

Python 程序设计语言分析

161220052 贾淑佳

一、语言特性

1.1 心理特性



Python 的设计哲学是“优雅”、“明确”、“简单”。因此其对手 Perl 语言中“总是有多种方法来做同一件事”的理念在 Python 开发者中通常是难以忍受的。Python 开发者的哲学是“用一种方法，最好是只有一种方法来做一件事”。在设计 Python 语言时，如果面临多种选择，Python 开发者一般会拒绝花俏的语法，而选择明确的没有或者很少有歧义的语法。由于这种设计观念的差异，Python 源代码通常被认为比 Perl 具备更好的可读性，并且能够支撑大规模的软件开发。这些准则被称为 Python 格言。在 Python 解释器内运行 `import this` 可以获得完整的列表。

1.2 工程特性

- 胶水语言 Glue Language

- 很容易和其他著名的程序语言连接 (C/C++)，集成封装

- 脚本语言 Script Language

- 高级脚本语言，比脚本语言只能处理简单任务强大

- 面向对象语言 Object-Oriented Language

- 完全支持继承、重载、派生、多继承

1.3 应用特性

Web 程序

Python 经常被用于 Web 开发。比如，通过 `mod_wsgi` 模块，Apache 可以运行用 Python 编写的 Web 程序。Python 定义了 WSGI 标准应用接口来协调 Http 服务器与基于 Python 的 Web 程序之间的沟通。一些 Web 框架，如 Django、TurboGears、web2py、Zope、flask 等，可以让程序员轻松地开发和管理复杂的 Web 程序。

Python 对于各种网络协定的支持很完善，因此经常被用于编写服务器软件、网络蠕虫。第三方库 Twisted 支持异步联机编写程序和多数标准的网络协定(包含客户端和服务端)，并且提供了多种工具，被广泛用于编写高性能的服务器软件。

操作系统

在很多作业系统里，Python 是标准的系统组件。大多数 Linux 发布版以及 NetBSD、OpenBSD 和 Mac OS X 都集成了 Python，可以在终端机下直接运行 Python。有一些 Linux 发布版的安装器使用 Python 语言编写，比如 Ubuntu 的 Ubiquity 安装器、Red Hat Linux 和 Fedora 的 Anaconda 安装器。Gentoo Linux 使用 Python 来编写它的 Portage 包管理系统。Python 标准库包含了多个调用作业系统功能的库。通过 `pywin32` 这个第三方软件包，Python 能够访问 Windows 的 COM 服务及其它 Windows API。使用 IronPython，Python 程序能够直接调用 .Net Framework。一般说来，Python 编写的系统管理脚本在可读性、性能、源代码重用度、扩展性几方面都优于普通的 shell 脚本。

其他

NumPy、SciPy、Matplotlib 可以让 Python 程序员编写科学计算程序。PyQt、PySide、wxPython、PyGTK 是 Python 快速开发桌面应用程序的利器。

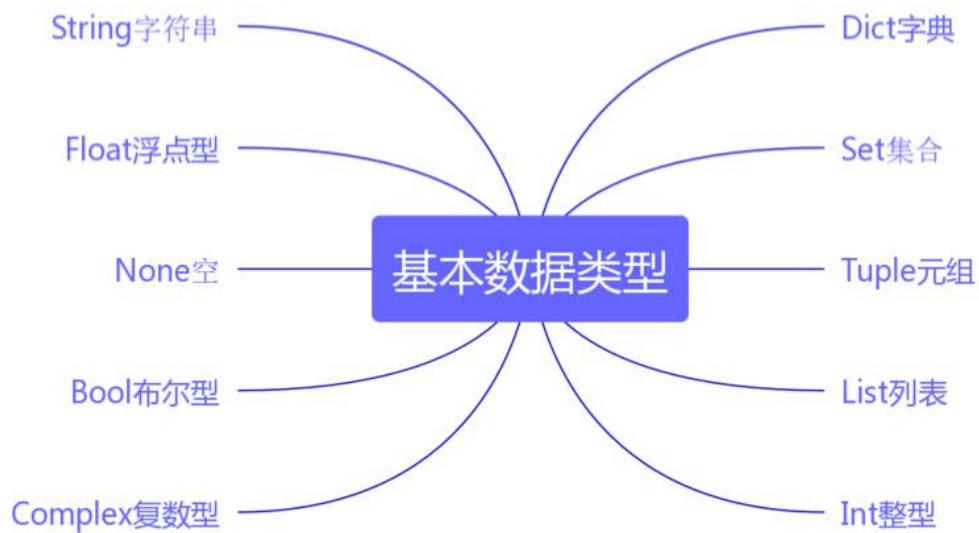
很多游戏使用 C++编写图形显示等高性能模块，而使用 Python 或者 Lua 编写游戏的逻辑

辑、服务器。相较于 Python，Lua 的功能更简单、体积更小；而 Python 则支持更多的特性和数据类型。很多游戏，如 EVE Online 使用 Python 来处理游戏中繁多的逻辑。

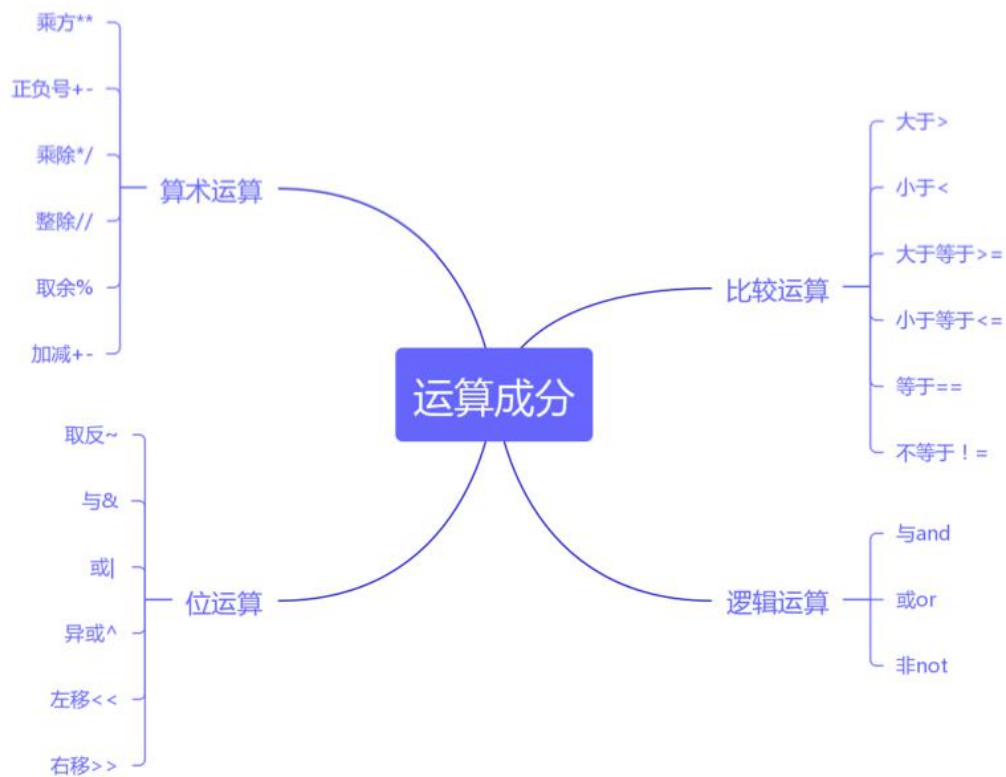
YouTube、Google、Yahoo!、NASA 都在内部大量地使用 Python。OLPC 的作业系统 Sugar 项目的大多数软件都是使用 Python 编写。

二、基本成分

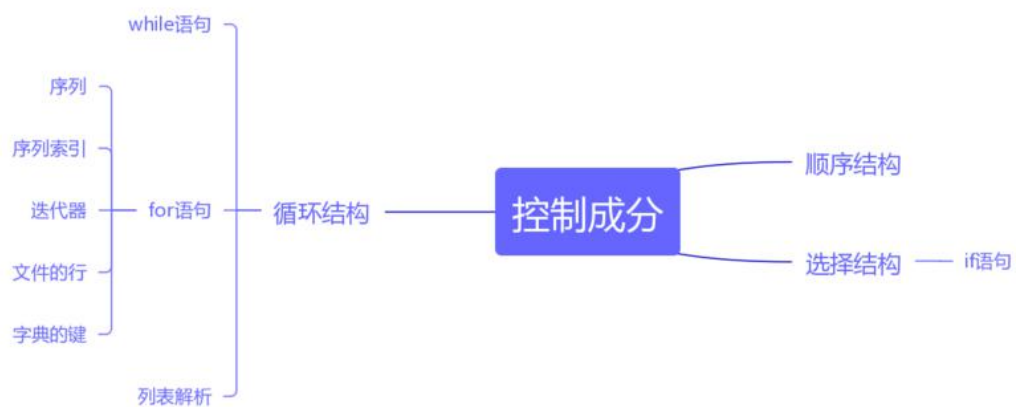
2.1 数据成分



2.2 运算成分



2.3 控制成分



2.4 传输成分

基本输入输出语句：input()、print()

三、注释分析

Github 500lines 项目中的 web-server 项目的注释分析。

程序语言为 python，从下表格中可以看出，代码中富含功能性注释，每个程序段的功能描述都很清晰，但在模块头部没有序言性注释，不能直接得到每个程序模块的功能信息。另外，除序言性注释和功能性注释外，代码中还包含“#.....”这样的注释，用来作为功能模块之间的分割线。

程序名	来源	代码行	程序语言	序言性注释	功能性注释
web-server	00-hello-web/server.py	8	Python		# Page to send back.
	01_echo_request_info/server.py	17			# Handle a GET request
		8			# Template for page to send back.
		24			# Handle a request by constructing an HTML page that echoes the
		25			# request back to the caller.
		30			# Create an information page to send.
		42			# Send the created page.
	02_serve_static/server_status_code.py	17			# How to display an error.
		27			# Classify and handle request.
		31			# Figure out what exactly is being requested.
		34			# It doesn't exist...
		38			# ...it's a file...
		42			# ...it's something we don't handle.
		46			# Handle errors.
		59			# Handle unknown objects.
		64			# Send actual content.
	02_serve_static/server.py	17			# How to display an error.
		27			# Classify and handle request.
		31			# Figure out what exactly is being requested.

		34		# It doesn't exist...
		38		# ...it's a file...
		42		# ...it's something we don't handle.
		46		# Handle errors.
		59		# Handle unknown objects.
		64		# Send actual content.
	03_handlers/server-index-page.py	70		# How to display an error.
		80		# Classify and handle request.
		84		# Figure out what exactly is being requested.
		87		# Figure out how to handle it.
		93		# Handle errors.
		106		# Handle unknown objects.
		111		# Send actual content.
	03_handlers/server-no-index-page.py	86		# How to display an error.
		96		# How to display a directory listing.
		107		# Classify and handle request.
		111		# Figure out what exactly is being requested.
		114		# Figure out how to handle it.
		120		# Handle errors.
		143		# Handle unknown objects.
		148		# Send actual content.

四、个人项目注释修改

以程序中生成随机验证码的函数模块为例：

```
//函数 verifyImage，用来生成随机的验证码
//调用格式 verifyImage($type=1,$length=4,$pixel=0,$line=0,$sess_name =
"verify")，其中 type 验证码类型，length 验证码长度，pixel 像素点个数，line 线个数
//该模块需要调用 string.func.php
<?php
require_once 'string.func.php';
//通过 GD 库做验证码
function verifyImage($type=1,$length=4,$pixel=0,$line=0,$sess_name =
"verify"){
```

```

session_start();
//创建画布
$width = 80;
$height = 28;
$image = imagecreatetruecolor ( $width, $height );
$white = imagecolorallocate ( $image, 255, 255, 255 );
$black = imagecolorallocate ( $image, 0, 0, 0 );
//用填充矩形填充画布
imagefilledrectangle ( $image, 1, 1, $width - 2, $height - 2, $white );
$chars = buildRandomString ( $type, $length );
$_SESSION [$sess_name] = $chars;
//选择允许在验证码中出现的字体样式
$fontfiles = array ( "MSYH.TTF", "MSYHBD.TTF", "SIMLI.TTF", "SIMSUN.TTC",
"SIMYOU.TTF", "STZHONGS.TTF" );
//随机生成字体的大小、角度、字体样式、颜色
for($i = 0; $i < $length; $i ++) {
    $size = mt_rand ( 14, 18 );
    $angle = mt_rand ( - 15, 15 );
    $x = 5 + $i * $size;
    $y = mt_rand ( 20, 26 );
    $fontfile = "fonts/" . $fontfiles [mt_rand ( 0, count ( $fontfiles ) -
1 )];
    $color = imagecolorallocate ( $image, mt_rand ( 50, 90 ), mt_rand ( 80,
200 ), mt_rand ( 90, 180 ) );
    $text = substr ( $chars, $i, 1 );
    imagefttext ( $image, $size, $angle, $x, $y, $color, $fontfile, $text );
}
//生成干扰因素像素点
if ($pixel) {
    for($i = 0; $i < 50; $i ++) {
        imagesetpixel ( $image, mt_rand ( 0, $width - 1 ), mt_rand ( 0, $height
- 1 ), $black );
    }
}
//生成干扰因素线
if ($line) {
    for($i = 1; $i < $line; $i ++) {
        $color = imagecolorallocate ( $image, mt_rand ( 50, 90 ), mt_rand
( 80, 200 ), mt_rand ( 90, 180 ) );
        imageline ( $image, mt_rand ( 0, $width - 1 ), mt_rand ( 0, $height
- 1 ), mt_rand ( 0, $width - 1 ), mt_rand ( 0, $height - 1 ), $color );
    }
}
header ( "content-type:image/gif" );

```

```
    imagegif ( $image );  
    imagedestroy ( $image );  
}
```