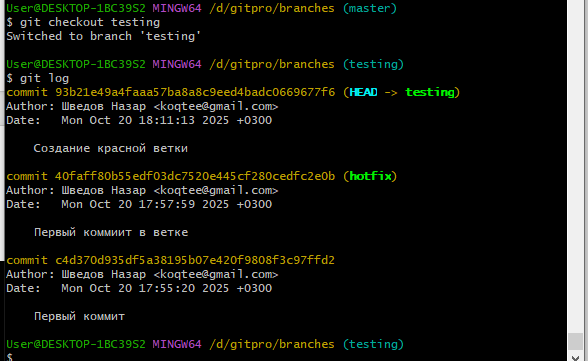
Команда git branch -v:



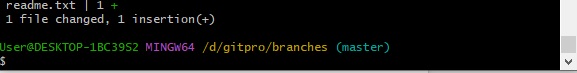
Проверка статуса и вывод списка коммитов главной ветки:



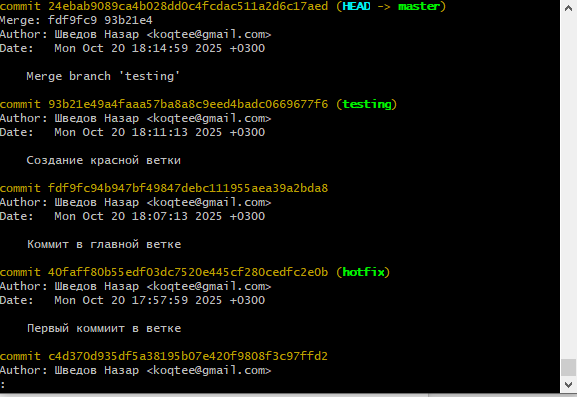
Переход в ветку testing и вывод списка ее коммитов:



Переход в главную ветку и слияние ее c testing:

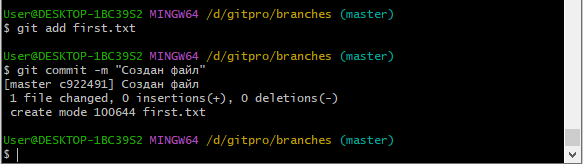


Список коммитов и коммит Merge branch ‘testing’ :

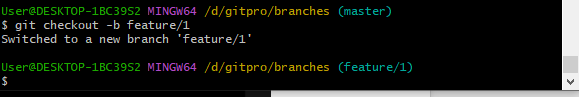


Решение конфликтов.

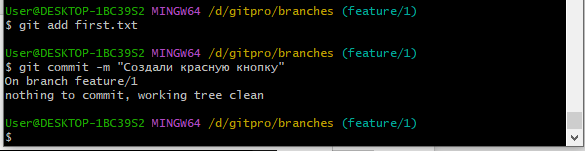
Создаем новый файл с текстом и делаем коммит:



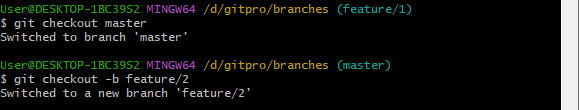
Создание новой ветки и переключение на ее:



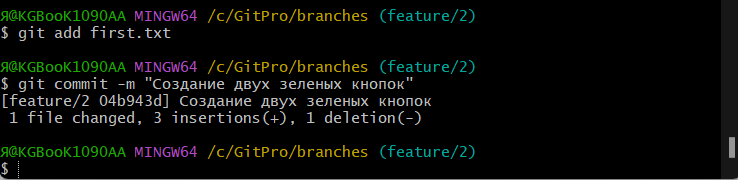
Добавление нового текста в файл, индексация и создание коммита:



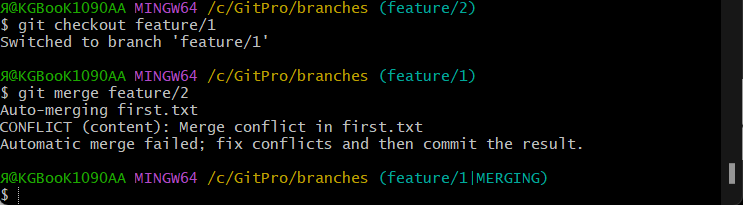
Переход на главную ветку и создание второй новой ветки:



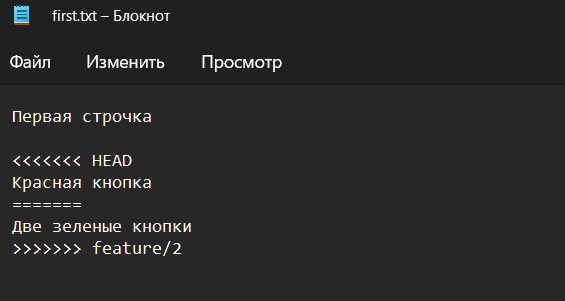
Добавление нового текста в файл, индексация и создание коммита:



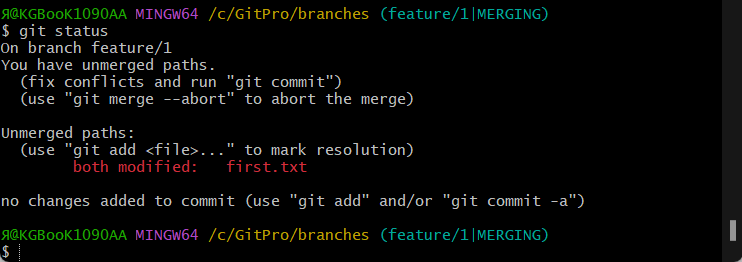
Переход на первую ветку и пытаемся их слить:



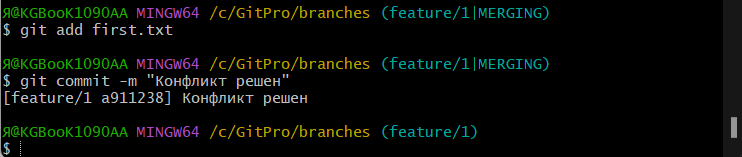
Причина ошибки слияния:



Убираем технические обозначения и сводим все к одной строчке. Проверяем статус репозитория:



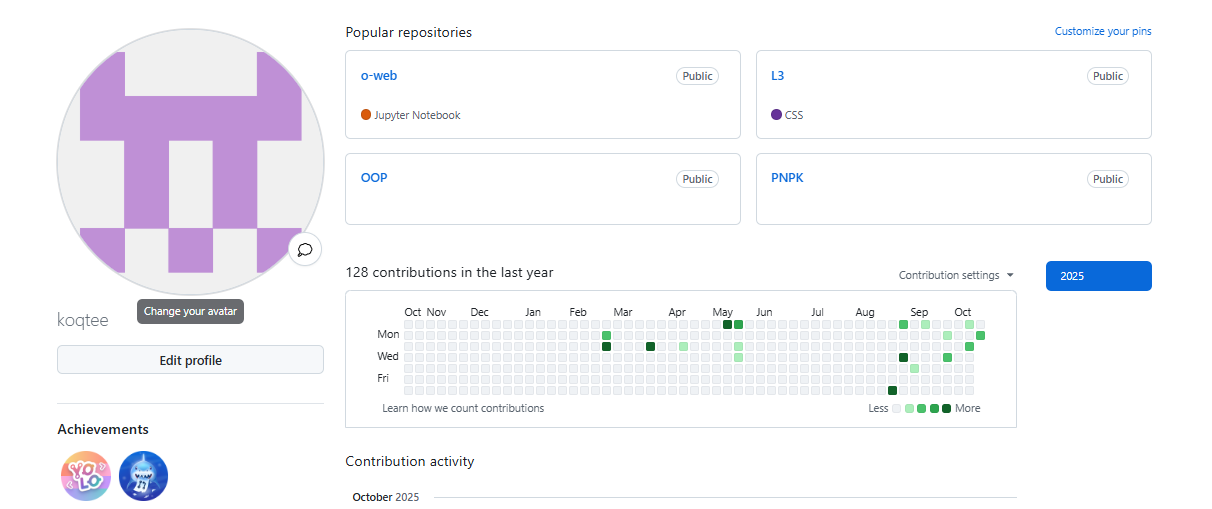
Индексация и создание коммита:



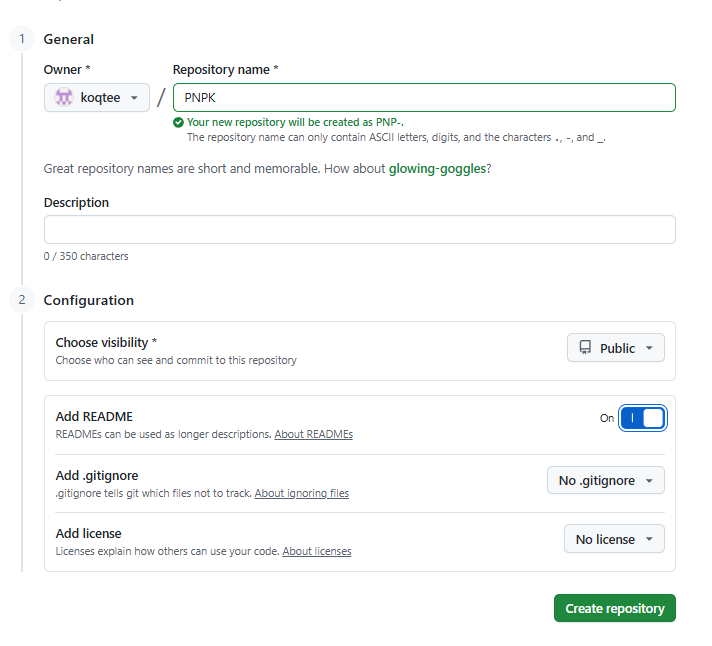
**Задание 1.7**

Работа с GitHub.

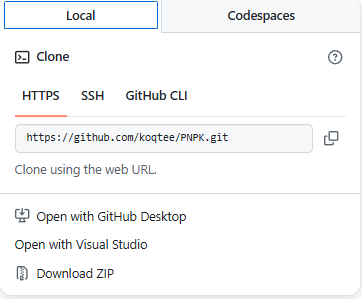
Созданный аккаунт на GitHub



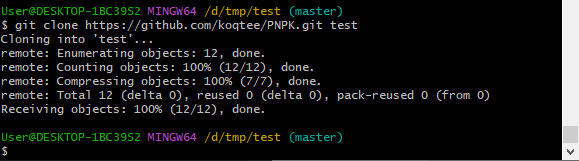
Создание удаленного репозитория на GitHub:



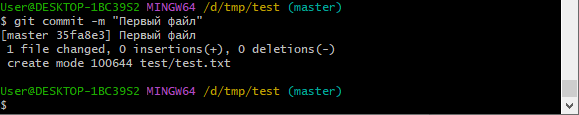
Получение ссылки на удаленный репозиторий:



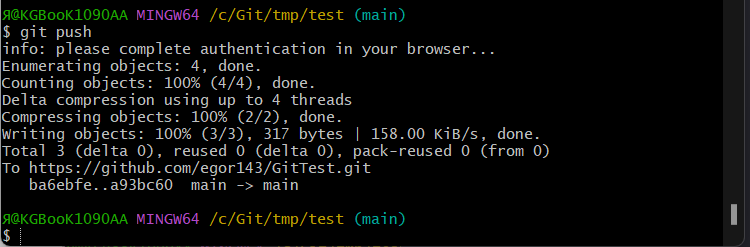
Подключаем наш репозиторий GitHub к локальному репозиторию:



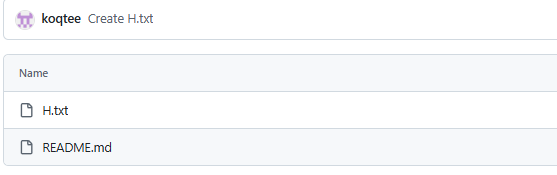
Создаем текстовый файл в папке test, индексируем и делаем коммит:



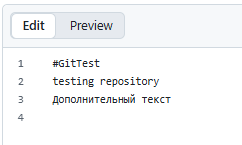
Команда gitpush:



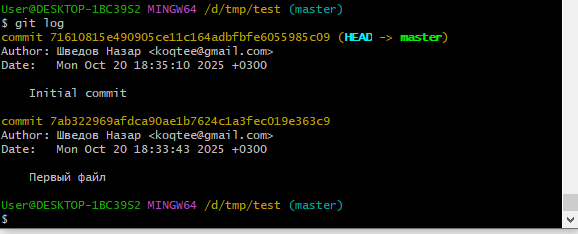
Файл успешно добавлен в удаленный репозиторий:



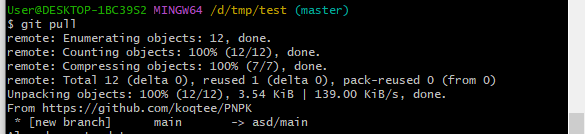
Добавляем текст README на GitHub:



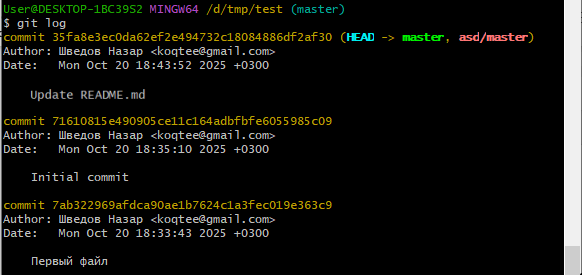
На GitHub у нас 3 коммита, а в локальном репозитории только 2:



Команда gitpull:



Выводим список коммитов:



Просмотр содержимого обновленного файла:

