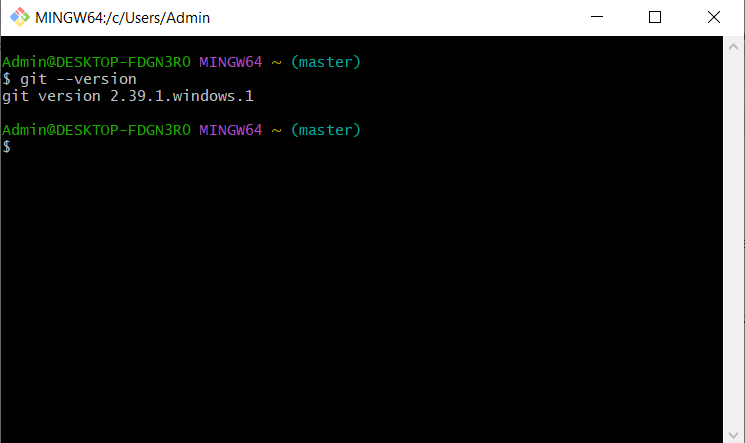
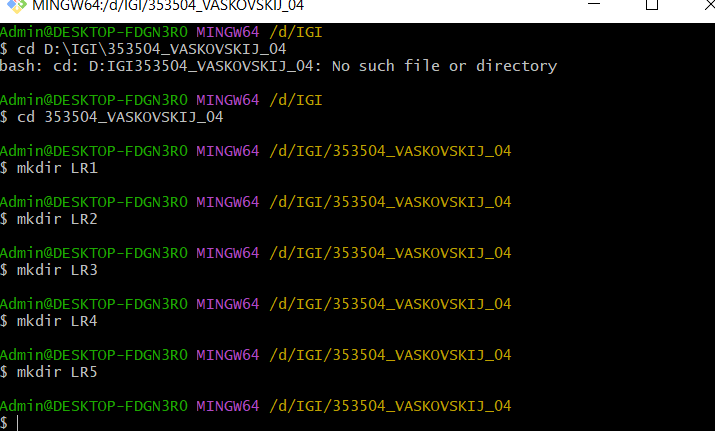
Васьковский Захар 353504 (Вариант 4)

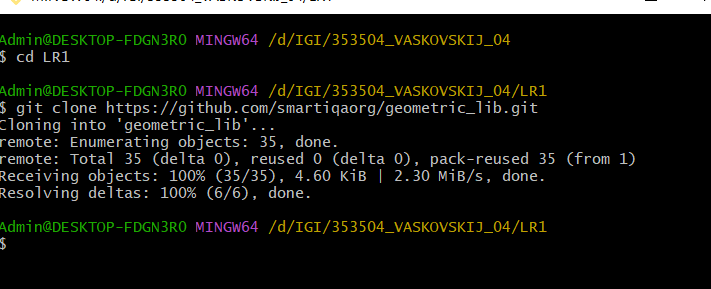
1. Настройка и установка Git (также проверка версии Git)



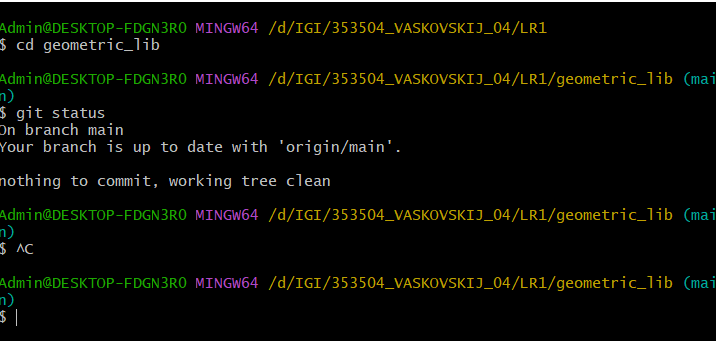
1. Далее создадим директории для лабораторных работ



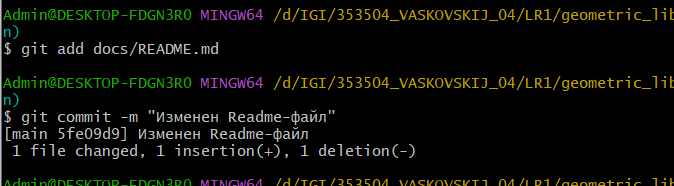
1. Далее клонируем remote-репозиторий в папке LR1



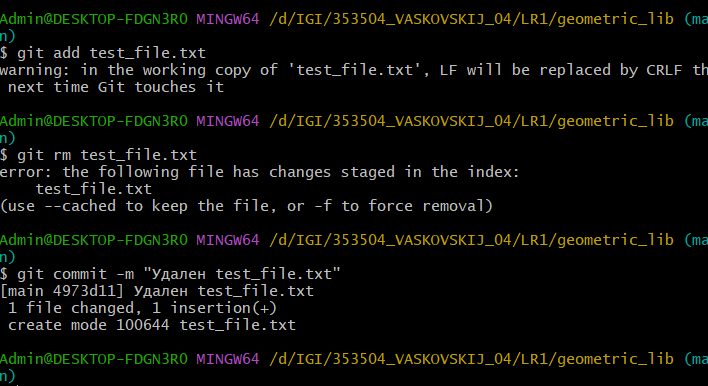
* 1. (Сначала выполним состояние репозитория)



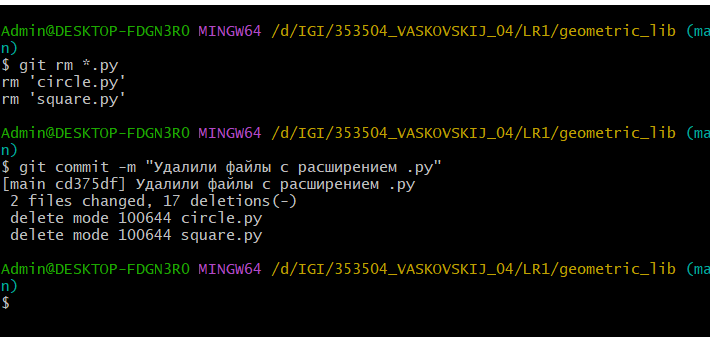
4.1 (Вносим изменения в файл Readme.md и создаем коммит, предварительно проиндексировав Readme.md)



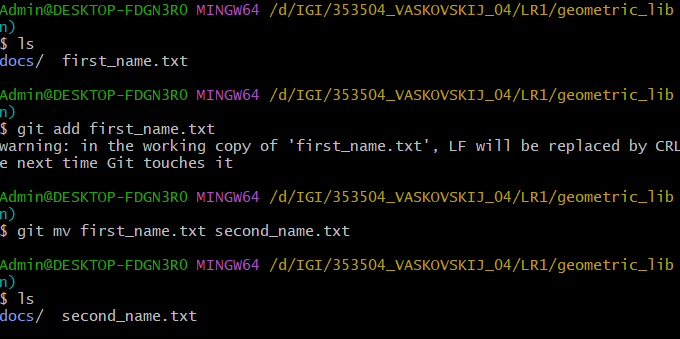
* 1. (Добавим текстовый файл и удалим его)

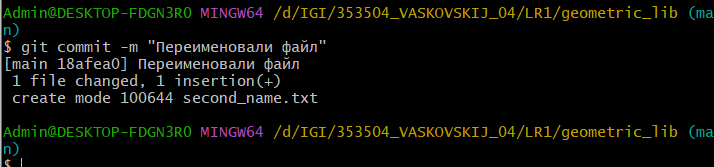


* 1. (Теперь удалим несколько файлов с одинаковым расширением .py)

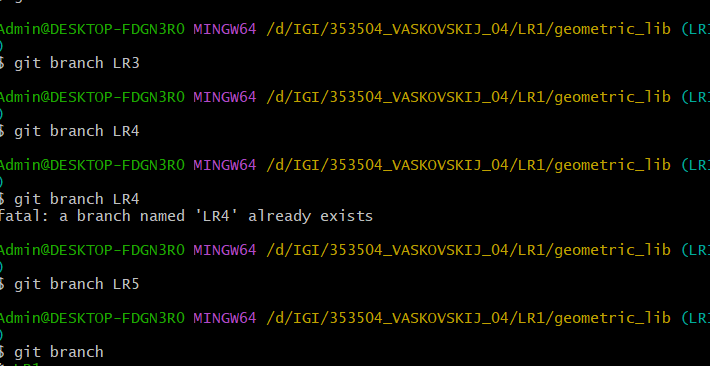


4.4 (Переименуем файл )

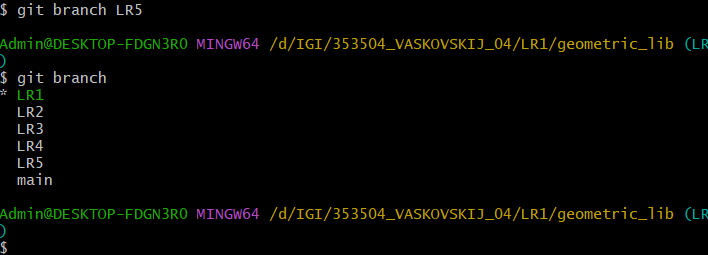




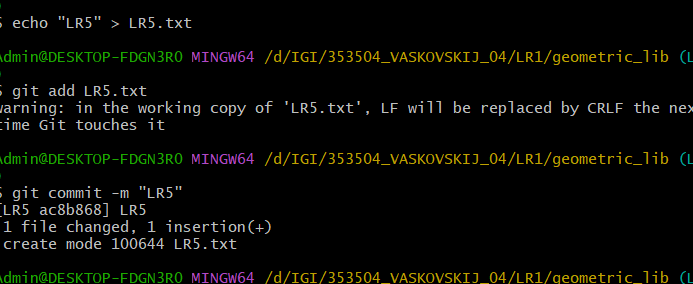
* 1. (Создадим ветки для каждой папки LR1-LR5)



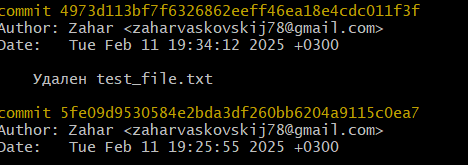
* 1. (Также можно посмотреть все доступные ветки командой git branch)

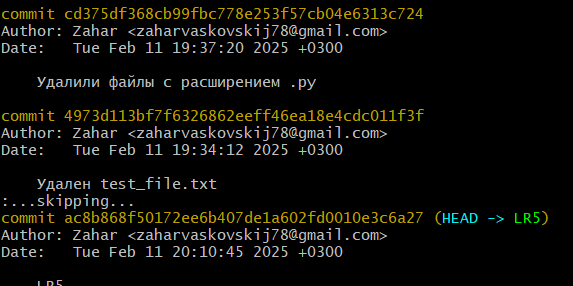


* 1. (также создадим в соответвующей ветке соответвующий текстовый файл)

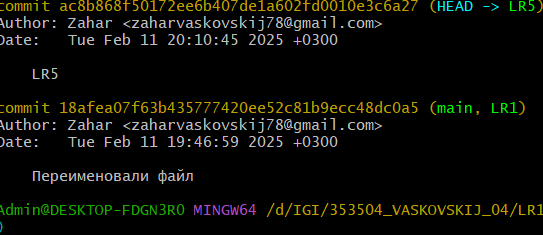


* 1. (выведем полную историю коммитов в репозитории)

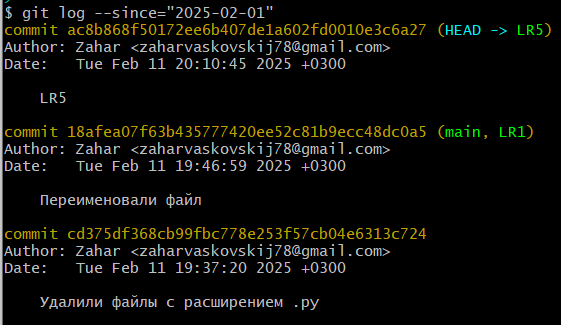




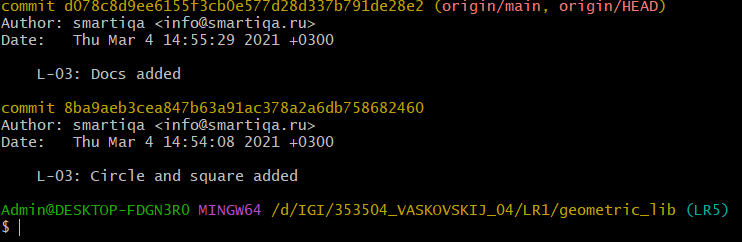
4.7(также при помощи комманды git log –n 2 можно получить только последние два)



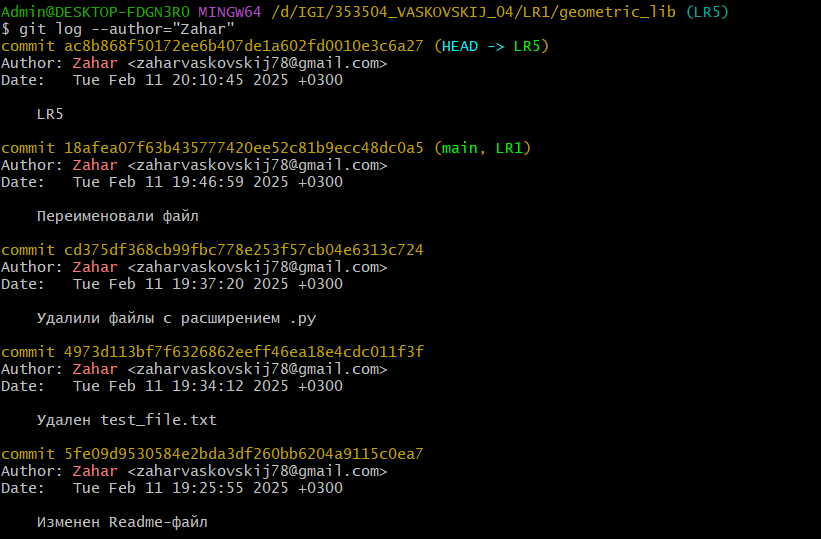
* 1. (коммиты после указанной даты)



* 1. (коммиты до указанной даты)



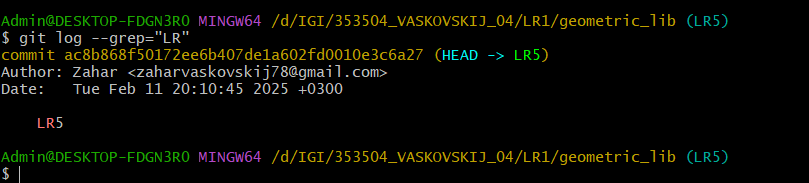
* 1. (получим коммиты по имени автора)



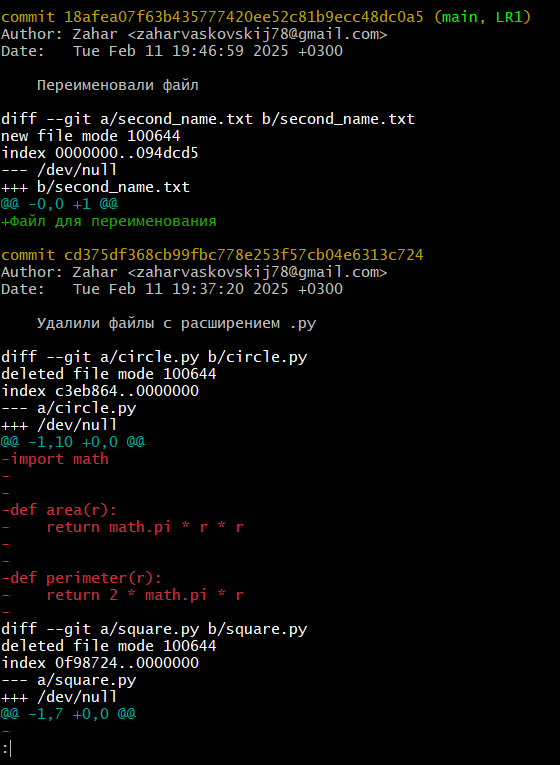
* 1. (получим коммиты по коммитеру)



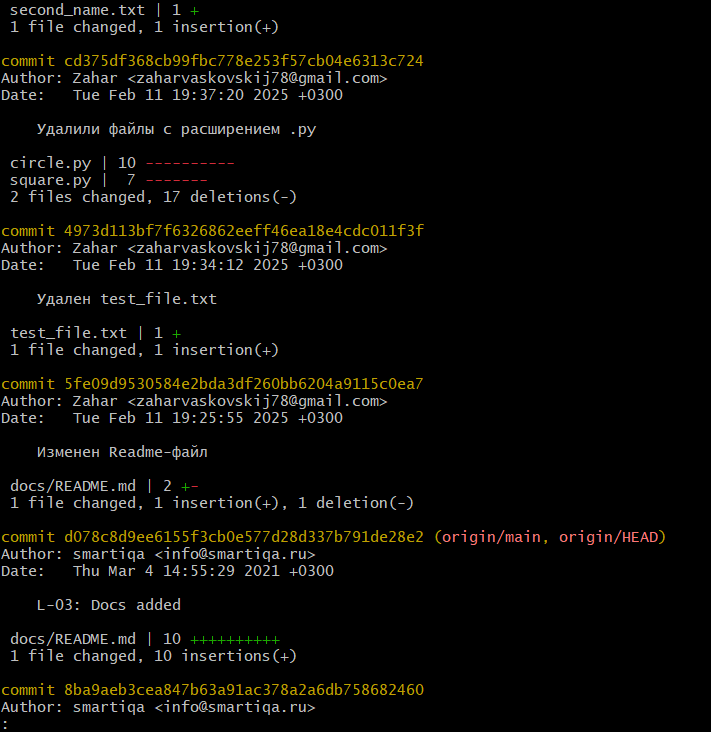
* 1. (поиск коммитов по ключевому слову в сообщениях комита)



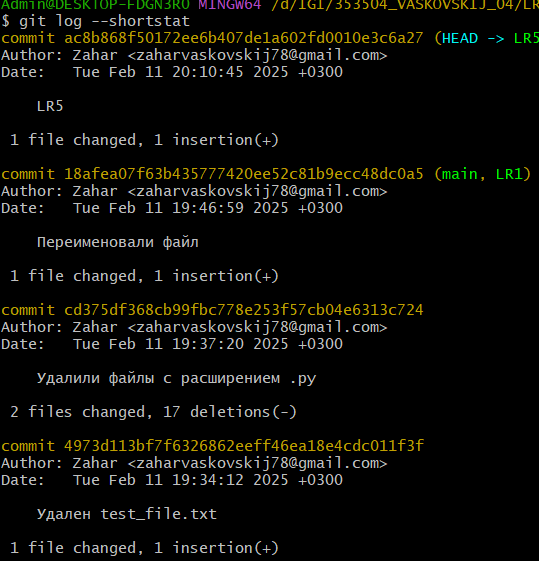
* 1. (показать изменения в файлах каждого коммита)



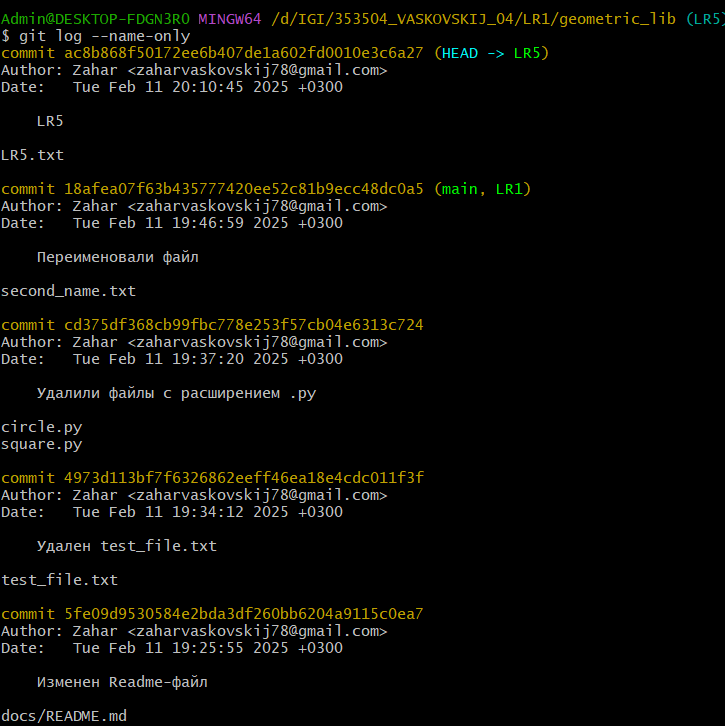
* 1. (показать статистику изменений)



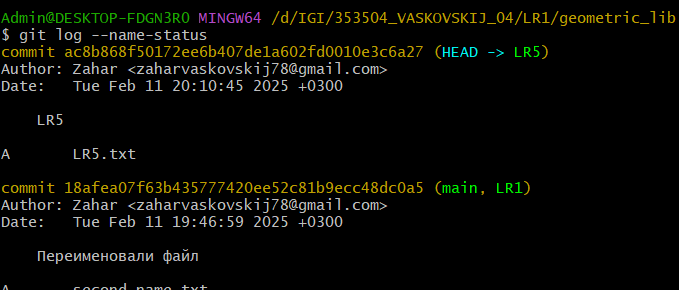
4.8(показать краткую статистику)



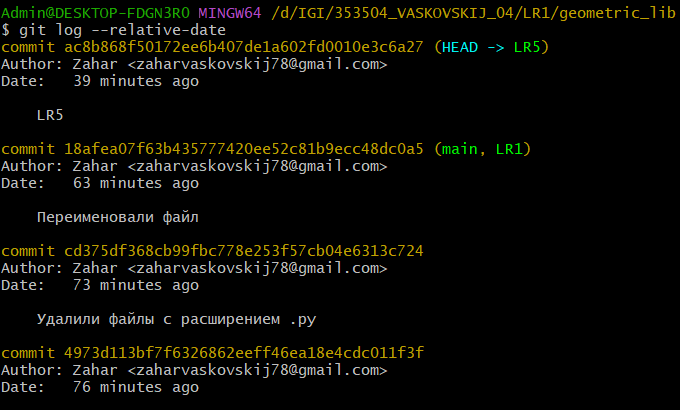
4.8(посмотреть только измененные файлы)



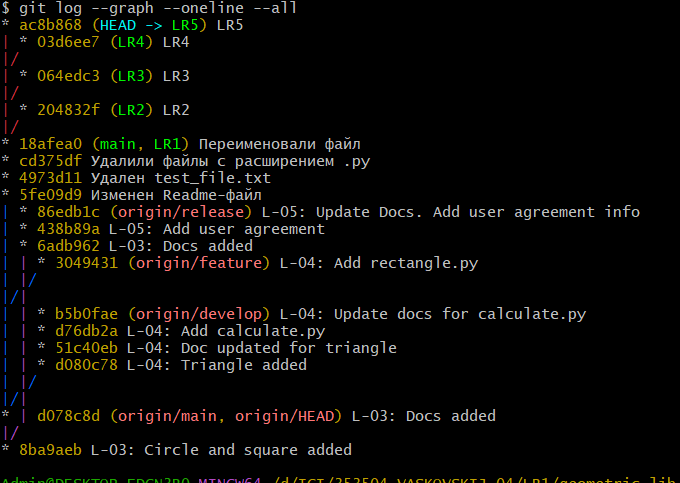
4.8(посмотреть статус измененных файлов)



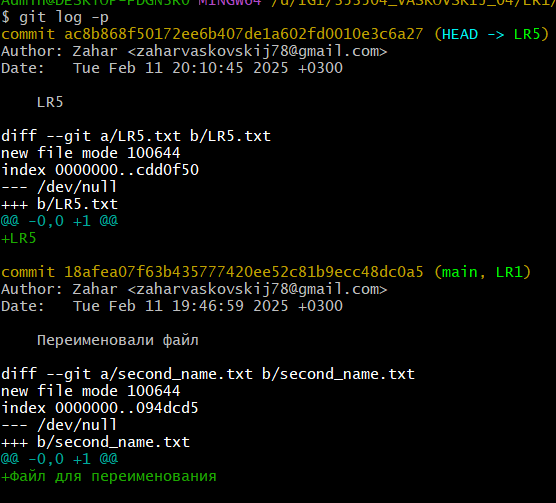
4.8(отображение дат в относительном формате)



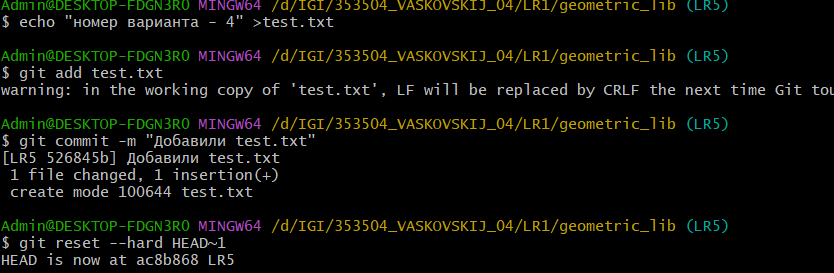
4.8(графическое отображение истории коммитов)



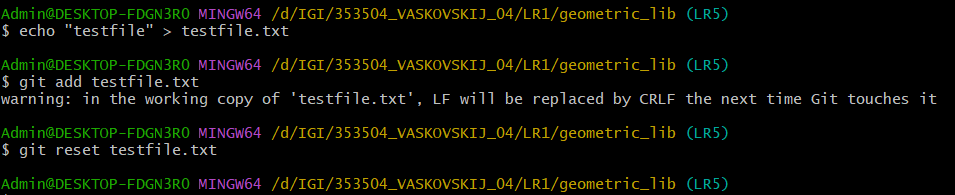
4.9(найти разницу в изменения для каждого коммита)



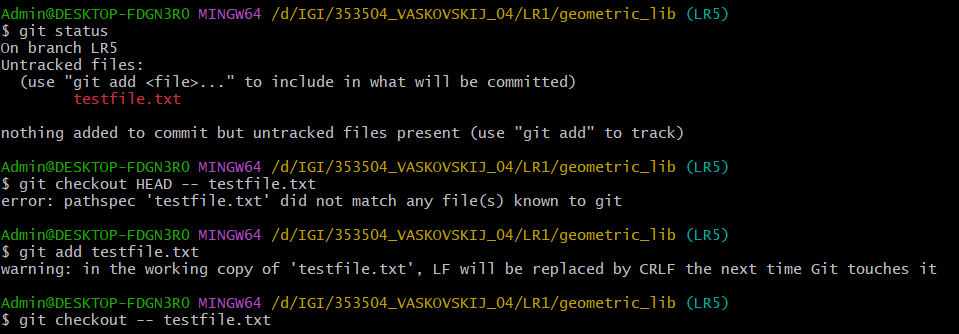
4.10(Создадим файл test.txt, закоммитим изменения и потом их отменим)



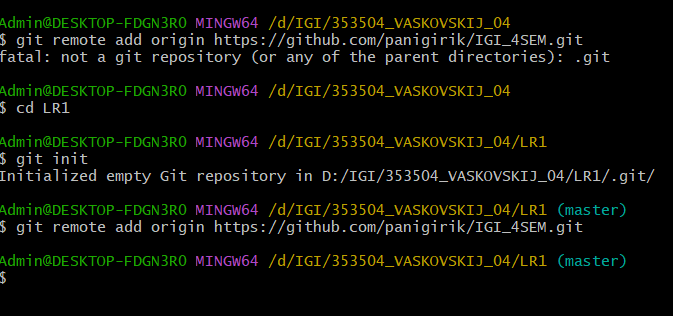
4.11 (проиндексировать файл и отменить его индексацию)



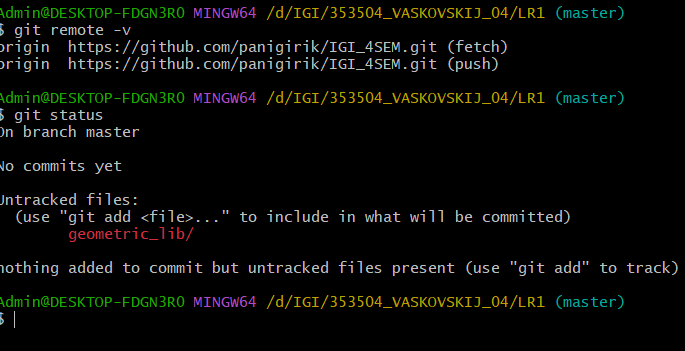
4.12(внести изменения в файл, но потом отменить все изменения)



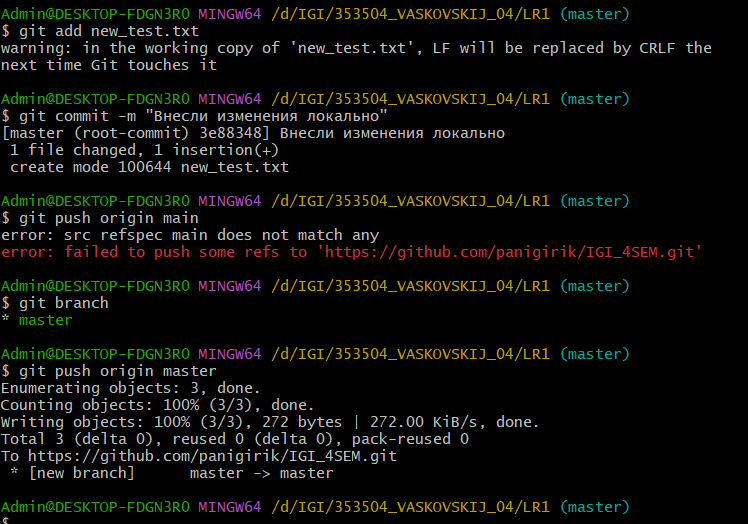
5.1(Создадим remote-репозитрий на github- хосте и привяжем его к локальному репозиторию)



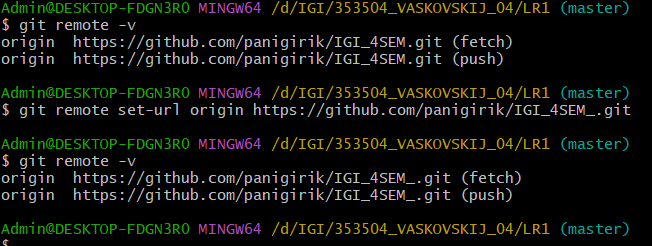
5.2 (просмотрим информацию о репозитории)



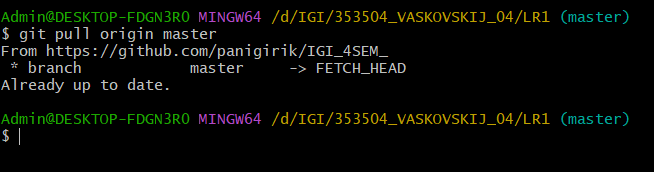
5.2(внесли некоторые изменения, добавив их в коммит и запушили его в remote-репозиторий )



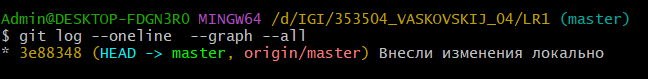
5.2(далее переименуем remote-репозиторий)



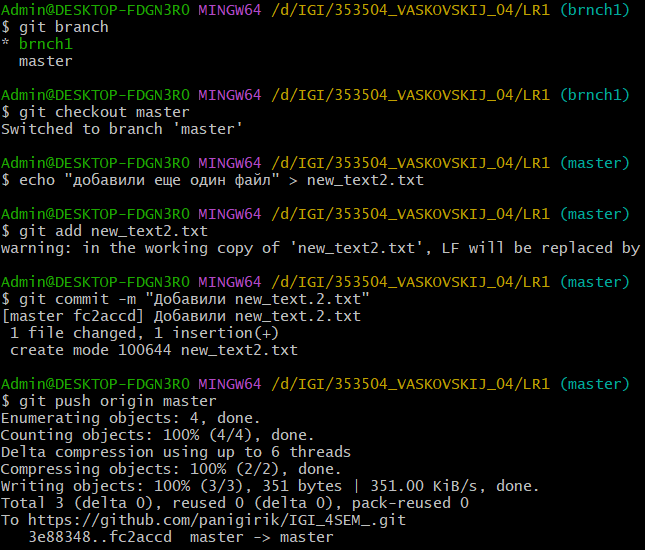
5.3(получим данные из remote-репозитория)



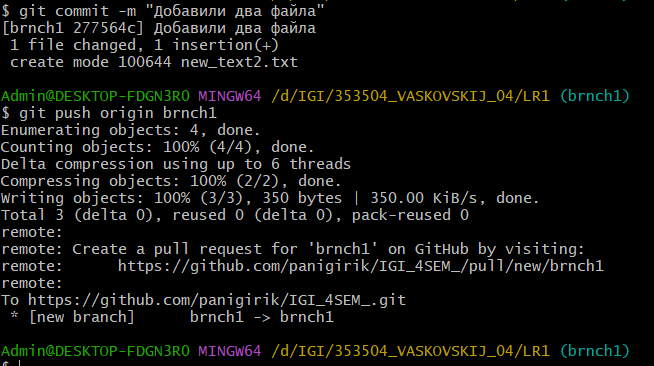
5.4 (просмотрим историю изменений графически)



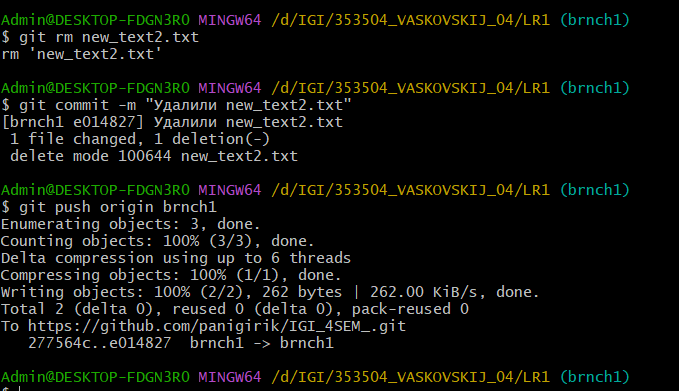
5.5(создадим новую ветку brnch1)



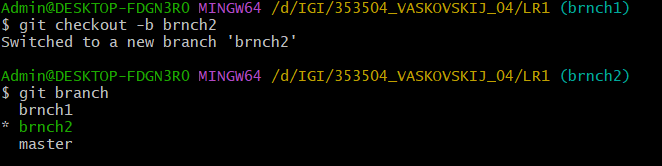
5.5(перенесем двай файла в новую ветку)



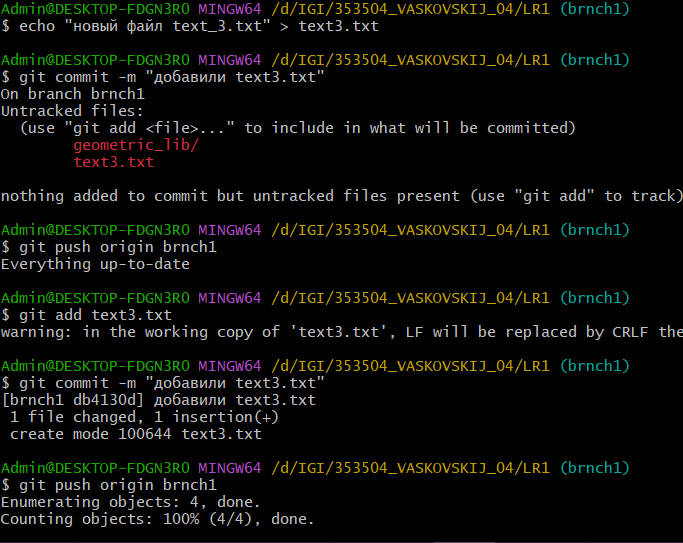
5.5(удалим один файл new\_text2.txt)



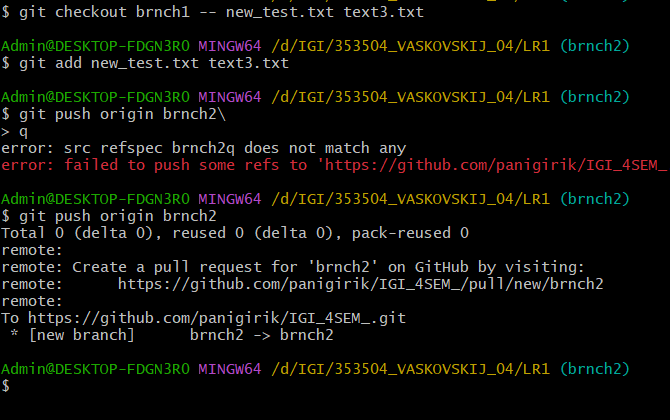
5.6(создадим ветку brnch2 и перейдем на нее)



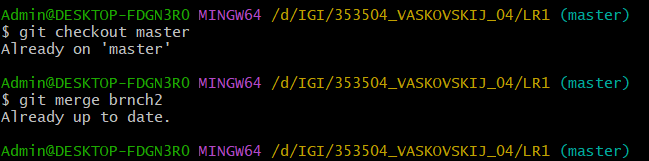
5.7(далее добавим в brnch1 новый файл)



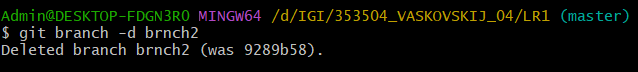
5.7(затем перенесем два файла из brnch1 в brnch2)

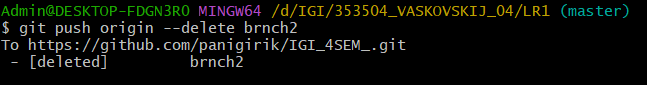


5.7(переключаемя на master и мержим с brnch2)



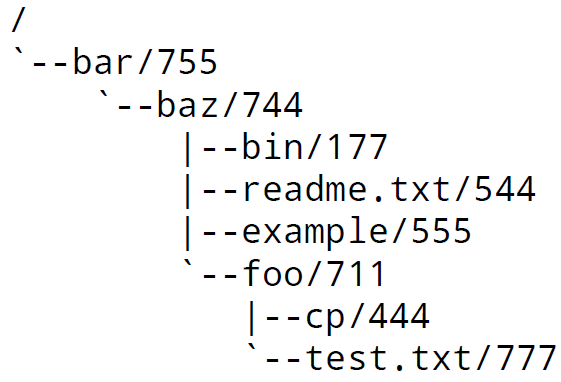
5.7(удаляем brnch2 локально и в remote-репозитории)





ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

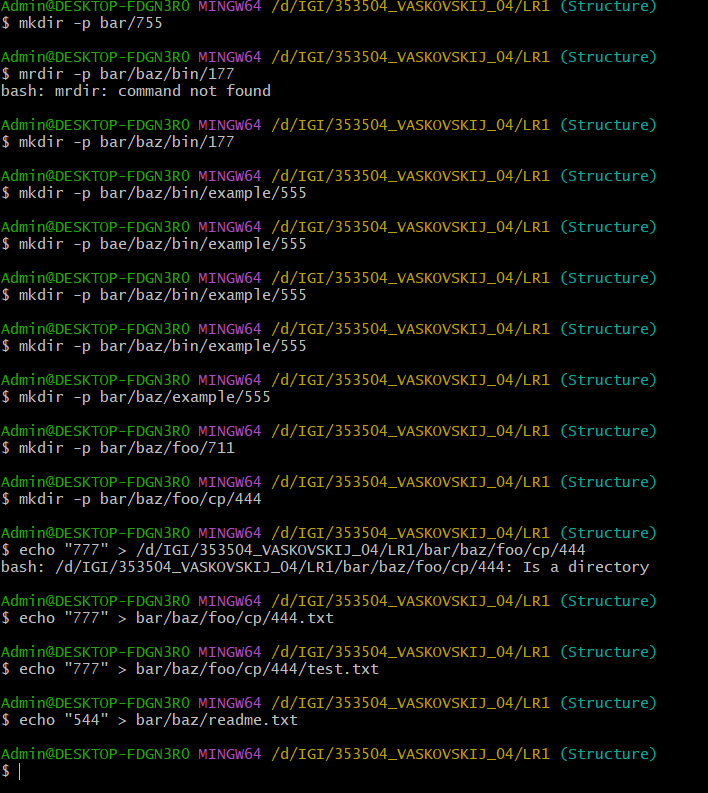
Создадим все необходимые файлы и директории, чтобы получилась структура, как в варианте



Для данной структуры создадим новую ветку Structure например



Далее создадим структуру папок в определенной последовательности



Далее закомитим все изменения