

Python 课程教学探讨

◆赵玉明

(山东服装职业学院信息工程系 山东泰安 271000)

摘要: python 被益为胶水语言, 简介高效适用场合多, 本文从课堂教学出发, 探讨了如何深入系统的掌握一门编程工具。

关键词: 编程模式; 教学重点; 标准库; 扩展库

据了解, 到了 2027 年, Python 会推出 5.0 版本, 开发效率极高、运行速度飞快, 会成为全球最受程序员欢迎的开发语言! 全球超过 50% 的 IT 系统是基于 Python 开发的。Python 在数据分析(大数据/金融分析/商业分析/科学分析等)领域, 在人工智能、深度学习领域应用。利用 Python 语言提高开发能力, 进行网站开发, 创建 GUI 应用、构建 2D/3D 的展示和游戏等。

1 Python 编程模式与教学方法

大部分讲授 Python 的老师都有其他语言的编程经验, 听课的学生也很可能接触过其他编程语言。在讲授、学习和应用 Python 时, 切记不要把其他语言的编程习惯和风格带到 Python 中来。不能使用其他语言尤其是 C 语言的思路去编写 Python 程序, 那会使得代码变得非常冗长、繁琐、可读性差并且难以维护, 浪费大量的宝贵时间, 无法发挥 Python 的优势, 甚至有可能写出看似正确实则错误的代码。

作为 Python 用户, 尽量尝试从最自然、最简洁的角度出发去思考和解决问题, 这样才能写出更加优雅、更加 Pythonic 的代码。在讲授 Python 程序设计课程时, 建议采用边讲边练、学生分组合作的教学模式, 精心设计一些案例带领学生一起完成并做好严格的过程控制与管理, 对出现的错误信息分析纠错, 通过分步测试解决出现的问题。同时经常鼓励和引导学生去想“还有没有更简洁更快速更直接更好的方法?” 类似的问题, 激发学生的潜能, 也充分挖掘 Python 语言的潜能。例如: 在计算 $1+2+3+\dots+99+100=?$ 时, 同学们普遍采用循环结构计算语句不够简洁, 要求学生用一行语句输出结果, 该语句为: `sum(range(1,101))`

2 Python 基础知识教学重点

2.1 代码规范

在编写选择结构、循环结构以及定义函数和类时, 一定要严格按照预期的逻辑关系对代码进行正确的缩进。

2.2 关键字、变量、运算符与表达式

在任何编程语言中, 关键字、变量、运算符、表达式都是最基本和最重要的内容, Python 也不例外。在 Python 中, 可以使用 `print(keyword.kwlist)` 语句查看所有关键字列表, 并且不允许使用关键字作为变量、函数或类名。Python 采用的是基于值的自动内存管理模式, 变量不直接存储值, 而是存储值的引用, 这使得 Python 变量的使用非常灵活方便, 例如在列表或元组等序列中可以同时存储不同类型的数据。在 Python 中, 变量不需要提前声明, 使用结束后也不需要刻意考虑变量的释放, 同时还支持动态改变变量类型。

2.3 Python 序列

Python 不仅提供了整数、实数、字符串等常见数据类型, 还内置了列表、元组、字典、集合等功能强大的序列类型, 同时 `range` 对象、`reversed` 对象、`map` 对象、文件对象以及其他可迭代对象也支持与 Python 序列类似的操作。序列可以说是 Python 最重要的一部分内容, 也是实现业务逻辑的重要途径, 这部分内容一定要精讲多练。

2.4 带有 else 的循环结构和异常处理结构

在 Python 中, `for` 循环、`while` 循环和异常处理结构可以带有 `else` 子句。对于循环结构而言, 当循环条件不成立时执行 `else` 中的代码, 如果是执行了 `break` 语句跳出循环则不执行 `else` 中的代码。对于异常处理结构而言, 如果 `try` 中的代码没有引发异常则执行 `else` 中的代码。带有 `else` 子句的循环结构和异常处理结构更符合人类的思维方式, 可读性较强, 同时也减少了代码量, 是应该熟练掌握和运用的。

2.5 Python 内置函数和标准库

教学中应培养学生经常浏览 Python 帮助文档的习惯, 熟悉 Python 提供的内置对象和标准库功能, 而不是遇到什么问题都首先想到自己写代码实现全部功能, 这样才能获得更高的开发效率。同时, 为培养学生自学能力和获取新知识的能力, 应要求熟练使用 `dir()` 函数查看模块中对象列表以及使用 `help()` 函数查看对象使用帮助。下面通过几个例子说明 Python 内置函数和标准库的强大功能。

列表对象已经提供了原地排序方法 `sort()`, 而内置函数 `sorted()` 则提供了更加强大的功能, 可以实现列表、元组、字典以及 `range` 和其他可迭代对象的复杂排序。然而, Python 标准库 `fractions` 已经提供了用于计算最大公约数的方法 `gcd()`, 直接调用即可。

例如: `>>> fractions.gcd(40, 32)` 结果为 8

3 Python 标准库和扩展库简介

Python 拥有大量涉及几乎所有领域的标准库和扩展库, 并且绝大部分扩展库都支持使用 `pip` 工具进行安装和管理。需要说明的是, 不论针对哪个领域的应用开发, 也不论选择哪个标准库或扩展库, Python 运算符、序列、各种推导式、程序控制结构、异常处理结构、文件操作等基本知识都是非常重要的, 只有完全理解和熟练掌握这些 Python 基础知识才能更有效地运用各种扩展库。在教学过程中可根据具体的学生专业进行适当选取。其中包括 GUI 编程, 网络编程, 大数据处理, 系统编程与多语言混合编程, 线程编程与进程编程, 数据库编程, 多媒体编程与游戏编程, 逆向工程与软件分析, 科学计算、统计分析与可视化, 移动端编程, 密码学编程, 人工智能与机器学习编程等内容。

4 结语

近几年来, 很多高校多个专业陆续开设了 Python 程序设计课程, 结合多年 Python 教学工作和实际开发中总结的经验, 对 Python 编程模式和教学重点进行了剖析, 对 Python 编程模式进行了深入剖析, 对 Python 的应用领域进行了认真梳理, 对国内 Python 程序设计教学工作有极大的促进作用。

参考文献:

- [1] 肖旻, 陈行. 基于 Python 语言编程特点及应用之探讨[J]. 电脑知识与技术, 2014, 10
- [2] 董付国. Python 程序设计基础[M]. 北京: 清华大学出版社, 2015.8

