Python в Visual Studio code

(автор: Вологжанин Е.А.)

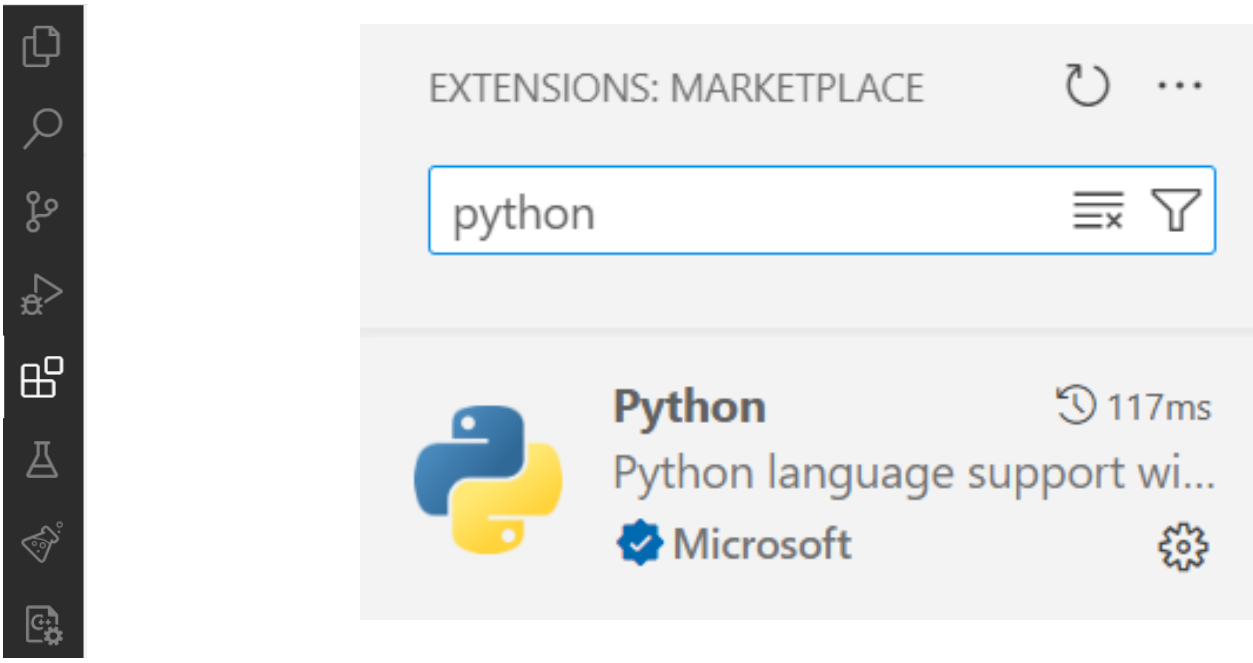
Работать с Python в Visual Studio Code с использованием расширения Microsoft Python просто, весело и продуктивно. Расширение делает VS Code отличным редактором Python и работает в любой операционной системе с различными интерпретаторами Python.

Перед началом установки python необходимо скачать и установить среду для редактирования кода: Visual Studio Code - Code Editing. Redefined

Установите Python и расширение Python

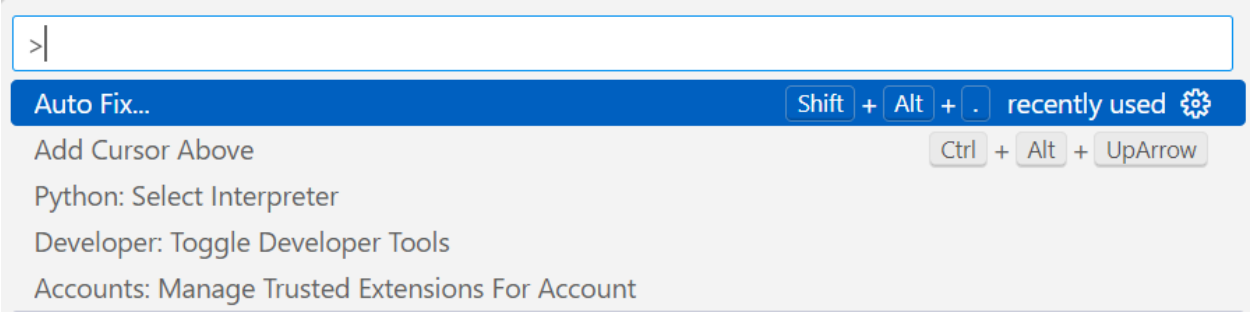
Вам необходимо установить интерпретатор Python самостоятельно, отдельно от расширения. Для быстрой установки используйте Python с python.org, а так же установите расширение из VS Code Marketplace, либо внутри VS Code.

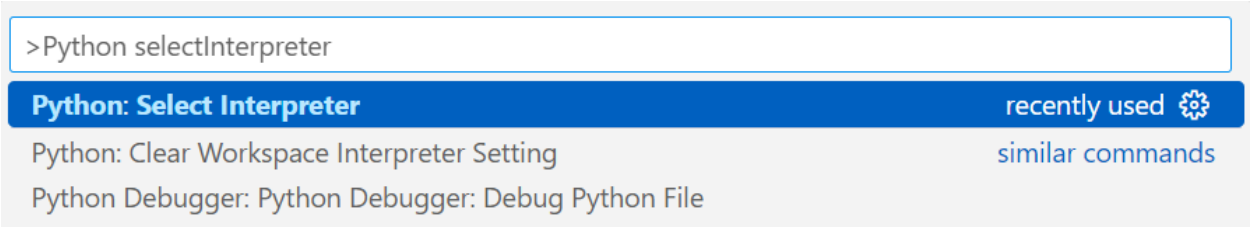
Войти в VS Code

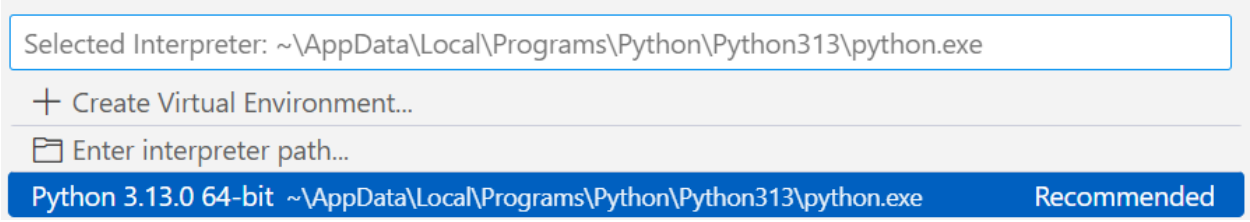


(Данного расширения будет достаточно для начала работы)

После установки версии Python выберите ее с помощью нажатия кнопки **F1** и ввода команды **«Python: Select Interpreter»**.





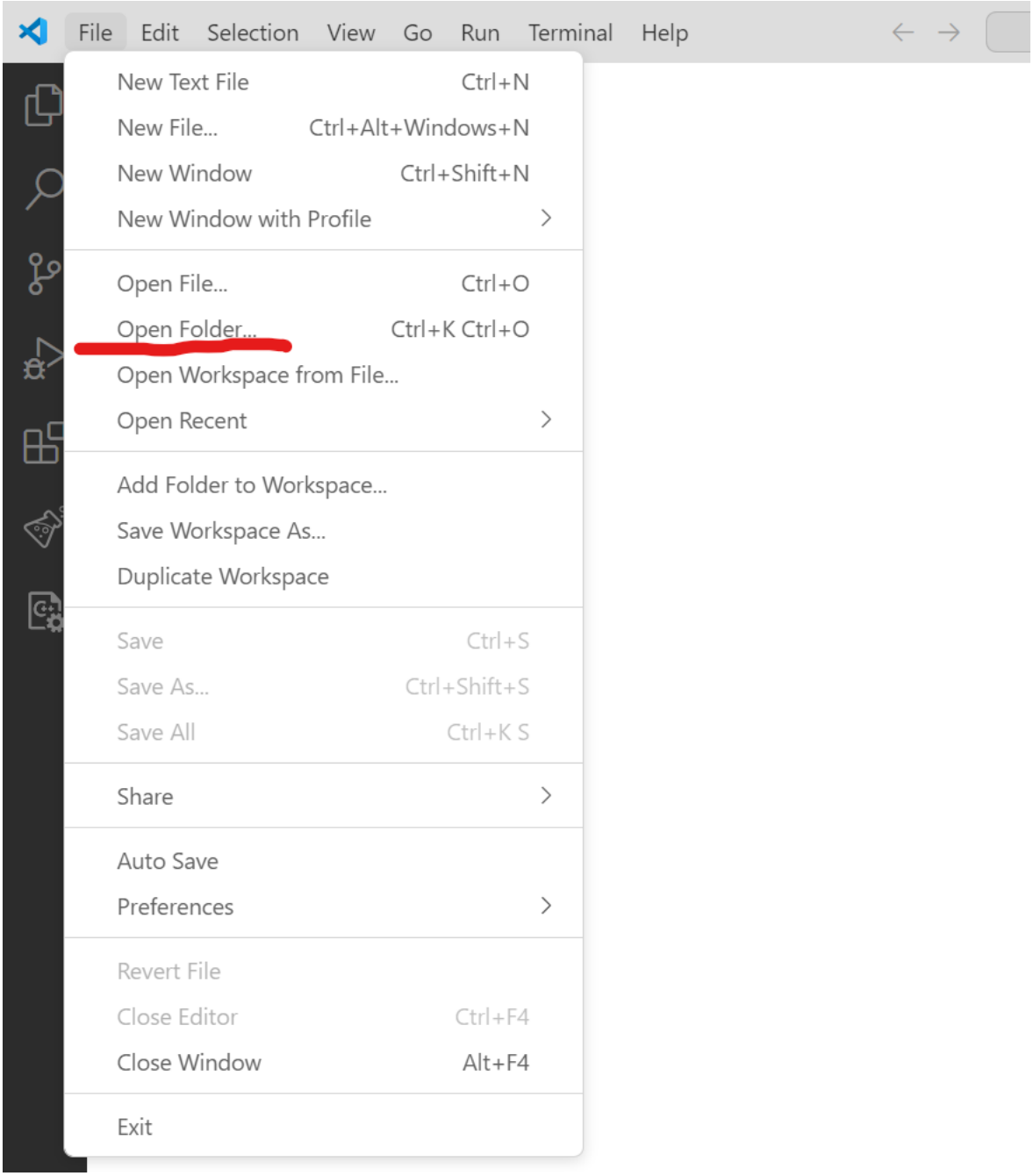


Если VS Code не находит автоматически нужный вам интерпретатор, обратитесь к разделу Environments - Manually Specify an interpretationr.

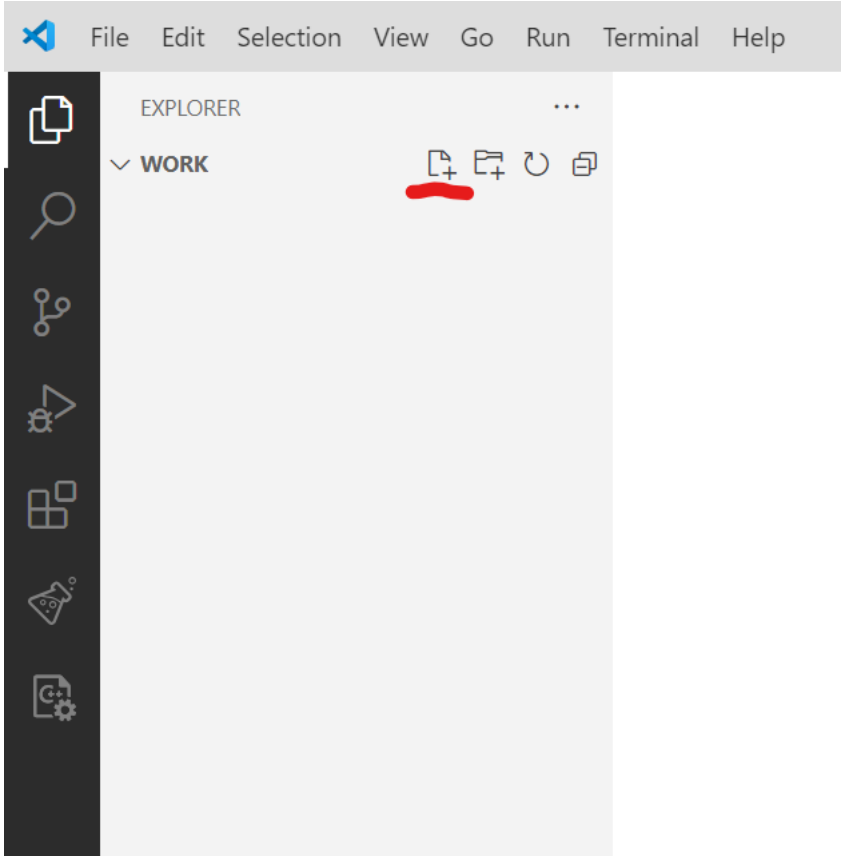
(При желании вы можете настроить расширение Python через настройки. Узнайте больше в справочнике настроек Python).

Запустить код Python

Чтобы опробовать Python, выберите папку, в которой собираетесь работать:



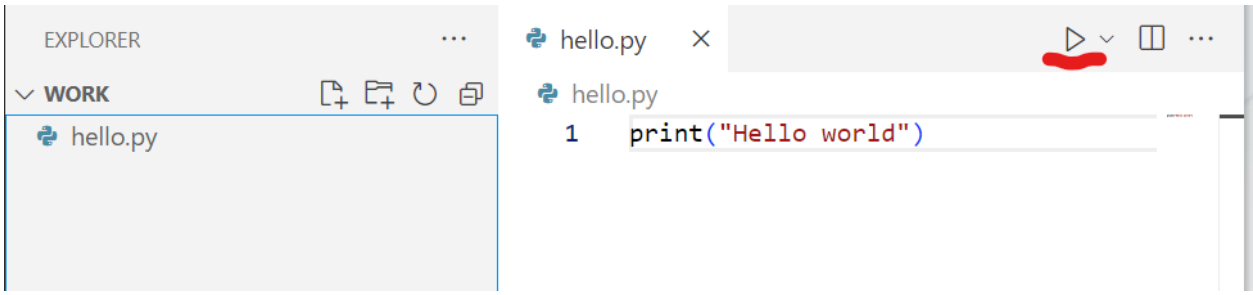
В выбранной папке создайте файл с именем **hello.py**



Вставьте в созданный файл следующий код: **print("Hello world")**

Расширение Python затем предоставляет ярлыки для запуска кода Python с использованием текущего выбранного интерпретатора.

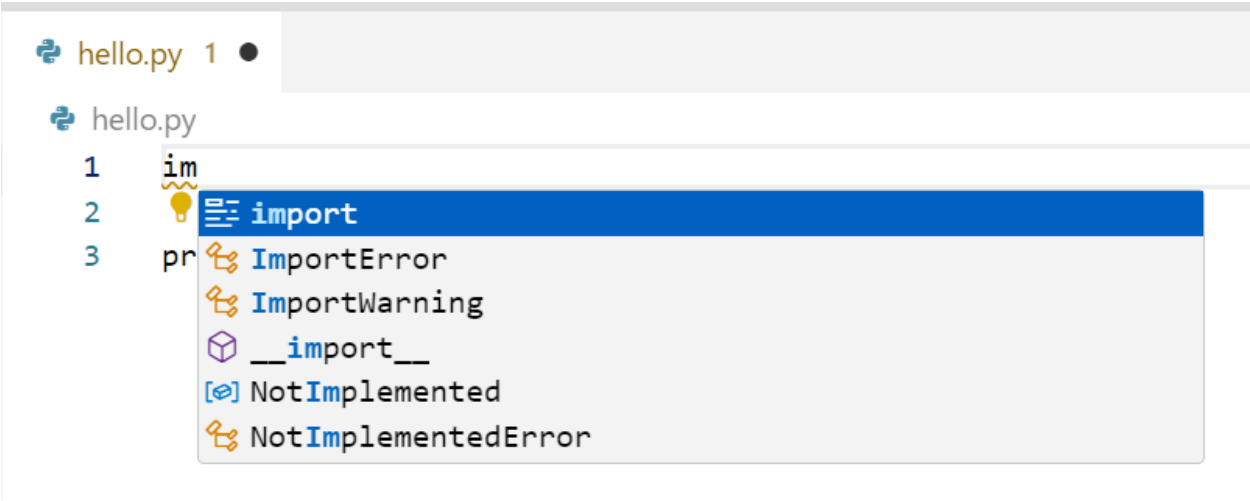
Чтобы запустить активный файл Python, нажмите кнопку воспроизведения **Run Python File** в правом верхнем углу редактора.



Автозаполнение и IntelliSense

Расширение Python поддерживает автодополнение кода и IntelliSense с использованием текущего выбранного интерпретатора. IntelliSense — это общий термин для ряда функций, включая интеллектуальное автодополнение кода (контекстный метод и предложения переменных) во всех ваших файлах и для встроенных и сторонних модулей.

Ctrl+Пробел



Установка и использование пакетов

Давайте продолжим пример, используя пакеты.

В Python пакеты — это способ получения любого количества полезных библиотек кода, обычно из PyPI , которые предоставляют дополнительную функциональность вашей программе. В этом примере вы используете numpy пакет для генерации случайного числа.

Вставьте следующий исходный код в файл hello.py:

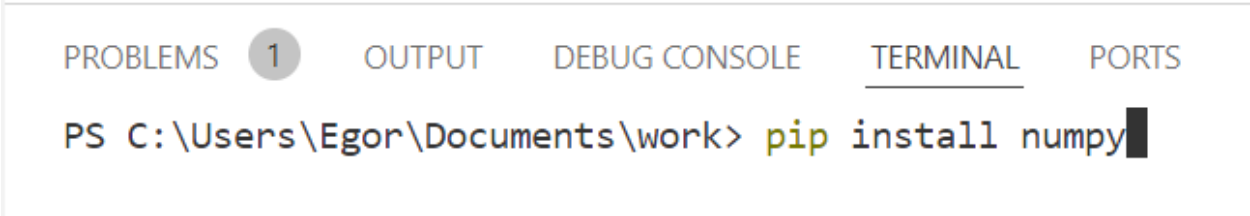
```
hello.py 1 ×
hello.py > ...
1   import numpy as np
2
3   msg = "Roll a dice!"
4   print(msg)
5
6   print(np.random.randint(1, 9))
```

При запуске программы вы должны увидеть сообщение **"ModuleNotFoundError: No module named 'numpy'"** . Это сообщение означает, что требуемый пакет недоступен в вашем интерпретаторе.

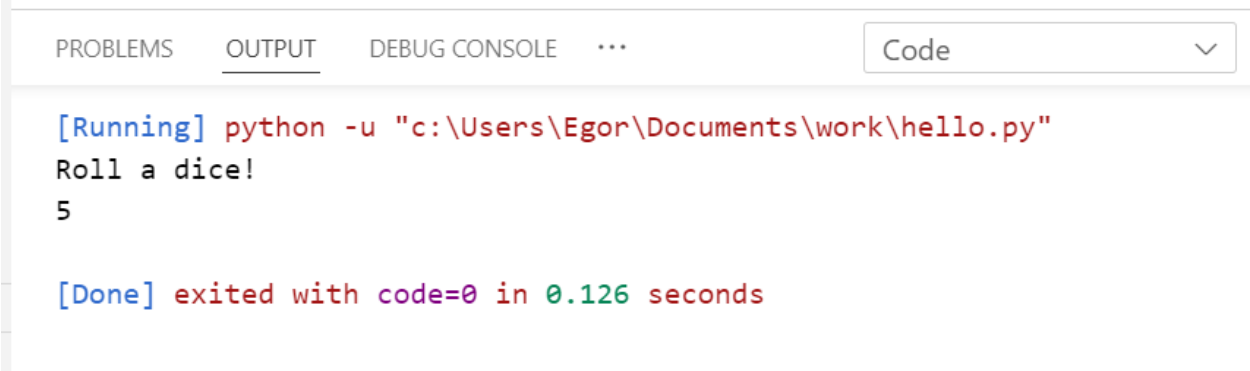
```
PROBLEMS 1   OUTPUT   DEBUG CONSOLE   TERMINAL   ···   Filter (e.g. text, **/*.ts, !*...
hello.py 1
  Import "numpy" could not be resolved  Pylance(reportMissingImports)  [Ln 1, Col 8]
```

Чтобы установить **numpy** пакет, используйте палитру команд для запуска терминала: Create New Terminal **(Ctrl+Shift+`)**. Эта команда открывает командную строку для выбранного вами интерпретатора.

Чтобы установить необходимые пакеты в вашей виртуальной среде, введите следующую команду:



Теперь при запуске программы не появляется ошибка и можно увидеть результат написанного кода:



Ссылки на оригинальные материалы:

- Python in Visual Studio Code
- Getting Started with Python in VS Code