

## 面向非计算机专业 Python 编程 案例教学方法研究

洪 樱 许小静

武汉纺织大学计算机与人工智能学院，武汉

**摘 要** | 程序设计类课程对于非计算机专业的学生来说是一种挑战。因此多数非计算机专业选择易学、代码高产的 python 语言作为程序设计课程的入门语言。如何结合非计算机各个专业的特点，设计出适合专业特点的案例，并应用于实际教学过程是目前亟待解决的问题。本文以培养学生的计算思维、创新及设计能力、快速适应社会和企业的工作效率等为目标，设计并研究了实用案例教学方法，取得了良好的教学效果。

**关键词** | 程序设计；python 语言；计算思维；案例教学

Copyright © 2021 by author (s) and SciScan Publishing Limited

This article is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/). <https://creativecommons.org/licenses/by-nc/4.0/>



### 1 引言

目前国内各大高校非计算机专业开设的程序设计课程包括 C 语言、VB、Access 数据库设计及应用、office 高级应用，其中艺术体育类专业大多开设 Access，文科管理类专业大多开设 office 高级应用，理工类专业开设 C 语言或 VB。C 语言和 VB 属于程序设计课程，目前教学过程中存在理论知识理解不够

作者简介：洪樱，武汉纺织大学计算机与人工智能学院，讲师。E-mail: 29602732@qq.com。

文章引用：洪樱，许小静. 面向非计算机专业 Python 编程案例教学方法研究 [J]. 现代计算机技术与应用, 2021, 3 (4) : 115-120.

<https://doi.org/10.35534/mcta.0304021>

深刻、编程缺乏算法思想、依赖教师指导示范、实践兴趣难以培养、学习积极性较差、课堂参与力度不够、自学能力不足等诸多问题。程序设计类课程对于非计算机专业的学生来说是一种挑战,学生之前没有编程和设计的概念,编程和设计思维难以形成,因此学生学习难度较高,教学效果也难以提高,面对这种情况,应选择一种简单、易学、上手快、能快速解决实际问题的程序设计语言。

由于 Python 的免费、开源、跨平台、丰富生态库等诸多优点,目前国内外各大学院都将 Python 程序设计这门课程作为计算机专业或非计算机专业的程序设计入门教学语言。因此,如何结合非计算机各个专业的特点,设计出快速掌握 python 知识框架的案例,并应用于实际教学过程是目前亟待解决的问题。

## 2 预期目标

非计算专业的案例教学法,既不是纯理论的教学,也不是一味地进行操作,而且跟实训课程有本质的区别。实训讲究的是操作,没有规划、设计的概念。目前已经有一些跟计算机相关的课程采用实训法进行教学,这种教学法训练的是操作过程,学生不能创新,只能机械性学习,不能深入、无法加深概念的理解,从而对日后的工作也无多大的作用。而案例教学法,是需要学生参与规划和设计,在解决实际问题的过程中锻炼动手能力,在完善的过程中进行分析与思考,在做出更好的解决方案时培养计算思维。

### 2.1 培养学生的计算思维

使用与专业相关的案例进行教学,需要学生在问题的提出后,寻找解决方案。在寻找解决方案的过程中,让非计算机专业的学生意识到他们的很多专业问题实际上都可以使用计算机来完成,拥有了这样的意识,利用计算机相关软件来解决学科问题或是开发相关软件来解决学科问题对学生来说就不是一件难事了。使学生养成利用计算机来解决各项工作问题的思维习惯,培养了学生的设计能力和处理问题的能力。

### 2.2 培养学生的创新及设计能力和提高学习兴趣

利用案例教学法,由学生自己就实际需求设计和制作软件,就需要学生参

与设计, 界面如何设置、知识点如何应用、安全性怎样保证、如何与客户沟通等方面都需要学生积极主动参与设计, 培养了学生的创新和设计能力。软件制作出来后, 学生非常有成就感, 从而提高了学生的兴趣。

## 2.3 培养学生快速适应社会和企业的工作效率

学生在案例教学的过程中了解了各种环境需要或客户需求, 在学习中适应此种环境和工作需求, 为日后毕业迅速融入工作岗位打下基础。

## 2.4 学生与老师的互动和交流增强了课堂学术气氛

在教学中, 经常需要老师和学生就案例中的某些问题进行分析、探讨, 借以寻找最佳方法, 一方面增强了学生与老师的互动和交流, 活跃了课程气氛, 另一方面也达到了学术研讨效果, 并且也提高了学生的动手能力。

# 3 实用案例教学方法举例

在函数与模块化知识点章节, 可以《信息管理系统》案例为例开展教学。此案例中所需要处理的信息数据可以根据所教授的专业进行选取。教学内容及顺序安排如图1:

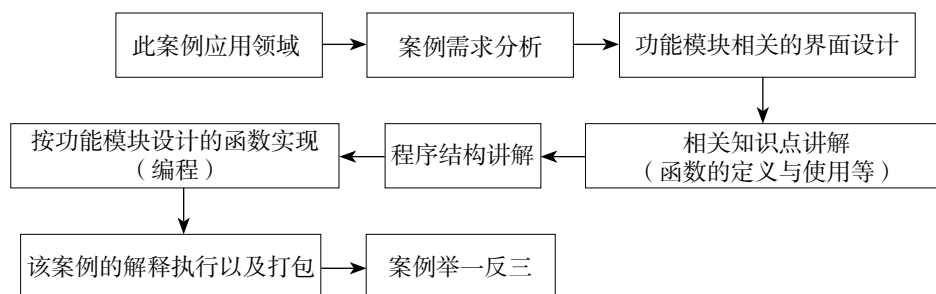


图1 《信息管理系统》案例教学法流程示意图

Figure 1 Schematic diagram of the case teaching process of Information Management System

案例需求分析和功能模块相关的界面设计由学生讨论完成。教师进行相关

知识点的讲解,并帮助学生分析并设计出程序结构,学生按功能模块运用相关知识点实现代码编写、调试、打包。程序完成后教师进行总结,布置相似项目作为课后作业从而使得学生能够举一反三。

此外,还可以适当扩充案例功能:如文件的读写功能(案例中的数据从文件中读入/系统处理结果写回文件等)、系统中信息数据的可视化展示(散点图、柱形图等)。此时 python 解释器可能需要安装 pandas、numpy、matplotlib 等扩展库。通过此案例将不同章节的知识点融合在一起,不仅是对基本知识体系结构的一种补充,同时也增强了学生的综合编程能力。

讲解字典知识点时,可以以《文本词频统计》案例为例开展教学。此案例中的文本来源于文件(如文本文件),文件数据也可以是与所授专业相关的。教学内容及顺序安排如图 2:

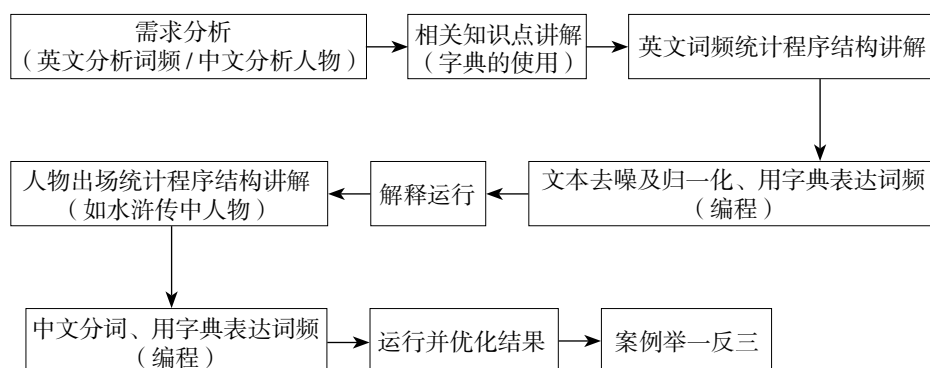


图 2 《文本词频统计》案例教学法流程示意图

Figure 2 Case teaching process diagram of Text Word Frequency Statistics

对于英文实例,根据空格和标点符号即可进行单词分隔;对于中文实例,则需要介绍用于中文分词的第三方库 jieba 库。讲解这两种衍生实例,有助于学生理解中英文文本处理的差异。引导学生将解决实际问题的主要精力从程序语法逐步过渡到业务逻辑实现上。

此案例也可以适当扩充其功能:中英文词频统计的结果以可视化的方式进行展示,如词云。此时 python 解释器需要额外安装 word cloud 扩展库。

## 4 结束语

应用案例教学法,不仅使学生学会了编程套路,而且有了一个完整的 Python 语法体系的知识框架,还学习了完整地利用 Python 进行系统开发的过程。在案例讲解和学习过程中,还体验到了丰富的生态库,体会到了 Python 编程的乐趣。同时,由于 Python 的免费、开源、跨平台、丰富的计算生态等诸多特点,使得不同专业的学生能够快速实现和验证自己的思路、创意或推测。在有些编程语言中需要编写大量代码才能实现的功能,在 Python 中直接调用内置函数或库方法用较少的代码即可实现,大幅度简化了代码的编写和维护。学生只需把主要精力放在业务逻辑的设计和实现上,在开发速度和运行效率之间达到了完美的平衡。

实践表明,通过设计制作案例程序的过程来学习程序设计课程,借以深刻理解 Python 语言的强大功能,从而提高学生应用计算机解决实际问题的水平,为日后的工作和生活打下基础,也为非计算机专业学生对计算机的学习和使用产生极大兴趣,形成计算思维,进而可以应用计算机来解决跟专业相关的工作问题,同时也激发了学生的创新意识和创造性。

## 参考文献

- [1] 张婷燕,王剑宇,卢爱臣. 递进式教学法在 Python 程序设计中的应用[J]. 电脑知识与技术, 2021, 11(20): 260-262.
- [2] 易发胜,李立,赵丽琴. 新工科背景下 Python 程序设计课程教学方法研究[J]. 计算机教育, 2021, 7: 148-151.
- [3] 赵广辉,李敏之,邵艳玲. Python 程序设计基础[M]. 北京:高等教育出版社, 2021.
- [4] 嵩天,黄天羽,礼欣. Python 语言:程序设计课程教学改革理想选择[J]. 中国大学教学, 2016(2): 42-47.

## Research on Python Programming Case Teaching Method for Non-Computer Majors

Hong Ying   Xu Xiaojing

*School of Computer Science and Artificial Intelligence, Wuhan Textile University, Wuhan*

**Abstract:** Programming courses are a challenge for non-computer majors. Therefore, most non computer majors choose Python language, which is easy to learn and high-yield code, as the entry language of programming course. How to combine the characteristics of non-computer majors, design cases suitable for professional characteristics and apply them to the actual teaching process is an urgent problem to be solved. Aiming at cultivating students' computing thinking, innovation and design ability, and quickly adapting to the work efficiency of society and enterprises, this paper designs and studies practical case teaching methods, and achieves good teaching results.

**Key words:** Programming; Python language; Computational thinking; Case teaching