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»

Git

Вариант № 14

Выполнил:

студент группы ИВТАПбд-21

Мухаметзянов Т. А.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

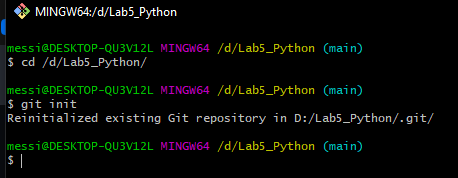
Необходимо научиться использовать все возможности GIT.

**Требования к программе:**

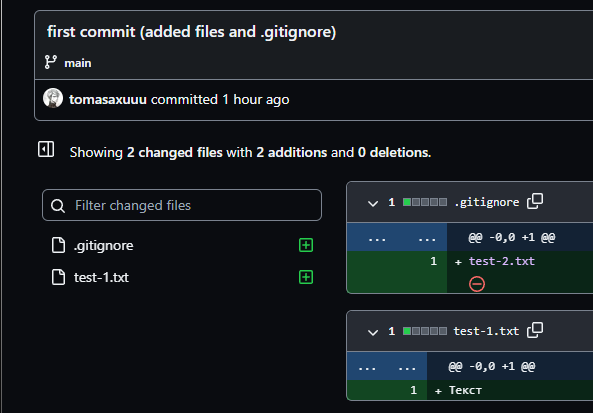
1. Создать локальный репозиторий
2. Добавить несколько файлов
3. Сделать один из файлов игнорируемым
4. Зафиксировать изменения
5. Внести изменения в файлы и зафиксировать изменения
6. Создать несколько бранчей, редактировать файлы в разных бранчах и фиксировать изменения. Осуществить слияние двух веток 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.

**Описание реализации**

1. Создание локального репозитория происходит при помощи создания папки на вашем компьютере и инициализации ее с помощью GIT



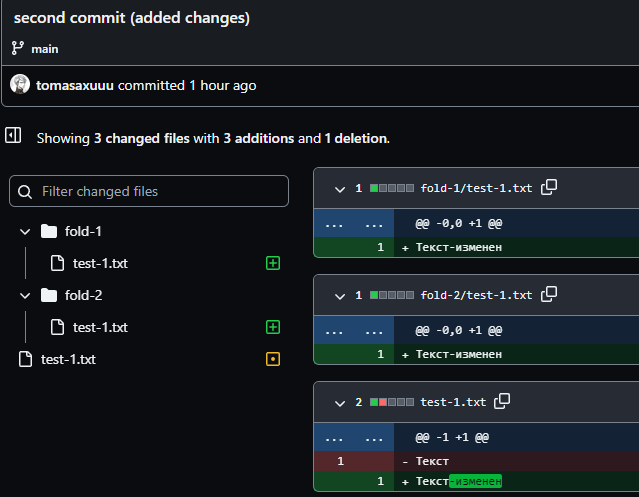
1. Добавление файлов, добавили в ручную либо с помощью команд touch <file>, echo “content” <file>, дальше перечень команд:  
   git add .  
   git commit -m “commit”  
   git push -U origin main (добавление сначала в основную ветку проекта)

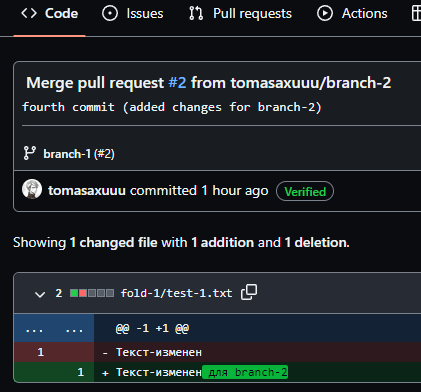


1. Игнорирование файла – создание файла .gitignore и вписываем туда название файла, который будет игнорироваться  
   
2. Внести изменения:

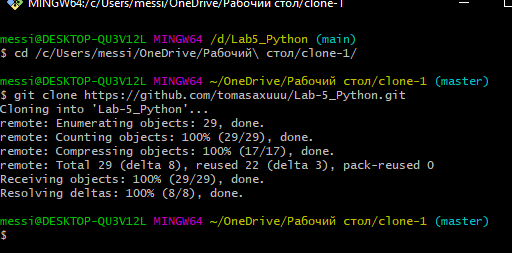
git add .  
git commit -m “commit”  
git push

1. Изменения в файлах и снова внести изменения (изменяем содержимое файлов или добавляем новые файлы и повторяем пункт 4)

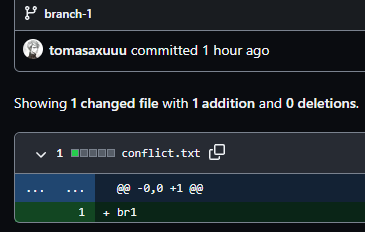


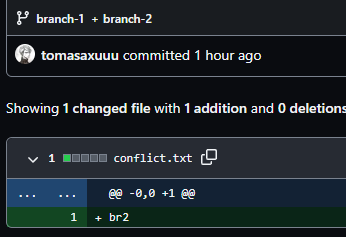
1. Создание нескольких бранчей, редакт файлов в бранчах и их слияние (веток)  
   создание новой ветки: git branch <name branch>  
   перейти в новую ветку: git checkout <name branch>  
   повторяем пункт 4 для внесения изменений в ветку, но вместо git push пишем git push origin <name branch> - повторяем так 2 раза в две разные ветки, дальше заходим в первую новую ветку командой: git checkout <name 1branch> и пишем git merge branch 2, чтобы слить в первую ветку нашу вторую ветку, в итоге получим все изменения в одной ветке – branch 1.  
   
2. Аккаунт был создан, создаем репо, дальше заливаем файлы командами 4 пункта.
3. Клонирование репозитория с помощью команды:

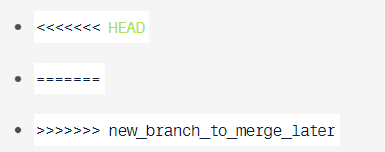
git clone https://github.com/tomasaxuuu/Lab-5\_Python.git



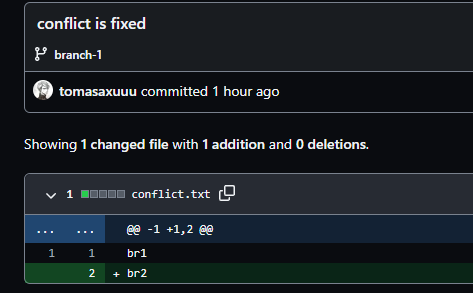
1. Конфликт файлов: создаем файл с одним и тем же именем в двух ветках (branch-1 и branch-2), с разным содержимым, дальше командой git checkout branch-1  
   git merge branch-2  
   сливаем изменения в ветку branch-1.  
   У нас будет конфликт, так будет 1 файл с разным содержимом в двух ветках, поэтому чтобы его разрешить мы заходим в конфликтный файл и меняем его так как нам надо, после чего командами из 4 пункта просто пушим изменения в репо.







(тут я забыл скрин, но после HEAD содержимое файла из branch-1 должно быть, а после середины из branch-2. Удаляем все кроме содержимого из двух файлов и получается что у нас теперь будет – br-1  
br-2)



Ссылка на репо с 5 лабой:  
<https://github.com/tomasaxuuu/Lab-5_Python>