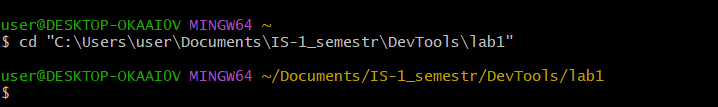
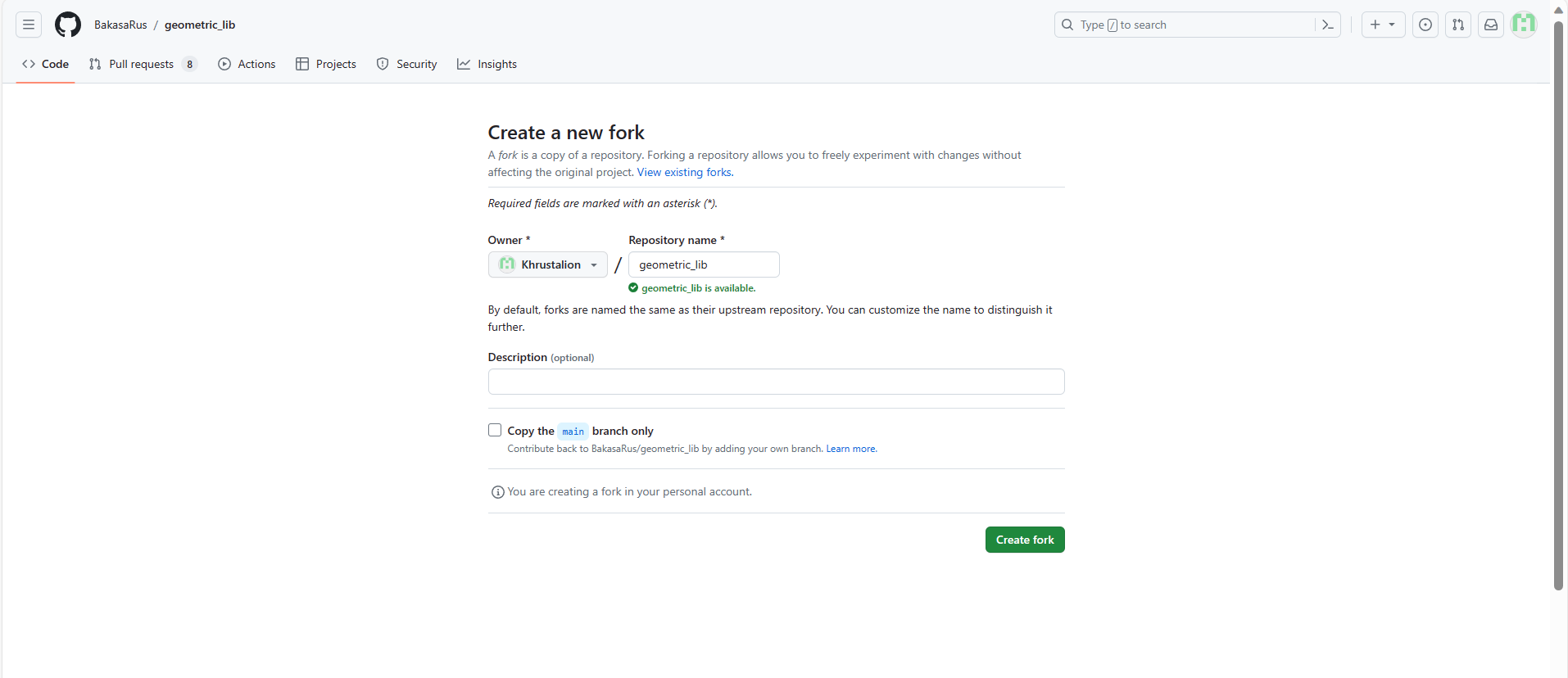
1. **Переходим в папку, в которую будем сохранять репозиторий**

Для этого введем команду «cd <путь к папке>»

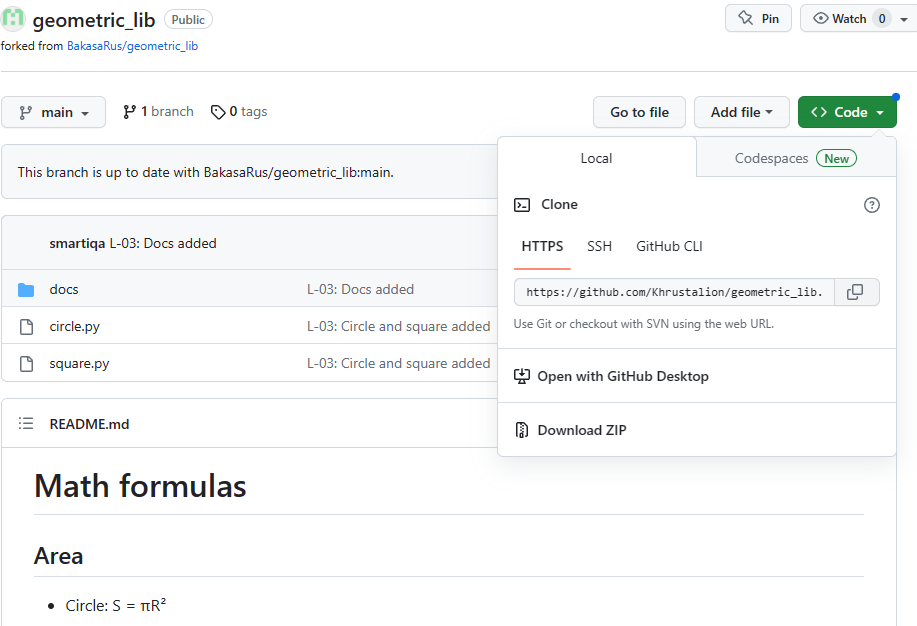
****

**Делаем копию репозитория к себе на github**



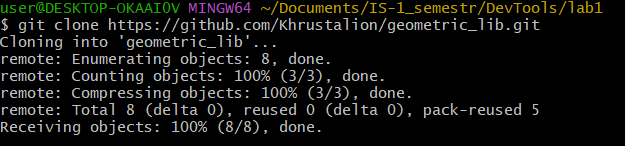
**Копируем ссылку на репозиторий**

Чтобы получить доступ к репозиторию, необходимо получить доступ к нему при помощи ссылки:

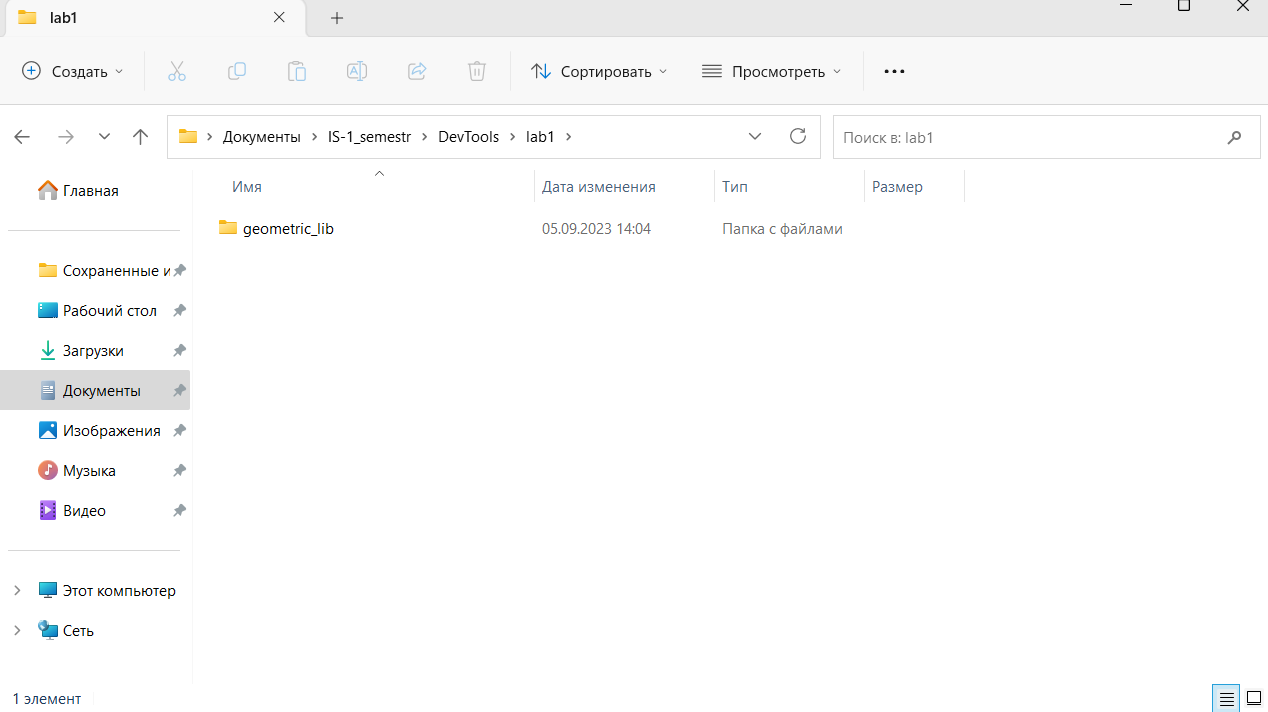


**Клонируем репозиторий к себе на устройство**

С помощью команды <git clone «ссылка на репозиторий»> клонируем репозиторий:

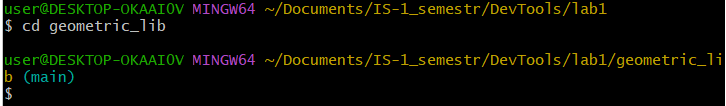


После выполнения этой команды в папке появятся все файлы, клонированные из репозитория:



**Переходим в папку клонированного репозитория**

Для этого выполняем команду <cd geometric\_lib>



1. **Добавим новую ветку**

Для этого выполним команду <git branch new\_features\_409807>

C:\Users\user\Pictures\Screenshots\Снимок экрана 2023-09-05 145837.png

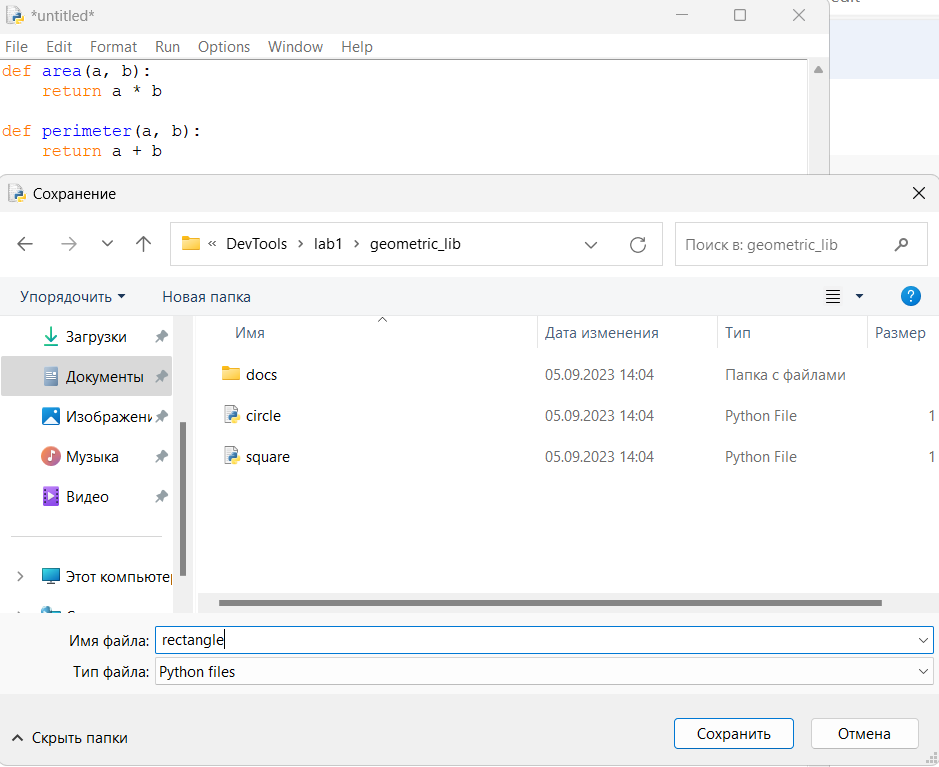
После этого создастся ветка с именем «new\_features\_409807»

**Сделаем новую ветку активной**

Для этого выполним команду <git checkout new\_features\_409807>

C:\Users\user\Pictures\Screenshots\Снимок экрана 2023-09-05 150255.png

1. **Создадим новый файл и добавим его в папку проекта**

****

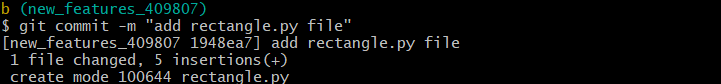
**Добавим новый файл в нашу ветку**

Для этого выполним команду <git add rectangle.py>

C:\Users\user\Pictures\Screenshots\Снимок экрана 2023-09-05 153001.png

1. **Сделаем коммит**

Для этого выполним команды <git commit –m “add rectangle.py file”>

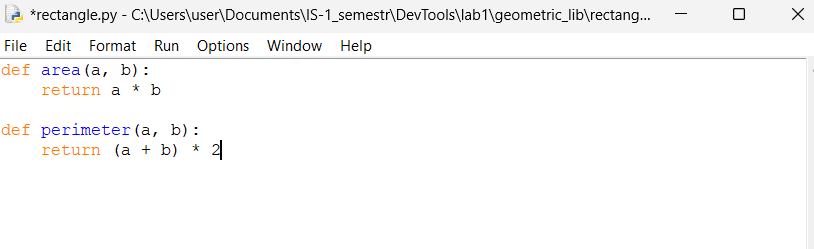


1. **Проделаем те же самые действия**

* Создадим файл triangle.py
* Добавим его в ветку

C:\Users\user\Pictures\Screenshots\Снимок экрана 2023-09-05 184350.png

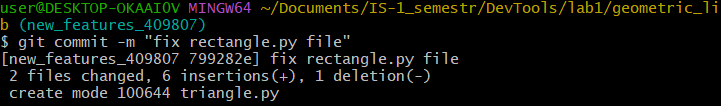
1. **Исправим ошибку в файле rectangle.py**

****

**Добавим исправленный файл в ветку**

C:\Users\user\Pictures\Screenshots\Снимок экрана 2023-09-05 184717.png

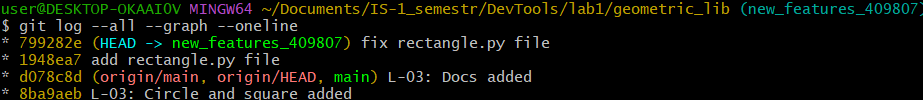
1. **Сделаем коммит**

****

1. **Построим граф истории всего репозитория**

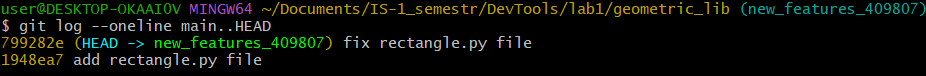
Для этого выполним команду <git log --all --graph --oneline >

* --oneline (то же самое, что и --pretty=oneline) – упрощает понимание информации коммитов(выводит только часть информации)
* -g (то же самое, что и --graph) – выводит график, отражающий структуру ветвления истории коммитов



1. **Построим граф истории текущей ветки**

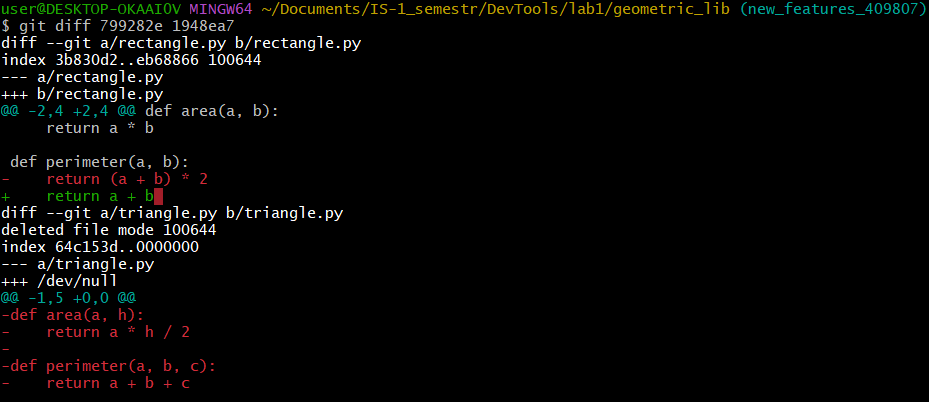
Для этого выполним команды <git log --oneline main..HEAD>



1. **Посмотрим изменения между двумя последними коммитами**

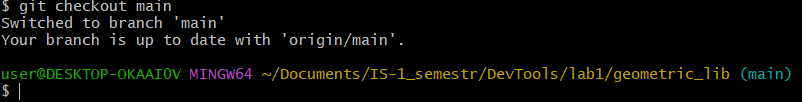
Чтобы посмотреть изменения между коммитами, необходимо знать их хеши. Их можно получить из предыдущего пункта (хеши написаны желтым цветом в начале строк)

После этого выполним команду <git diff 7992822e 1948ea7>

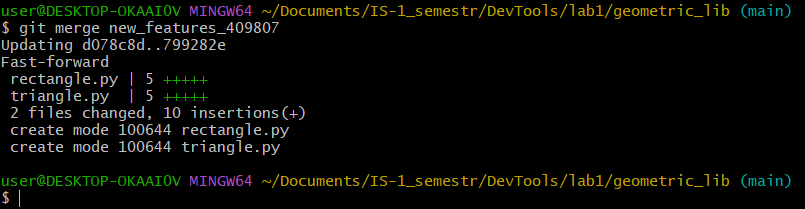


1. **Выполним merge в ветку main**

Для этого перейдем в ветку main с помощью команды <git checkout main>

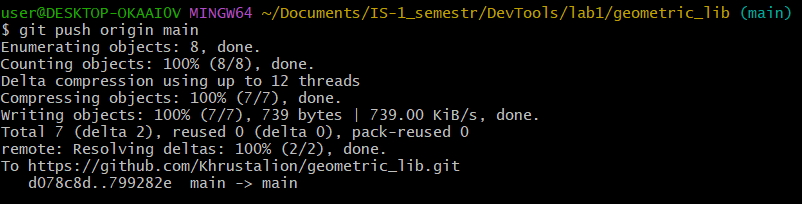


И выполним merge c помощью команды <git merge new\_features\_409807>



1. **Сделаем pull requests**

Для этого выполним команду <git push origin main>



После этого перейдем в наш репозиторий на github и обновим страницу. После этого в репозитории появятся новые файлы

