

ЗНАКОМСТВО С ЯЗЫКОМ ПРОГРАММИРОВАНИЯ



Компьютерная программа



Компьютерная программа — список команд (инструкций) для компьютера. Команды могут быть любыми, например:

- ▶ считать информацию с клавиатуры;
- ▶ произвести арифметические вычисления (+, −, *, /);
- ▶ вывести информацию на экран.

Для создания программ используются языки программирования. Выбор языка программирования, как правило, продиктован особенностями самой программы.

Язык программирования

- ▶ Язык программирования — набор определенных правил, по которым компьютер может понимать команды (инструкции) и выполнять их. Текст программы на любом языке программирования, называется программным кодом.
- ▶ Языки программирования бывают компилируемые и интерпретируемые. Если программа написана на компилируемом языке (C, C++, Pascal), то перед выполнением её нужно полностью проверить на наличие синтаксических ошибок и уже после этого перевести в понятную для компьютера форму — машинный код. Это делает специальная программа, которая называется компилятором.
- ▶ Если программа написана на интерпретируемом языке (Python, PHP, Ruby), она не переводится в машинный код целиком. Вместо этого специальная программа, которая называется интерпретатором, идет по коду, анализирует его и выполняет каждую отдельную команду.

Python - это

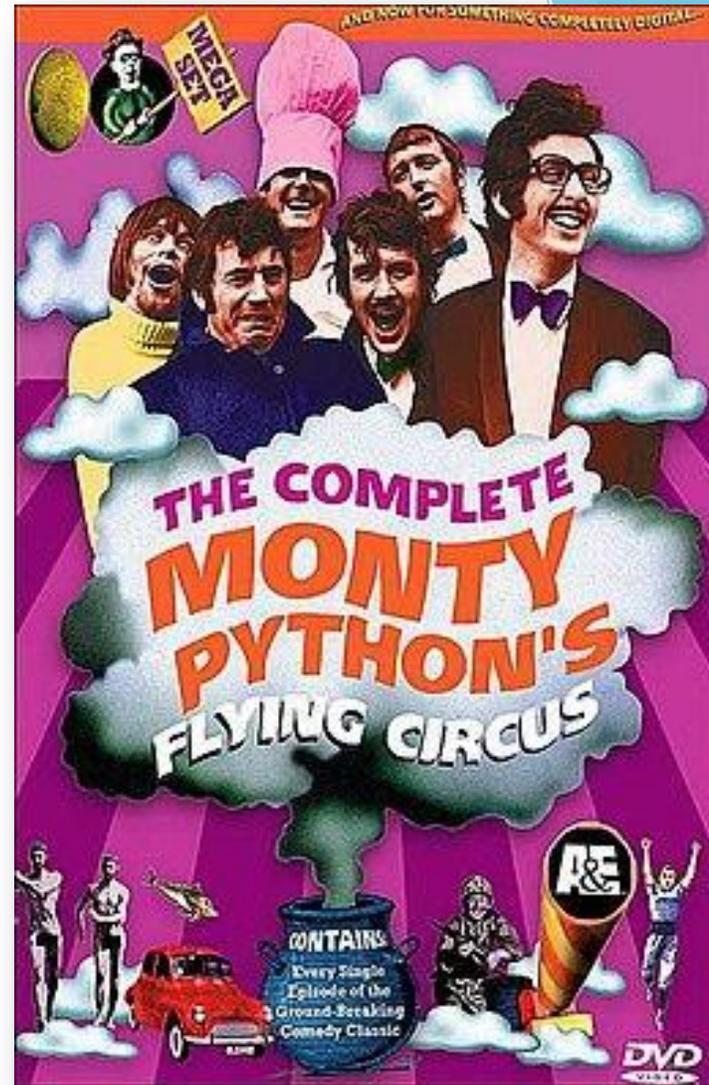


Мощный и простой язык программирования, на котором можно писать проекты любого масштаба.

Язык активно развивается и новые версии выходят каждые 2-3 года.

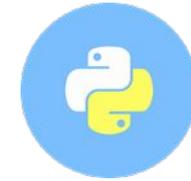


Официальный сайт <http://python.org>.



Язык программирования Python был создан в 1991 году голландцем Гвидо ван Россумом

Питон или пайтон как правильно произносить?



Часто язык называют просто «питон», так же название языка и переводится.

Но в сообществе на такое произношение могут обратить внимание, потому что правильным и каноничным считается говорить «пайтон». Ведь создатель языка Гвидо Ван Россум назвал язык в честь шоу «Летающий цирк Монти Пайтона», которое он очень любил.

Для чего используют



Основные области, в которых используется Python:

- веб-разработка;
- машинное обучение;
- проекты с искусственным интеллектом, нейросети;
- Data Science, аналитика;
- создание игр.

1. *Кроссплатформенный*, потому что Python работает почти на всех известных операционных системах, включая: Linux, Window, FreeBSD, Macintosh, Solaris и т.д.
2. *Интерактивный*, потому что позволяет в режиме реального времени взаимодействовать с интерпретатором.
3. *Интерпретируемый*, потому что не требует компиляций для выполнения кода

«Hello, World!»		
C++	Java	Python
<pre># include < iostream> using name space std ; int main () { cout<< " Hello , world ! " <<endl ; return 0; }</pre>	<pre>classMyClass{ public static void main (String[] args){ System.out.println ("Hello, World!"); } }</pre>	<pre>Print("Hello, World!")</pre>

Проекты, в которых используется Python



- ▶ Компания Google использует Python в своей поисковой системе;
- ▶ Компании Intel, Cisco, Hewlett-Packard, Seagate, Qualcomm и IBM, используют Python для тестирования аппаратного обеспечения;
- ▶ Сервис YouTube в значительной степени реализован на Python;
- ▶ Агентство национальной безопасности (NSA) использует Python для шифрования и анализа данных;
- ▶ Компании JPMorgan Chase, UBS, Getco и Citadel применяют Python для прогнозирования финансового рынка;
- ▶ Программа BitTorrent для обмена файлами в пиринговых сетях написана на языке Python;
- ▶ NASA, Los Alamos, JPL и Fermilab используют Python для научных вычислений.

Плюсы Python



- Хорошо подходит для новичков.
- Простой синтаксис: код легко писать, читать и поддерживать.
- Большая стандартная библиотека и много дополнительных библиотек.
- Поддерживает объектно-ориентированное программирование.
- Кроссплатформенность и поддержка почти всех современных систем.

Минусы Python



- Низкая скорость.
- Плохо подходит для разработки мобильных приложений.
- Из-за динамической типизации выше вероятность ошибки при запуске, нужно больше тестов.
- Не подходит для работы с памятью на низком уровне.

Философия Python в оригинале



- ▶ Although practicality beats purity;
- ▶ Errors should never pass silently;
- ▶ Unless explicitly silenced;
- ▶ Beautiful is better than ugly;
- ▶ Explicit is better than implicit;
- ▶ Simple is better than complex;
- ▶ Complex is better than complicated;
- ▶ Flat is better than nested;
- ▶ Sparse is better than dense;
- ▶ Readability counts;
- ▶ Special cases aren't special enough to break the rules;
- ▶ In the face of ambiguity, refuse the temptation to guess;
- ▶ There should be one – and preferably only one – obvious way to do it;
- ▶ Although that way may not be obvious at first unless you're Dutch;
- ▶ Now is better than never;
- ▶ Although never is often better than *right* now;
- ▶ If the implementation is hard to explain, it's a bad idea;
- ▶ If the implementation is easy to explain, it may be a good idea;
- ▶ Namespaces are one honking great idea – let's do more of those!

Философия Python с переводом

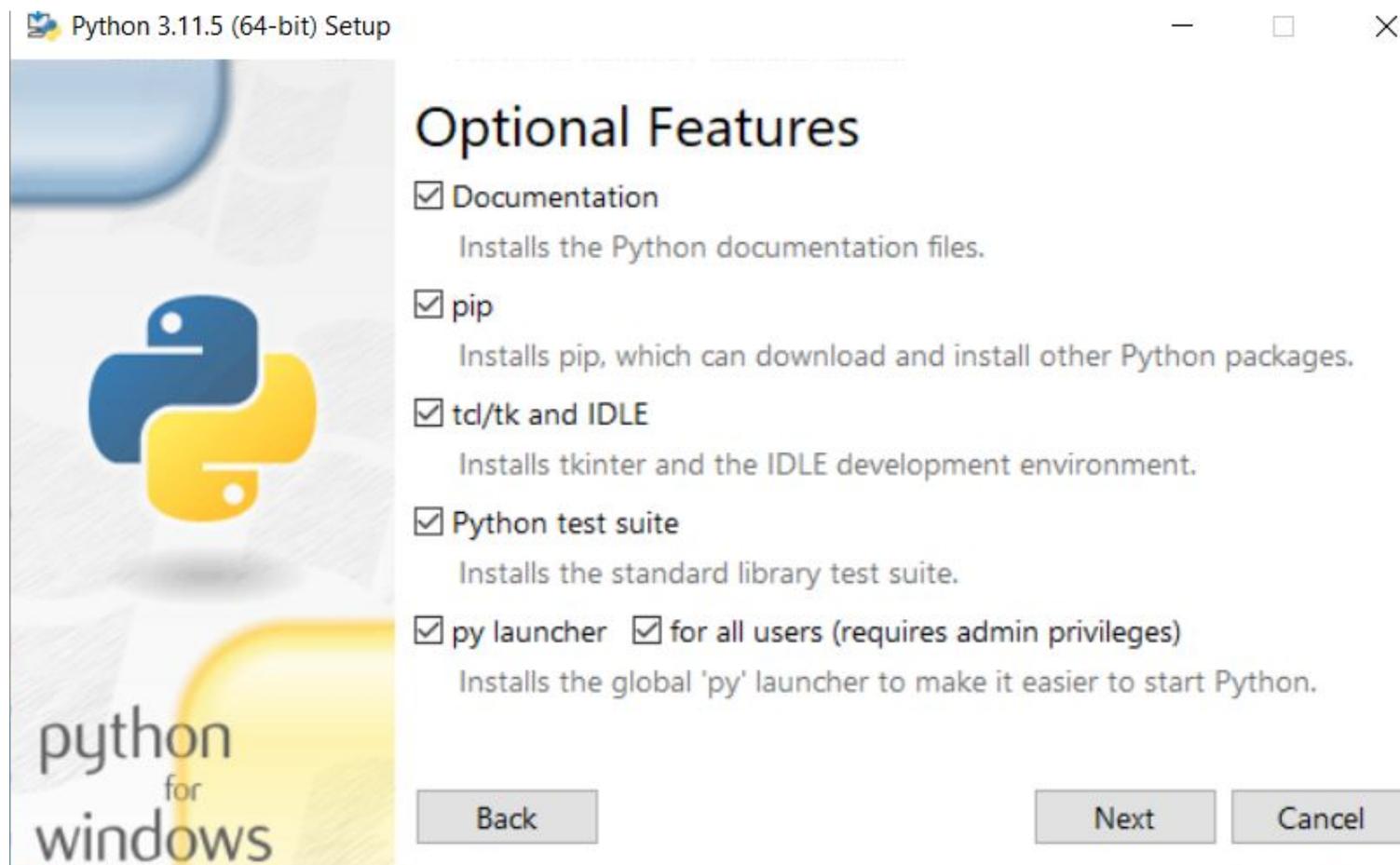


- ▶ Красивое лучше, чем уродливое;
- ▶ Явное лучше, чем неявное;
- ▶ Простое лучше, чем сложное;
- ▶ Сложное лучше, чем запутанное;
- ▶ Плоское лучше, чем вложенное;
- ▶ Разреженное лучше, чем плотное;
- ▶ Читаемость имеет значение;
- ▶ Особые случаи не настолько особые, чтобы нарушать правила;
- ▶ При этом практичность важнее безупречности;
- ▶ Ошибки никогда не должны замалчиваться;
- ▶ Если не замалчиваются явно;
- ▶ Встретив двусмысленность, отбрось искушение угадать;
- ▶ Должен существовать один — и, желательно, только один — очевидный способ сделать это;
- ▶ Хотя он поначалу может быть и не очевиден, если вы не голландец;
- ▶ Сейчас лучше, чем никогда;
- ▶ Хотя никогда зачастую лучше, чем прямо сейчас;
- ▶ Если реализацию сложно объяснить — идея плоха;
- ▶ Если реализацию легко объяснить — идея, возможно, хороша;
- ▶ Пространства имён — отличная вещь! Давайте будем делать их больше!

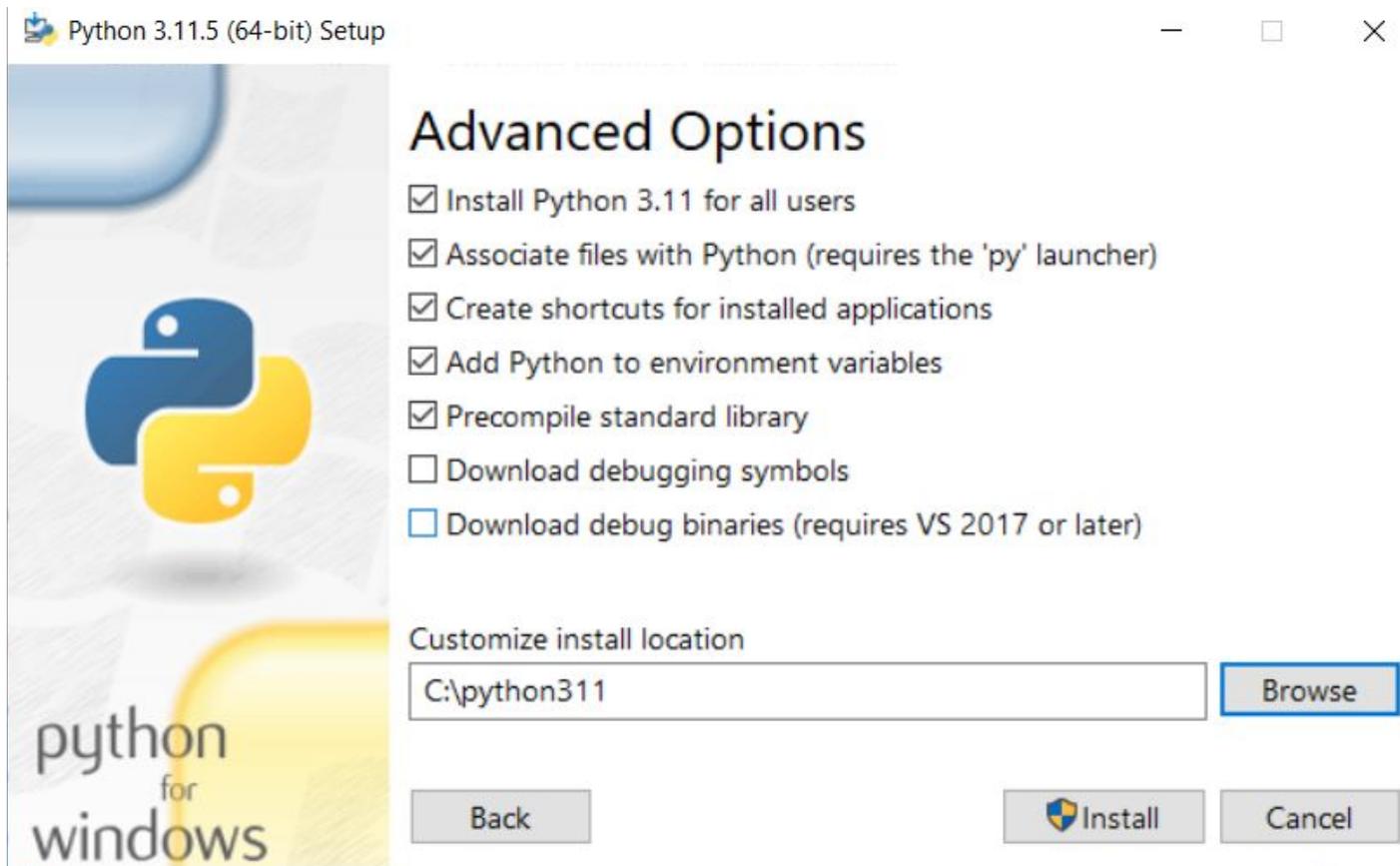
Установка среды



Отметьте необходимые опции установки



Выберете место установки



После успешной установки вас ждет следующее сообщение.



