Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра Информатики

Дисциплина «Избранные главы информатики»

**ОТЧЕТ**

к лабораторной работе №1

БГУИР 6-05-0612-02 118

|  |
| --- |
| Выполнил студент группы 353501  СЛЮСАРЬ Станислав Юрьевич |
|  |
| (дата, подпись студента) |
| Проверил доцент каф. Информатики  ЖВАКИНА Анна Васильевна |
|  |
| (дата, подпись преподавателя) |

Минск 2025

# 1 Индивидуальное задание

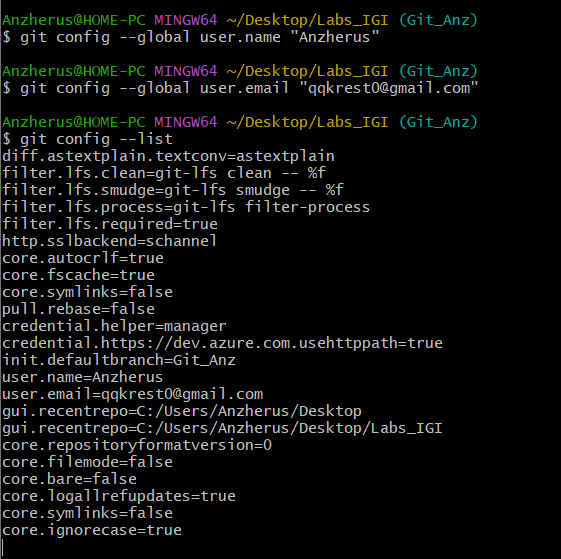
**Задание 1. Вариант 22.** Познакомиться с возможностями и получить практические навыки работы с системой контроля версий Git.

# 2 Выполнение работы

Сначала проверяем версию загруженного github, с помощью команды git –version:

Рисунок 1 — Проверка версии

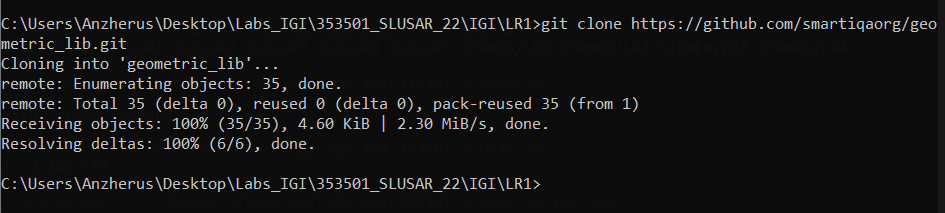
Проводим базовую настройку Git:

Рисунок 2 — Базовая настройка Git

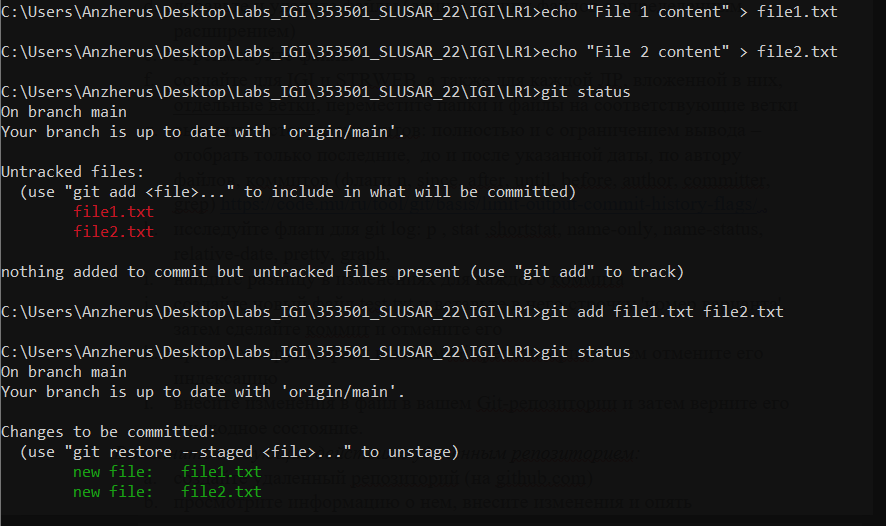
Далее создаём структуру папок:

Рисунок 3 — Создание структуры папок

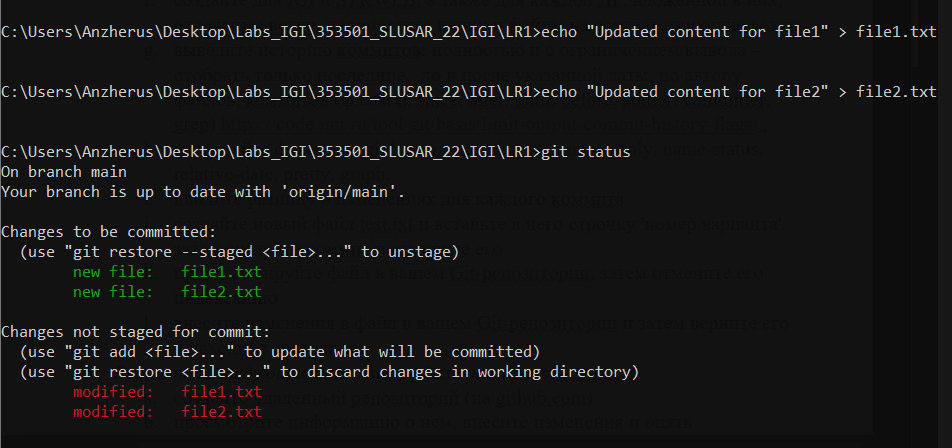
Создаём репозиторий( клонирум с гита) в структуре .../IGI/LR1:

Рисунок 4 — Клонируем репозиторий

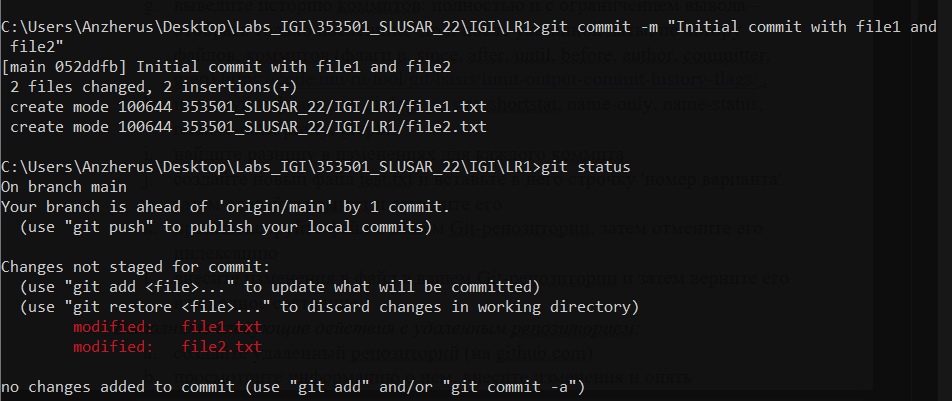
Проводим индексацию нескольких файлов:

Рисунок 5 — Индексация нескольких файлов

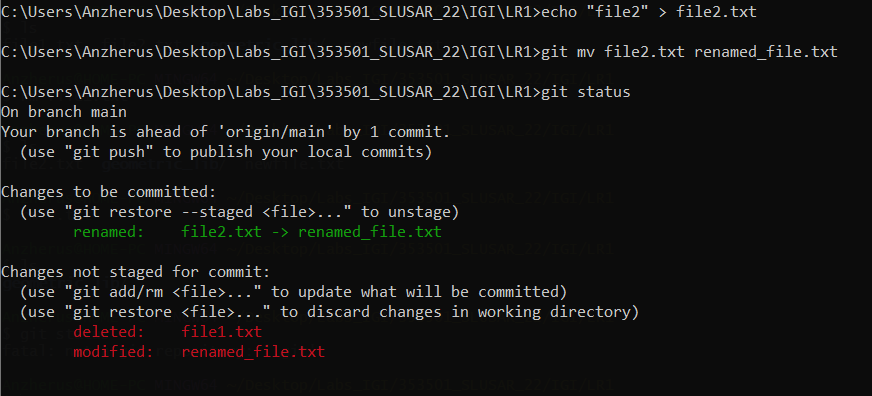
Вносим изменения в файлы:

Рисунок 6 — Внесение изменений в файлы

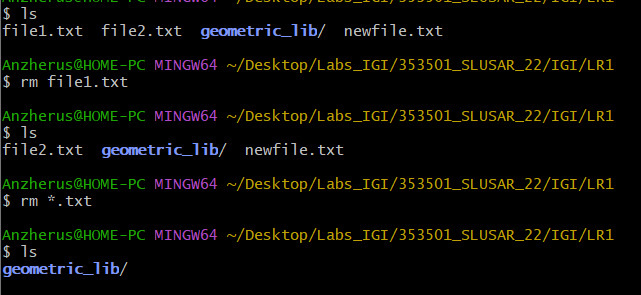
Выполняем коммит:

Рисунок 7 — Выполняем коммит

Переименовываем файлы:

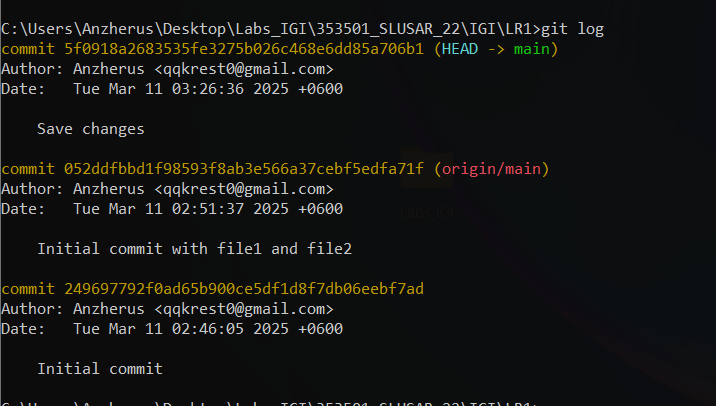
Рисунок 8 — Переименовываем файлы

Удаление файлов :

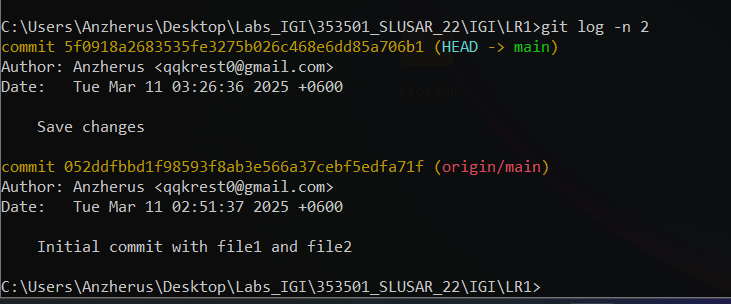
Рискунок 9 — Удаление файлов

Теперь проходим по истории коммитов:

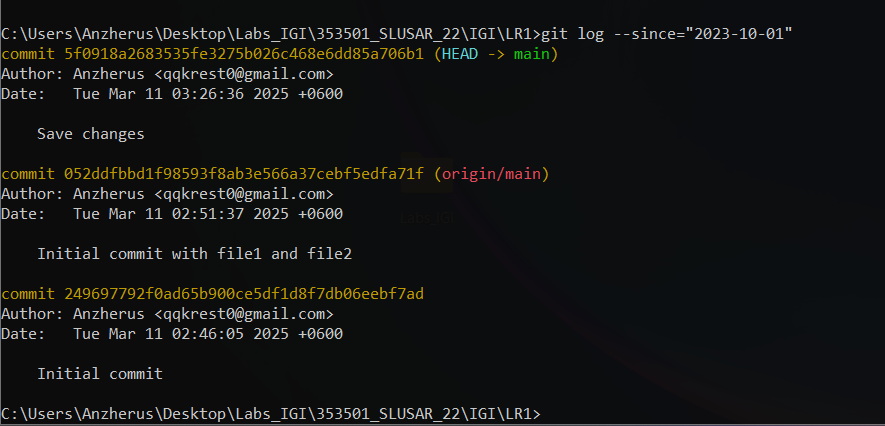
Полная история :

Рисунок 10 — git log

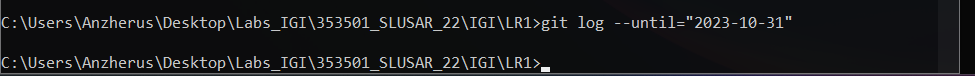
Ограничение в 2 :

Рисунок 11 — git log -n 2

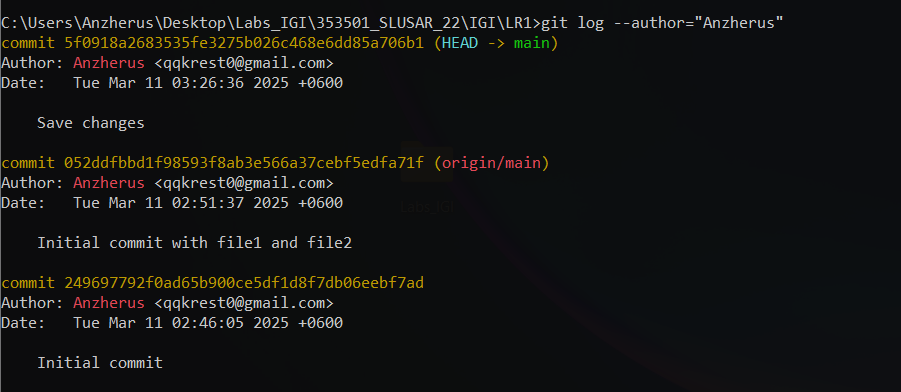
История с определённого времени :

Рисунок 12 — git log –since=”…”

История с определённого времени:

Рисунок 13 — git log –until=”..”

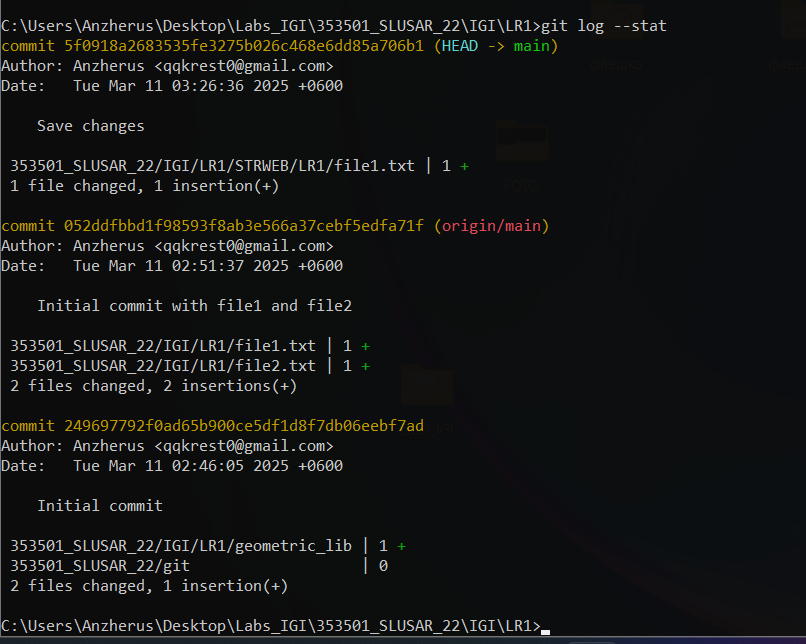
Вывод истории по автору:

Рисунок 13 — вывод по автору

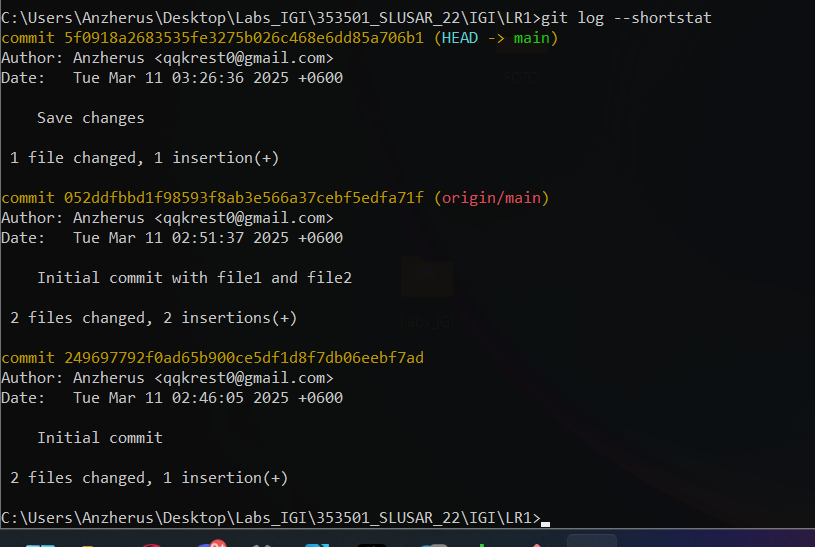
Показ изменений в каждом коммите:

Рисунок 14 — Изменения в коммитах

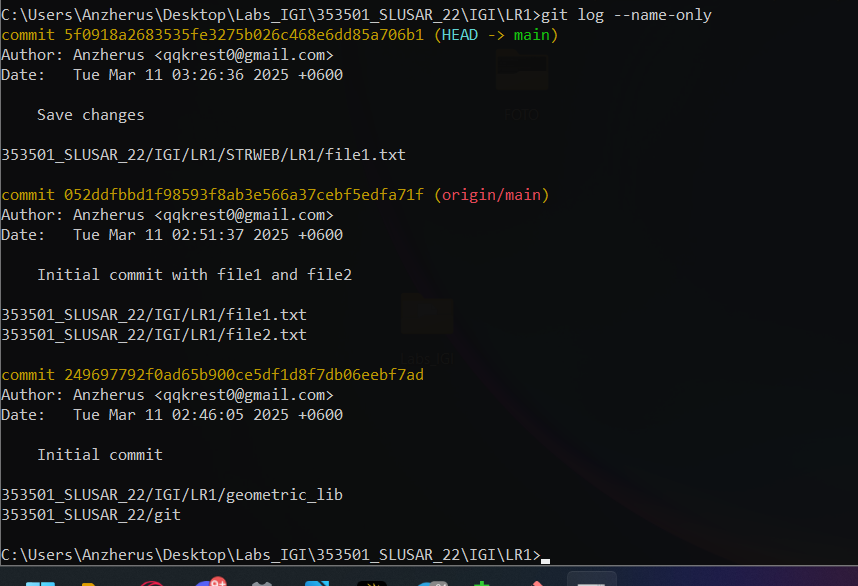
Статистика изменений:

Рисунок 15 — Cтатистика изменений

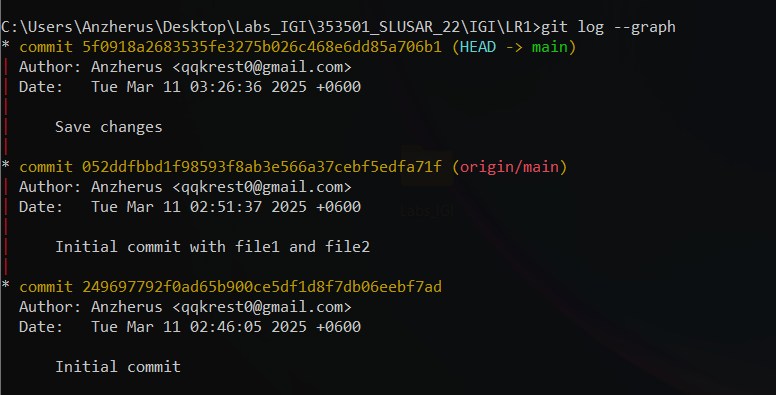
Выводим краткую статистику:

Рисунок 16 — Краткая статистика

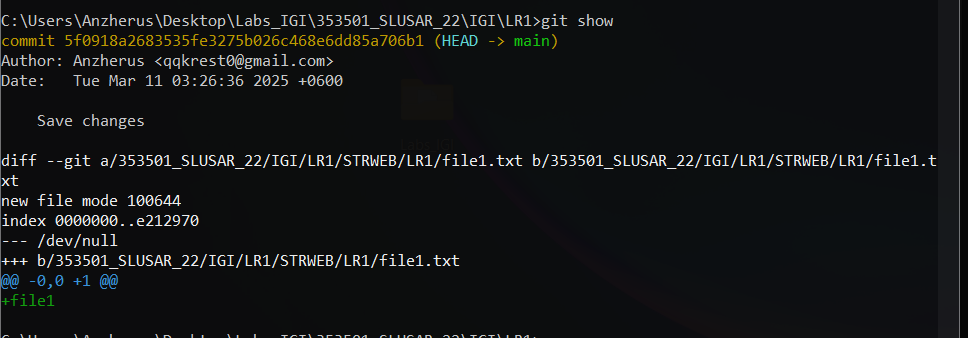
Только имена изменённых файлов:

Рисунок 17 — только имена изменённых файлов

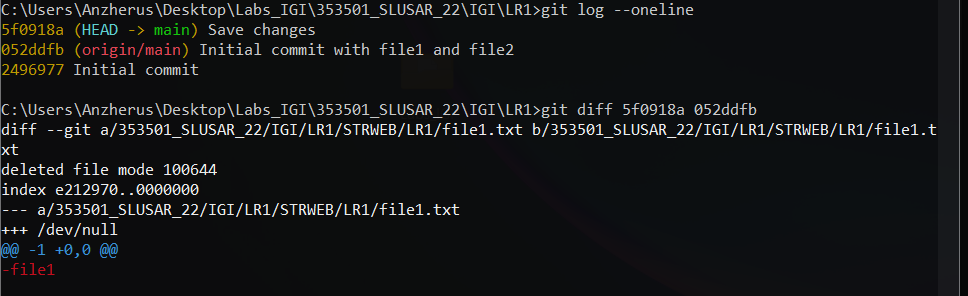
Графическое представление:

Рисунок 18 — графическое представление

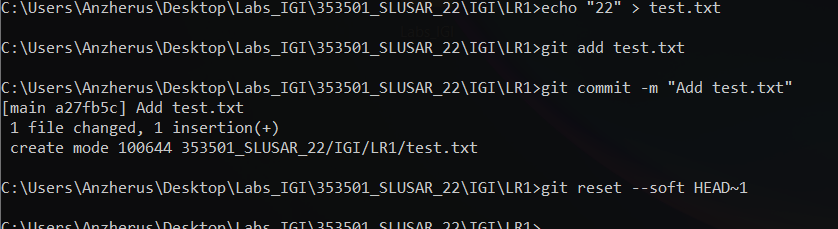
Показ разницы для последнего коммита:

Рисунок 19 — Git show

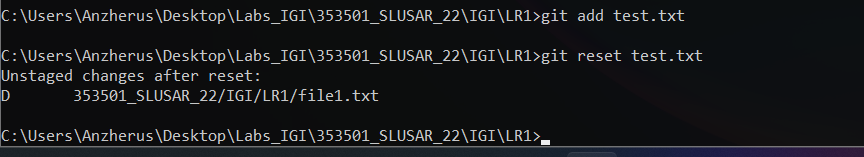
Разница между 2-мя коммитами:

Рисунок 20 — Git diff

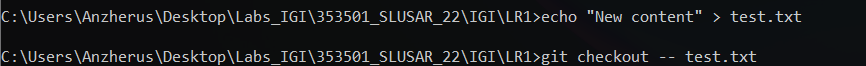
Создание и отмена коммита:

Рисунок 21 — Создание и отмена коммита

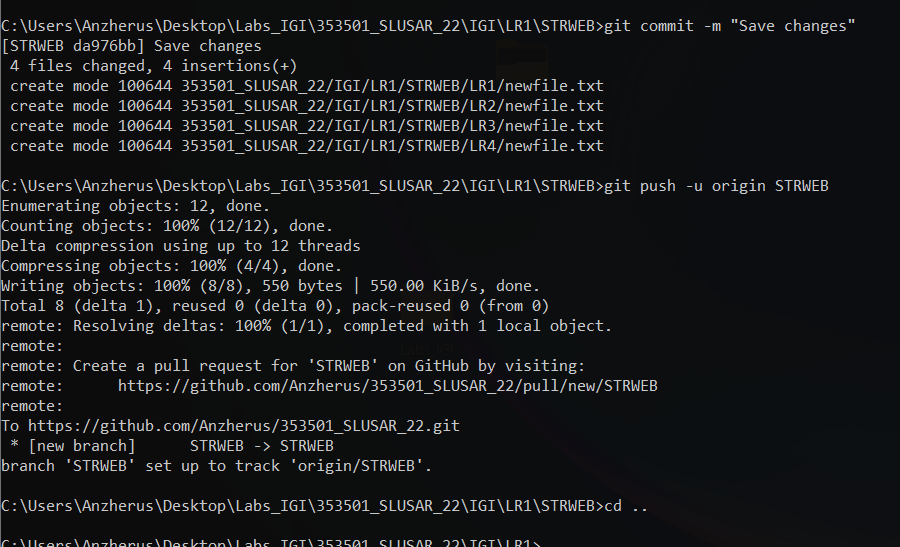
Отмена индексации файла:

Рисунок 22 — Отмена индексации файла

Возврат файла в исходное состояние:

Рисунок 23 — Возврат файла в исходное состояние

Делаем коммит и пушим в GitHub:

Рисунок 24 — commit + push

Далее загружаем всё остальное также в репозиторий.

Удалить ветку git branch -d/-D <branch>

Удалить удалённую git push origin –delete <branch>

Перенести файлы на ветку, на которой находимся git checkout <с которой хотим перенести> -- <Files>

Соеденить с веткой на которой находимся git merge <branch>

git remote -v

git branch / -r/ -a

delete file: git rm …

Удалить папку : git rm -r …

Удалить только в git: git rm -r –cached ...

git commit -m “edrfghj”

git push origin <branch>

local, delet, sistem

git push -u origin <branch>

Переименовать git mv <old\_name> <New\_name>

git config --list

# Вывод

В ходе лабораторной работы получили навыки проектирования приложения, состоящих из нескольких взаимосвязанных классов. Использование полиморфизма позволяет создавать более гибкие и расширяемые программные системы, а также реализовывать общий интерфейс для работы с различными объектами.