**Введение в Python и его установка**

Надо будет поставить в VS code python.

Низкоуровневые языки работают непосредственно с аппаратной частью компьютера, поэтому скорость их работы, в отличии от высокоуровневых, выше.

**Процессор** – различает только два сигнала. Принято говорить о них как 1 и 0. Но по факту сигнал либо есть либо нет. Процессор работает только с командами состоящими из нулей и единиц – **машинным кодом.**

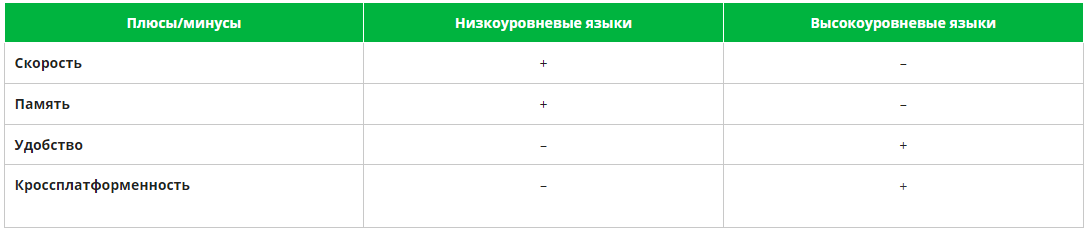
Низкоуровневые языки уже состоят из набора таких команд, поэтому эти команды и понятны компьютеру.

**Но эти языки не являются кроссплатформенными.** Ну то есть машинный код на **windows,** не получится просто запустить на **Linux.** Связано с архитектурой этих платформ.

**Высокоуровневые языки** наоборот могут запускать код не зависимо от операционной системы. И синтаксис этих языков более понятный.

**Python** кстати к синтаксису довольно строг. Особенно к табуляции.

Также программы написанные на высокоуровневых языках, требуют для своей работы и хранения значительно больше памяти, чем низкоуровневые.



**Ещё языки программирования делятся на компилируемые и интерпретируемые.**

В **компилируемых** языках код программы преобразуется в машинный код с помощью компилятора. На выходе получается новая скомпилированная программа, которую выполнит компьютер.

**Интерпретатор** принимает удобный для восприятия человеком код и начинает построчно исполнять его, преобразуя каждую строку в машинный код. Новой программы по итогу не случится.

**JS** интерпретируемый, как и **python** и **PHP.**

**C++, C#, Java, Golang –** компилируемые.

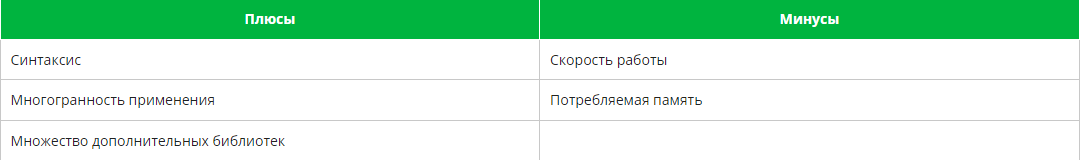
**Версии Python**

В январе 2020 года прекратилась поддержка **Python 2.7.18**

Сейчас самая актуальная версия это **Python 3.** Но и среди них есть устаревшие версии.

**Python 3.4 включительно и старше постепенно перестают поддерживаться разработчиками**

**Актуальные:** любая из версий 3.5, 3.6, 3.7, 3.8. Постепенно ранние версии также прекращают свою поддержку, поэтому старайтесь пользоваться наиболее поздней из доступных версий. В данном курсе будем пользоваться **Python 3.8.**



Ну зато честно.

**А вот и нет. Хотя ладно. Python 3.10.6 и 3.8 Скачаю оба. А на ноутбуке у меня вообще 3.9 еще стоит.**

Если что-то будет не работать, просто включу другой интерпретатор.

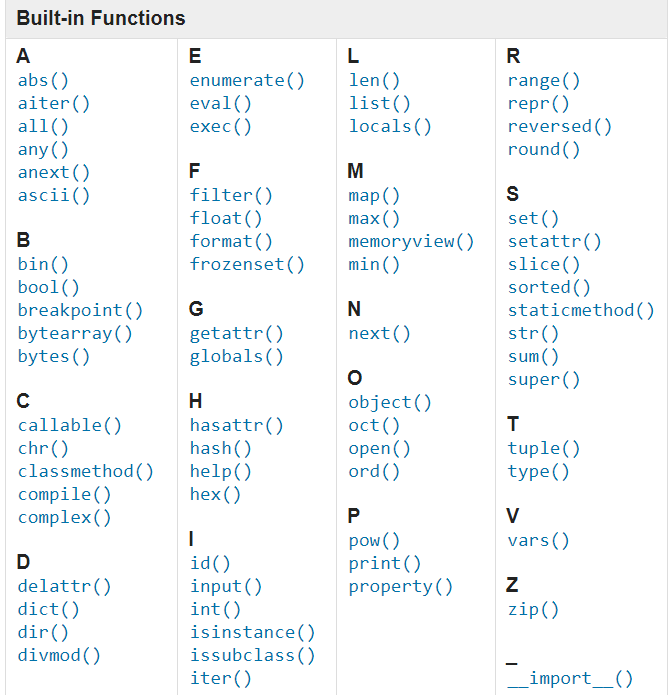
**Как работать с документацией по Python**

Тупо ссылки

* [Модули для работы с текстовым форматом данных](https://docs.python.org/3/library/text.html). Например, строки, регулярные выражения.
* [Расширенные типы данных](https://docs.python.org/3/library/datatypes.html). Позволяют работать с датами, очередями, массивами.
* [Модули для работы с числами](https://docs.python.org/3/library/numeric.html). Модули с расширенными математическими операциями.
* [Взаимодействие с файловой системой](https://docs.python.org/3/library/filesys.html).
* [Работа с операционной системой](https://docs.python.org/3/library/allos.html).
* [Простейший графический интерфейс](https://docs.python.org/3/library/tk.html).

Тут чуть полезнее. Список встроенных функций с их описанием.

<https://docs.python.org/3/library/functions.html>



Встроенных типов данных меньше чем этих функций. Основные: целые числа, числа с плавающей запятой, строки, списки и словари. Словари, насколько помню, что-то вроде объектов. Могу ошибаться.