

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | Институт цифровых интеллектуальных систем | Кафедра  компьютерных систем управления | |

Дисциплина «Основы системного программного обеспечения»

**Отчет по лабораторной работе №\_1\_**

|  |  |  |
| --- | --- | --- |
| **Выполнил**  **студент гр. АДБ-20-07:** | \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_  *(дата) (подпись)* | **Мельников Н. А** |
| **Проверил**  **к.т.н., доцент** | \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_  *(дата) (подпись)* | **Ковалев И.А.** |

**Москва 2023 г.**

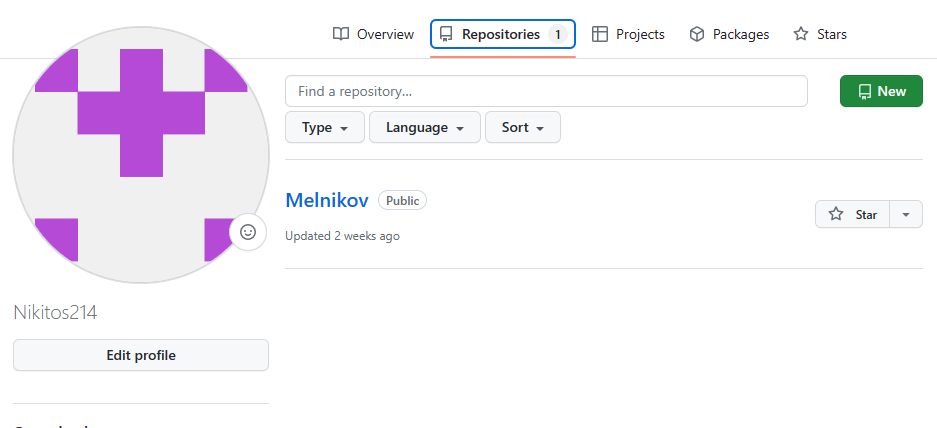
**Содержание**

1. **Цель**
2. **Задачи работы**
3. **Описание и выполнение тестовых (если присутствуют) и индивидуальных заданий**

**Цель:** Работа с системами контроля версий на примере Git Hub

**Задачи:** создать рабочую область и модифицировать версии файлов

Создание нового репозитория

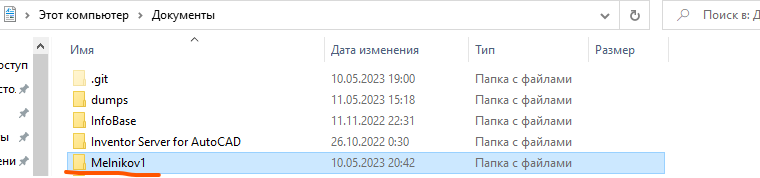


Создадим теперь локальную папку, которая будет связана с нашим

облачным git хранилищем. В общем случаем облачный git может представлять

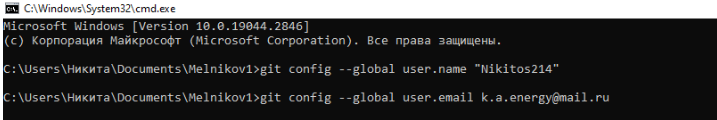
также локальный сервер, объединенный в единую информационную сеть со

всеми ПК.



Работа с командной строкой:

Указываем имя пользователя и адрес электронной почты:

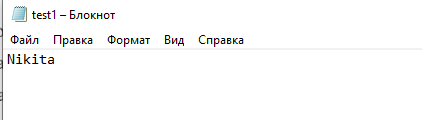


Перейдите в созданную вами папку в командной строке используя команду cd.

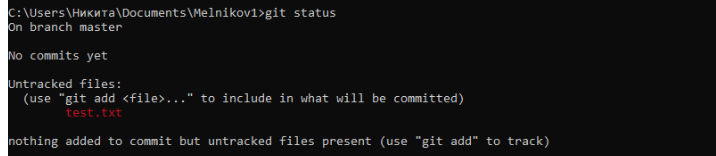


Теперь используя проводник windows создайте в своей папке любой

текстовый файл и пропишите в него ваше имя.



Вводим команду git stаtus и вам отобразиться сообщение, что есть новый файл, но он не отслеживается.

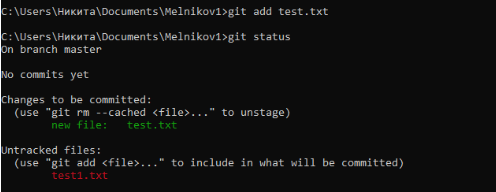


Это значит, что файл новый и система еще не знает, нужно ли следить за

изменениями в файле или его можно просто игнорировать. Для того, чтобы

начать отслеживать новый файл, нужно его специальным образом объявить.

Если хотим добавить только один файл, можно написать: git add test.txt

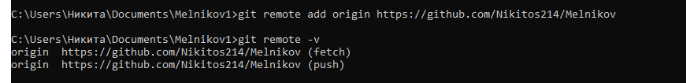


Файл готов к коммиту.

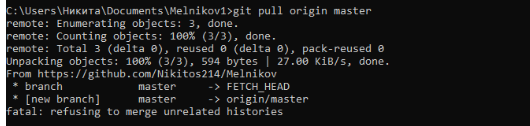


Чтобы связать наш локальный репозиторий с репозиторием на GitHub,

выполним следующую команду в терминале.

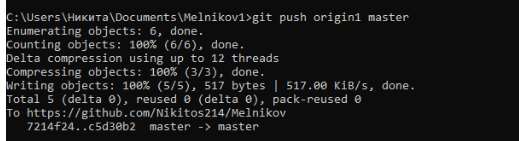


у нас есть файл и на сервере и локальном репозитории. Если вы сделали изменения в вашем репозитории (а мы их сделали, создав файл readme), другие пользователи могут скачать изменения при помощи команды git pull origin master

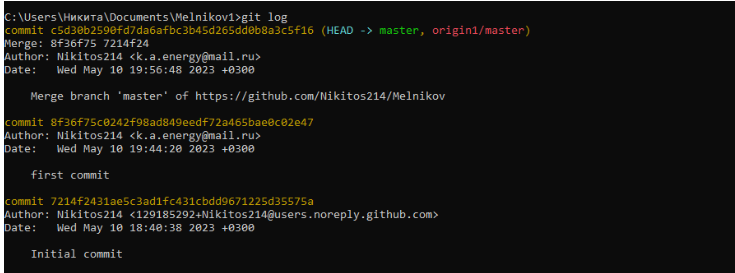


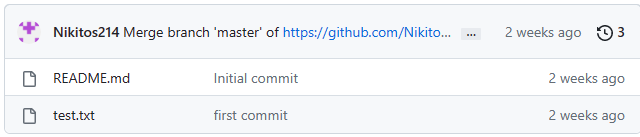
Теперь отравим коммит на сервер, команда, предназначенная для этого — push. Она принимает два параметра: имя удаленного репозитория (мы назвали наш origin) и ветку, в которую необходимо внести изменения (master — это ветка по умолчанию для всех репозиториев). git push origin master.





Посмотрим наши изменения с использованием команды git log



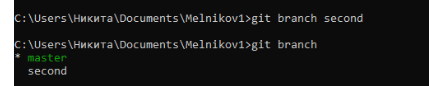


Создание новой ветки:

Создадим новую ветку second – команда используемая git branch second.

Можно посмотреть в какой ветки находимся, используя команду

git branch



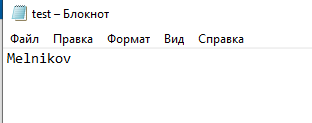
Переключимся на другую ветку, используя команду git checkout second.

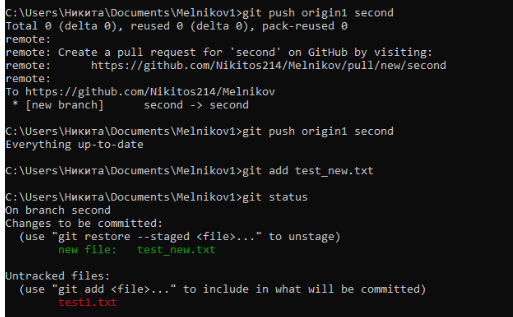


Создайте новый файл в нашем локальном репозитории и напишете в нем свою

фамилию, добавьте в область подготовленных файлов, закомитьте и отправьте

на сервер, только учтите, что у вас теперь ветка не master, а second





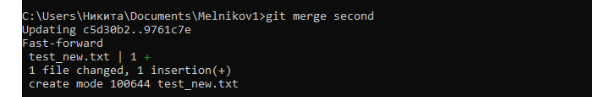
Слияние веток:



Чтобы он появился, нужно воспользоваться merge

для объединения веток (применения изменений из ветки second к основной

версии проекта).



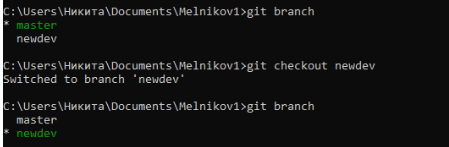
можно удалить ветку git branch –d second



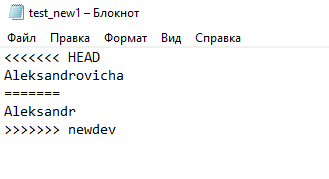
Создайте ветку с названием newdev



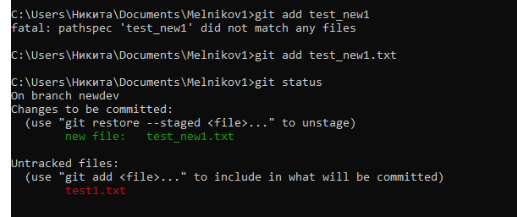
Переключитесь на нее



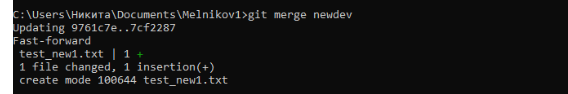
Добавьте в файл с именем отчество



Зафиксируем изменения. Закомитьте изменения

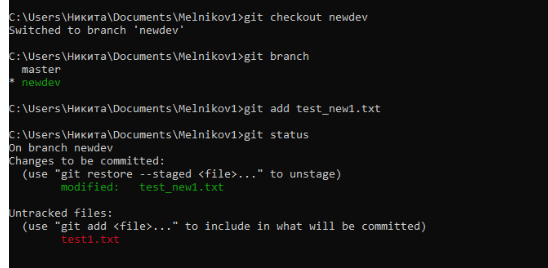


Объединим ветки



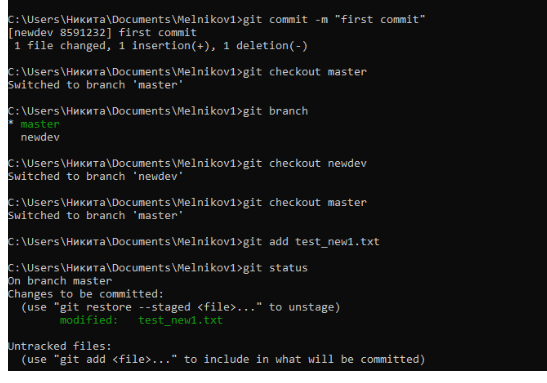
Переключитесь на ветку newdev и удалите в отчестве несколько букв

Зафиксируйте, закоммите

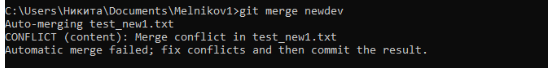


Переключимся на ветку master, добавьте к отчеству несколько букв

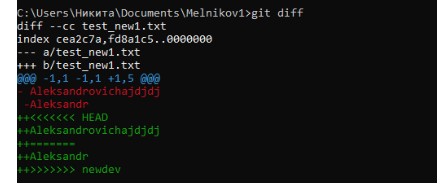
Зафиксируйте, закоммите

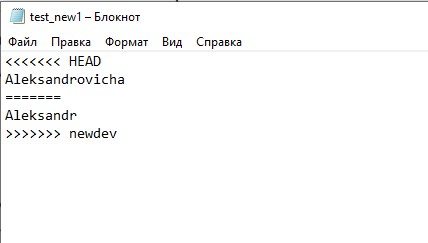


Объединим ветки git merge newdev.

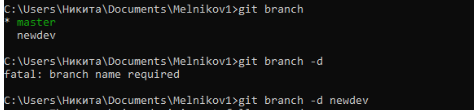


Наберите команду, для просмотра изменений git diff.





Удалите ветку newdev





Давайте удалим их из github

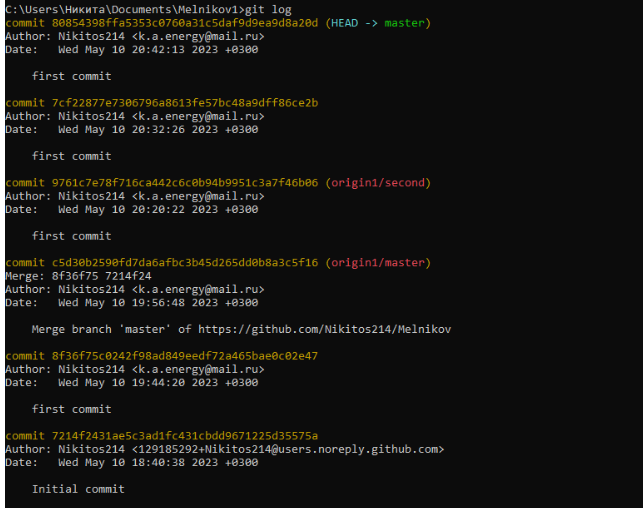
git push origin --delete newdev

git push origin --delete second



Чтобы посмотреть все комиты, можно использовать команду

git log

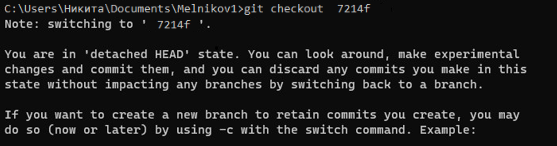


Выберете любой коммит, на который хотите откатится, достаточно указать его

первые несколько символов (перед выполнением посмотрите свои файлы,

чтобы запомнить, что там находится)

git checkout [hash commit]



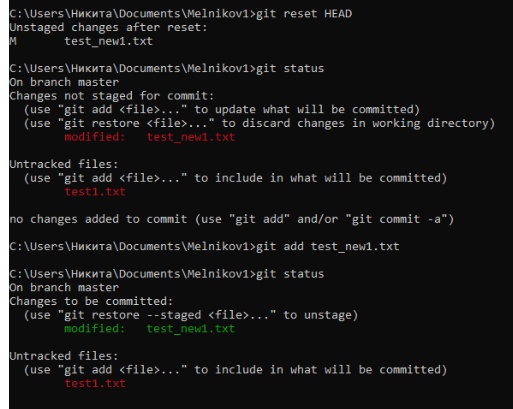
Исправление коммита

Возможно ситуация, что вы закрепили файлы (поменяйте что-то в файле и

закрепите их, вызовите git status), но еще не коммитили, хотите убрать файлы

из области закрепления, вызовите команду:

git reset HEAD





Вывод:

В данной лабораторной работе были приобретены навыки работы с системами контроля версий, а именно с Git Hub, мы познакомились ближе с файловой системой, с технологией работы веток и коммитов.