

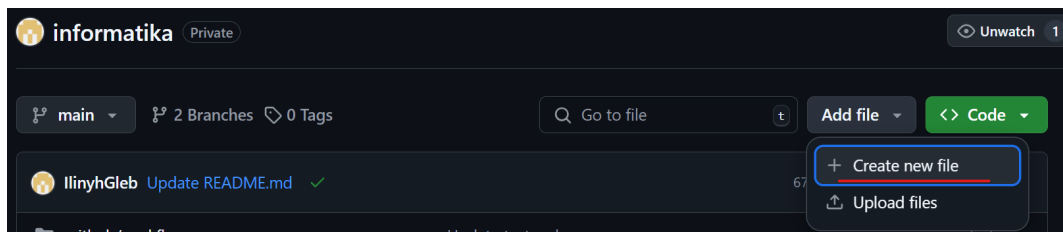
# Лабораторная работа: Запуск Python

Правила выполнения лабораторной работы:

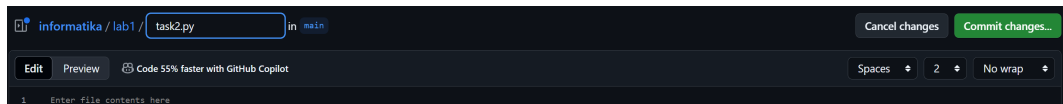
- В процессе выполнения лабораторной работы собирается отчёт. Представление каждого выполненного задания в отчете включает следующие элементы:
  1. Постановка задачи в виде текста задания.
  2. Код решения задачи (сохранить исходный вид при копировании из IDE).
  3. Скриншоты выполнения программы. Можно использовать программу Ножницы, скриншот окна (комбинация клавиш Alt+Prt Scr) или скриншот фрагмента экрана (комбинация клавиш Ctrl+Win+s, для Windows 10 и выше).
- В конце отчёта необходимо добавить раздел **Выводы**, в котором в виде нумерованного списка пишутся выводы по работе: что было сделано, какие задачи были решены, что следует отметить по итогу выполнения задач и т.п..

## Задание 0. Создание репозитория GitHub

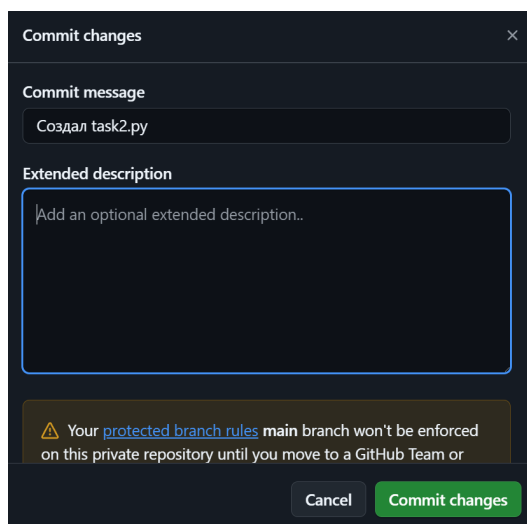
1. Создайте репозиторий `GitHub.com` для курса Информатики.
2. Создайте новый пустой файл:



3. В поле названия укажите `lab1/task2.py`. Таким образом будет создана папка `lab1`, в которой будет создан файл `task2.py`



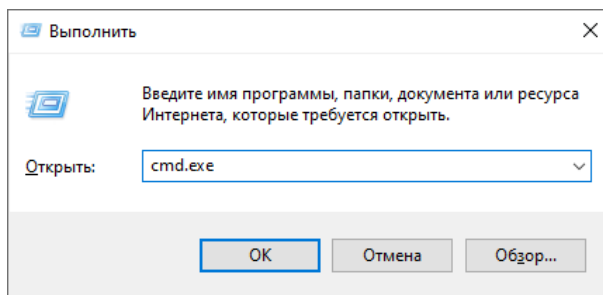
4. Зафиксируйте изменения с помощью кнопки `Commit changes...`
5. В качестве сообщения коммита напишите: `Создал(-а) task2.py`



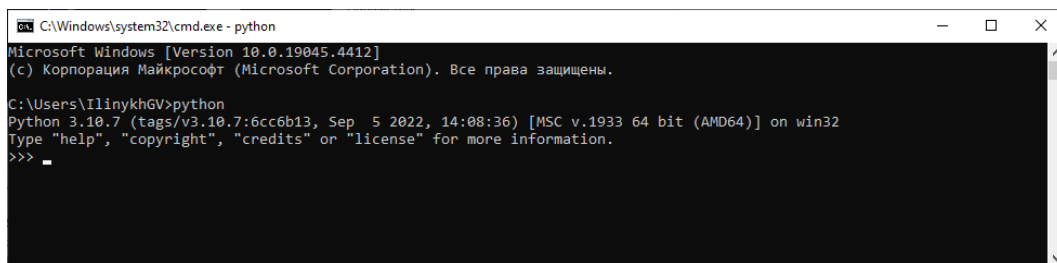
## Задание 1. Запуск Python в командной строке

Запустите командную строку:

1. В Windows: нажмите комбинацию клавиш Win+R
2. В появившемся окне введите название программы командной строки: cmd.exe



3. В появившемся окне введите название программы интерпретатора Python: python и нажмите Enter

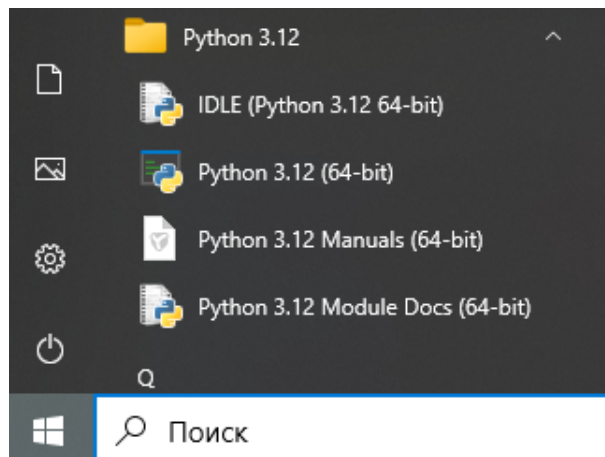


4. В командной строке открылся Python в интерактивном режиме. В таком режиме можно пользоваться Python как калькулятором. Введите выражение  $75 \times 35$  и нажмите Enter.
5. Задайте переменную  $x=5$ . Введите выражение  $x/7+5 \times 3$  и нажмите Enter.
6. Для более сложных расчётов необходимо импортировать дополнительную библиотеку Math. Введите `import math` и нажмите Enter.
7. Добавьте переменную  $y$  и задайте ей любое значение.
8. Посчитайте выражение  $\cos(y)+3 \times \sin(x)$ . Для ввода функции  $\cos(y)$  используется код `math.cos(y)`
9. Для выхода из Python используйте комбинацию клавиш Ctrl+z.
10. Сохраните скриншот консоли и добавьте его в отчёт.

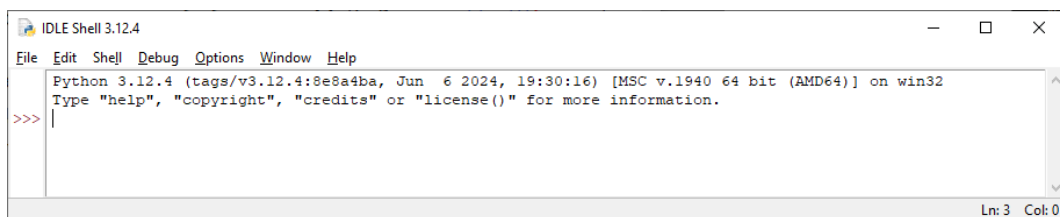
## Задание 2. Запуск Python IDLE

Запустите программу IDLE Python - простую среду для работы с Python:

1. В Windows: нажмите клавишу Win и введите в поиске IDLE, запустите найденную программу IDLE (Python). Также программу можно найти в панели программ в папке Python:



- Открывшееся окно представляет собой оболочку для запуска Python, аналогичную той, что была запущена в командной строке в Задании 1:



- Нажмите File→New File чтобы создать текстовый файл.
- В проводнике Windows создайте свою папку `lab1` в документах пользователя и сохраните редактируемый текстовый файл под именем `task2.py` в этой папке.
- Для запуска программы из текстового файла нажмите Run→Run Module или клавишу F5 в выбранном окне файла.
- Напишите программу для вычисления следующих математических функций:

$$a = \sqrt{\frac{x^3}{2}} - \sin y$$

$$b = \frac{e^2}{3} - \cos y + z + \ln y$$

- Первая строка программы включает импорт функций. Выражение для импорта косинуса и натурального логарифма имеет вид: `from math import cos, log`. Добавьте в это выражение импорт функции синуса и экспоненты и напишите его в программе. В таком случае к функции `cos` можно обратиться по её имени без указания имени библиотеки `math.cos`
- Для ввода переменной `x` пользователем добавьте код:

```
x = float(input("Введите значение переменной x: "))
```

Напишите аналогичные строки кода для ввода переменных `y` и `z`.

- После подсчёта функции `a` выведите её значение в консоль командой:

```
print(f"Получено значение функции a={a}")
```

- Сохраните скриншот текстового файла и консоли IDLE и добавьте его в отчёт.
- Скопируйте полученный код в файл `task2.py`, созданный в Задании 0 и зафиксируйте изменения с сообщением коммита: `обновил task2.py`.

## Задание 3. Запуск Python в онлайн-интерпретаторе

1. Запустите онлайн-интерпретатор <https://www.online-python.com/>,
2. Удалите стандартный код и посчитайте в нём значение функции:

$$z = \frac{\sqrt[4]{x^3}}{x+a} + x^b$$

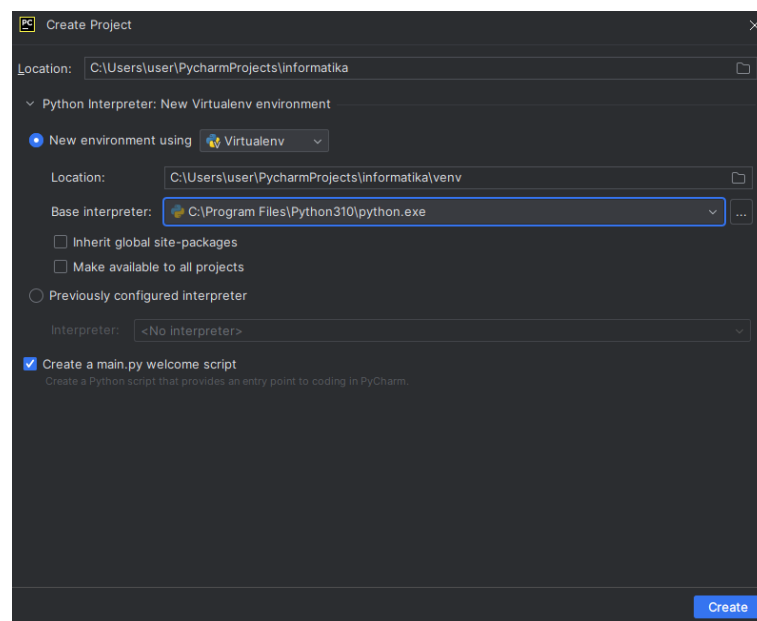
используя пользовательский ввод и вывод, как в Задании 2.

3. Сохраните скриншот страницы и добавьте его в отчёт.
4. Скопируйте полученный код в новый файл `task3.py` в удалённом репозитории

## Задание 4. Запуск Python в интегрированной среде разработки Pycharm

Запустите программу Pycharm:

1. В Windows: нажмите клавишу Win и введите в поиске Pycharm, запустите найденную программу Pycharm Community Edition. Также программу можно найти в панели программ в папке JetBrains
2. Создайте новый проект File→New Project



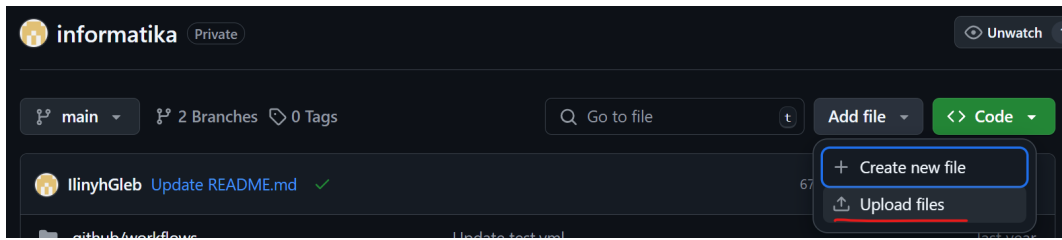
При этом автоматически будет создано новое виртуальное окружение в папке `venv`

3. Создайте новый файл в дереве проекта (левая панель) с помощью ПКМ→New File... с именем `task4.py`  
ПКМ - правая клавиша мыши
4. Запишите в новом файле код вычисления значения функции:

$$z = 2\sqrt[4]{x^6}(x+a) + y^b$$

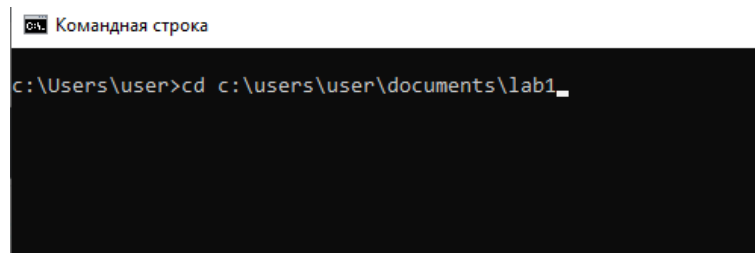
используя пользовательский ввод и вывод, как в Задании 2.

5. Запускайте программы комбинацией клавиш Ctrl+Shift+F10 при первом запуске и Shift+F10 при последующих.
6. Сохраните скриншоты консоли при запусках программы и добавьте их в отчёт.
7. Загрузите `task4.py` в удалённый репозиторий



## Задание 5. Создание виртуального окружения в командной строке

1. Откройте командную строку `cmd.exe`
2. Перейдите в папку, созданную в Задании 2 с помощью команды `cd`:

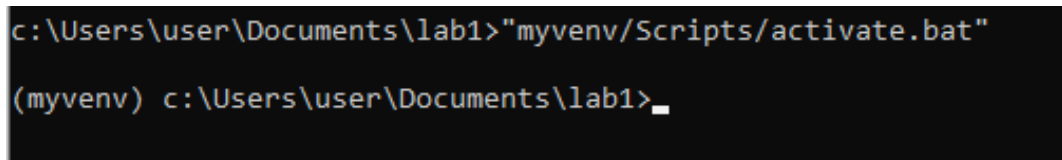


3. Введите команду:

```
python -m venv myenv
```

Ключ `-m` значит, что Python будет запускать модуль. В данном случае будет запущен модуль `venv`, который создаёт виртуальное окружение. Будет создано виртуальное окружение под названием `myenv`. Папка с соответствующим названием будет создана в текущей директории. Можно это проверить зайдя в папку в Проводнике Windows или введя команду `dir` в консоли

4. Активируйте созданное виртуальное окружение с помощью запуска программы `activate.bat`:

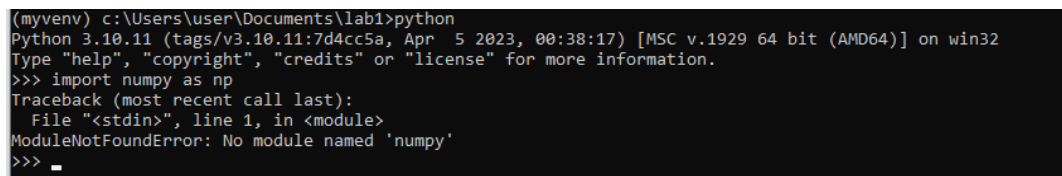


Если окружение активировано, то слева появится его имя в круглых скобках. Значит теперь вы работаете именно в этом окружении в рамках текущей консоли (командной строки). Если вы хотите выключить виртуальное окружение, то аналогичным образом запустите программу `deactivate.bat`

5. Откройте консоль Python с помощью соответствующей команды `python`
6. Попробуйте импортировать пакет `numpy`:

```
import numpy as np
```

Так как в данном окружении пакет не установлен, то появится ошибка



7. Закройте консоль Python с помощью `Ctrl+z`
8. Установите пакет `numpy` в данной виртуальное окружение с помощью команды

```
pip install numpy
```

Запустится процесс установки пакета

```
(myvenv) c:\Users\user\Documents\lab1>pip install numpy
Collecting numpy
  Downloading numpy-2.1.2-cp310-cp310-win_amd64.whl (12.9 MB)
----- 12.9/12.9 MB 2.8 MB/s eta 0:00:00
Installing collected packages: numpy
Successfully installed numpy-2.1.2

[notice] A new release of pip is available: 23.0.1 -> 24.2
[notice] To update, run: python.exe -m pip install --upgrade pip

(myvenv) c:\Users\user\Documents\lab1>_
```

9. Повторите шаги 5-6
10. Сохраните скриншоты консоли при запусках программы и добавьте их в отчёт.