

# Лабораторная работа №1: Введение в Python и Git

---

## 1. Устанавливаем Python

- Открываем терминал
- Вводим следующие команды:
  - **sudo apt update** – обновляем список пакетов, доступных в репозиториях
  - **sudo apt install software-properties-common**
  - **sudo add-apt-repository ppa:deadsnakes/ppa** – добавляем deadsnakes PPA к списку источников в нашей системе
  - **sudo apt install python3.9** – устанавливаем Python 3.9
  - **python3.9 --version** – проверяем корректность установки

```
[MBP-Arshinov:~ arshegor$ Python3.9 --version  
Python 3.9.6
```

Если на экране мы видим сообщение, подобное показанному на предыдущем снимке экрана, Python 3.9 установлен в нашей Ubuntu, и мы можем начать его использовать.

## 2. Устанавливаем Git

- В терминале вводим следующую команду:
  - **sudo apt install git** – устанавливаем Git

## 3. Устанавливаем Pycharm

- В терминале вводим следующую команду:
  - **sudo snap install pycharm-community --classic**
- Ищем в приложениях PyCharm и открываем его

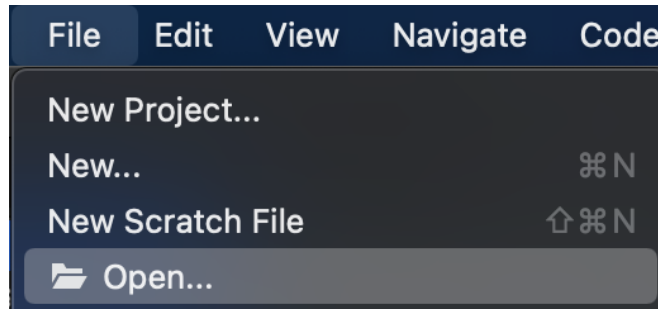
## 4. Создаем и настраиваем директорию проекта

- В терминале вводим следующие команды:
  - **mkdir MyApp** – создаем директорию с именем "MyWebApp"
  - **cd MyApp** – переходим в директорию
- После создания директории создаем и активируем виртуальное окружение Python

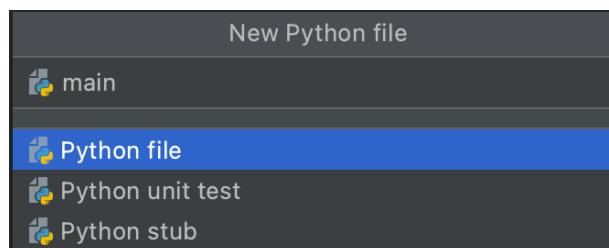
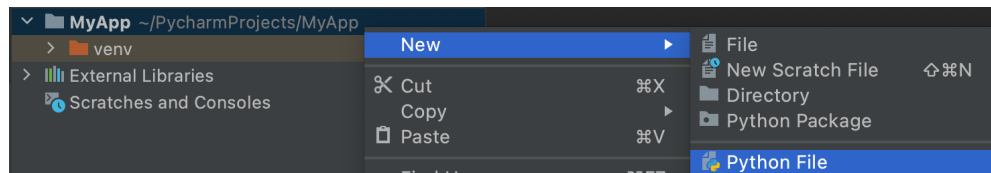
- **python3 -m venv venv** – создаем виртуальное окружение
- **source ./venv/bin/activate** – активируем виртуальное окружение

## 5. Открываем директорию в PyCharm:

- Открываем нашу директорию с именем MyApp:



## 6. Создаем новый файл с именем main.py в папке MyApp



## 7. Решим простую задачу

Задача: В программе содержится некий список с длинами сторон треугольника. Необходимо найти комбинацию из трех длин, совместив которые получается максимальная площадь треугольника.

Решение:

- Копируем следующий код в файл main.py

```
sides = [3, 2, 4, 7, 5, 12, 11, 13, 15, 16, 14, 14]
```

```
sides = sorted(sides, reverse=True)
```

```
smax = 0
```

```
for i in range(len(sides)):
```

```
    for j in range(i + 1, len(sides)):
```

```
        for k in range(j + 1, len(sides)):
```

```
            a = sides[i]
```

```
            b = sides[j]
```

```
            c = sides[k]
```

```
            if a + b > c and a + c > b and b + c > a:
```

```
                p = (a + b + c) / 2
```

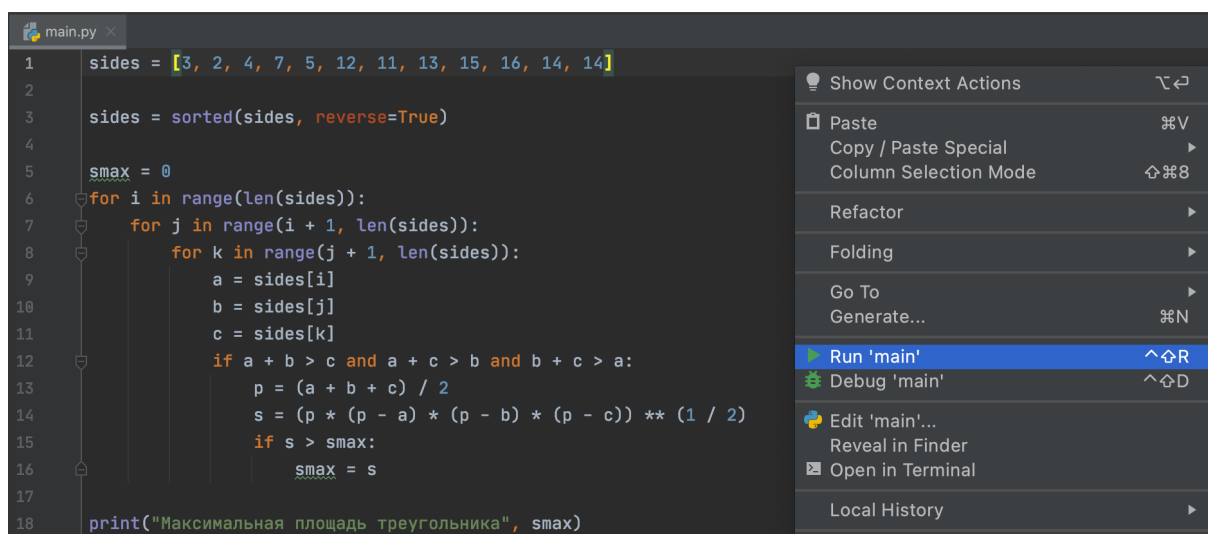
```
                s = (p * (p - a) * (p - b) * (p - c)) ** (1 / 2)
```

```
                if s > smax:
```

```
                    smax = s
```

```
print("Максимальная площадь треугольника", smax)
```

- Запускаем нашу программу



## 8. Зафиксируем изменения в нашем проекте в репозитории Git

- В терминале, в нашей директории инициализируем пустой репозиторий – вводим команду **git init**

```
MBP-Arshinov:MyApp arshegor$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/arshegor/MyApp/.git/
```

- Проверяем статус нашего репозитория – **git status**

```
MBP-Arshinov:MyApp arshegor$ git status
On branch master

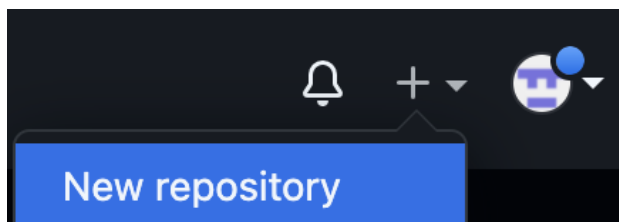
No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
.idea/
main.py
venv/
```

- Добавим в индекс файл main.py – **git add main.py**  
Мы работали с только с файлом main.py поэтому папки ./idea и venv/ мы не будем добавлять в индекс.
- Зафиксируем изменения - **git commit -m "Find max square of triangles"**

## 9. Зафиксируем изменения локального репозитория в удаленном репозитории

- Регистрируемся на сайте github.com
- Создаем новый репозиторий

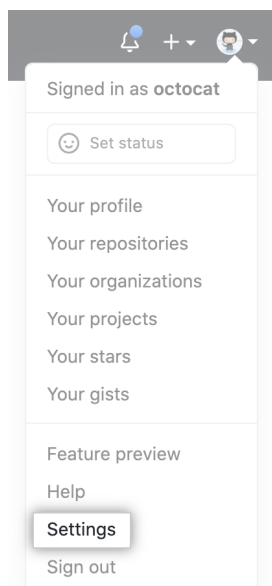


- Копируем url-адрес репозитория
- Добавляем удаленный репозиторий – **git remote add origin <url-адрес репозитория>**

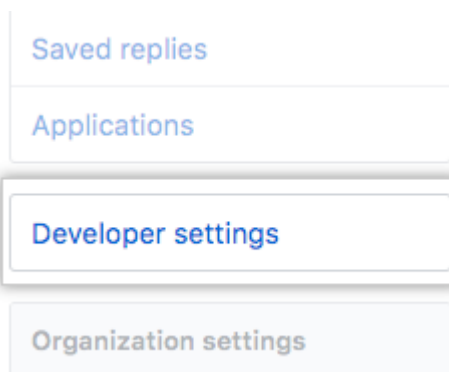
**Если Вас просят ввести токен то выполните следующие шаги:**

Подтвердите свой адрес электронной почты, если он еще не был подтвержден.

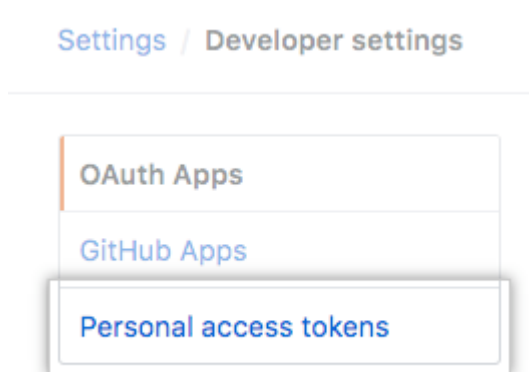
В правом верхнем углу любой страницы щелкните фотографию своего профиля, затем нажмите "Настройки".



На левой боковой панели нажмите Настройки разработчика.



На левой боковой панели выберите Маркеры личного доступа.



Нажмите кнопку Создать новый токен.

## Personal access tokens

Generate new token

Дайте вашему токенту описательное имя.

### Note

My bash script

What's this token for?

Нажмите кнопку Создать токен.

☐ write:gpg\_key

Write user gpg keys

☐ read:gpg\_key

Read user gpg keys

Generate token

Cancel

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your new personal access token now. You won't be able to see it again!

✓ ghp\_IqIMN0ZH6z0wIEB4T9A2g4EHMy8Ji42q4HA5 

Enable SSO ▾

Delete

- Фиксируем изменения – **git push origin master**

### Домашнее задание:

Задача: На вход программе подаются 3 коэффициента квадратного уравнения. Программа должна находить корни квадратного уравнения.

Залить данную задачу на локальный Git и GitHub.