Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра Информатики

Дисциплина «Избранные главы информатики»

**ОТЧЕТ**

к лабораторной работе №1

на тему:

**«СИСТЕМА КОНТРОЛЯ ВЕРСИЙ GIT»**

БГУИР 6-05-0612-02 20

|  |
| --- |
| Выполнил студент группы 353504  СИЛЬЧЕНКО Анна Андреевна |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  ЖВАКИНА Анна Васильевна |
|  |
| (дата, подпись преподавателя) |

Минск 2024

# 1 Индивидуальное задание

**Задание**:

1. *Подготовьте рабочее окружение в соответствии с типом вашей операционной системы(<https://smartiqa.ru/courses/git/lesson-1> ):*
   1. Установите Git
   2. Выполните базовую настройку
2. *Создайте структуру папок (использовать командную строку) и поместите в репозиторий Git на своем локальном компьютере:*

(Наименование для базовой папки: **253501\_IVANOV\_12**, где 12 – порядковый номер в журнале)

GROUP\_NUMBER\_YOUR\_LAST\_NAME\_ NUMBER\_IN\_LIST

**IGI**

LR1

Папки и файлы в соответствии с индивидуальным заданием



Отчет по ЛР1 со скринами действий и результатов

LR2

LR3

LR4

LR5

**STRWEB**

LR1

LR2

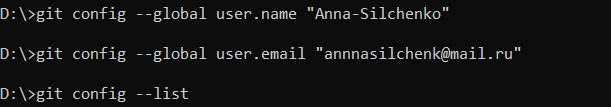
LR3

LR4

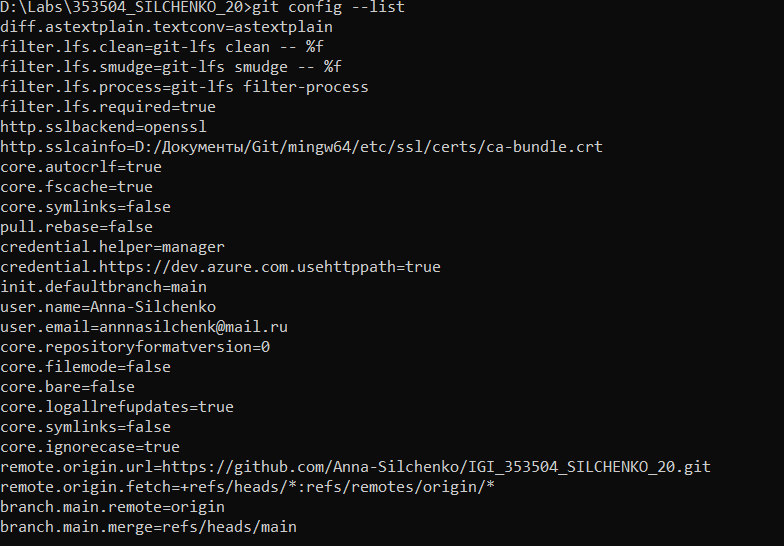
1. *Создайте репозиторий внутри папки* LR1 *с помощью клонирования проекта с сайта github.com (*https://github.com/smartiqaorg/geometric\_lib*)*
2. *Выполните следующие действия, каждый раз проверяя состояние файлов в локальном репозитории до и после выполнения операции:*
   1. сделайте индексацию нескольких файлов в вашем Git-репозитории (git add …),
   2. внесите изменения в файлы,
   3. выполните коммиты к проиндексированным файлам,
   4. добавьте и удалите файлы (один и группу файлов с определенным расширением)
   5. переименуйте файлы
   6. создайте для IGI и STRWEB, а также для каждой ЛР, вложенной в них, отдельные ветки, переместите папки и файлы на соответствующие ветки
   7. выведите историю коммитов: полностью и с ограничением вывода – отобрать только последние, до и после указанной даты, по автору файлов, коммитов (флаги n, since, after, until, before, author, committer, grep) <https://code.mu/ru/tool/git/basis/limit-output-commit-history-flags/> ,
   8. исследуйте флаги для git log: p , stat ,shortstat, name-only, name-status, relative-date, pretty, graph,
   9. найдите разницу в изменениях для каждого коммита
   10. создайте новый файл test.txt и вставьте в него строчку 'номер варианта'. затем сделайте коммит и отмените его
   11. проиндексируйте файл в вашем Git-репозитории, затем отмените его индексацию
   12. внесите изменения в файл в вашем Git-репозитории и затем верните его в исходное состояние.
3. *Выполните следующие действия с удаленным репозиторием:*
   1. создайте удаленный репозиторий (на github.com)
   2. просмотрите информацию о нем, внесите изменения и опять просмотрите (remote, fetch).
   3. измените несколько файлов в вашем проекте и перенесите их в удаленный репозиторий,
   4. переименуйте свой удаленный репозиторий
   5. получите данные из удаленного репозитория
   6. просмотреть историю изменений
   7. создайте дополнительную ветку 1 в удаленном реппозитории, перенесите в нее два файла из другой ветки, удалите один из файлов
   8. создайте дополнительную ветку 2 в удаленном реппозитории, перенесите в нее два файла из вашего проекта, слить изменения в основную ветку, удалить дополнительную ветку 2

# 2 Выполнение работы

Подготовьте рабочее окружение в соответствии с типом вашей операционной системы.

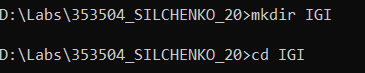


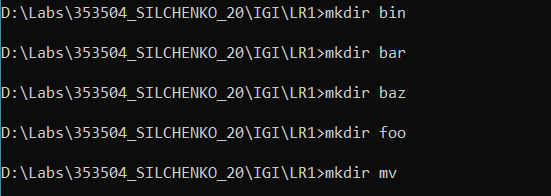
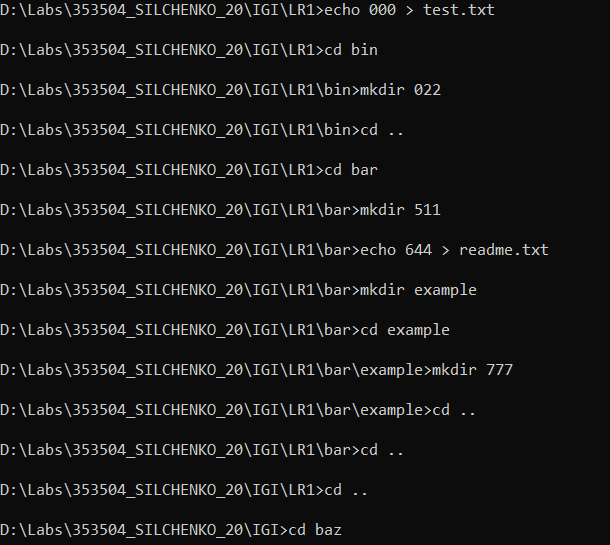


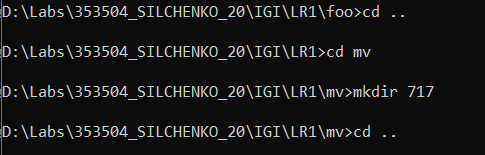
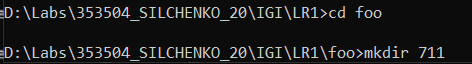
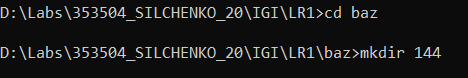


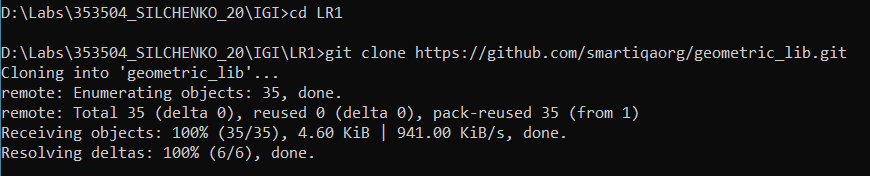
Создайте структуру папок и выполните индивидуальное задание (использовать командную строку) и поместите в репозиторий Git на своем локальном компьютере.

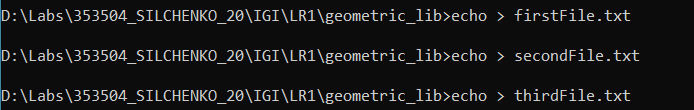
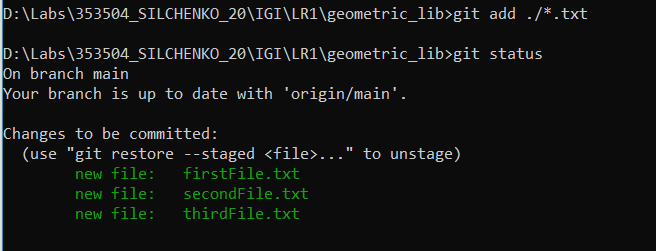


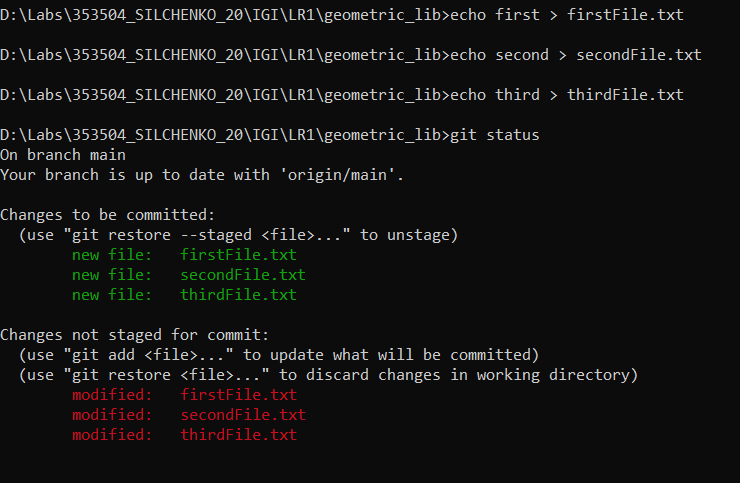
 

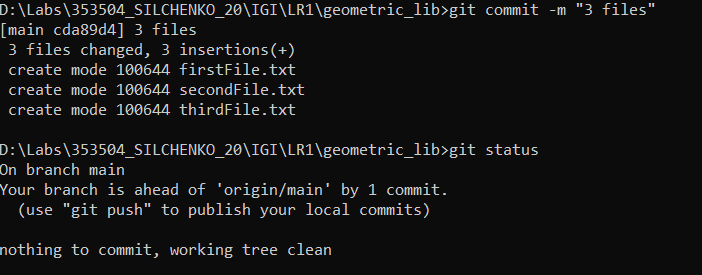
 

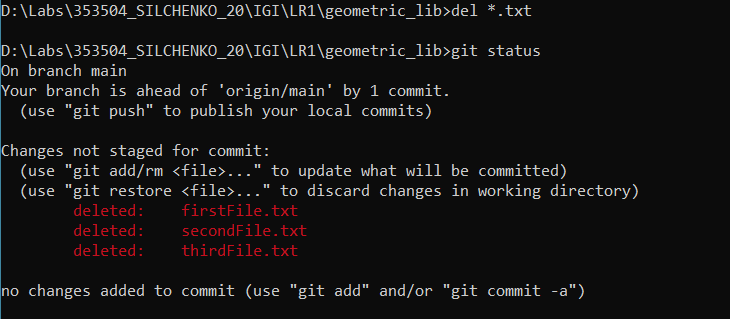


Создайте репозиторий внутри папки LR1 с помощью клонирования проекта с сайта github.com. 

Сделайте индексацию нескольких файлов в вашем Git-репозитории (git add …). 

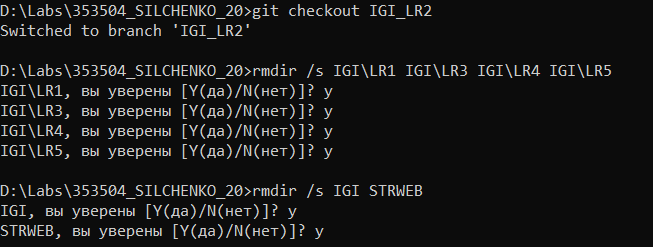
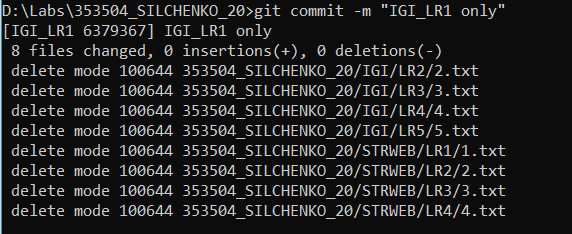
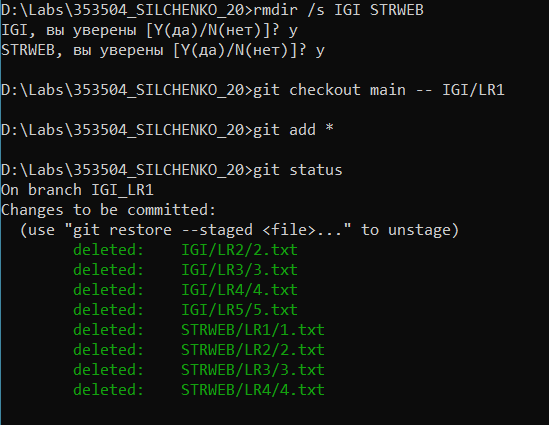
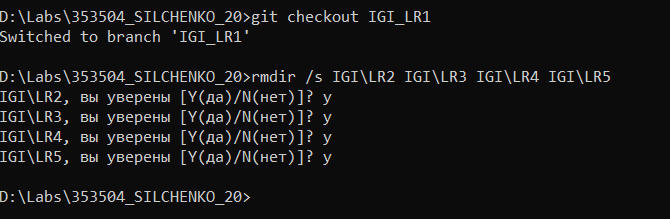
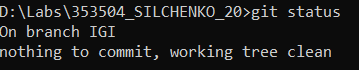
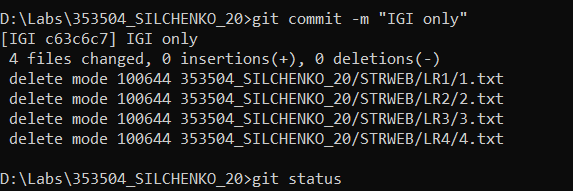
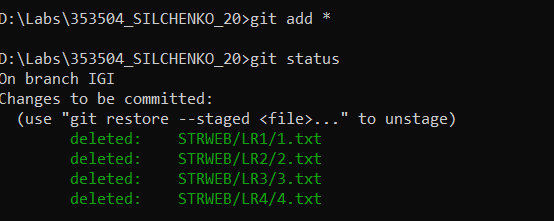
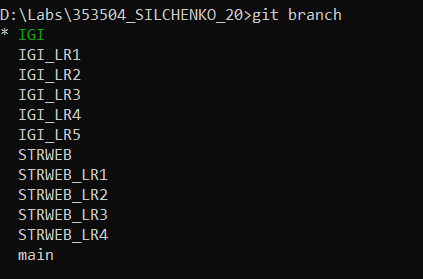
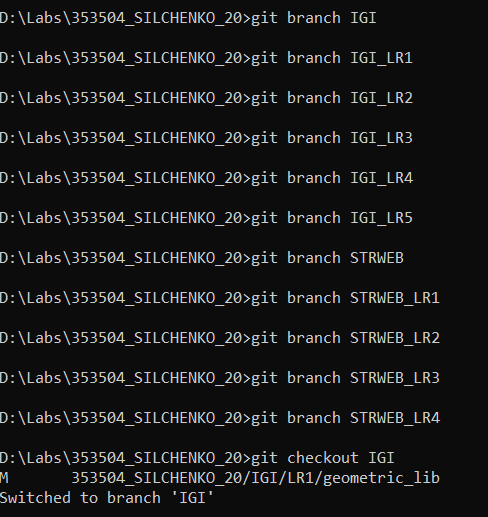
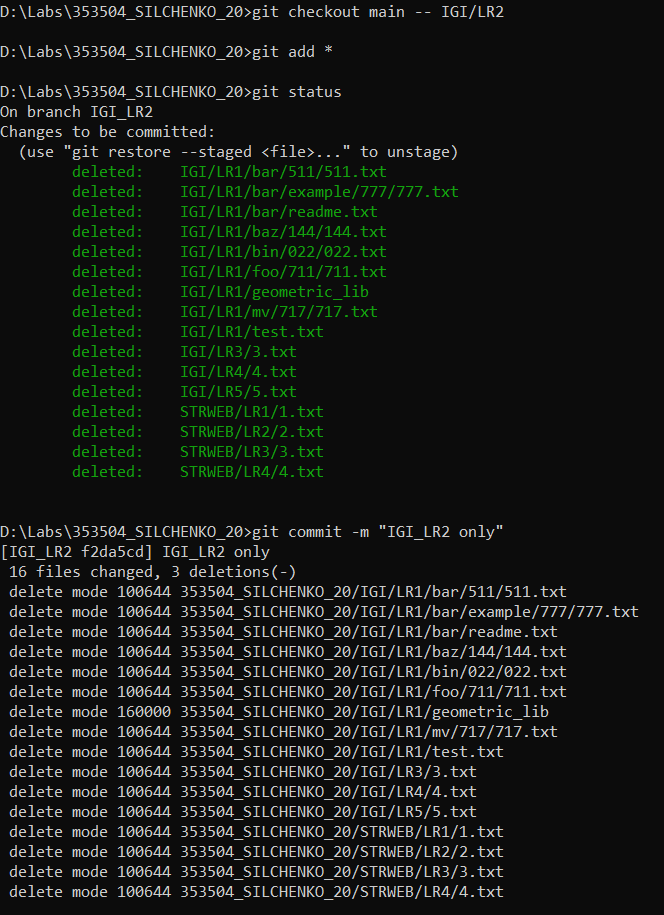
Внесите изменения в файлы. 

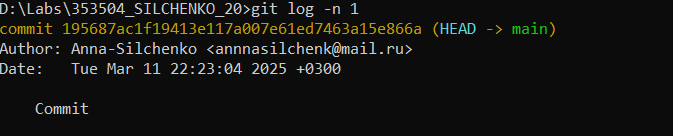
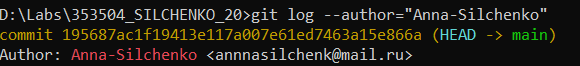
Выполните коммиты к проиндексированным файлам.

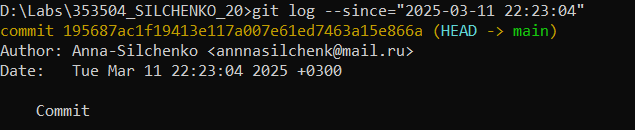
Добавьте и удалите файлы (один и группу файлов с определенным расширением).

Переименуйте файлы.

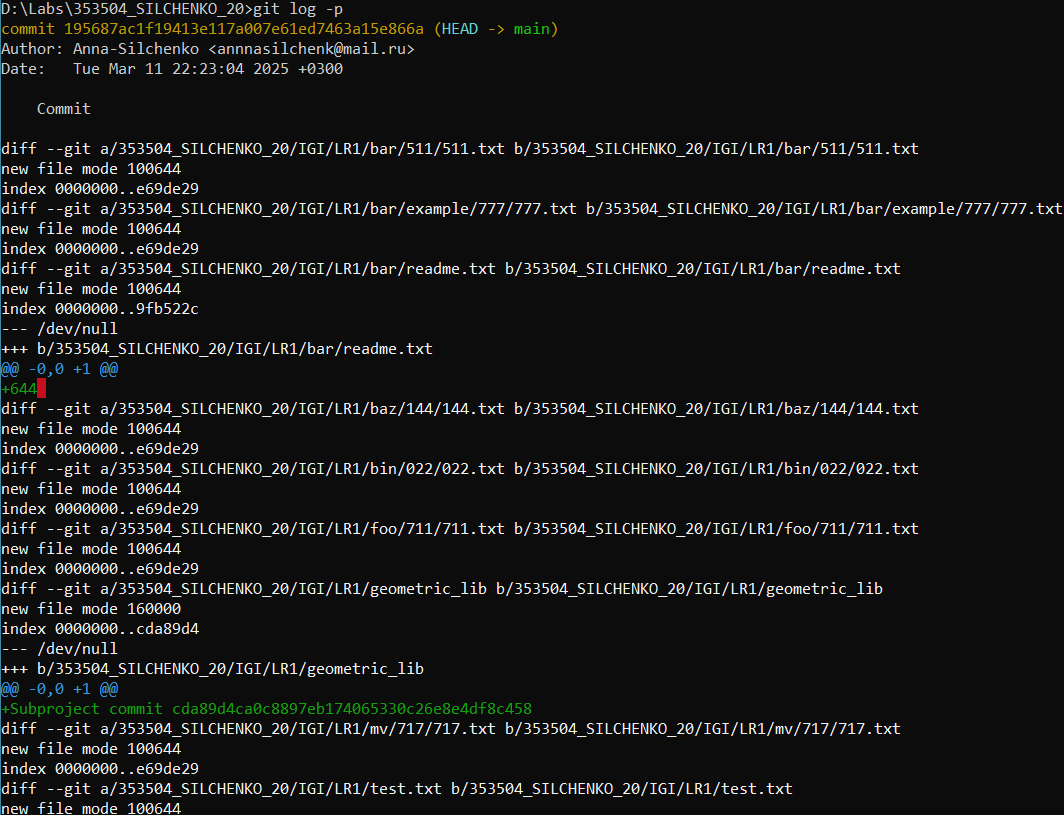
# 

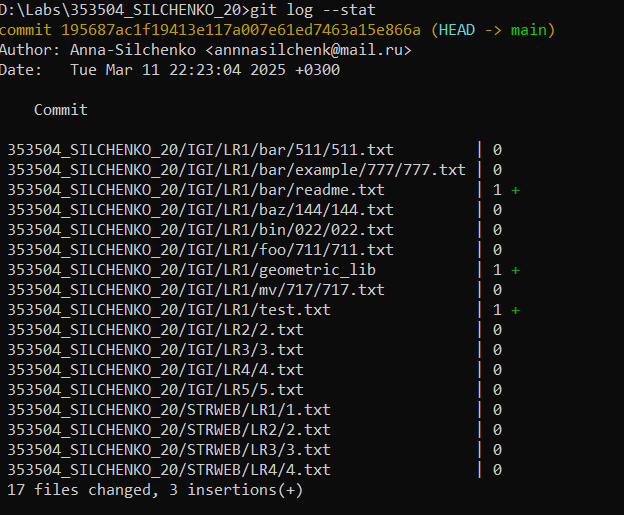
Создайте для IGI и STRWEB, а также для каждой ЛР, вложенной в них, отдельные ветки, переместите папки и файлы на соответствующие ветки (чтобы это реализовать, мне пришлось заполнить конечные для всех папок подпапки файлами .gitkeep , чтобы папки сохранялись при коммите). 

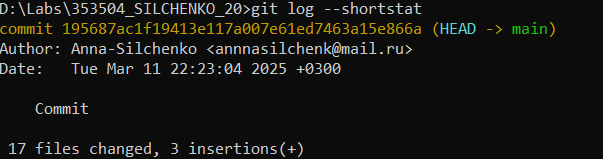
Выведите историю коммитов: полностью и с ограничением вывода – отобрать только последние, до и после указанной даты, по автору файлов, коммитов (флаги n, since, after, until, before, author, committer, grep). 

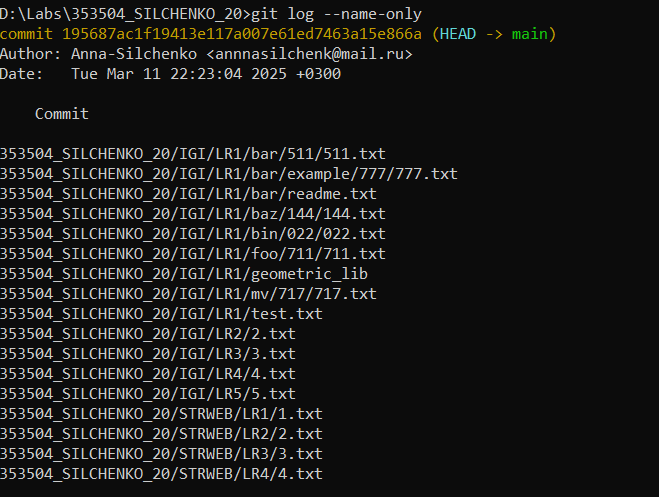


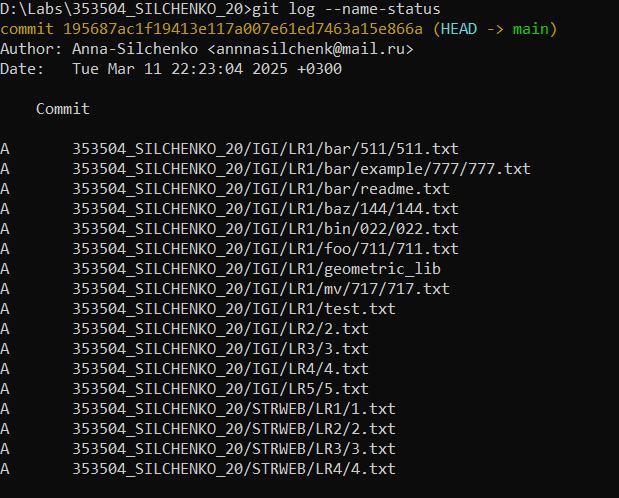


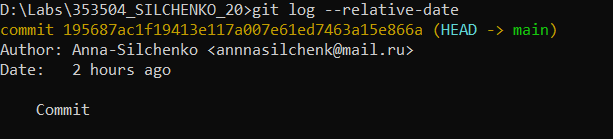
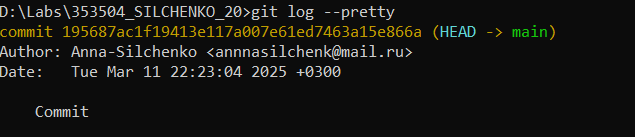
Исследуйте флаги для git log: p , stat ,shortstat, name-only, name-status, relative-date, pretty, graph.

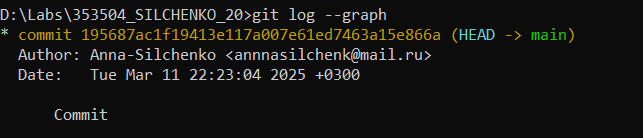




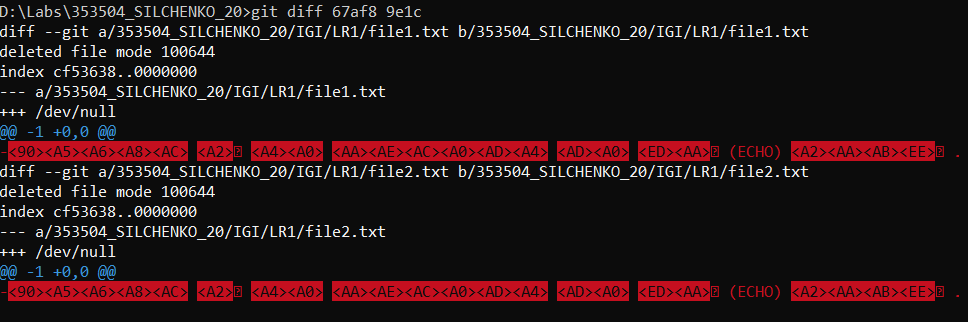




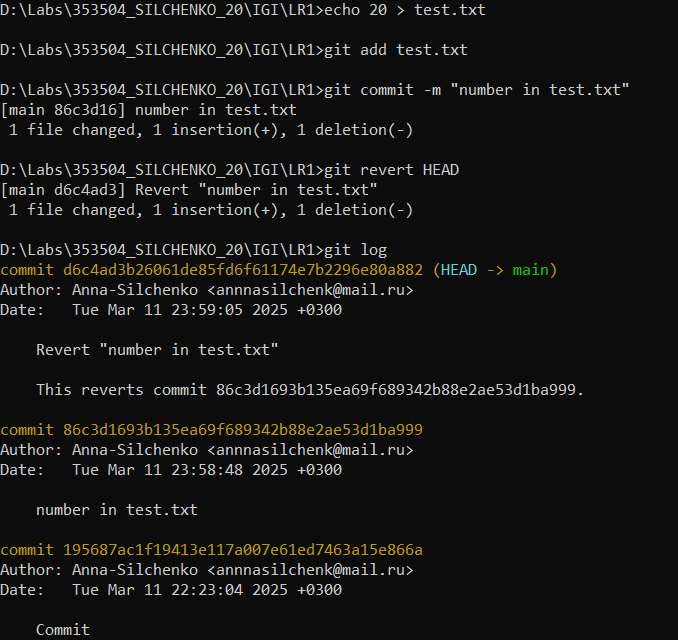
 

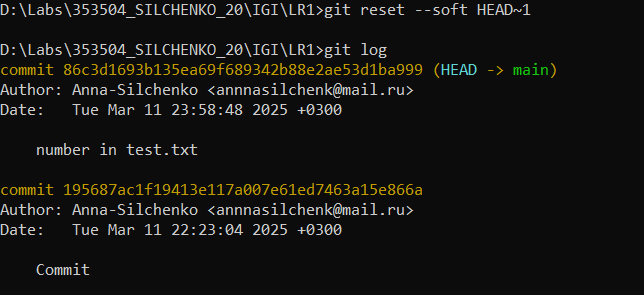


Найдите разницу в изменениях для каждого коммита.

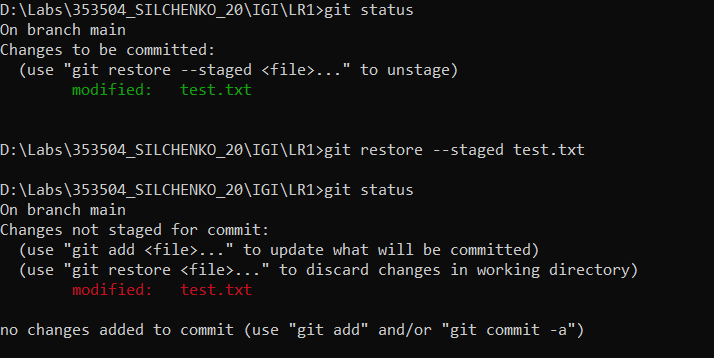


Создайте новый файл test.txt и вставьте в него строчку 'номер варианта'. Затем сделайте коммит и отмените его.

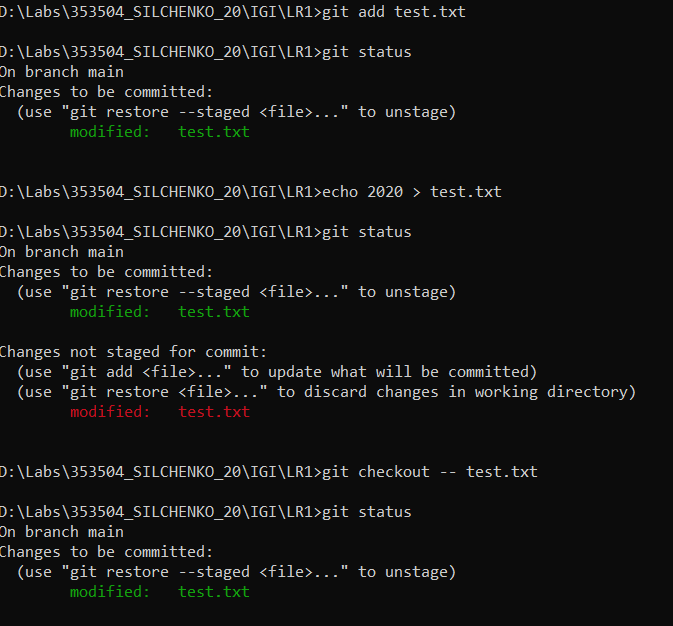




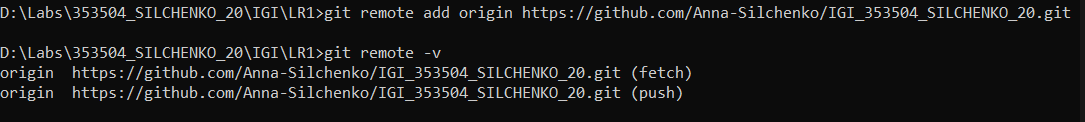
Проиндексируйте файл в вашем Git-репозитории, затем отмените его индексацию.



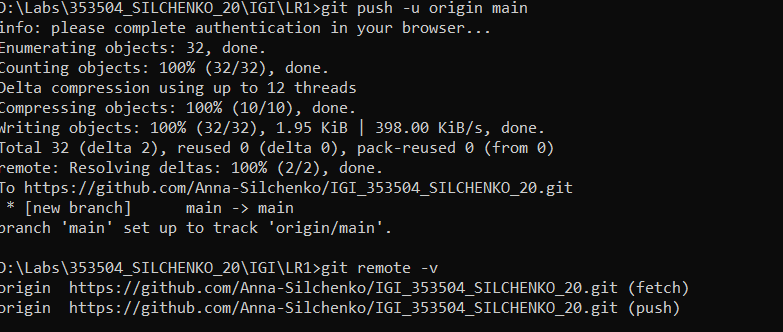
Внесите изменения в файл в вашем Git-репозитории и затем верните его в исходное состояние.

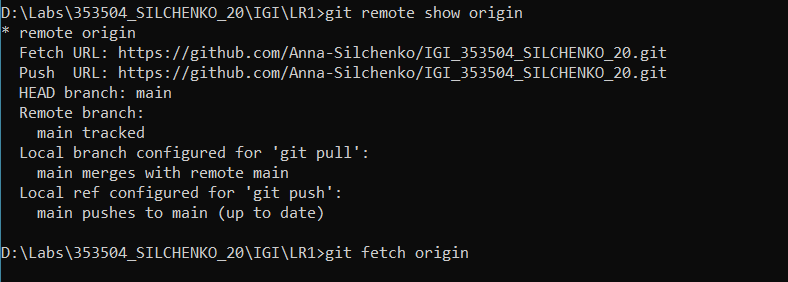


Создайте удаленный репозиторий (на github.com).

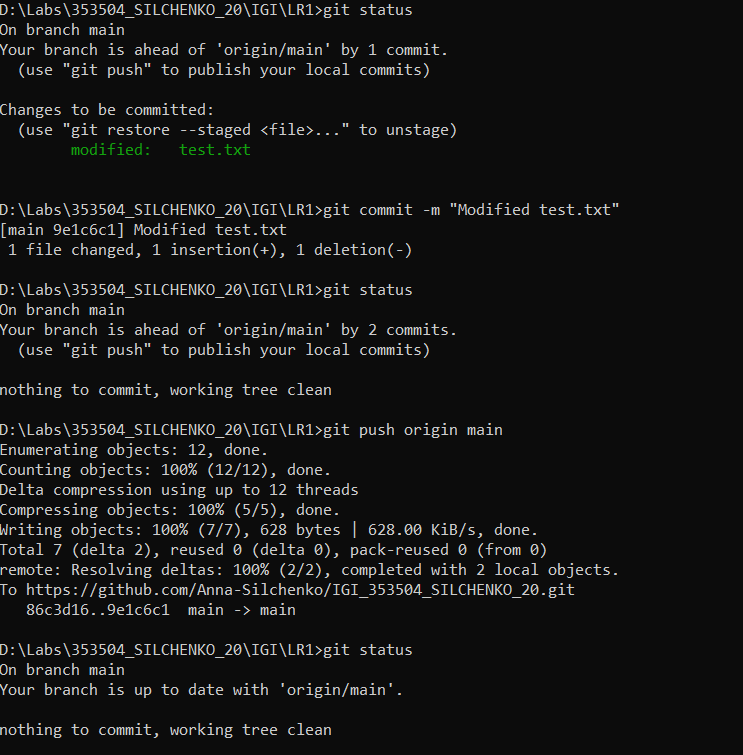


Просмотрите информацию о нем, внесите изменения и опять просмотрите (remote, fetch).

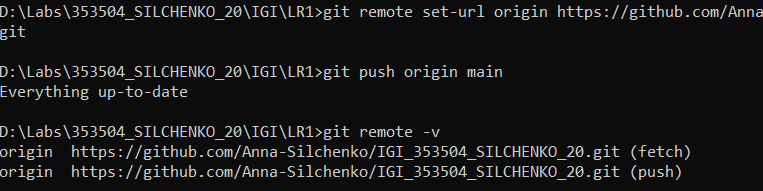




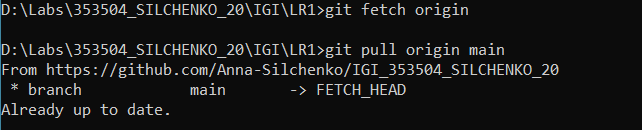
Измените несколько файлов в вашем проекте и перенесите их в удаленный репозиторий.

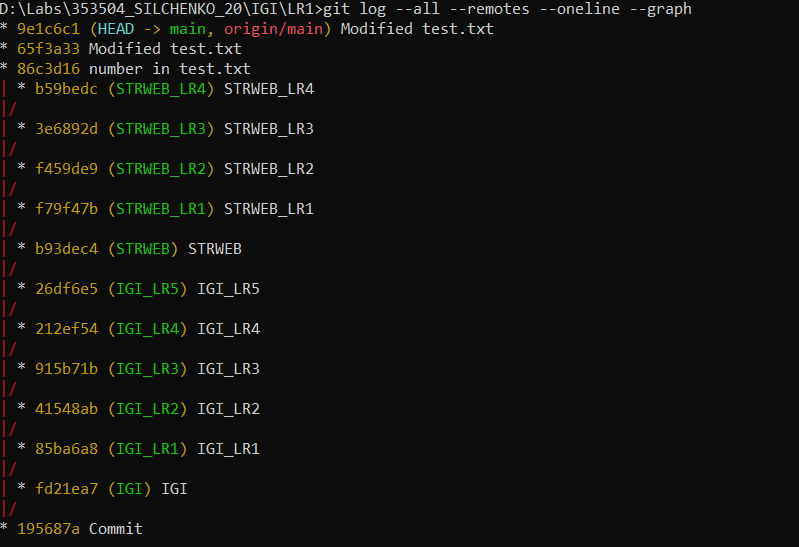


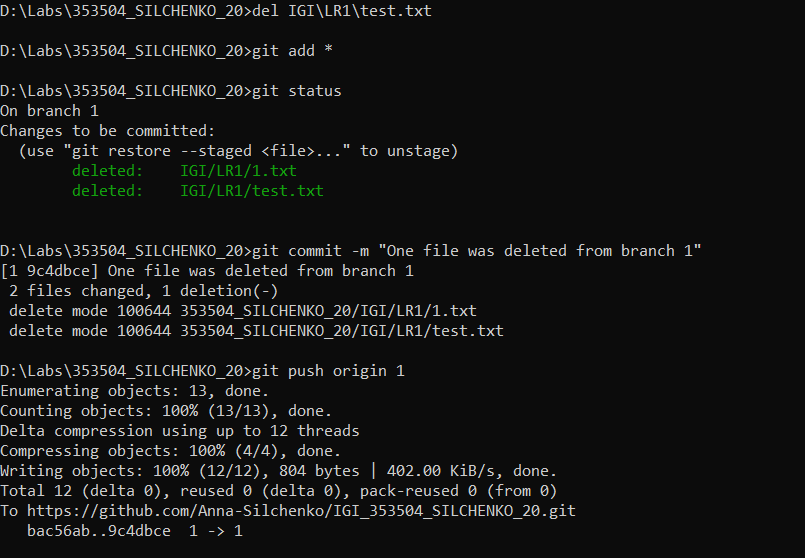
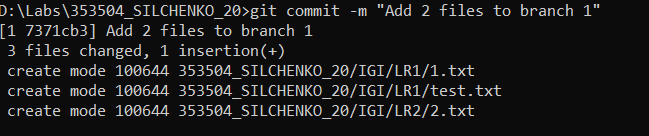
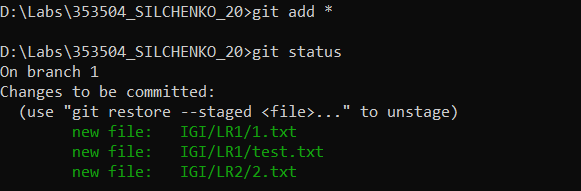
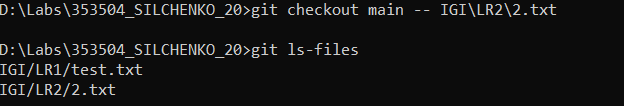
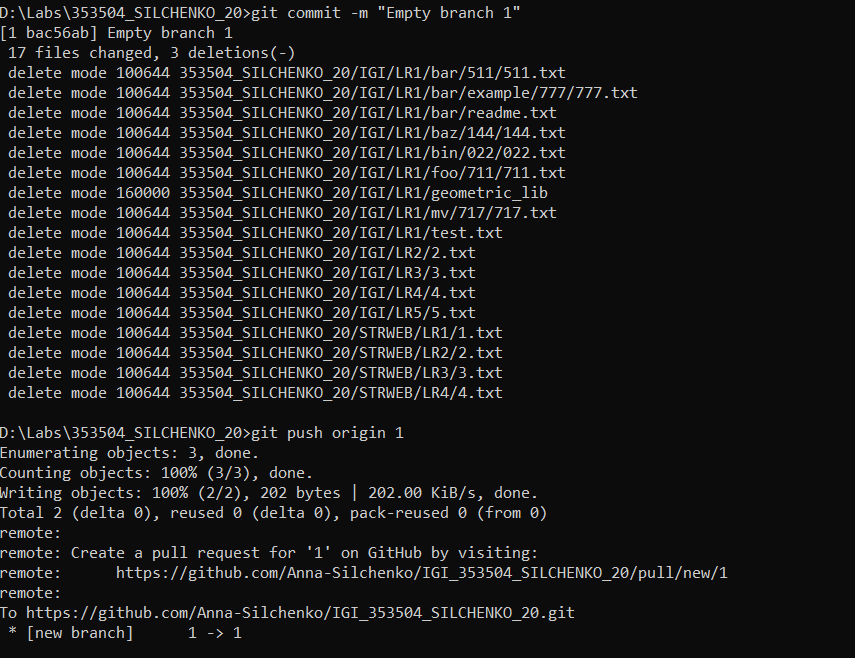
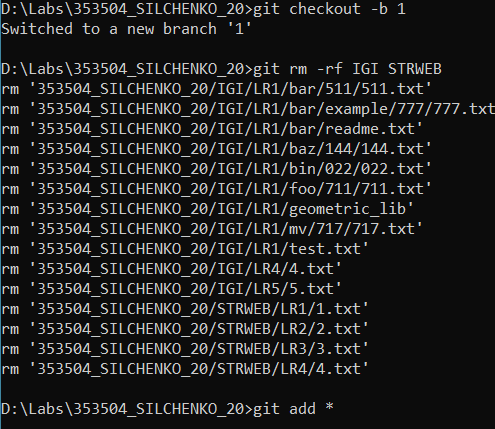
Переименуйте свой удаленный репозиторий.

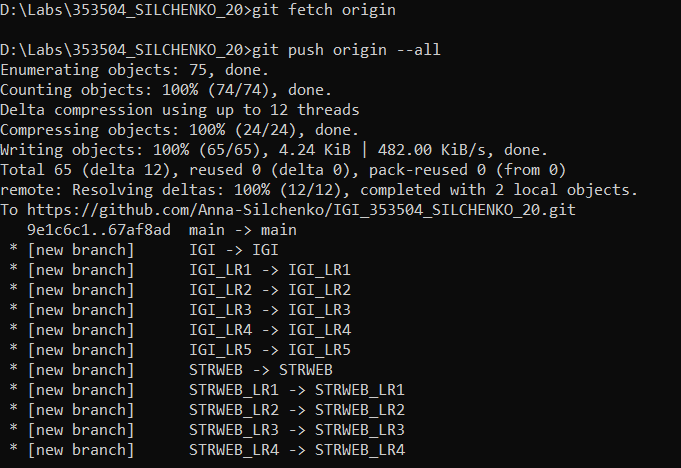
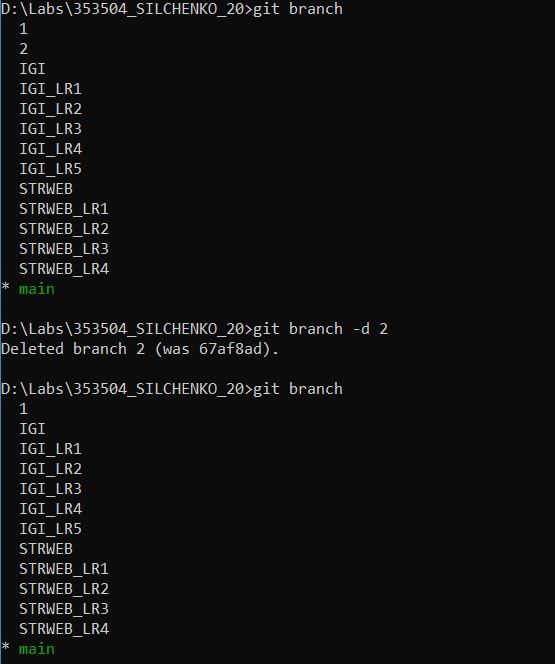
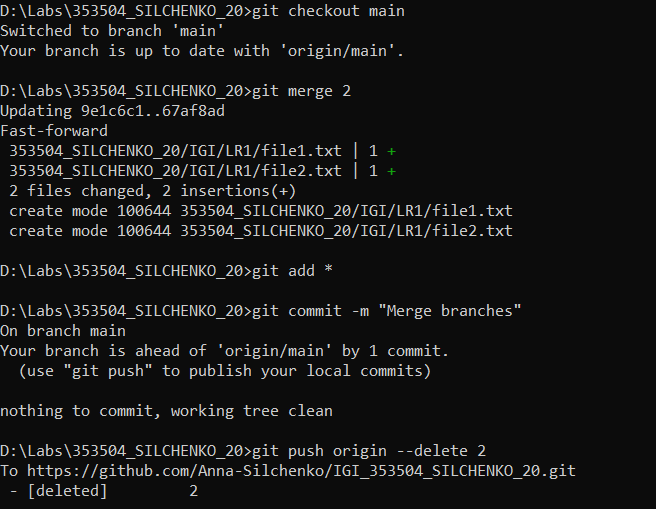
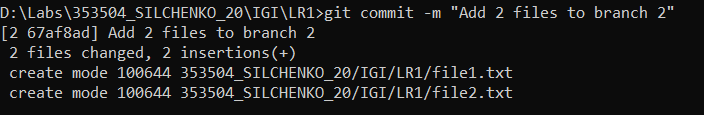
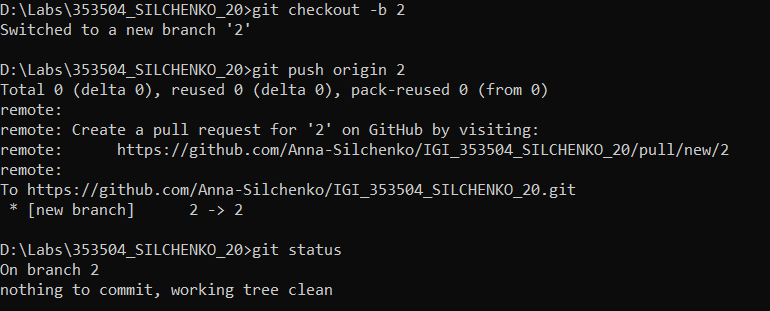


Получите данные из удаленного репозитория



Просмотреть историю изменений.

Создайте дополнительную ветку 1 в удаленном репозитории, перенесите в нее два файла из другой ветки, удалите один из файлов.

Cоздайте дополнительную ветку 2 в удаленном репозитории, перенесите в нее два файла из вашего проекта, слить изменения в основную ветку, удалить дополнительную ветку 2.

**ВЫВОД**

В ходе выполнения данной лабораторной работы мы познакомились с возможностями системы контроля версий Git, изучили основные команды по работе с ней, получили практические навыки, работая с локальными и удаленными репозиториями.