

Python 基础教程



Python是一种解释型、面向对象、动态数据类型的高级程序设计语言。

Python由Guido van Rossum于1989年底发明，第一个公开发行人版发行于1991年。

像Perl语言一样, Python 源代码同样遵循 GPL(GNU General Public License) 协议。

本教程主要针对Python 2.x版本的学习，如果你使用的是Python 3.x版本请移

步至[Python 3.X版本的教程](#)。

谁适合阅读本教程？

本教程适合想从零开始学习Python编程语言的开发人员。当然本教程也会对一些模块进行深入，让你更好的了解Python的应用。

学习本教程前你需要了解

在继续本教程之前，你应该了解一些基本的计算机编程术语。如果你学习过PHP，ASP等编程语言，将有助于你更快的了解Python编程。

执行Python程序

对于大多数程序语言，第一个入门编程代码便是"Hello World！"，以下代码为使用Python输出"Hello World！"：

实例(Python 2.0+)

```
#!/usr/bin/python
print "Hello, World!"
```

[运行实例 »](#)

Python 3.0+版本已经把print作为一个内置函数，正确输出"Hello World！"代码如下：

实例(Python 3.0+)

```
#!/usr/bin/python3
print("Hello, World!")
```

[运行实例 »](#)

相关文章

[Python2.7.14 中文手册](#)[Python IDE\(集成开发环境\)介绍](#)[Python 编码规范\(Google\)](#)[Python 3.X版本的教程](#)[Python 简介 →](#)

1 篇笔记

[写笔记](#)

关于脚本第一行的 `#!/usr/bin/python` 的解释，相信很多不熟悉 Linux 系统的同学需要普及这个知识，脚本语言的第一行，只对 Linux/Unix 用户适用，用来指定本脚本用什么解释器来执行。

有这句的，加上执行权限后，可以直接用 `./` 执行，不然会出错，因为找不到 python 解释器。

`#!/usr/bin/python` 是告诉操作系统执行这个脚本的时候，调用 `/usr/bin` 下的 python 解释器。

`#!/usr/bin/env python` 这种用法是为了防止操作系统用户没有将 python 装在默认的 `/usr/bin` 路径里。当系统看到这一行的时候，首先会到 `env` 设置里查找 python 的安装路径，再调用对应路径下的解释器程序完成操作。

`#!/usr/bin/python` 相当于写死了 python 路径。

`#!/usr/bin/env python` 会去环境设置寻找 python 目录，可以增强代码的可移植性，推荐这种写法。

分成两种情况：

(1) 如果调用 python 脚本时，使用：

```
python script.py
```

`#!/usr/bin/python` 被忽略，等同于注释

(2) 如果调用python脚本时，使用：

```
./script.py
```

`#!/usr/bin/python` 指定解释器的路径

PS：shell 脚本中在第一行也有类似的声明。

dragonv 2个月前 (01-02)