

Инструменты и средства программирования

Тема 3. Знакомство с Python.



Немного истории

Автор: Guido van Rossum

Создателю нравилась британская комедийная передача Monty Python's flying circus.

Язык появился в 1991 году.

Мажорные версии:

- 1994 — Python 1.0
- 2000 — Python 2.0
- 2008 — Python 3.0

Версии и Интерпретаторы Python

Существует 2 версии языка Python:

- Python 2.x.x — не поддерживается с 2020 года
- Python 3.x.x — активная версия 3.9.1

Интерпретаторы:

- CPython
- IronPython
- Jython
- PyPy
- etc..

Что такое Python ?

Высокоуровневый динамический строго типизированный
интерпретируемый язык программирования общего
назначения.

Высокоуровневый

Высокий уровень абстракции от деталей исполняющей системы.

Динамический

Типы выводятся и проверяются во время выполнения (run-time).

С версии 3.5 можно также использовать compile-time проверки.

Строго типизированный

Сильно ограничены неявные приведения типов
(строка + число и т.п.).

Интерпретируемый

Команды языка интерпретируются во время работы специальной программой-интерпретатором.

Общего назначения

Нет ограниченной области применения
(в противоположность domain-specific языкам).

Характерные черты

- Кроссплатформенный
- Мультипарадигменный (процедурное, ООП, функциональное, метапрограммирование, ...)
- Автоматическое управление памятью

Pros & Cons



- Легкая разработка — краткость и выразительность
- Легко читать код — читабельность как одна из целей в основе дизайна языка
- Легко отлаживать — интерактивная проверка, можно всё динамически посмотреть



- Медленный
- Нет статических проверок компилятора*

The Zen of Python, by Tim Peters

- 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.
- Although practicality beats purity.
- Errors should never pass silently.
- Unless explicitly silenced.
- 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!

Где используется и кому нужен

В значительном количестве крупных компаний разрабатывающих ПО или обрабатывающих данные так или иначе используется Python.

Пруфы:

- [ru.wiki] <http://bit.ly/12ycKDm>
- [en.wiki] <http://bit.ly/1iAj5Gy>.

Кому пригодится:

- Backend engineer
- Machine Learning Engineer
- Data Scientist
- DevOps Engineer
- QA Engineer

Установка и настройка

Установка Python:

- Установка с официального сайта - <https://www.python.org/downloads/>
- С использованием пакетных менеджеров

PIP - пакетный менеджер.

Также есть Anaconda и Conda(мини версия анаконды), которая упростит вашу жизнь в работе с Python и установкой зависимостей.