



目录

CONTENTS



- | | |
|------------------|---------------------------|
| 1. Python基础知识介绍 | 9. Python文件操作 |
| 2. Python运算符、表达式 | 10. Python异常处理 |
| 3. Python变量、数据类型 | 11. Python时间模块介绍 |
| 4. Python流程控制 | 12. Python正则表达式 |
| 5. Python函数介绍 | 13. Python excel操作 |
| 6. Python浅拷贝和深拷贝 | 14. Python Mysql操作 |
| 7. Python类介绍 | 15. Python 网络编程 |
| 8. Python模块和包 | 16. Python应用pyh模块生成html表格 |

更多测试资讯，请关注光荣之路微信公众号：

gloryroadtrain



Python基础知识介绍

--Python入门篇

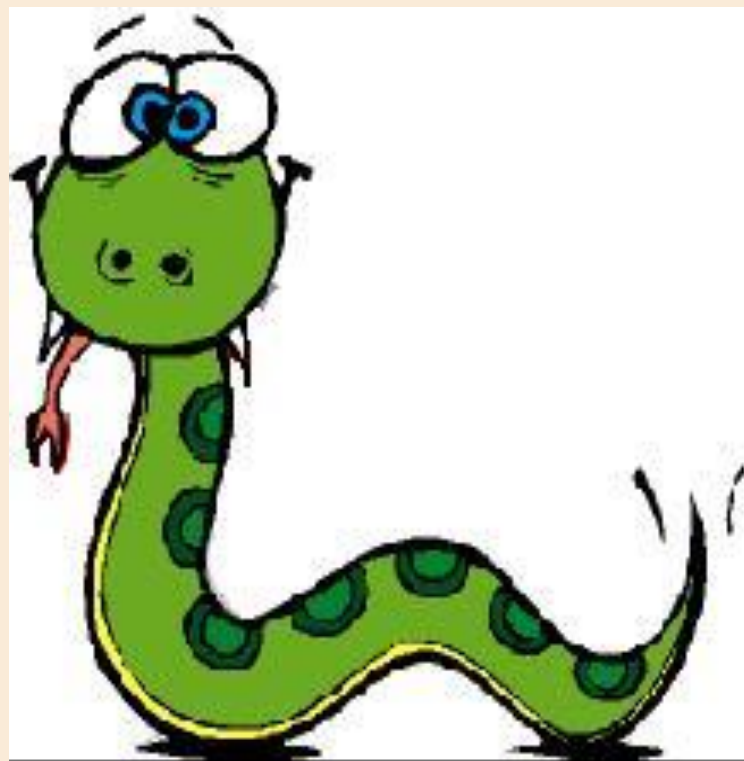


什么是Python



Python是一种解释型的、面向对象的、带有动态语义的高级程序设计语言。

特点：让你在编程时能够保持自己的风格，轻松实现你想要的功能，并且编写的程序清晰易懂。



Python起源



Python的作者，Guido von Rossum，荷兰人。1982年，Guido从阿姆斯特丹大学(University of Amsterdam)获得了数学和计算机双硕士学位。然而，尽管他算得上是一位数学家，但他更加享受计算机带来的乐趣。用他的话说，尽管拥有数学和计算机双料资质，他总趋向于做计算机相关的工作，并热衷于做任何和编程相关的活儿。



Python起源



Guido希望有一种语言，这种语言能够像C语言那样，能够全面调用计算机的功能接口，又可以像shell那样，可以轻松的编程。ABC语言让Guido看到希望。ABC是由荷兰的CWI (Centrum Wiskunde & Informatica, 数学和计算机研究所)开发的。Guido在CWI工作，并参与到ABC语言的开发。ABC语言以教学为目的。与当时的大部分语言不同，ABC语言的目标是“让用户感觉更好”。ABC语言希望让语言变得容易阅读，容易使用，容易记忆，容易学习，并以此来激发人们学习编程的兴趣。但是由于ABC语言需要比较高配置的电脑才能运行，在设计上，ABC语言不是模块化语言，可拓展性差；不能直接操作文件系统；传播困难，ABC编译器很大，必须被保存在磁带上。所以最终没有流行起来。

Python起源



1989年，为了打发圣诞节假期，Guido开始写Python语言的编译/解释器。Python来自Guido所挚爱的电视剧Monty Python's Flying Circus (BBC1960-1970年代播放的室内情景幽默剧，以当时的英国生活为素材)。他希望这个新的叫做Python的语言，能实现他的理念（一种C和shell之间，功能全面，易学易用，可拓展的语言）。

1991年，第一个Python编译器诞生。它是用C语言实现的，并能够调用C语言的库文件。从一出生，Python已经具有了：类，函数，异常处理，包含列表和词典在内的核心数据类型，以及模块为基础的拓展系统。

Python简史



Python的发展，可以分成几个重要的阶段：

--CNRI时期：CNRI是资助Python发展初期的重要单位，Python1.5版之前的成果大部分都在此期间完成。

--BeOpen时期：Guido van Rossum与BeOpen公司合作，此期间将Python2.0推出，甚至Python1.6也同时问世，但原则上已经分别维护了。

Python简史



--DC时期：此时Guido已离开BeOpen公司，将开发团队带到了Digital Creations (DC) 公司，该公司以发展Zope系统闻名，由于Python创始人的加入，因此这项合作也颇受关注。

--Python 3.0

Python应用



Python的应用



INDUSTRIAL
LIGHT & MAGIC



Python应用



Python的应用



实现Web爬虫和搜索引擎中的很多组件



Yahoo使用它（包括其他技术）管理讨论组



NASA在它的几个系统中既用了Python开发，又将其作为脚本语言



视频分享服务大部分是由Python 编写的

Python下载



Python最新源码，二进制文档，新闻资讯等可以在Python的官网查看到：

Python官网：<https://www.python.org/downloads/>

你可以在以下链接中下载Python的文档，你可以下载 HTML、PDF 和 PostScript 等格式的文档。

Python文档下载地址：www.python.org/doc/



群名称：Python零基础交流群
群 号：457561756

Python安装



Windows平台安装Python的简单步骤：

1. 打开WEB浏览器访问<http://www.python.org/download/>
2. 在下载列表中选择Window平台安装包，包格式为：python-XYZ.msi 文件， XYZ 为你要安装的版本号。
3. 下载后，双击下载包，进入Python安装向导，使用默认的设置一直点击“下一步”直到安装完成即可。

注意：安装之后配置环境变量！

Python运行方式



1. 交互模式
2. 命令行脚本模式(文本模式)
3. 集成开发环境(IDE)模式

练习：3种模式下完成第一个Hello World程序

Python编码



`#coding:utf-8`

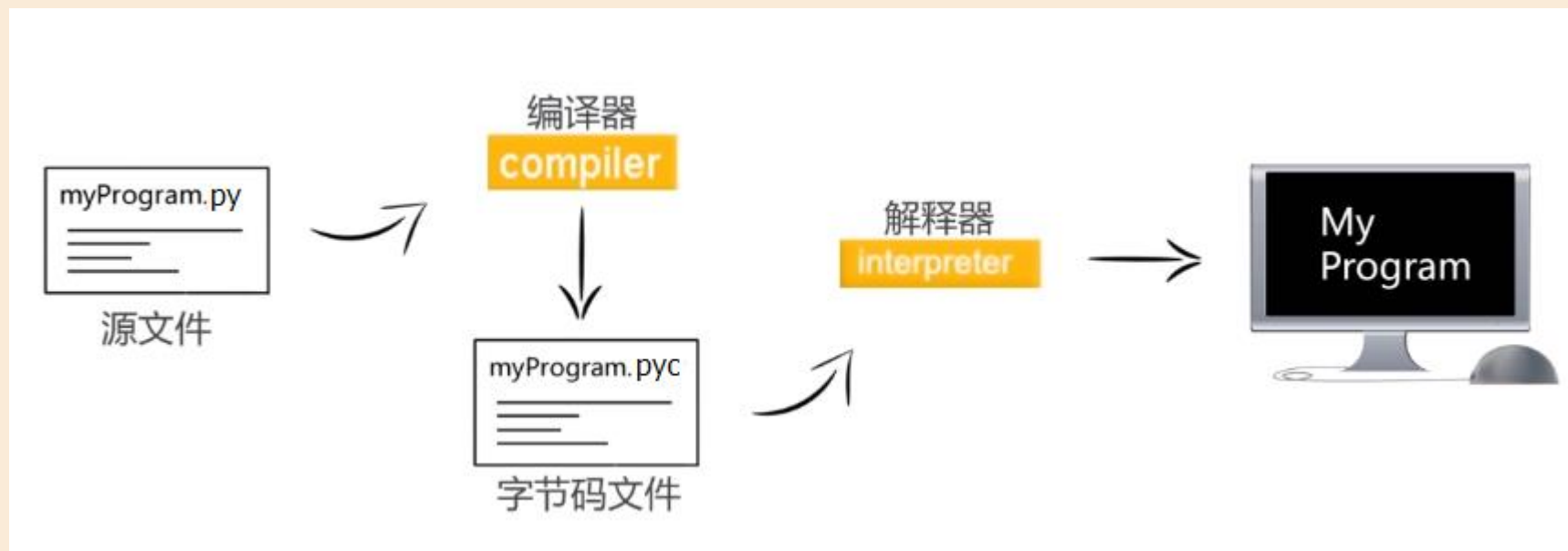
`#coding=utf-8`

`#-*- coding:utf-8 -*-`

Python文件类型



PVM(Python Vitual Machine)



Python文件类型



1.源代码

python源代码文件以".py"为扩展名，由python程序解释，不需要编译。

2.字节代码

Python源文件经编译后生成的扩展名为".pyc"的文件，有两种编译方法。

方法一：import py_compile

```
py_compile.compile("hello.py")
```

方法二：python -m py_compile hello.py

3.优化代码

经过优化的源文件，扩展名为".pyo"。

```
python -O -m py_compile hello.py
```

以上三种文件类型均可直接运行

Python基础语法



1. 缩进
2. 标识符
3. 多行语句
4. 引号
5. 注释
6. 代码组
7. 等待用户输入



Python基础语法--缩进

学习Python与其他语言最大的区别就是，Python的代码块不使用大括号（{}）来控制类，函数以及其他逻辑判断。

python最具特色的就是用缩进来写模块。**缩进的空白数量是可变的，但是所有代码块语句必须包含相同的缩进空白数量，这个必须严格执行，否则报错。**

Python基础语法--标识符



标识符是用户编程时使用的名字，如：变量、常量、函数、语句块等的名字都叫做标识符。在python里，标识符由字母、数字、下划线组成，但不能以数字开头。

例如：test_01,_test01(不推荐)都是正确的标示符，01_test则报错

python中的标识符是区分大小写的。

例如：Test和test代表两个不同的标识符

Python基础语法--标识符



以下划线开头的标识符是有特殊意义的。

以单下划线开头（`_foo`）的代表不能直接访问的类属性，需通过类提供的接口进行访问，不能用"`from xxx import *`"而导入；

以双下划线开头的（`__foo`）代表类的私有成员；

以双下划线开头和结尾的（`__foo__`）代表python里特殊方法专用的标识，如`__init__`（）代表类的构造函数。

Python基础语法--多行语句



Python语句中一般以换行作为语句的结束符。

但是我们可以使用斜杠（ \ ）将一行的语句分为多行显示，如下所示：

```
total = item_one + \
        item_two + \
        item_three
```

语句中包含[], {} 或 () 括号就不需要使用多行连接符。如下实例：

```
days = ['Sunday', 'Monday', 'Tuesday', 'Wednesday',
        'Thursday', 'Friday', 'Saturday']
```



Python基础语法--引号

Python 接收单引号('), 双引号("), 三引号('' ''') 来表示字符串。其中三引号可以由多行组成, 编写多行文本的快捷语法, 常用于文档字符串, 在文件的特定地点, 被当做注释。

```
word = 'word'
```

```
sentence = "This is a sentence."
```

```
paragraph = """This is a paragraph. It is  
made up of multiple lines and sentences."""
```

Python基础语法--注释



python中单行注释采用 # 开头

python没有块注释，所以现在推荐的多行注释也是采用的 #

```
#print "I'm ZZ."
```

```
#print "I'm 18 years old."
```

```
"""
```

```
I'm ZZ.
```

```
I'm 18 years old.
```

```
"""
```

Python基础语法--代码组



多个语句构成代码组，缩进相同的一组语句构成一个代码块。像if、while、def和class这样的复合语句，首行以关键字开始，以冒号(:)结束，该行之后的一行或多行代码构成代码组。我们将首行及后面的代码组称为一个子句(clause)。如下实例：

```
if expression :
```

```
    suite
```

```
elif expression :
```

```
    suite
```

```
else :
```

```
    suite
```


Python基础语法--等待用户输入



`raw_input()`与`input()`均是 python 的内建函数，通过读取控制台的输入与用户实现交互。区别如下：

1. `raw_input()` 将所有输入作为字符串看待，返回字符串类型；
`input()` 在输入数字时能够返回所输入数字的类型(`int`,`float`)；
2. `raw_input()` 直接读取控制台的输入,没有类型的限时；`input()` 的输入内容必须是一个合法的 python 表达式，如果输入字符串，必须使用引号将它括起来，否则它会引发`SyntaxError`。



微信公众号：
gloryroadtrain

吴老qq：745821166
小Z qq：836646629

光荣之路官网：
www.gloryroad.cn



群名称：Python零基础交流群
群 号：457561756