Лабораторная работа №1

**Тема**: Система контроля версий Git.

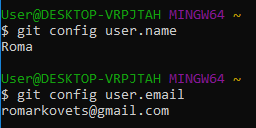
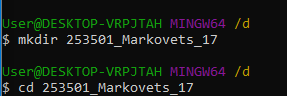
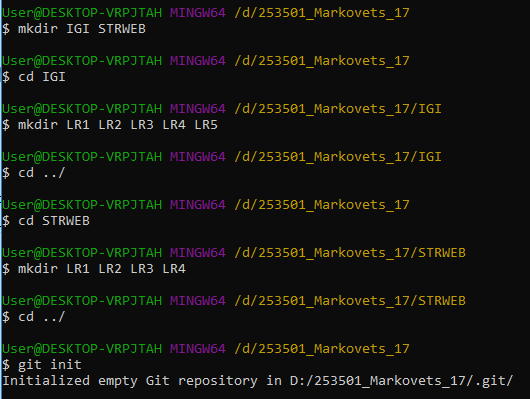
**Цель**: Познакомиться с возможностями и получить практические навыки работы с системой контроля версий Git.

**Cсылки**:

1. <https://smartiqa.ru/courses/git> - подробное описание : теория и практика
2. <https://code.mu/ru/tool/git/> - практические примеры
3. <https://learngitbranching.js.org/>
4. <https://githowto.com/ru>
5. <https://git-scm.com/book/ru/v2/>
6. <https://www.udemy.com/course/git-expert-4-hours/>

Для защиты ЛР необходимо оформить Отчет со скринами кода в командной строке и результатов его выполнения

**Задание**:

1. *Подготовьте рабочее окружение в соответствии с типом вашей операционной системы(*[*https://smartiqa.ru/courses/git/lesson-1*](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

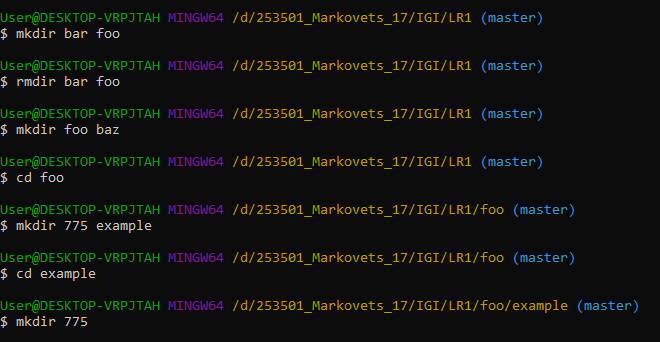
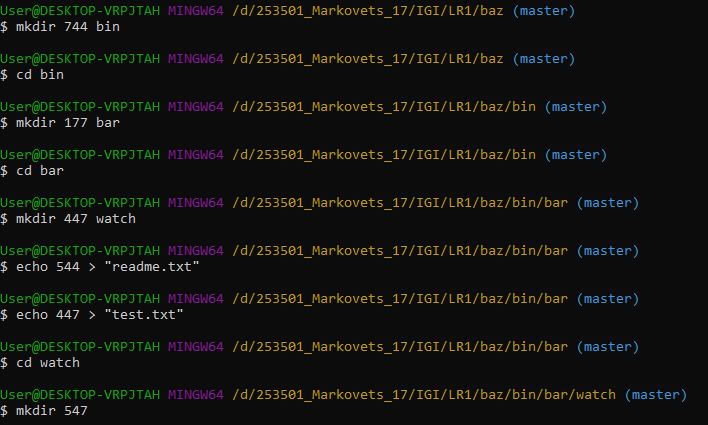
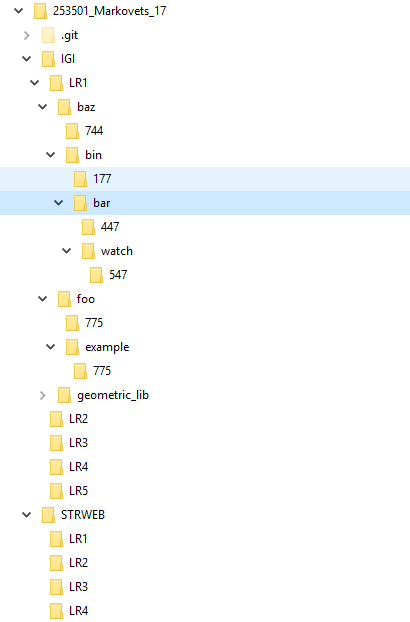
LR5

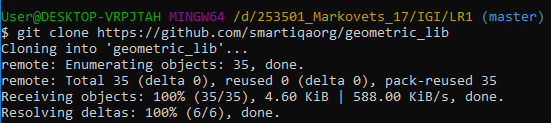
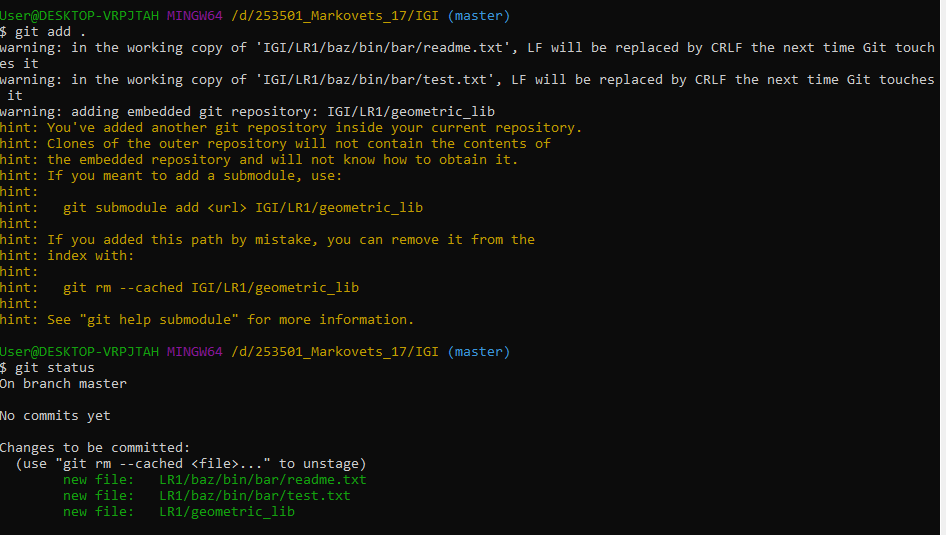
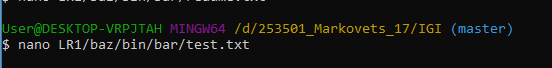
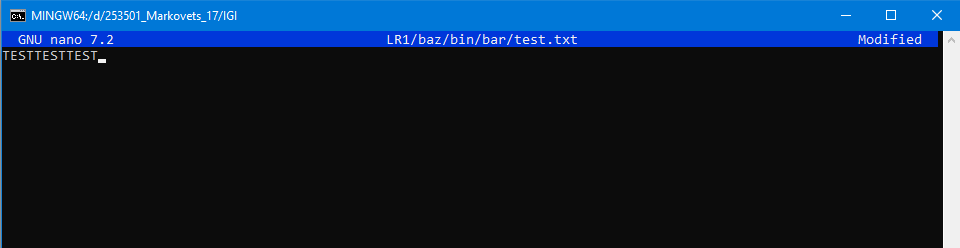
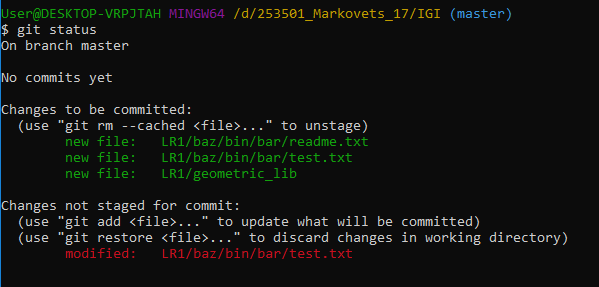
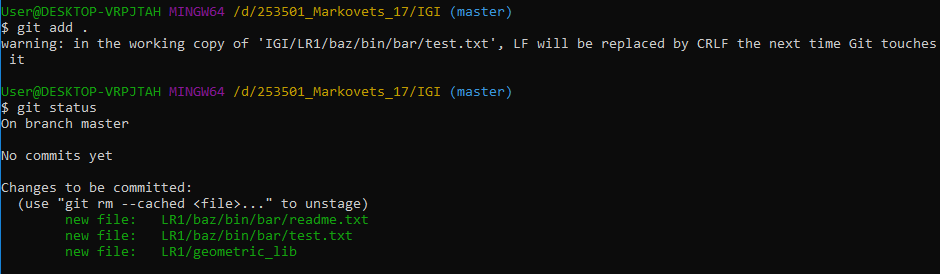
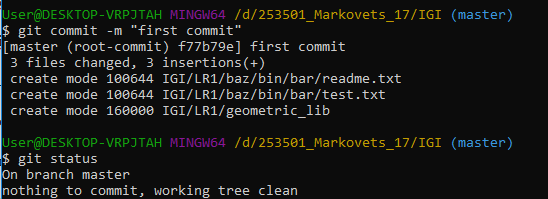
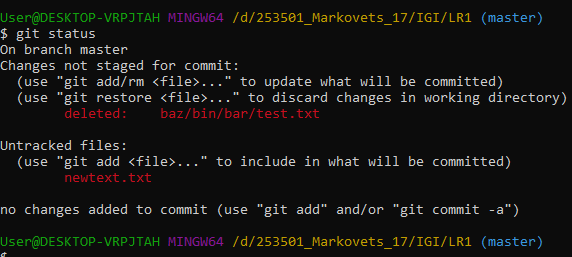
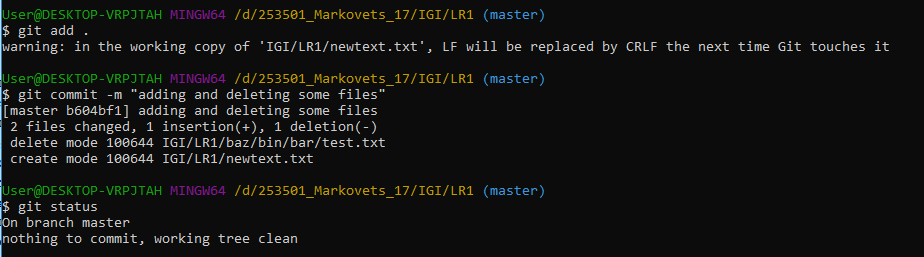
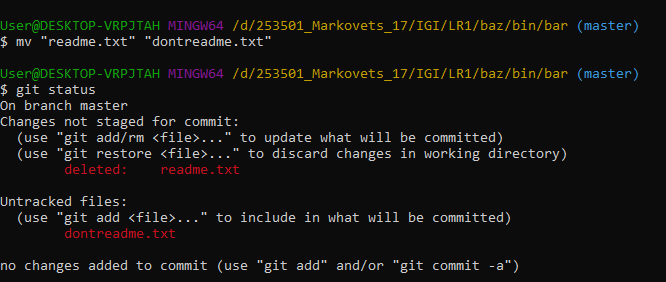
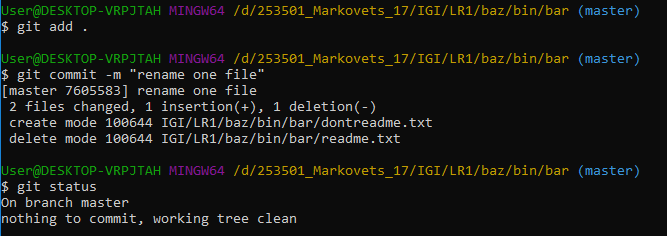
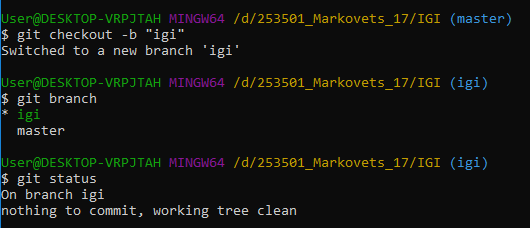
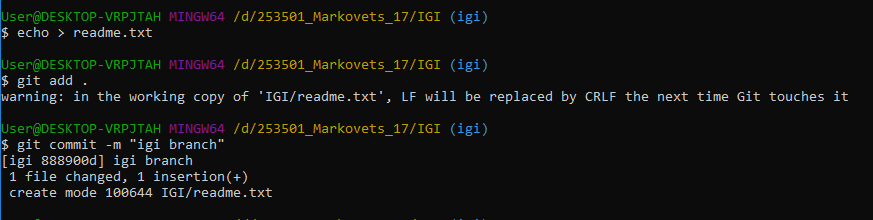
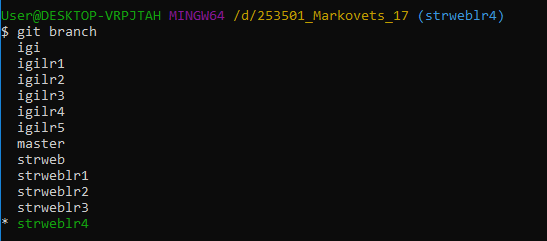
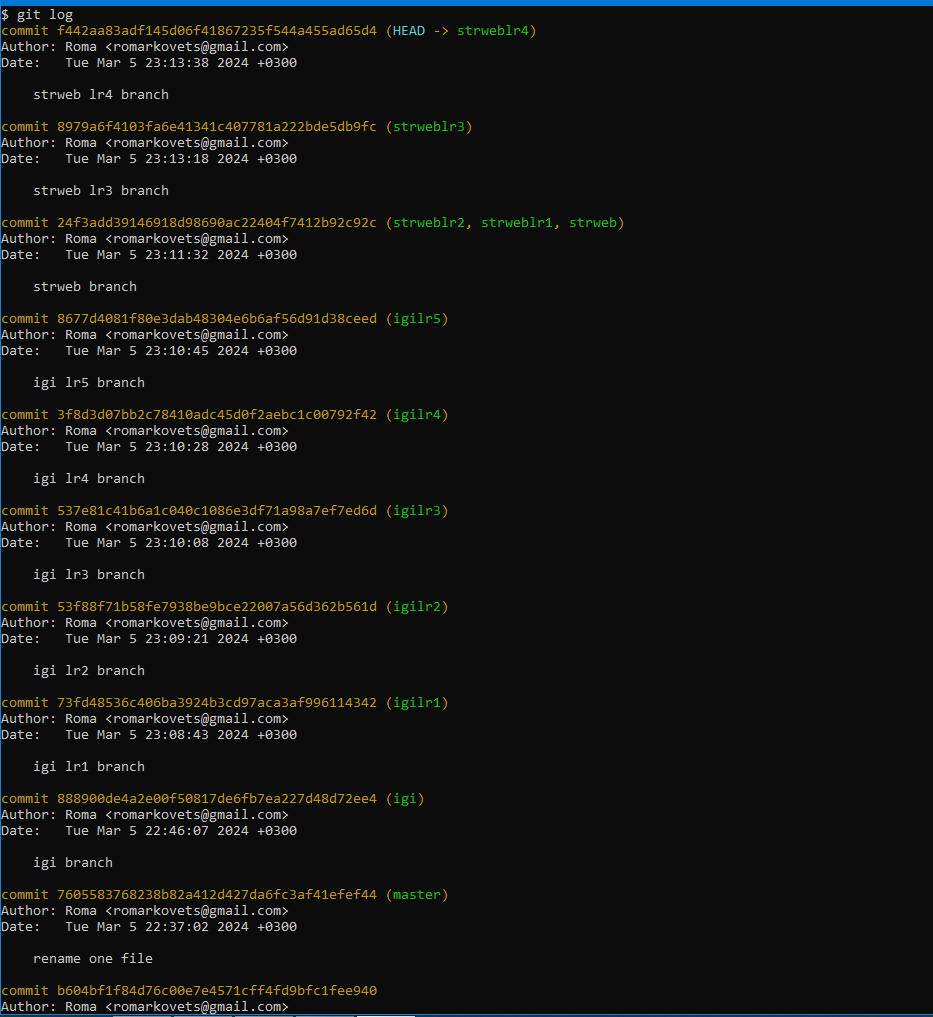
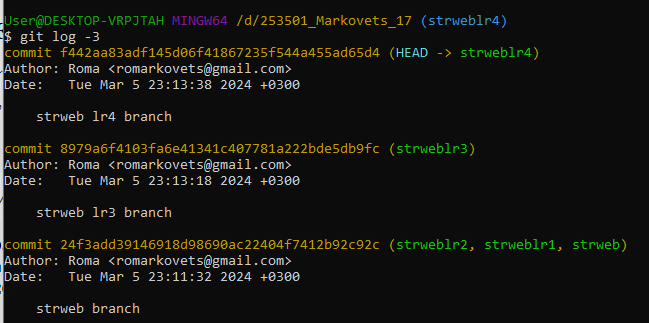
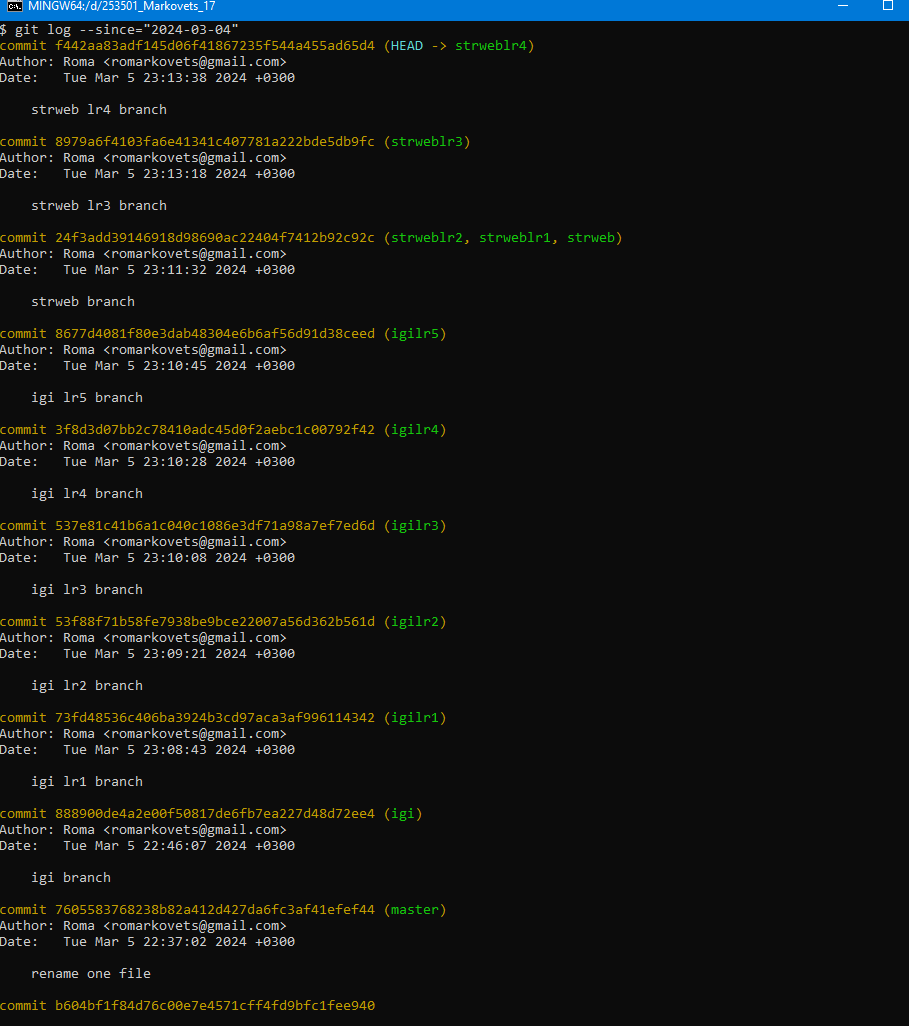
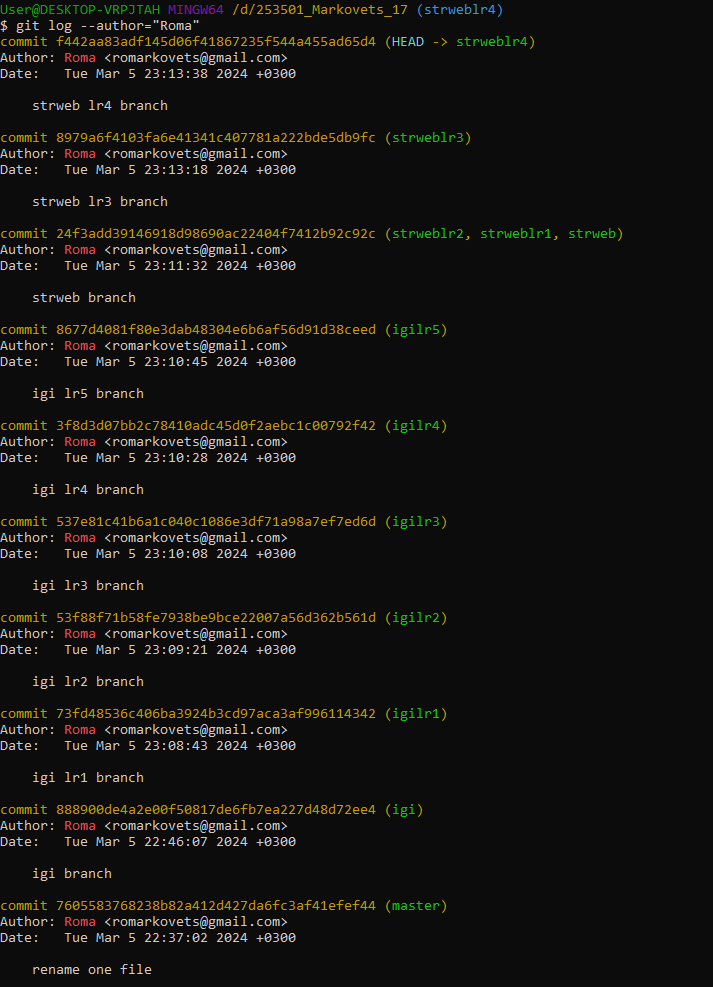
**STRWEB**

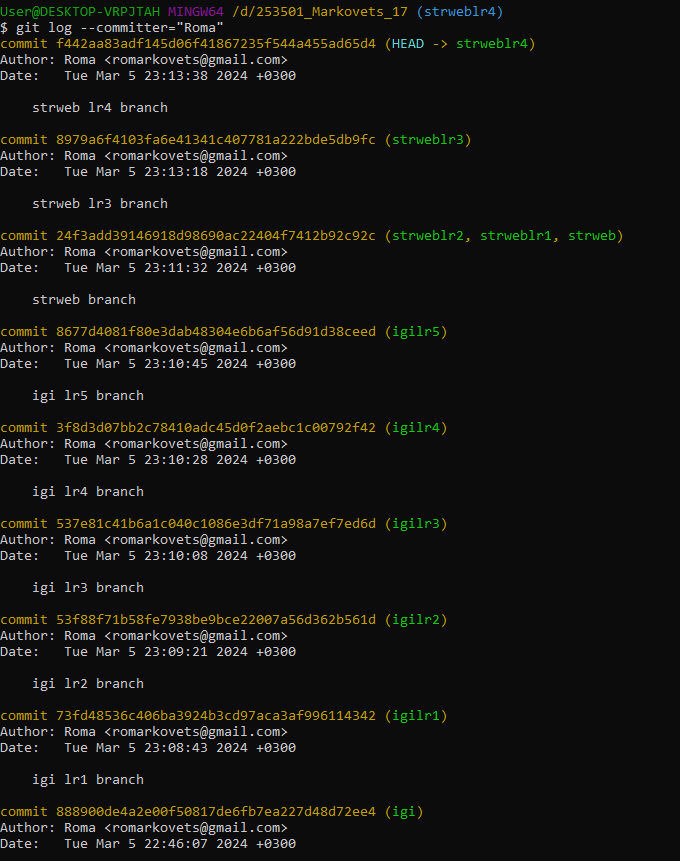
LR1

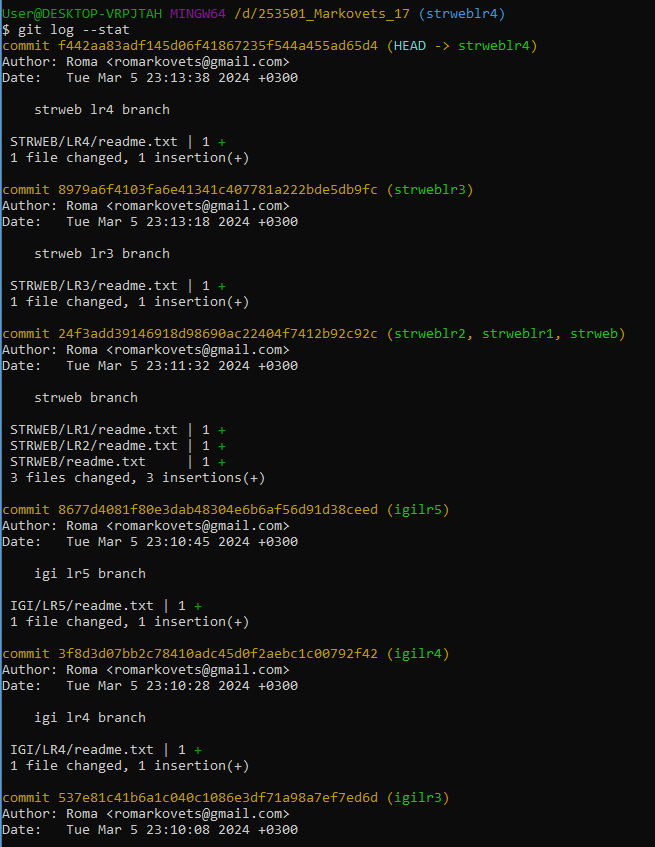
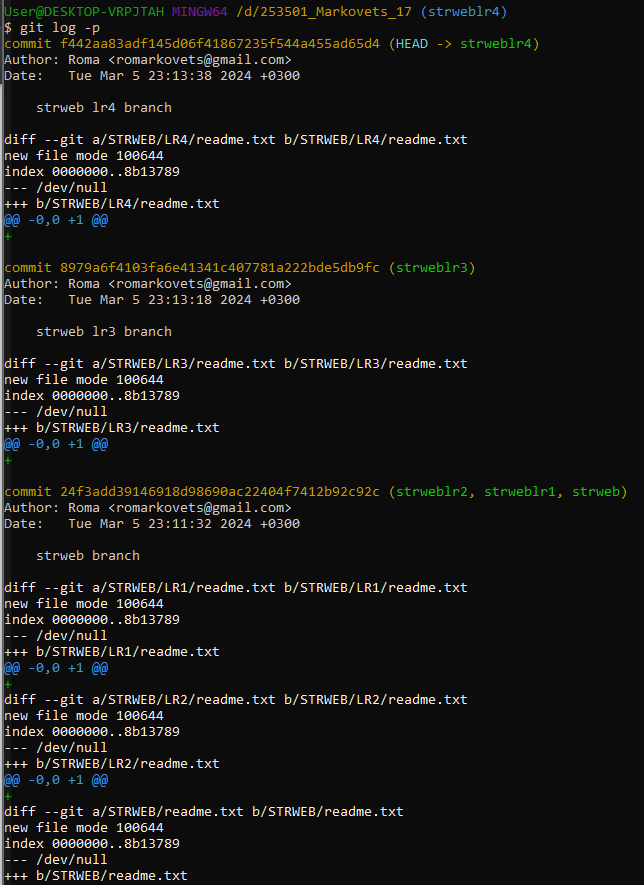
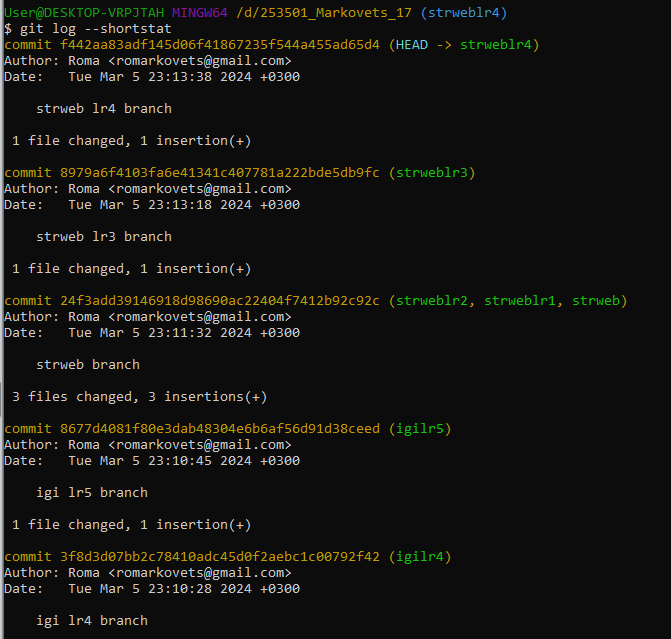
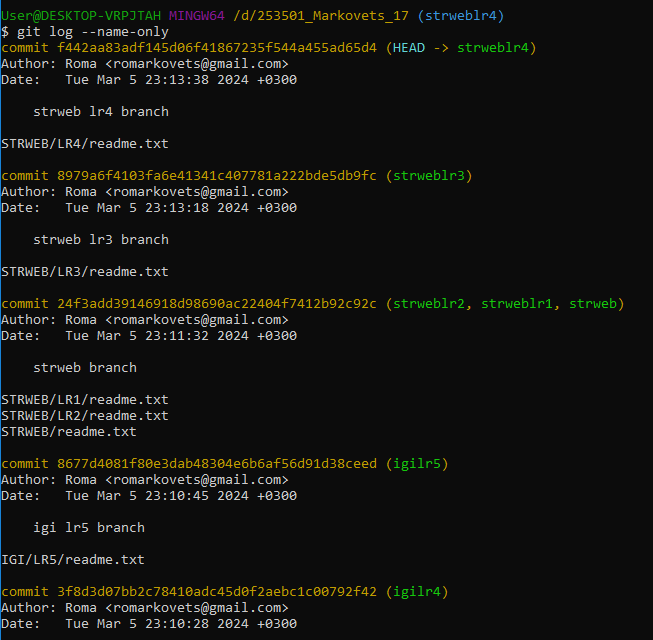
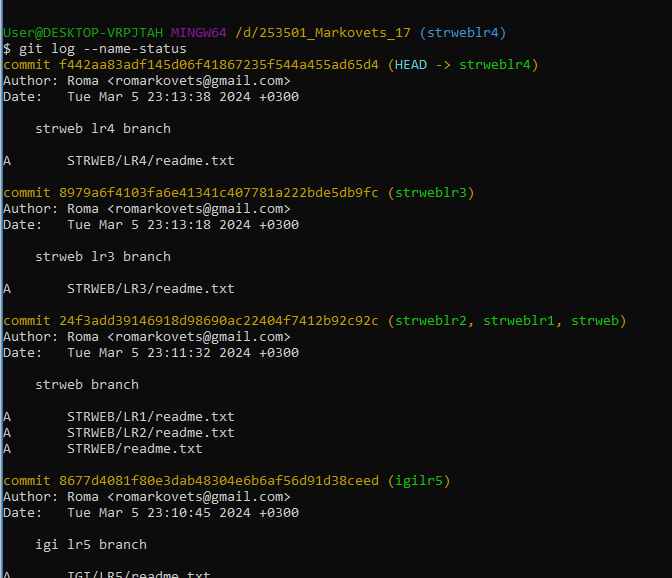
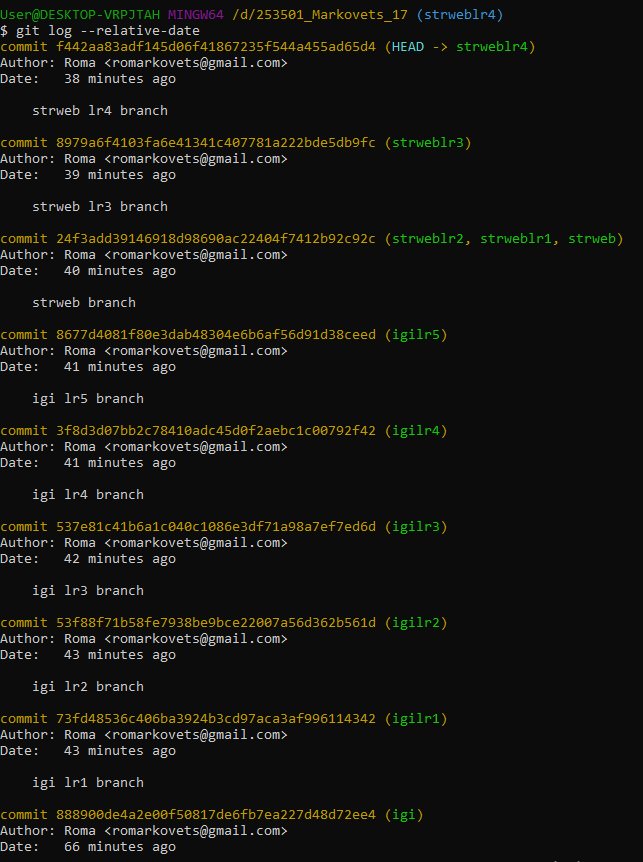
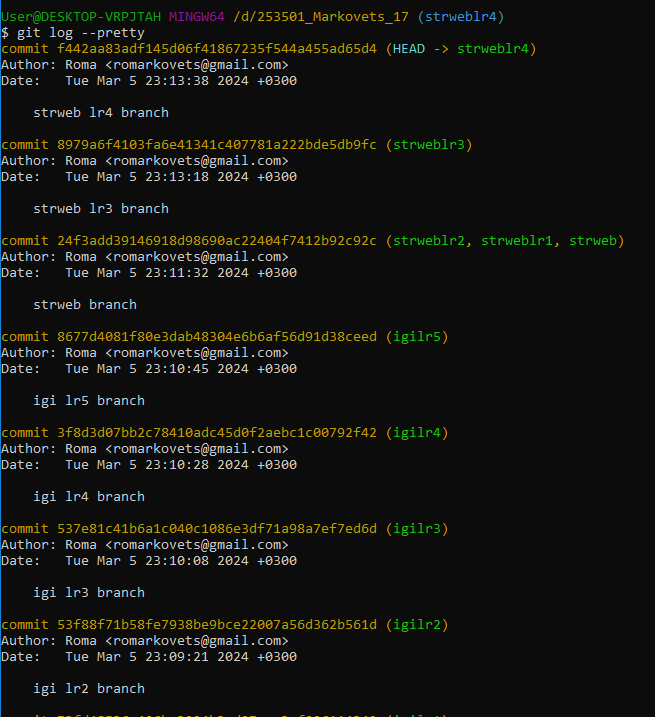
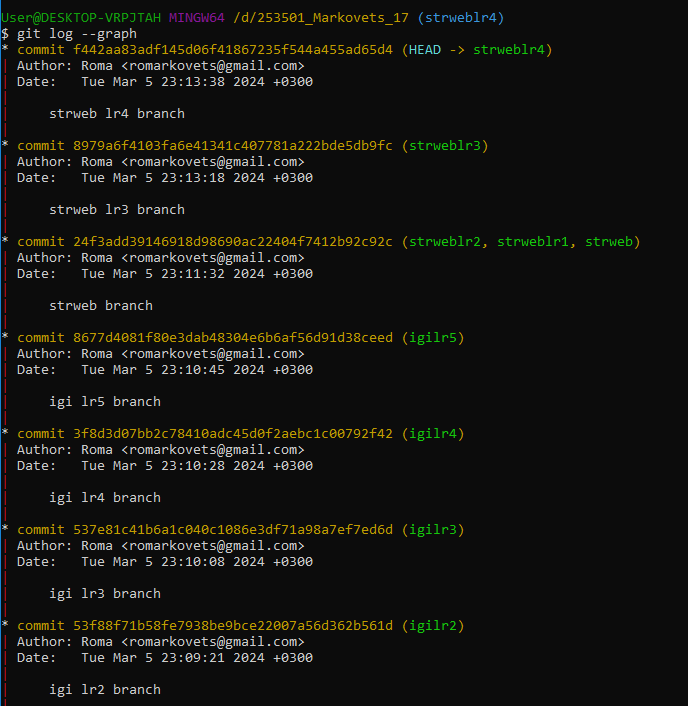
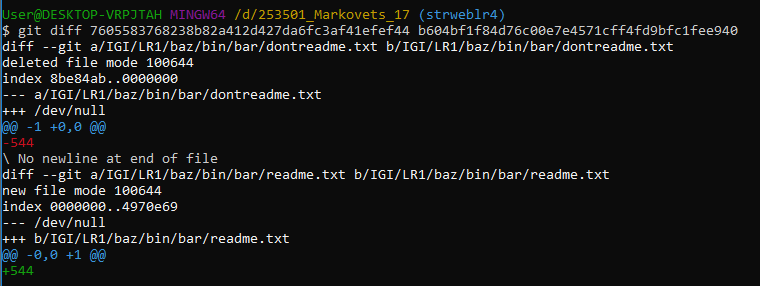
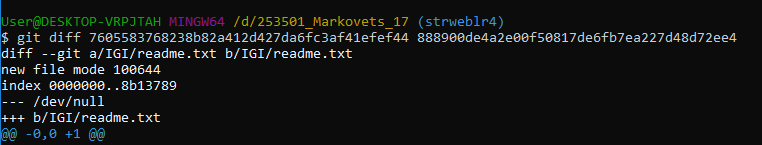
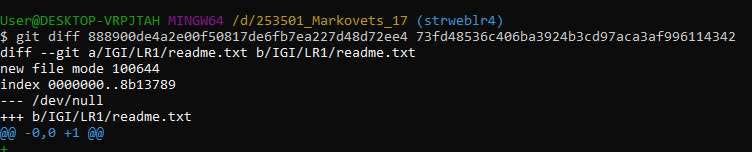
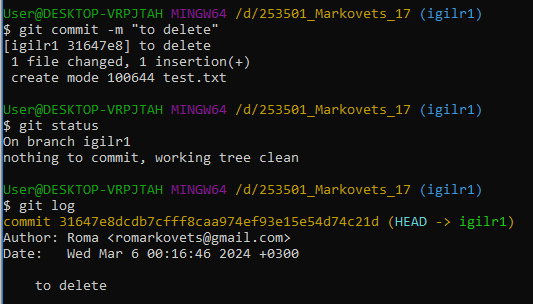
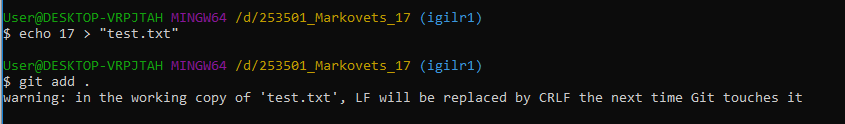
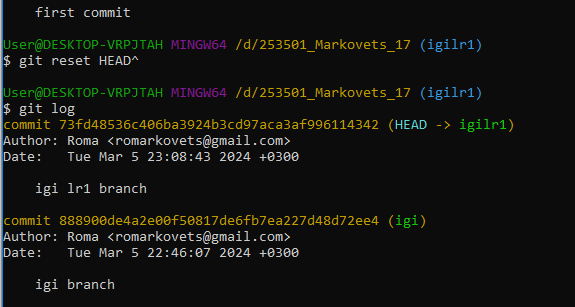
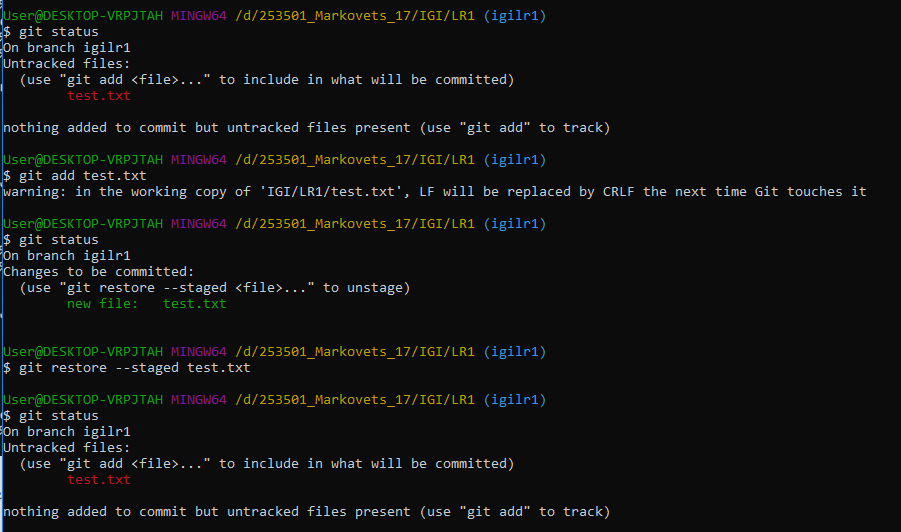
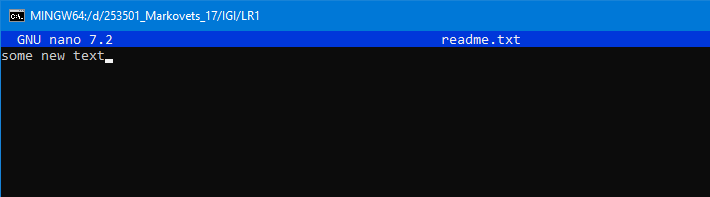
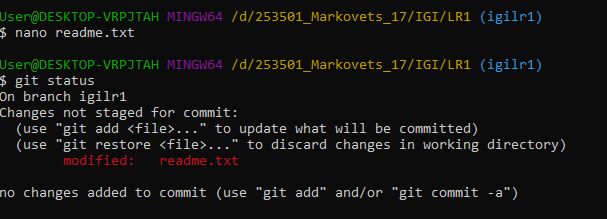
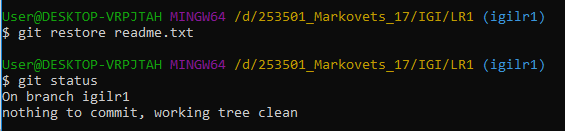
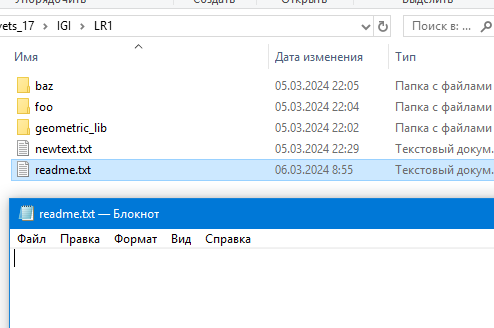
LR2

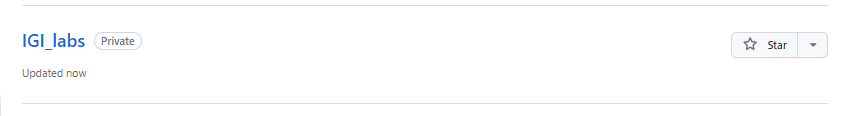
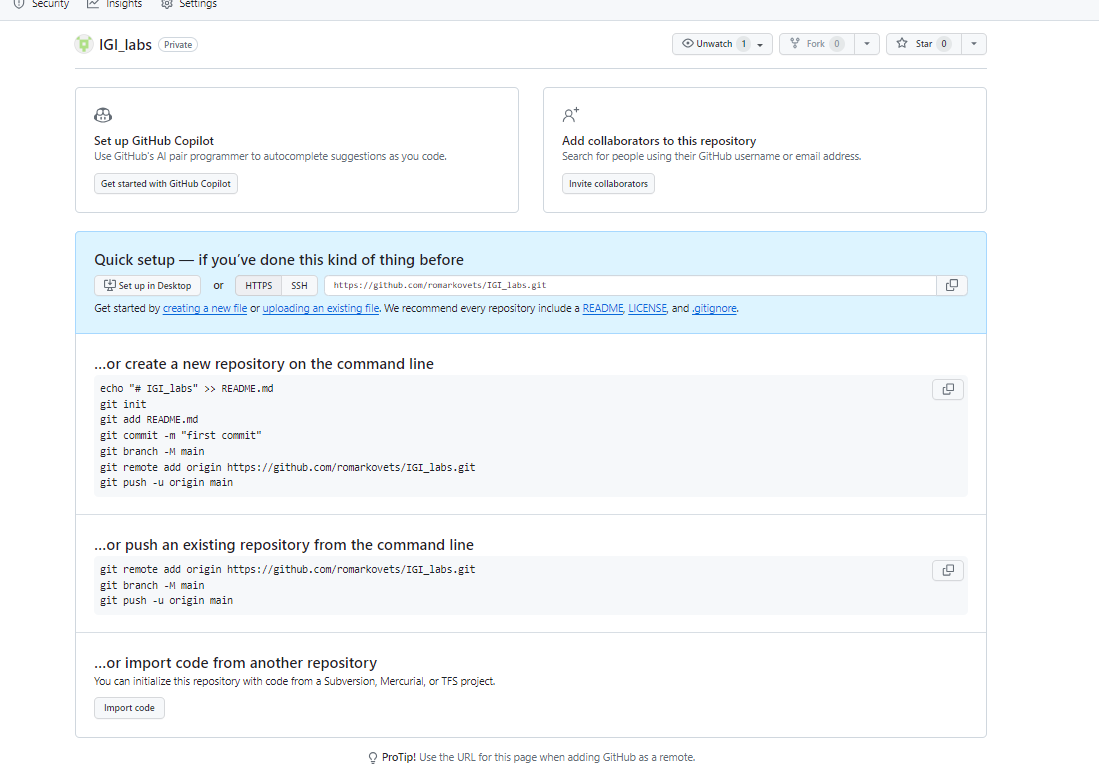
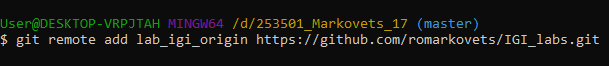
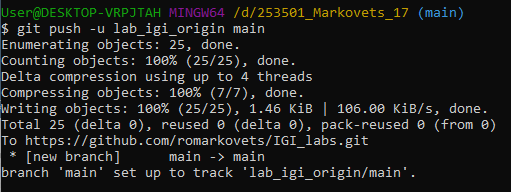
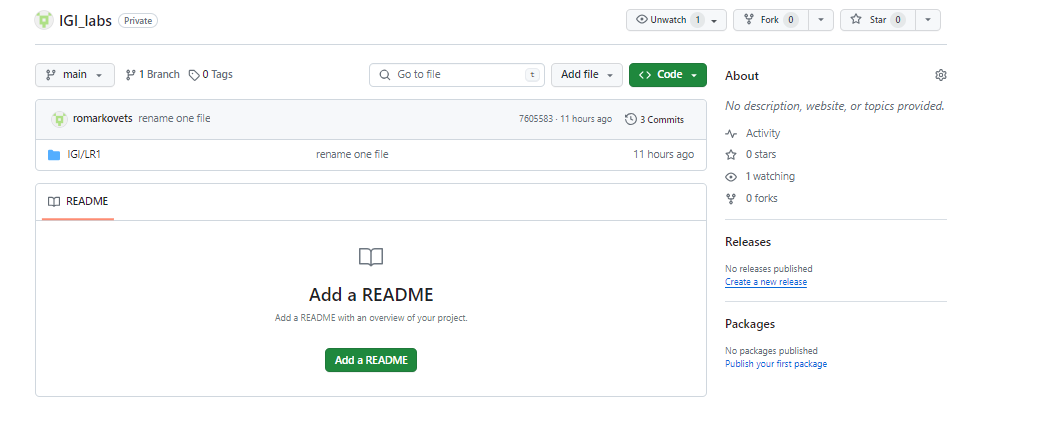
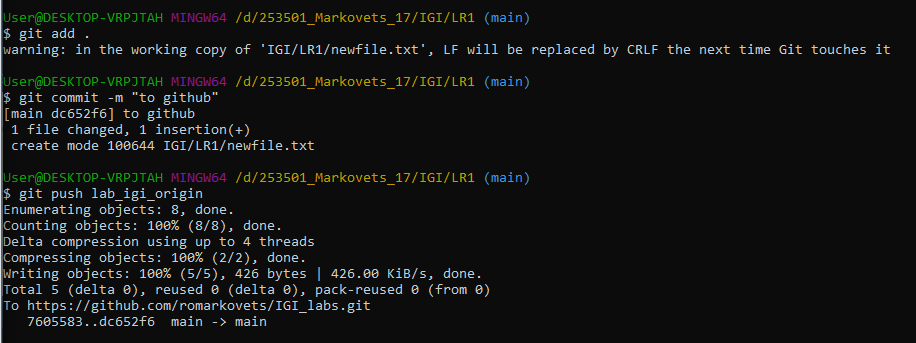
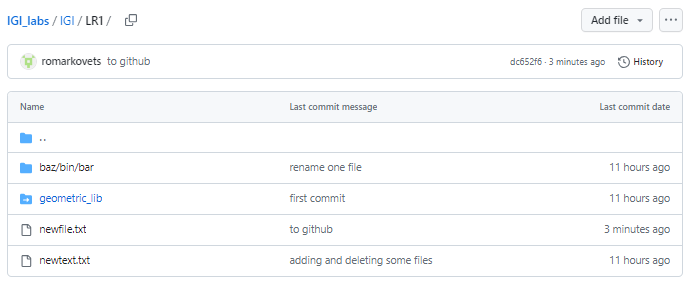
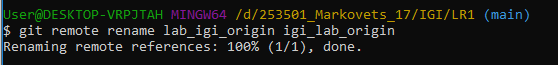
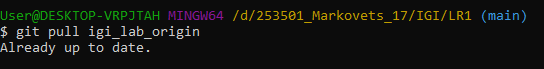
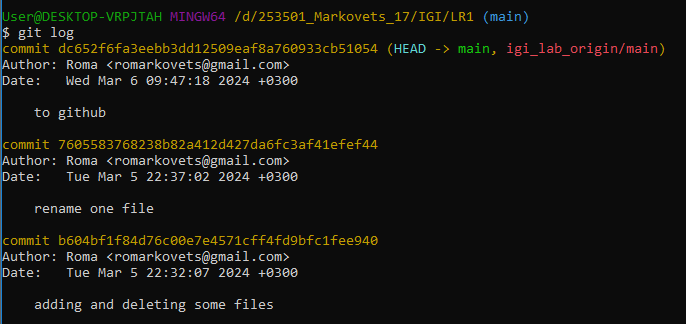
LR3

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  
        
        
      дальше аналогично для каждой 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/> ,  
      все коммиты:  
      Последние 3 коммита:  
        
      Коммиты строго позже 4 марта 2024:  
        
      Коммиты до 4 марта 2024 года включительно (их нет):   
      Коммиты над файлами с автором Roma:  
      

Коммиты от коммитера Roma:  


* 1. исследуйте флаги для git log: p , stat ,shortstat, name-only, name-status, relative-date, pretty, graph,   
       
       
       
       
       
       
     
  2. найдите разницу в изменениях для каждого коммита  
       
       
     
  3. создайте новый файл test.txt и вставьте в него строчку 'номер варианта'. затем сделайте коммит и отмените его  
       
       
     или же git revert <hash>
  4. проиндексируйте файл в вашем Git-репозитории, затем отмените его индексацию  
     
  5. внесите изменения в файл в вашем Git-репозитории и затем верните его в исходное состояние.  
       
       
       
       
     

1. *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  
      

Индивидуальное задание:

Структура папок

Примечание:

**bar/708** – в папке **bar** нужно расположить папку **708**

**test.txt/007** – в файле **test.txt** нужно поместить текст **007**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вариант** | **Структура папок** | **Вариант** | | **Структура папок** |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |