



Python初探

田宇伟 (fishtai0)

2017年3月

TIOBE Index for March 2017

Mar 2017	Mar 2016	Change	Programming Language	Ratings	Change
1	1		Java	16.384%	-4.14%
2	2		C	7.742%	-6.86%
3	3		C++	5.184%	-1.54%
4	4		C#	4.409%	+0.14%
5	5		Python	3.919%	-0.34%
6	7	^	Visual Basic .NET	3.174%	+0.61%
7	6	v	PHP	3.009%	+0.24%
8	8		JavaScript	2.667%	+0.33%
9	11	^	Delphi/Object Pascal	2.544%	+0.54%
10	14	^^	Swift	2.268%	+0.68%

Source: www.tiobe.com

什么是Python

n. 巨蟒； 大蟒

计算机编程语言

名字由来

Python语言的创始人，吉多·范罗苏姆（Guido van Rossum）是自七十年代风靡全球的英国六人喜剧团体巨蟒剧团（Monty Python）的忠实粉丝，因此其给自己新创造的计算机语言起名Python。

为什么选择Python

- 简单
- 免费、开源
- 高级语言
- 可移植性
- 交互性
- 解释性
- 面向对象
- 可扩展性
- 可嵌入性
- 丰富的库

Python缺点

- 性能相对不是特别好，运行效率差，速度慢
- Python的多线程由于其解析器GIL（Global Interpreter Lock，全局解释器锁）存在无法并行运行
- 代码缩进决定代码的逻辑关系

为什么是Python3

"Python 2.x is legacy, Python 3.x is the present and future of the language."^{note}

安装Python3

Ubuntu上使用 **linuxbrew** 包管理器安装 python3:

```
$ brew install python3
```

检查python3版本:

```
$ python3 --version  
Python 3.6.0
```

运行Python

启动Python，在其 **REPL**^注 中直接输入相应的命令

```
$ python3
Python 3.6.0 (default, Mar  4 2017, 12:32:34)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

可以立刻查看命令运行结果，方便进行一些快速实验验证

注: **REPL** —— **Read-Eval-Print-Loop** 读取-执行-输出-循环

运行Python

- 将完整的代码写成 **.py** 脚本，比如 `hello.py` ，然后通过 `python hello.py` 执行

```
# hello.py '#' 开头的行会被注释掉  
print("Hello World!")
```

IPython

为什么不用Python自带交互式解释器

- 不能在退出时保存历史记录以备未来查询
- 不支持Tab自动补全
- 不能快速获得模块/函数/类的信息，如参数、文档、原始代码等
- 不方便在交互环境下执行Shell命令

安装IPython: `$ pip3 install ipython`

```
$ ipython --version
5.3.0
```

IPython

- **获得对象信息**：输入你想查看的对象，然后加上一个或者两个 **?**，就能获得多种对象信息。比如“**exit?**”
- **Magic函数**：IPython有很多Magic函数，分为两种类型。一种是**Line magics**，单行函数，需要使用 **%** 开头；另一种是**Cell Magics**，多行函数或者希望执行其他语言的代码，需要使用 **%%** 开头。
- **调用系统Shell命令**。只需在命令前加**!**即可。
- **Tab自动补全**。
- **历史记录**。

Python开发工具

编辑器

- Sublime Text
- Visual Studio Code (开源, 免费)
- Atom (开源, 免费)
- Vim (开源, 免费)
- Emacs (开源, 免费)

集成开发环境 (IDE)

- PyCharm (有免费版)

Python之禅

```
In [1]: import this
```

```
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.
```

```
... ..
```