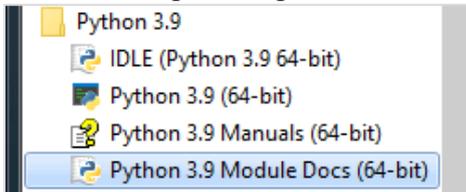


Создаем первую программу

В среде программирования IDLE

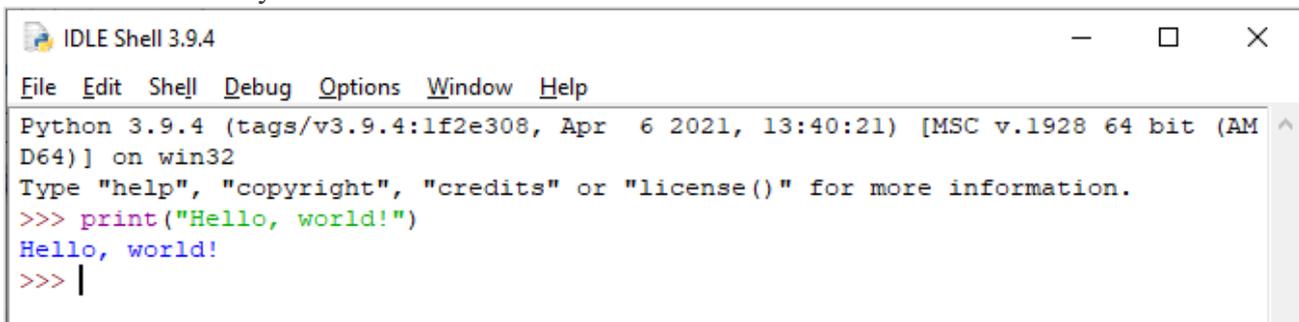
Свою первую программу – по традиции, это программа, печатающая на экране *Hello, world!* вы сможете набрать в среде IDLE. Запустите ее из главного меню:



Среда запустилась в интерактивном режиме и ждет ваших команд. Наберите в строке:

```
print("Hello, world!")
```

и нажмите клавишу *<Enter>*.



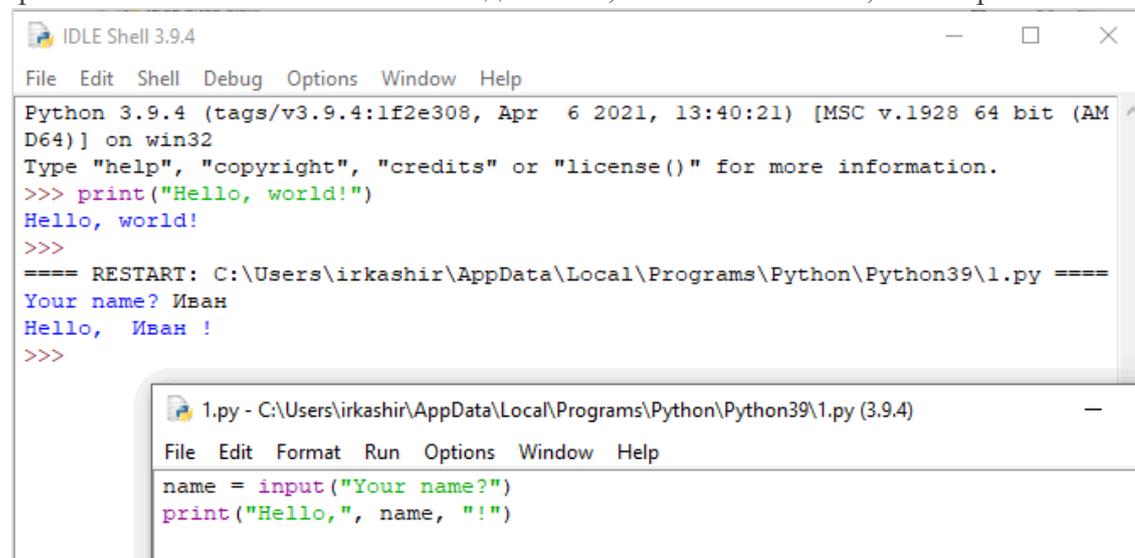
Поздравляем – вы написали свою первую программу! 🙌

Но в интерактивном режиме писать и сразу выполнять по одной команде в каждой строке не очень удобно, поэтому лучше записывать текст программы в отдельном файле и сохранять его, а потом запускать сразу все команды этого файла.

Для этого выберите пункт меню *File—New file* (или нажмите *<Ctrl-N>*). В открывшемся окне введите текст:

```
name = input("Your name? ")
print("Hello,", name, "!")
```

Теперь нажмите *<F5>* (или выберите пункт меню *Run—Run Module*). Среда предложит вам сохранить файл. Сохраните его в папку для ваших будущих программ. Программа запустится, и вы увидите приглашение ввести ваше имя. Введите имя, нажмите *<Enter>*, и на экране появится

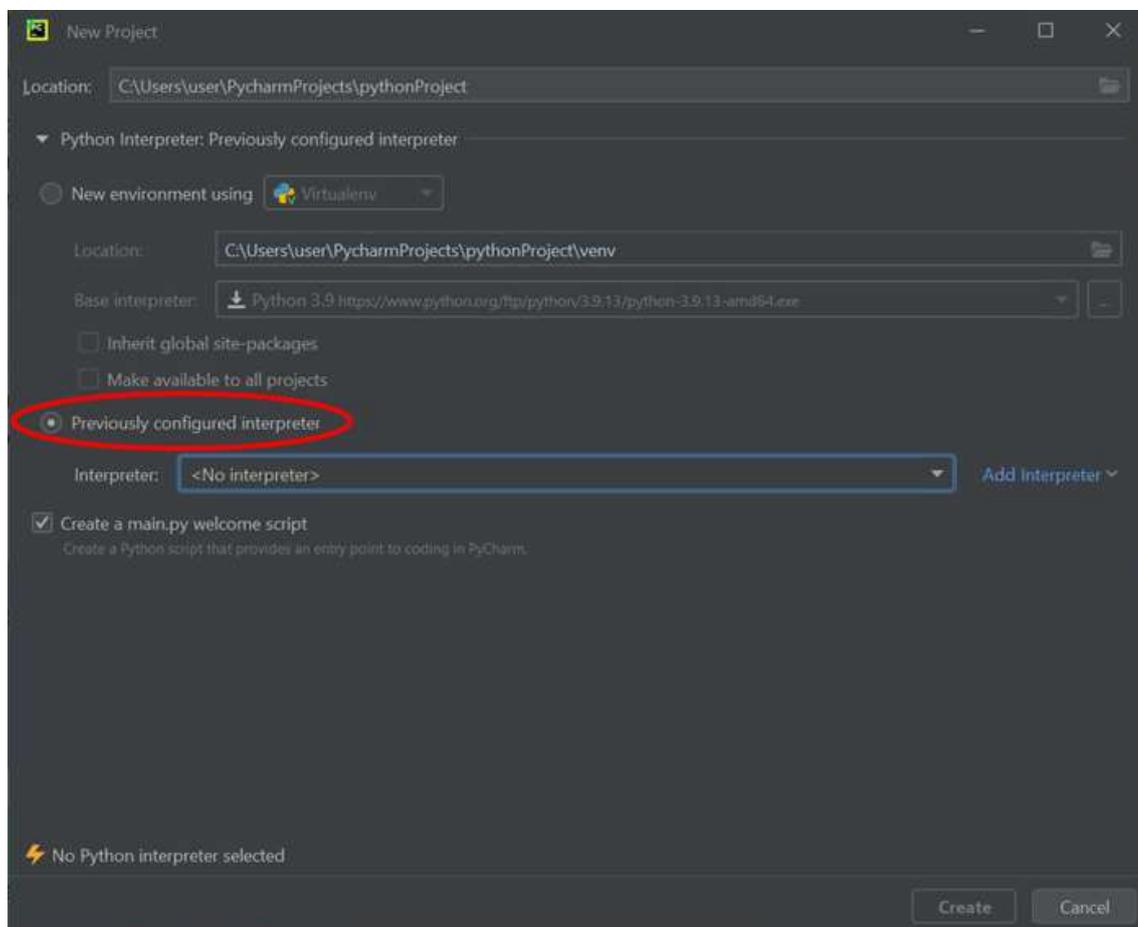


В среде программирования PyCharm

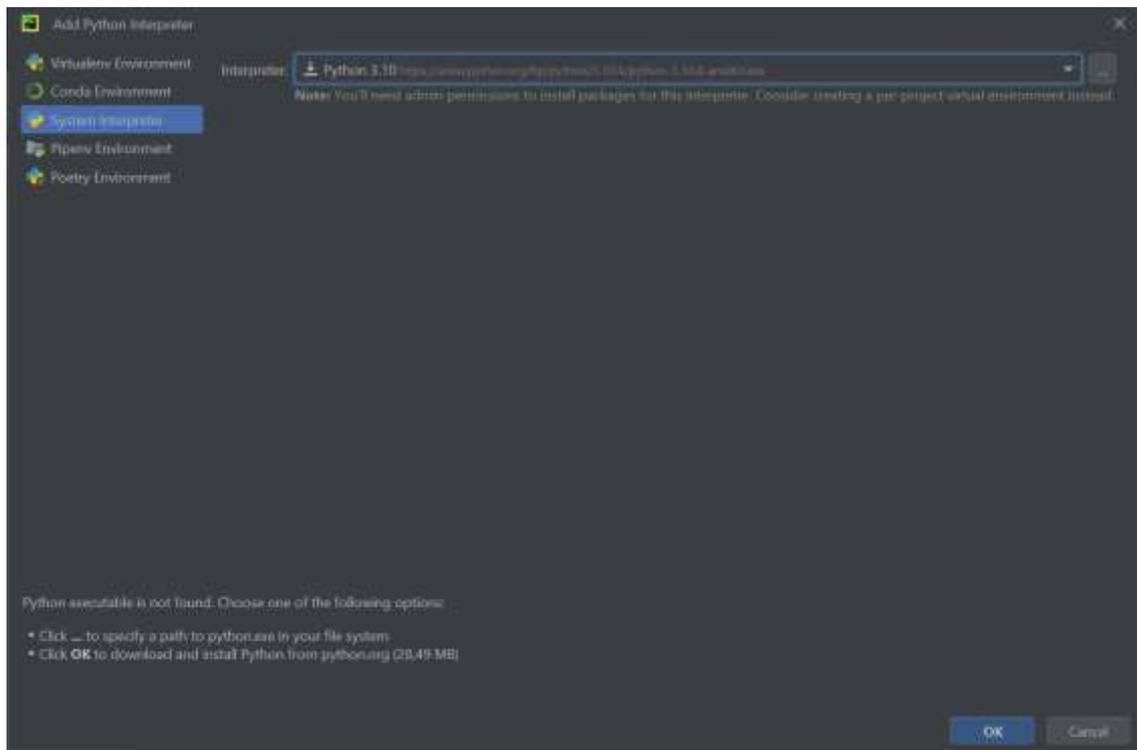
Откройте среду PyCharm и выберите New Project



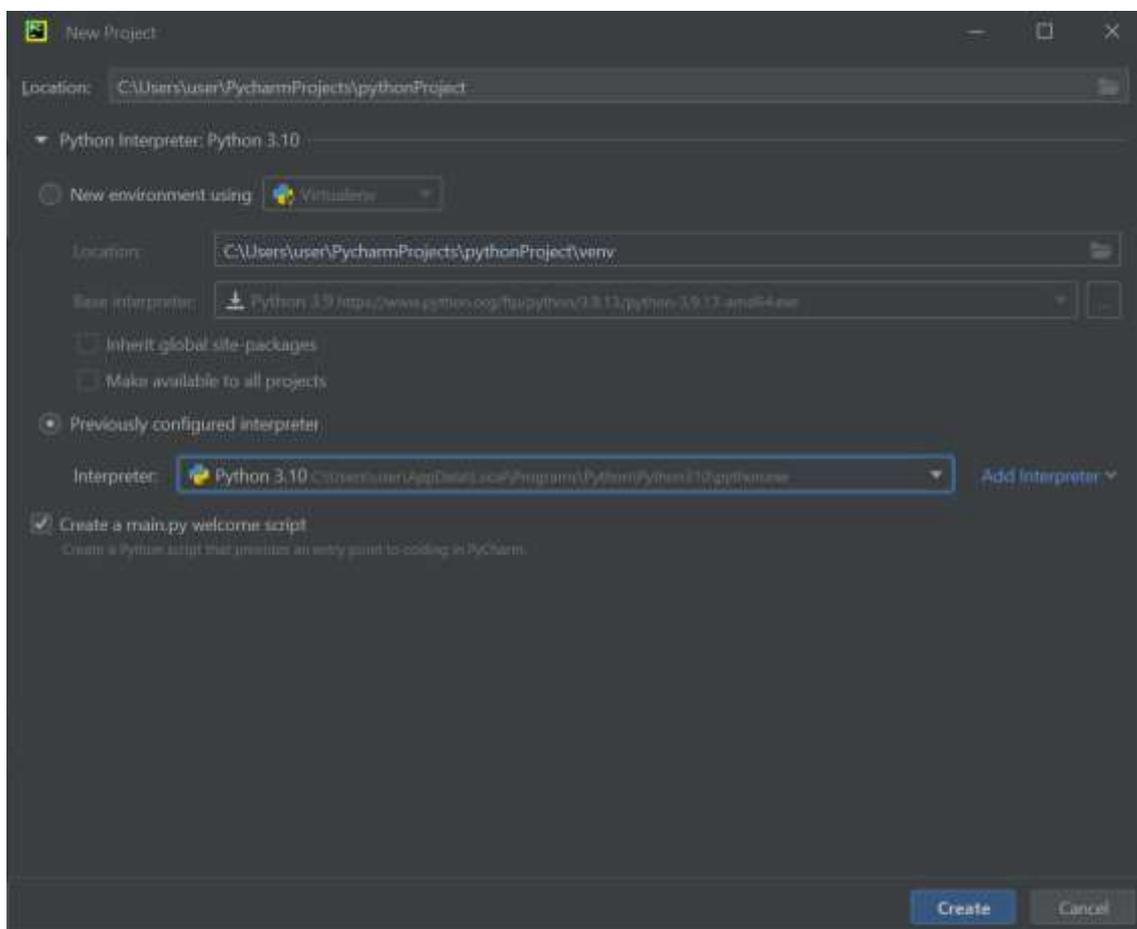
Откройте среду *PyCharm* и выберите *New Project*. В появившемся окне выберите радиокнопку *Previously configured interpreter*. Если вы предварительно установили Питон, то в выпадающем списке вы увидите установленную версию. Если же Питон не установлен, то перейдите по гиперссылке *Add Interpreter*



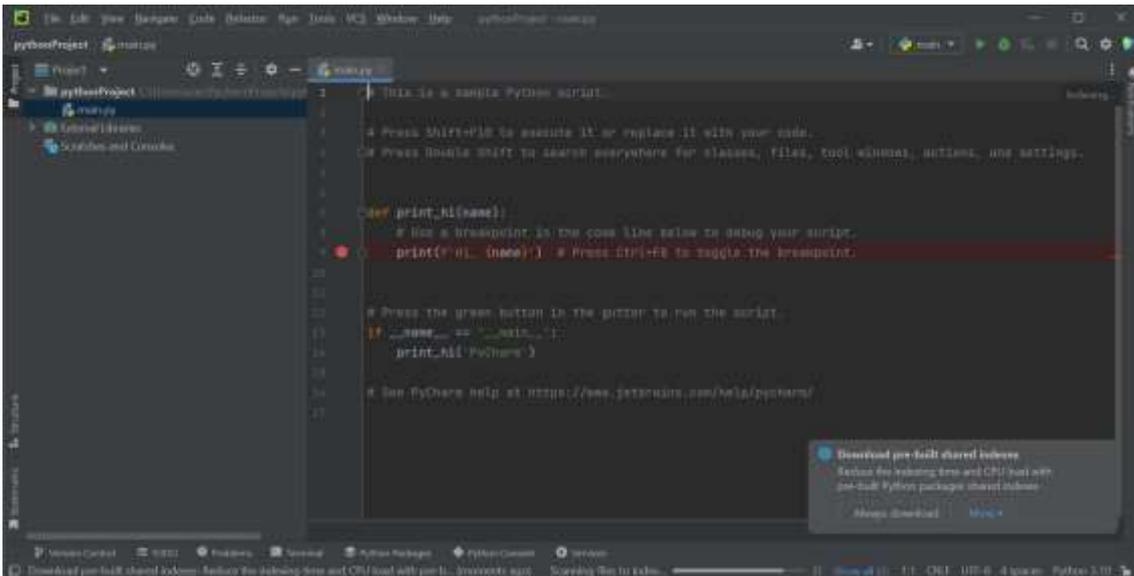
Откройте среду *PyCharm* и выберите *New Project*. В появившемся окне выберите радиокнопку *Previously configured interpreter*. Если вы предварительно установили Питон, то в выпадающем списке вы увидите установленную версию. Если же Питон не установлен, то перейдите по гиперссылке *Add Interpreter* в диалоговом окне в левом списке выберите *System Interpreter*, а затем в выпадающем списке *Interpreter* нужную версию Питона. По нажатию кнопки *Ok* начнется процесс установки.



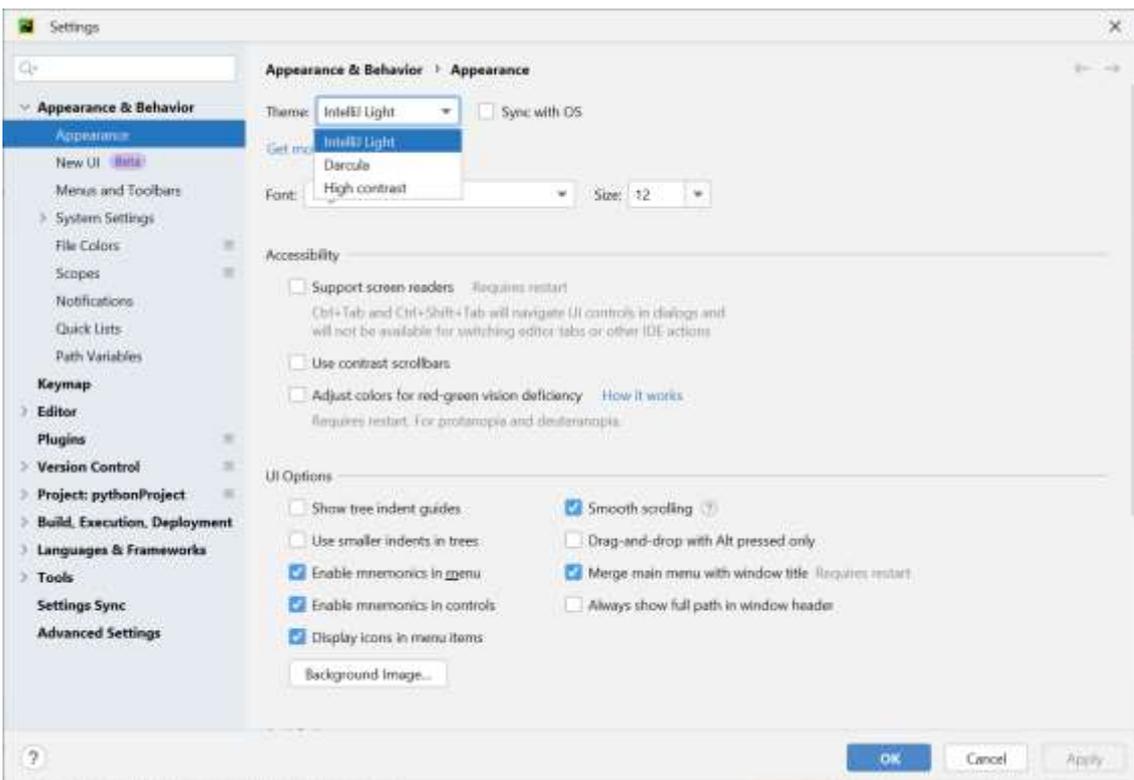
Появится окно с информацией об установленном Питоне. Нажмите кнопку *Create*



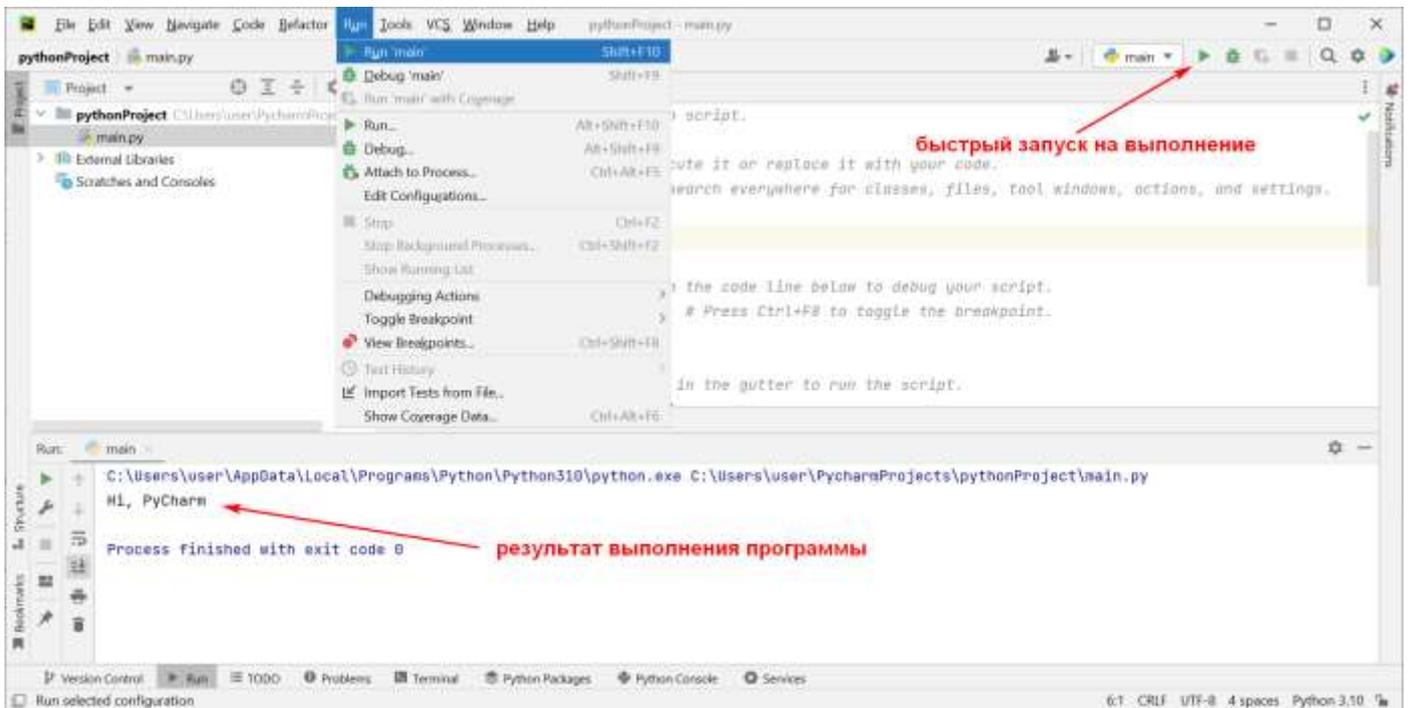
Вы создали новый проект Питона и файл кода с названием *main.py*



Темную схему среды (она называется *Darcula*) вы можете изменить на светлую, выбрав пункт меню *File-Settings...-Appearance & Behavior-Appearance*, а затем в выпадающем списке *Theme* выбрать светлую тему *IntelliJ Light*

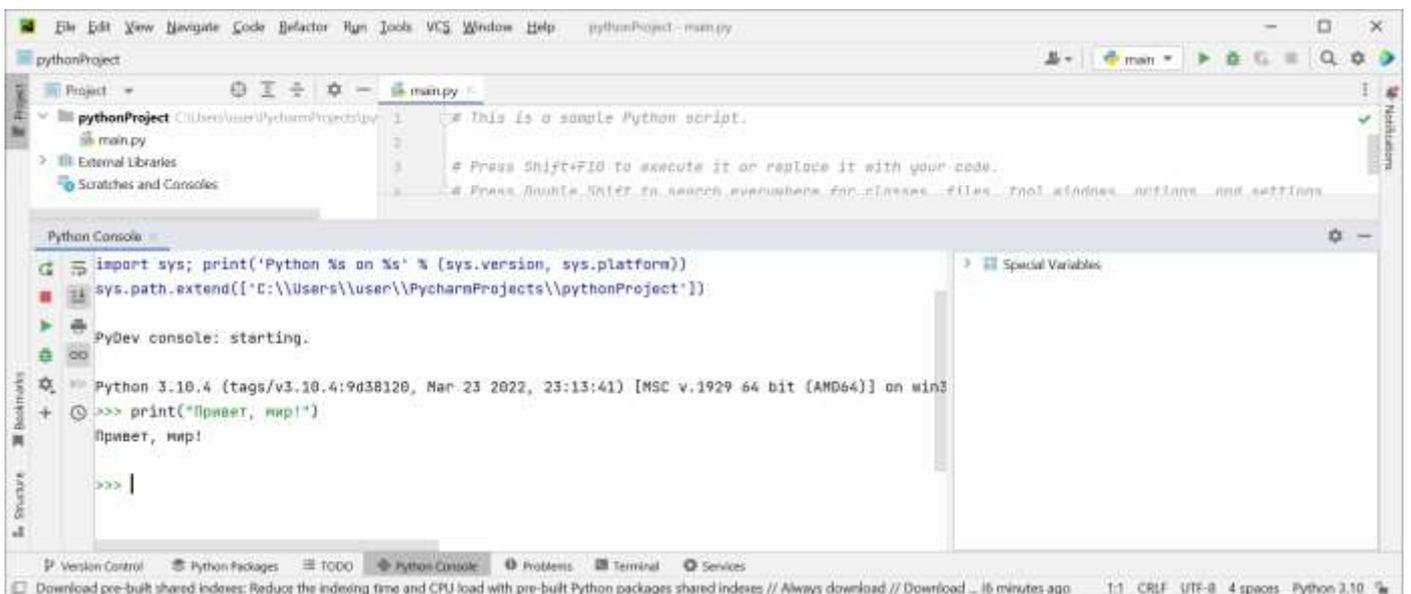


Созданный файл уже содержит некий код. Можете запустить эту программу на выполнение, выбрав пункт меню *Run-Run "имя программы"* или нажав клавиши *<Shift-F10>* или нажав кнопку с зеленым треугольником в правой части окна. Результат выполнения программы вы увидите на вкладке *Run* (переключение между вкладками в нижней части окна)



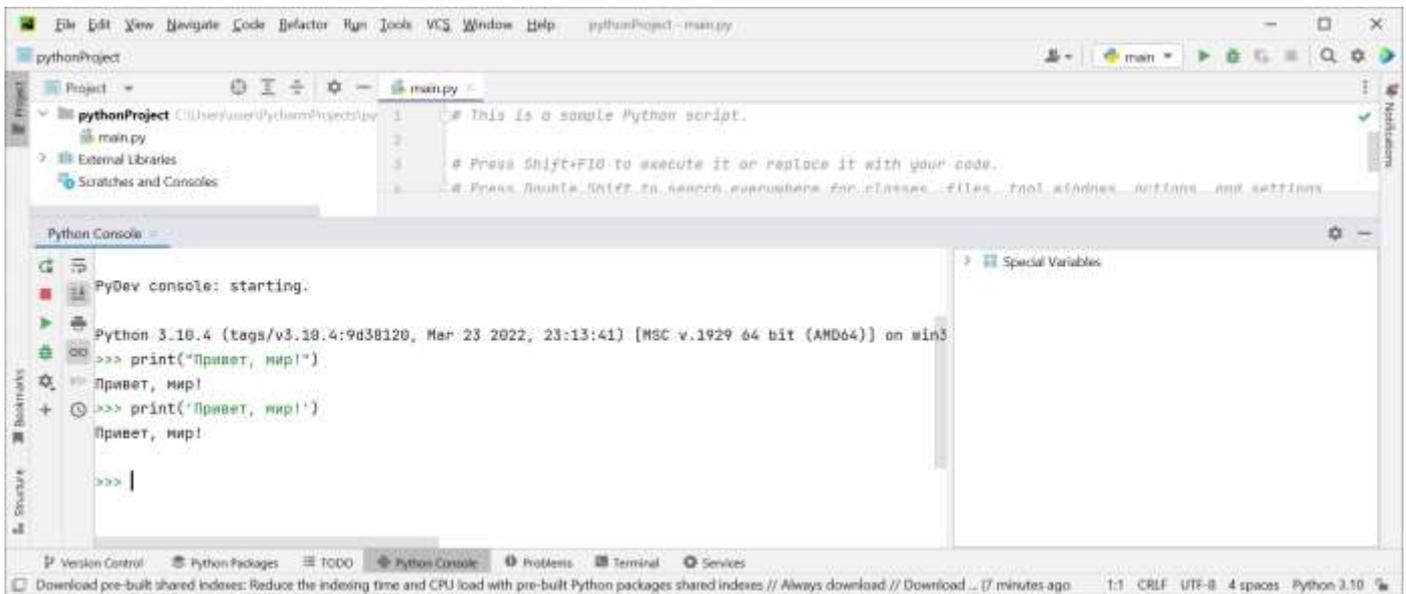
В среде можно работать в консольном режиме – в этом режиме в строке-приглашении к вводу >>> можно набирать операторы языка. В нем же отображаются вывод программы (то, что она печатает) и сообщения об ошибках. Для этого надо переключиться на вкладку *Python Console*

В строке-приглашении наберите традиционную первую программу – печать сообщения «Привет, мир!» и нажмите клавишу <Enter>.



В консоли появится этот текст. А строка-приглашение ожидает от вас новой команды

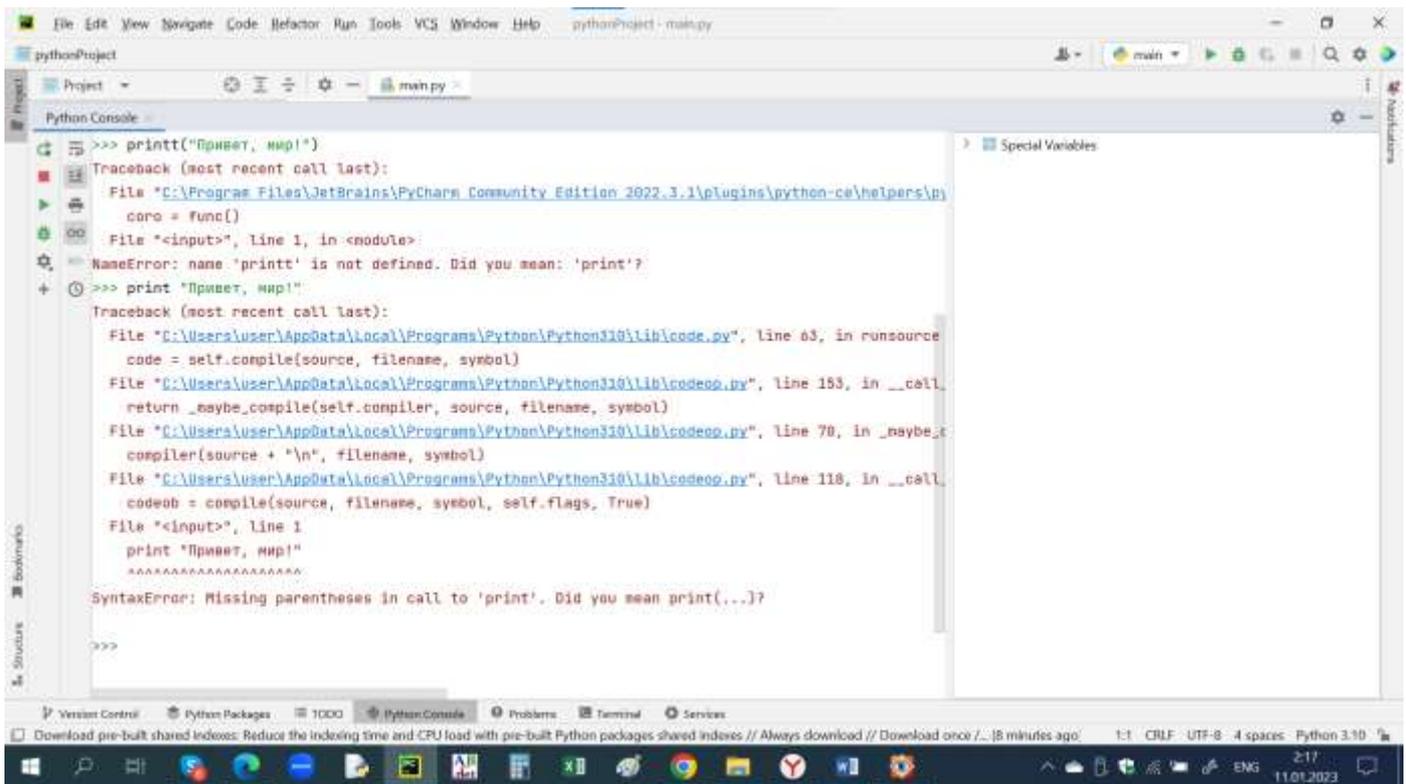
Попробуем еще раз ввести ту же самую команду, только заменив двойные кавычки на одинарные. **Обратите внимание, что Питон позволяет брать текст и в одинарные, и в двойные кавычки.**



Посмотрим, как Питон реагирует на синтаксические ошибки. Наберите команду с ошибкой `printt("Привет, мир!")`

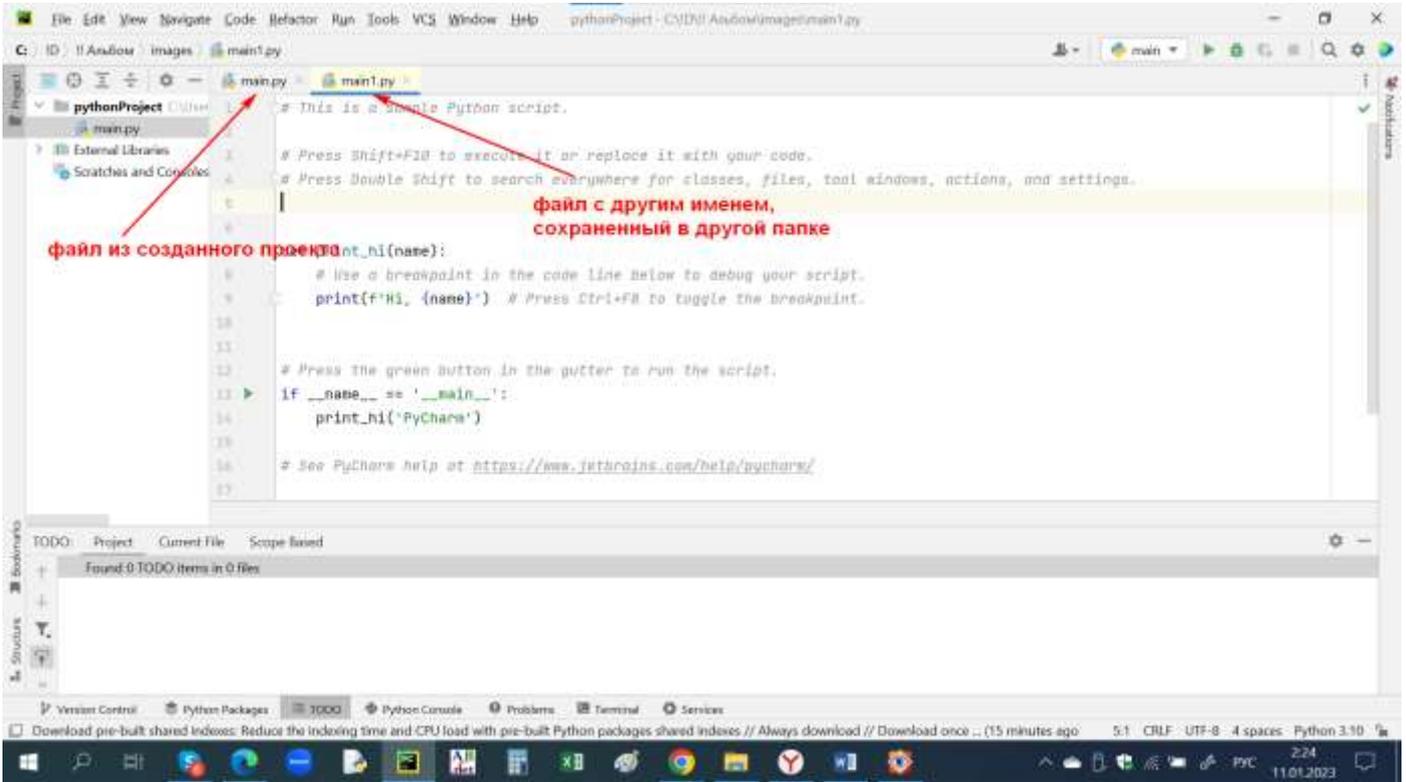
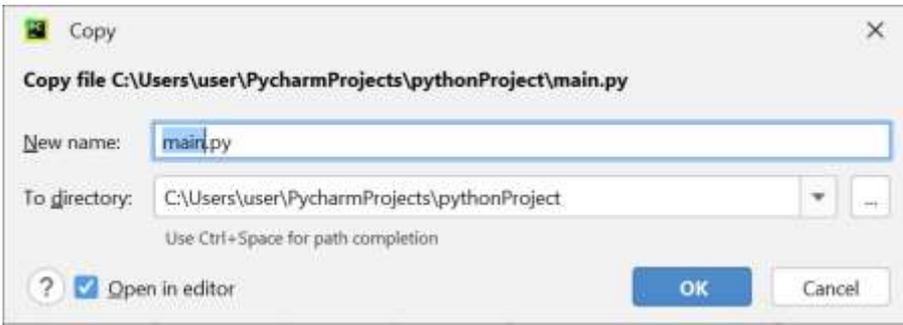
Произошла ошибка времени выполнения – Питон не нашел объекта с таким именем.

Наберите еще одну команду – здесь мы ошибемся, забыв набрать скобки. Питон отреагирует другим сообщением – сообщением о синтаксической ошибке – и скажет нам, что пропущены скобки

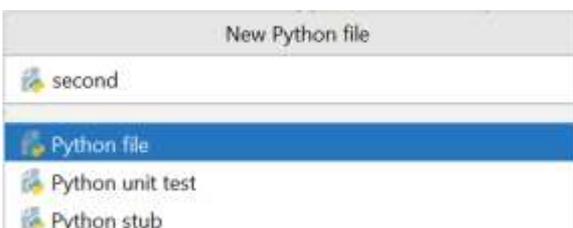
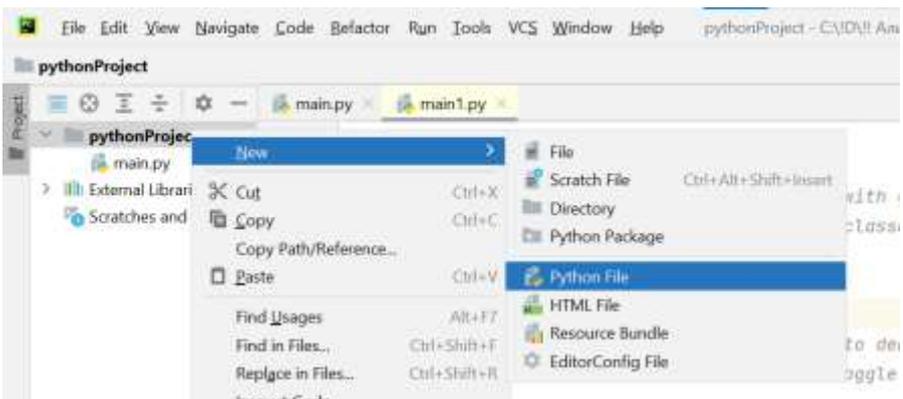


В интерактивном режиме окна консоли писать и сразу выполнять по одной команде в каждой строке не очень удобно, поэтому лучше записывать текст программы в отдельном файле и сохранять его, а потом запускать сразу все команды этого файла. Такой файл у нас уже создан по умолчанию – `main.py`. Сохранен он в папке проекта по умолчанию.

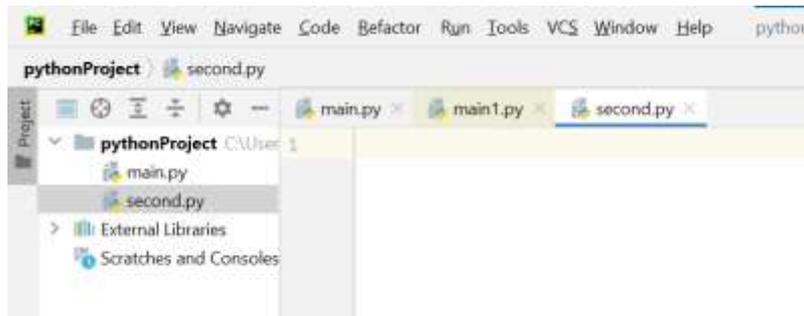
Вы можете сохранить его в другой папке, выбрав пункт меню `File–Save As...` – обратите внимание, что при этом происходит копирование файла в выбранную папку, а не в папку проекта.



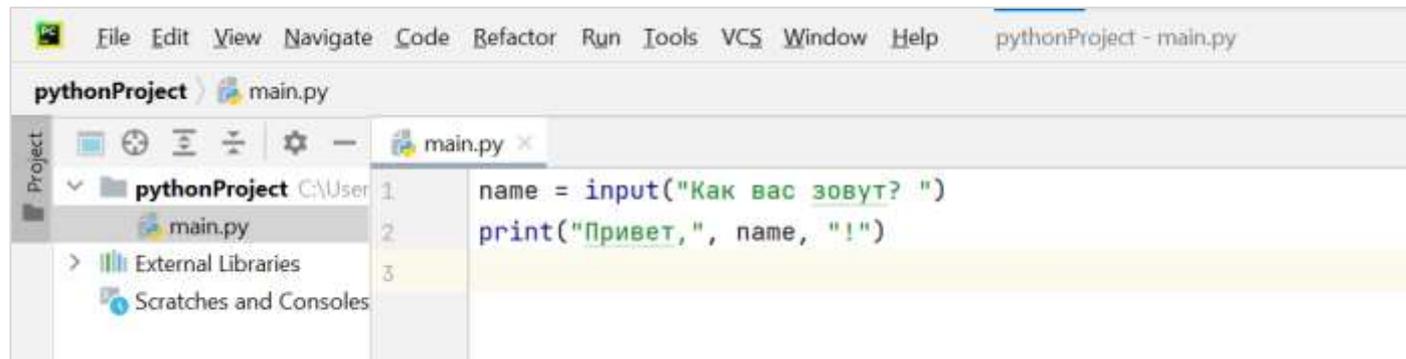
Добавить же новый файл в текущий проект можно, выбрав в контекстном меню проекта пункт *New–Python File* и задав затем его имя.



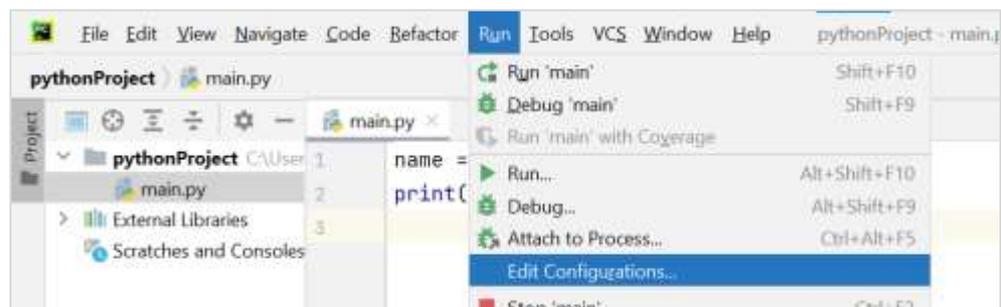
Созданный файл появляется в текущем проекте



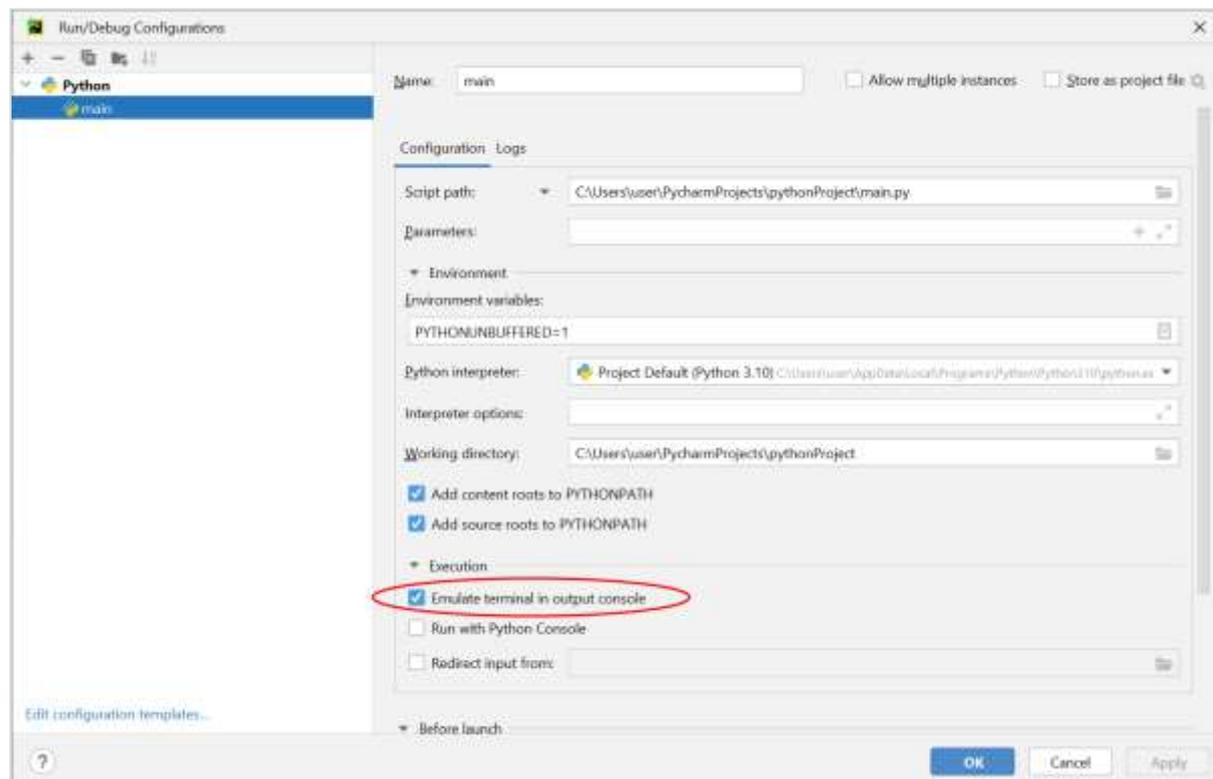
Попробуем создать программу. В файле *main.py* наберите ее текст.



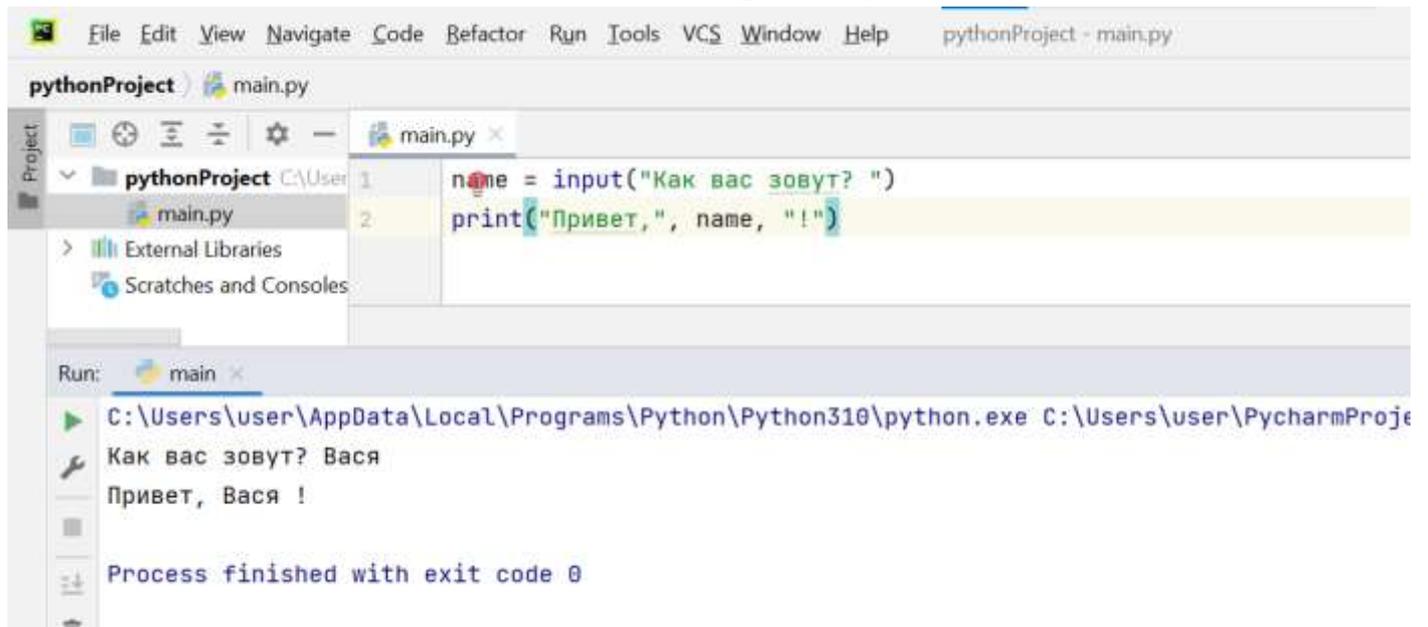
Перед запуском на выполнение (чтобы не было проблем с вводом данных) выберите пункт меню *Run–Edit Configurations...*



и установите флажок *Emulate terminal in output console*



Запустите программу на выполнение. Вы увидите приглашение ввести ваше имя. Введите имя, нажмите *<Enter>*, и на вкладке *Run* появится текст, приветствующий именно вас.



Теперь вы научились вводить данные и выводить их на экран!