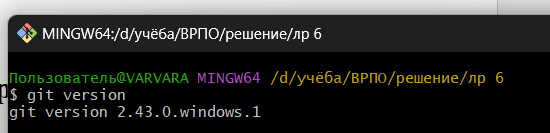
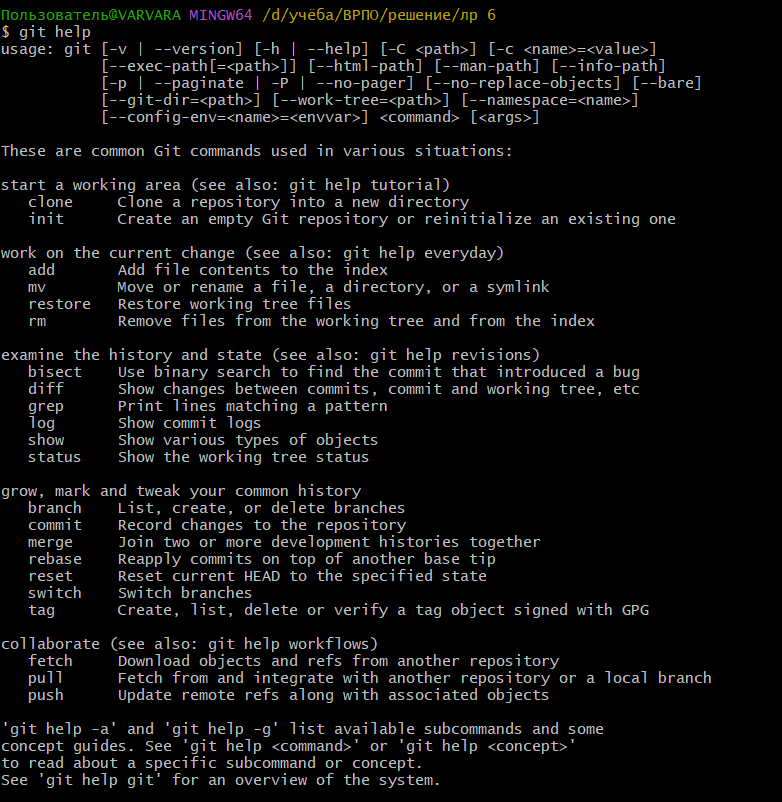
**Отчёт по лабораторной работе №6**

**Системы контроля версий. Основы работы в Git**

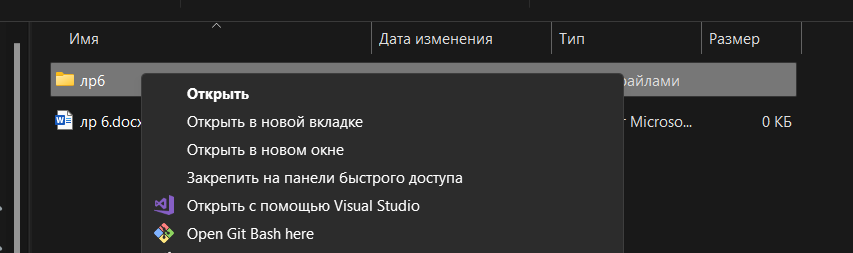
1. Проверка установленной версии

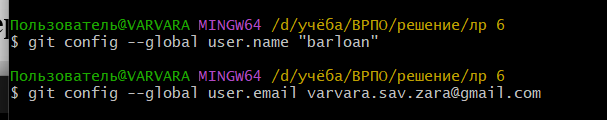


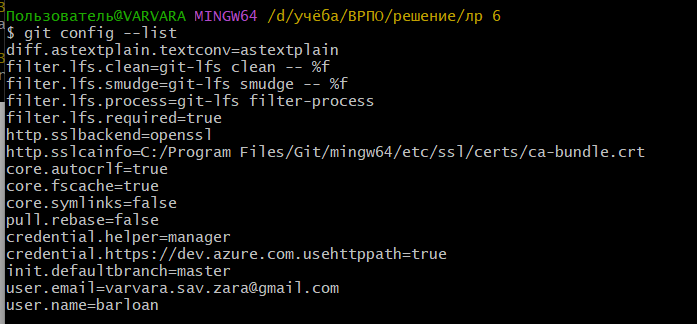
1. Получить справку по **Git**. Ознакомиться со списком командам с помощью команды git help.



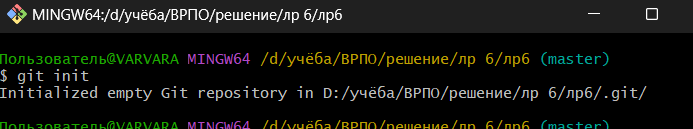
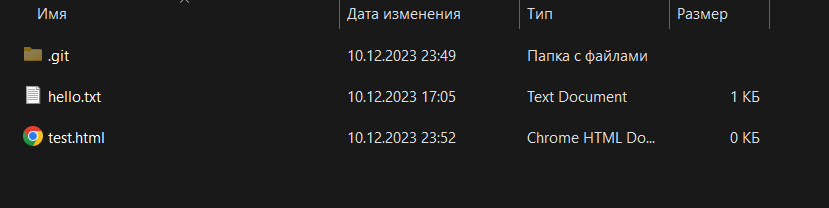
1. Создание локального репозитория. Запустить нужную папку в проводнике с помощью git bash.

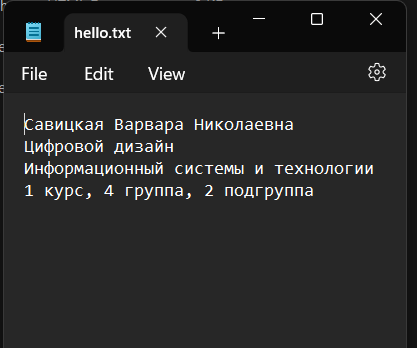


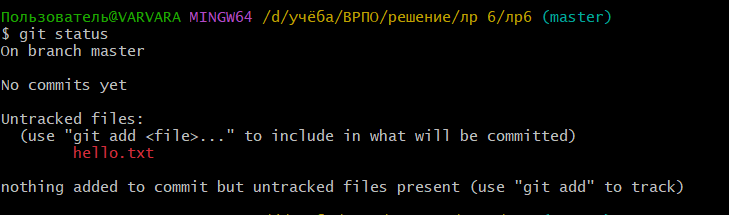
1. Выполните конфигурацию гит. Настройка конфигурационного файла
2. Проверка настройки конфигурационного файла командой git config --list



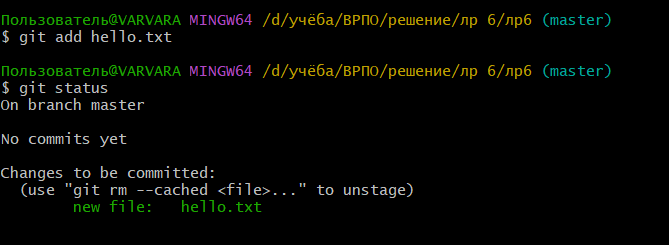
1. Инициализировать репозиторий с помощью команды git init

Появилась в проводнике новая скрытая папка .git

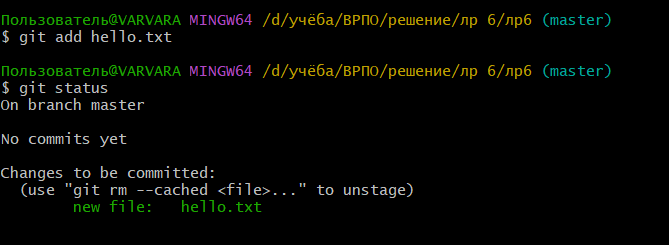
1. С помощью блокнота создать в папке текстовый файл Hello.txt.
2. Получить информацию о статусе репозитория с помощью команды git status. В окне отображается в какой ветки сейчас находимся, новые или модифицированные файлы, проиндексированные файлы



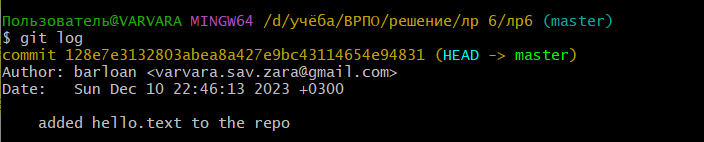
1. Проиндексировать файл hello.txt, то есть добавить его в репозиторий, с помощью команды git add hello.txt

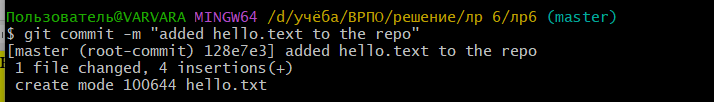


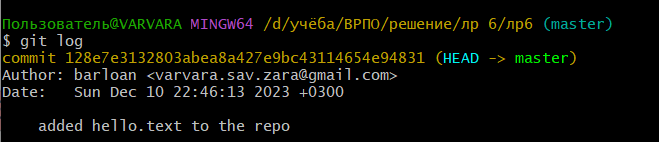
1. Для того, чтобы зафиксировать изменения индексированных файлов в репозитории используем еще раз команду git status. Можно увидеть, что коммит был успешно создан и добавлен новый файл hello.txt в ветку master.



1. Просмотр журнала с помощью команды git log



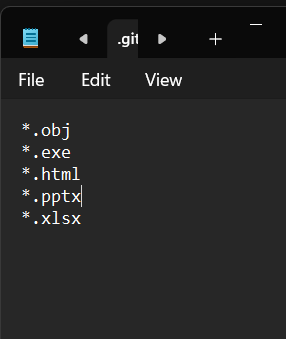
1. Зафиксировали изменения в репозитории с помощью git commit -m
2. С помощью команды можно git log журнал с логами commits начиная с указателя HEAD

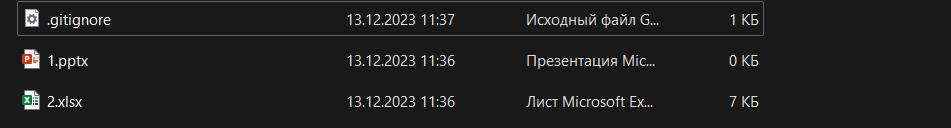


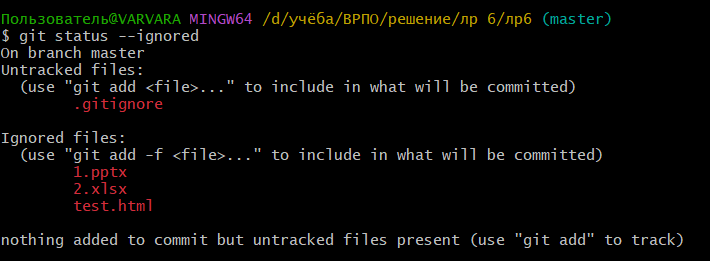
1. Ознакомление с шаблонами .gitignore и настройка этого файл для вашего репозитория

Создание текстового документа, занесение туда расширение тех файлов, которые будут игнорироваться

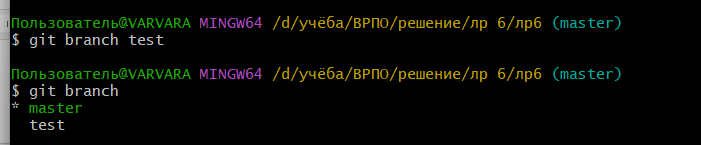
С помощью функции git status --ignored можно увидеть какие файлы игнорируются

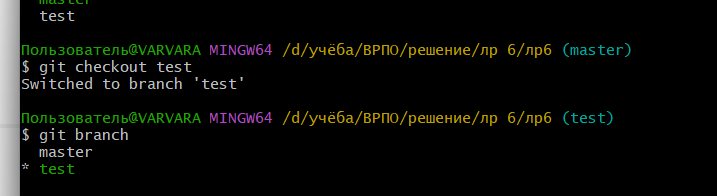




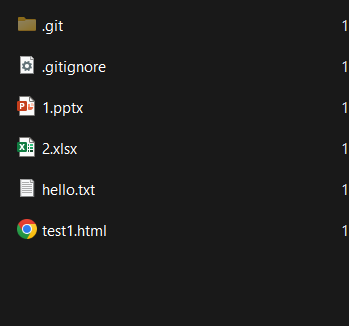


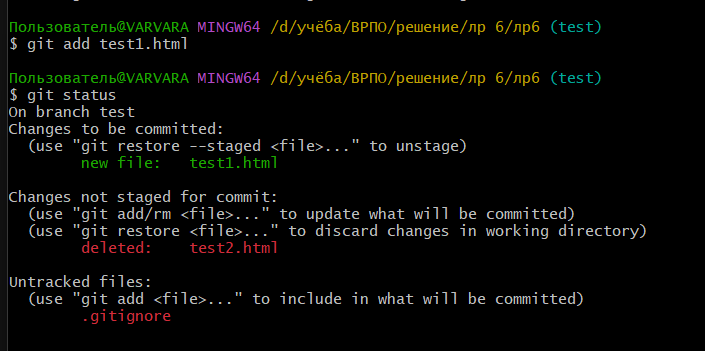
1. Создание новой ветки с именем test и переключение в нее. С помощью функций git branch и git checkout

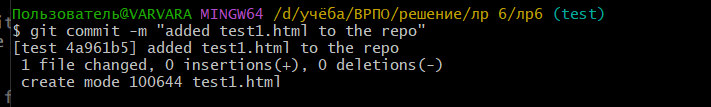




1. Создание нового текстового файл test2.html и фиксирование его (commit).

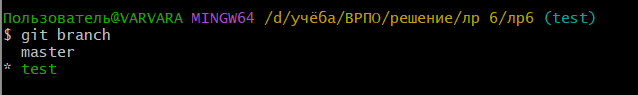




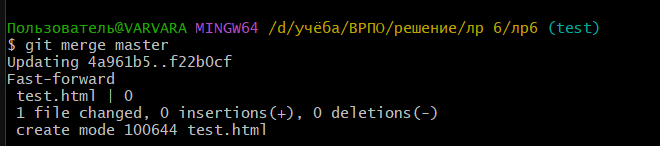


1. Просмотр списка всех веток и определение текущей

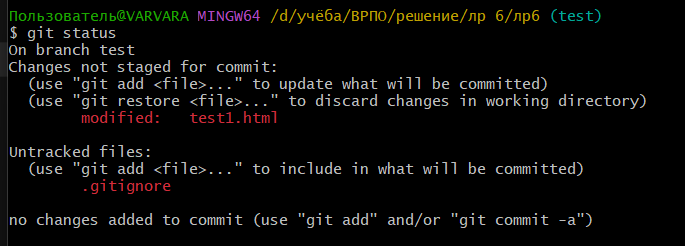
Текущей веткой является test

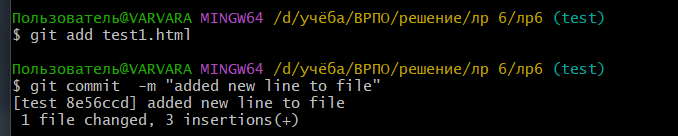
****

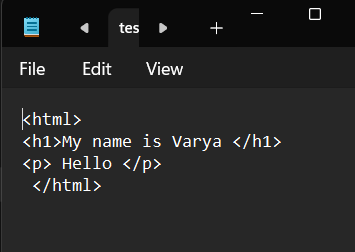
1. Выполнение слияние веток с помощью функции

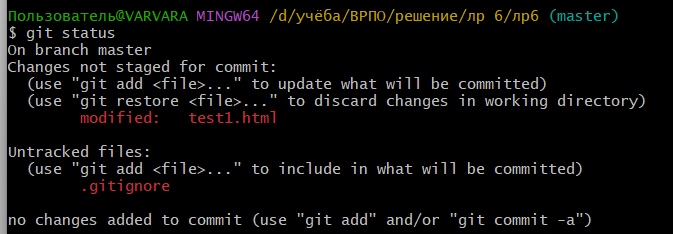


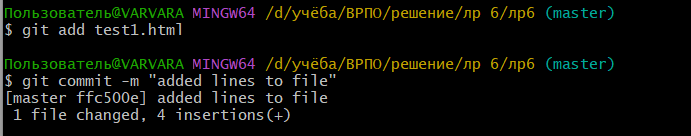
1. По очереди отредактировать файл test.html в ветках master и test. Выполните commit. Выполните слияние в одной и второй ветках. Разрешите ситуацию конфликта слияний

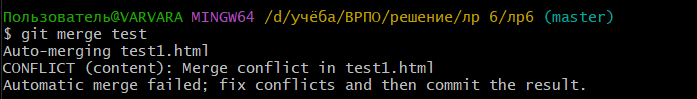


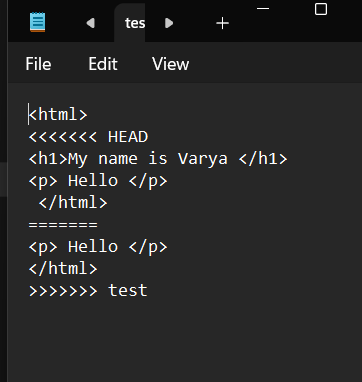








Ошибка при слиянии



Разрешение конфликта вручную

