Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №5**

Дисциплина: «Разработка профессиональных приложений»

Название: «GIT»

Вариант №6

Выполнил:

студент группы ИВТАПбд-22

Матюнин И.С.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск

2023

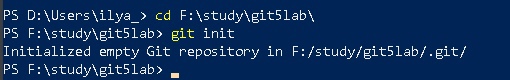
**Здание:**

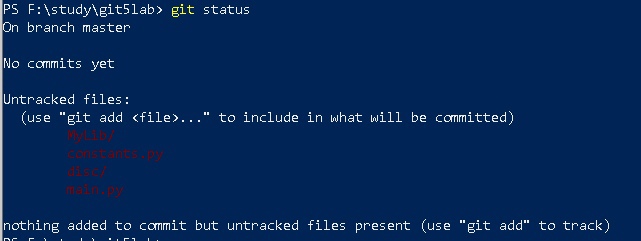
1. Установить Git (<https://git-scm.com/>)
2. Командами в КОНСОЛИ:
   1. Создать локальный репозиторий
   2. Добавить несколько файлов
   3. Сделать один из файлов игнорируемым
   4. Зафиксировать изменения
   5. Внести изменения в файлы и зафиксировать изменения
   6. Создать несколько бранчей, редактировать файлы в разных бранчах и фиксировать изменения. Осуществить слияние двух веток в одну.
   7. Создать аккаунт github (если его нет, а если уже имеется, то использовать его), отправить исходные коды в удаленный репозиторий.
   8. Склонировать удаленный репозиторий в папку, отличную от оригинальной.
   9. Добиться конфликта файлов при применении изменений в оригинальном репозитории.
   10. Разрешить конфликт.
   11. Подготовить отчет о проделанной работе.

**Описание Реализации:**

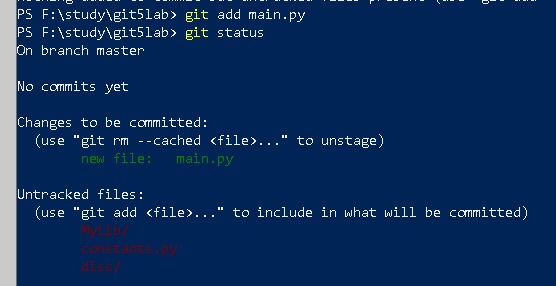
Лабораторная работа была выполнена в программе **Windows PowerShell.** В первую очередь все файлы из 4 лабораторной работы были перенесены в отдельную директорию.

Далее был инициализирован гит репозиторий в новой папке, определён статус репозитория (проверка на новые файлы).

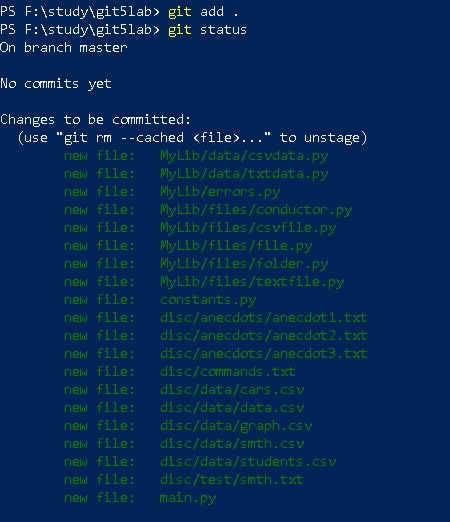




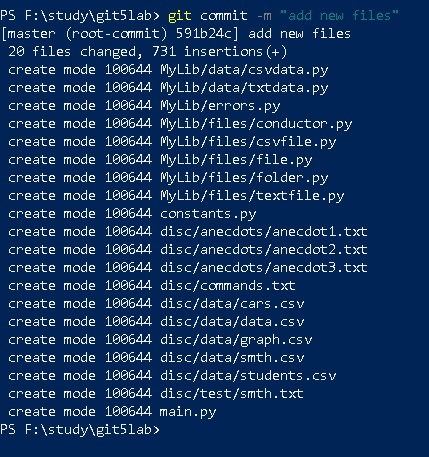
Добавляем файл main.py в репозиторий:



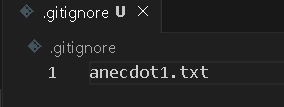
Добавляем все файлы в репозиторий и проверяем их статус:

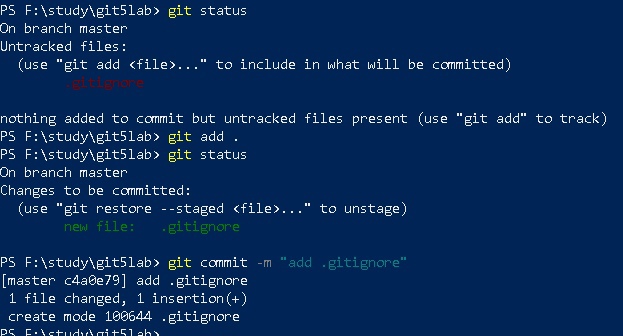


Создаём commit означающий изменение в репозиторие, а точнее добавление новых файлов.

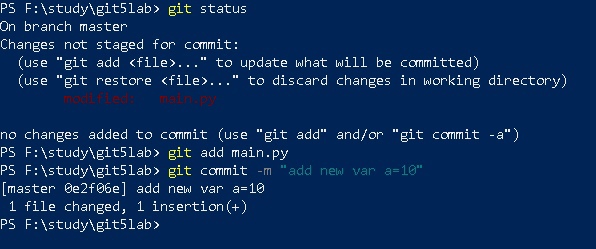


Далее был создан файл .gitignore. В данный файл записываются те файлы и директории, которые не нужно добавлять в удалённый репозиторий. Занесли в созданный файл название файла, который не должен попасть в git:





Далее был изменён файл main.py и заново занесёт в git:



Далее была вызвана команда просмотра доступных веток. Эта же команда позволяет создать новую ветку в удалённом репозитории. Так же был совершён переход на новую ветку.

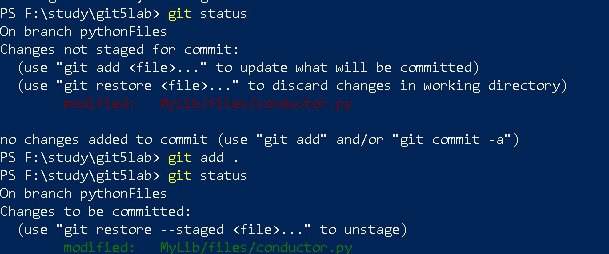


В новую ветку были добавлены файлы и создан commit:

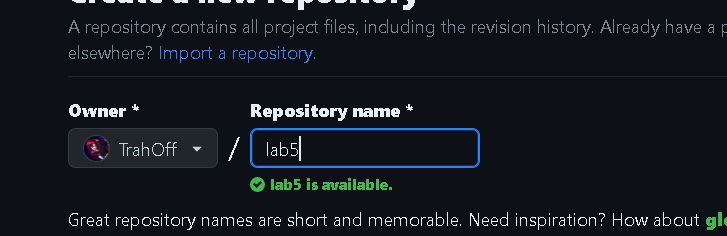
https://sun9-3.userapi.com/impg/nJHXIPobir4z9gDqIFkzAUtsHSA71tidCKhMfw/8lJnLrZ4j00.jpg?size=362x42&quality=96&sign=26303c6ce3985c2c4e9969af7b85fbde&type=album

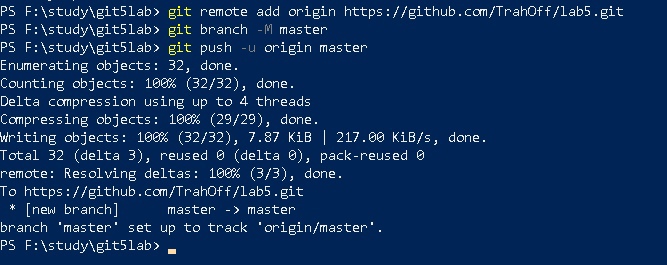
https://sun9-20.userapi.com/impg/HwQkk4O2mnSNglPN65texpFB-i-CKRzn6Rec3A/LUr2jby8jXA.jpg?size=341x20&quality=96&sign=b997cf0a5cde7954ca595f152eb7af63&type=album

В файл conductor.py были внесены изменения и так же проверен статус git.



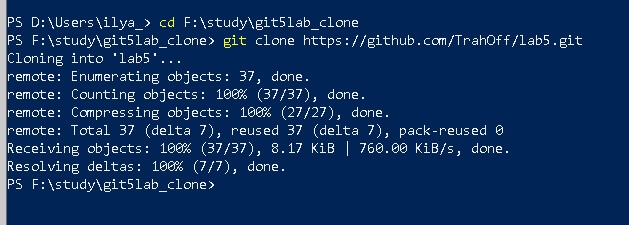
Далее в аккаунте GitHub был создан репозиторий, в который все файлы в дальшейнем должны будут быть загружены.

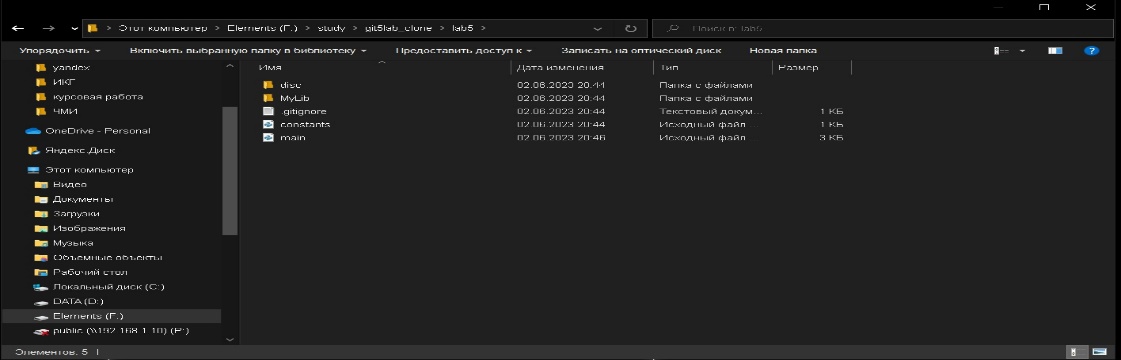




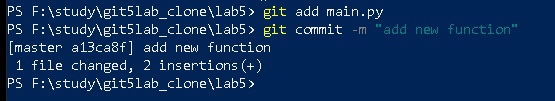
Таким образом были загружены все созданные ветки.

Далее на локальном компьютере была создана новая директория, в которую был клонирован удалённый репозиторий:



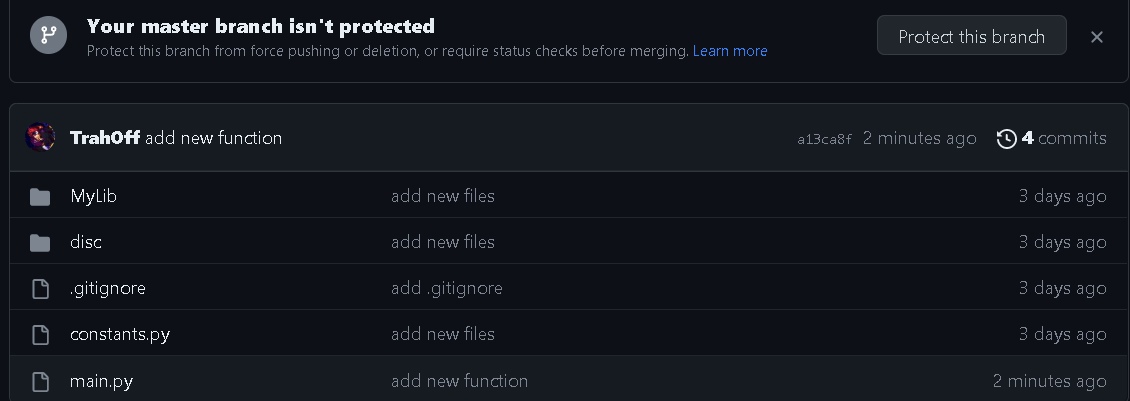


Далее в репозиторий был отправлен локальный файл main.py и создан commit:



После всех операций изменения были загружены в аккаунт GitHub:





**Описание альтернативных способов решения:**

Альтернативные решения найдены не были.