

《Python语言程序设计》

图书基本信息

书名：《Python语言程序设计》

13位ISBN编号：9787111412342

10位ISBN编号：7111412346

出版时间：2013-3

出版社：机械工业出版社

作者：（美）Y. Daniel Liang

页数：557

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu111.com

《Python语言程序设计》

内容概要

《Python语言程序设计》

作者简介

Y. Daniel Liang (梁勇) 阿姆斯特朗亚特兰大州立大学计算机科学系Yamacraw教授。之前曾是普度大学计算机科学系副教授，并曾两次获得普度大学杰出研究奖。他所编写的Java教程在美国大学Java课程中采用率最高。

Chapter 1 Introduction to Computers, Programs, and Python 1.1 Introduction 1.2 What Is a Computer? 1.3 Programming Languages 1.4 Operating Systems 1.5 The History of Python 1.6 Getting Started with Python 1.7 Programming Style and Documentation 1.8 Programming Errors 1.9 Getting Started with Graphics Programming

Chapter 2 Elementary Programming 2.1 Introduction 2.2 Writing a Simple Program 2.3 Reading Input from the Console 2.4 Identifiers 2.5 Variables, Assignment Statements, and Expressions 2.6 Simultaneous Assignments 2.7 Named Constants 2.8 Numeric Data Types and Operators 2.9 Evaluating Expressions and Operator Precedence 2.10 Augmented Assignment Operators 2.11 Type Conversions and Rounding 2.12 Case Study: Displaying the Current Time 2.13 Software Development Process 2.14 Case Study: Computing Distances

Chapter 3 Mathematical Functions, Strings, and Objects 3.1 Introduction 3.2 Common Python Functions 3.3 Strings and Characters 3.4 Case Study: Minimum Number of Coins 3.5 Introduction to Objects and Methods 3.6 Formatting Numbers and Strings 3.7 Drawing Various Shapes 3.8 Drawing with Colors and Fonts

Chapter 4 Selections 4.1 Introduction 4.2 Boolean Types, Values, and Expressions 4.3 Generating Random Numbers 4.4 if Statements 4.5 Case Study: Guessing Birthdays 4.6 Two-Way if-else Statements 4.7 Nested if and Multi-Way if-elif-else Statements 4.8 Common Errors in Selection Statements 4.9 Case Study: Computing Body Mass Index 4.10 Case Study: Computing Taxes 4.11 Logical Operators 4.12 Case Study: Determining Leap Years 4.13 Case Study: Lottery 4.14 Conditional Expressions 4.15 Operator Precedence and Associativity 4.16 Detecting the Location of an Object

Chapter 5 Loops 5.1 Introduction 5.2 The while Loop 5.3 The for Loop 5.4 Nested Loops 5.5 Minimizing Numerical Errors 5.6 Case Studies 5.7 Keywords break and continue 5.8 Case Study: Displaying Prime Numbers 5.9 Case Study: Random Walk

Chapter 6 Functions 6.1 Introduction 6.2 Defining a Function 6.3 Calling a Function 6.4 Functions with/without Return Values 6.5 Positional and Keyword Arguments 6.6 Passing Arguments by Reference Values 6.7 Modularizing Code 6.8 Case Study: Converting Decimals to Hexadecimals 6.9 The Scope of Variables 6.10 Default Arguments 6.11 Returning Multiple Values 6.12 Case Study: Generating Random ASCII Characters 6.13 Function Abstraction and Stepwise Refinement 6.14 Case Study: Reusable Graphics Functions

Chapter 7 Objects and Classes 7.1 Introduction 7.2 Defining Classes for Objects 7.3 UML Class Diagrams 7.4 Immutable Objects vs. Mutable Objects 7.5 Hiding Data Fields 7.6 Class Abstraction and Encapsulation 7.7 Object-Oriented Thinking

Chapter 8 More on Strings and Special Methods 8.1 Introduction 8.2 The str Class 8.3 Case Study: Checking Palindromes 8.4 Case Study: Converting Hexadecimals to Decimals 8.5 Operator Overloading and Special Methods 8.6 Case Study: The Rational Class

Chapter 9 GUI Programming Using Tkinter 9.1 Introduction 9.2 Getting Started with Tkinter 9.3 Processing Events 9.4 The Widget Classes 9.5 Canvas 9.6 The Geometry Managers 9.7 Case Study: Loan Calculator 9.8 Displaying Images 9.9 Menus 9.10 Popup Menus 9.11 Mouse, Key Events, and Bindings 9.12 Animations 9.13 Scrollbars 9.14 Standard Dialog Boxes

Chapter 10 Lists 10.1 Introduction 10.2 List Basics 10.3 Case Study: Lotto Numbers 10.4 Case Study: Deck of Cards 10.5 Deck of Cards GUI 10.6 Copying Lists 10.7 Passing Lists to Functions 10.8 Returning a List from a Function 10.9 Case Study: Counting the Occurrences of Each Letter 10.10 Searching Lists 10.11 Sorting Lists 10.12 Case Study: Bouncing Balls

Chapter 11 Multidimensional Lists 11.1 Introduction 11.2 Processing Two-Dimensional Lists 11.3 Passing Two-Dimensional Lists to Functions 11.4 Problem: Grading a Multiple-Choice Test 11.5 Problem: Finding the Closest Pair. 11.6 GUI: Finding the Closest Pair 11.7 Problem: Sudoku 11.8 Case Study: Sudoku GUI 11.9 Multidimensional Lists

Chapter 12 Inheritance and Polymorphism 12.1 Introduction 12.2 Superclasses and Subclasses 12.3 Overriding Methods 12.4 The object Class 12.5 Polymorphism and Dynamic Binding 12.6 The isinstance Function 12.7 Case Study: A Reusable Clock 12.8 Class Relationships 12.9 Case Study: Designing the Course Class 12.10 Designing a Class for Stacks 12.11 Case Study: The FigureCanvas Class

Chapter 13 Files and Exception Handling 13.1 Introduction 13.2 Text Input and Output 13.3 File Dialogs 13.4 Case Study: Counting Each Letter in a File 13.5 Retrieving Data from the Web 13.6 Exception Handling 13.7 Raising Exceptions 13.8 Processing Exceptions Using Exception Objects 13.9 Defining Custom Exception Classes 13.10 Binary IO Using Pickling 13.11 Case Study: Address Book

Chapter 14 Tuples, Sets, and Dictionaries 14.1 Introduction 14.2 Tuples 14.3 Sets 14.4 Comparing the Performance of Sets and Lists 14.5 Case Study: Counting Keywords 14.6 Dictionaries 14.7 Case Study: Occurrences of Words

Chapter 15 Recursion 15.1 Introduction 15.2 Case Study: Computing

《Python语言程序设计》

Factorials 15.3 Case Study: Computing Fibonacci Numbers 15.4 Problem Solving Using Recursion 15.5 Recursive Helper Functions 15.6 Case Study: Finding the Directory Size 15.7 Case Study: Towers of Hanoi 15.8 Case Study: Fractals 15.9 Case Study: Eight Queens 15.10 Recursion vs. Iteration 15.11 Tail Recursion
A detailed table of contents for the Web chapters is available on the companion Website:
Chapter 1 6 Developing Efficient Algorithms
Chapter 17 Sorting
Chapter 18 Linked Lists, Stacks, Queues, and Priority Queues
Chapter 19 Binary Search Trees
Chapter 20 AVL Trees
Chapter 21 Hashing: Implementing Dictionaries and Sets
Chapter 22 Graphs and Applications
Chapter 23 Weighted Graphs and Applications
APPENDIXES
Appendix A Python Keywords
Appendix B The ASCII Character Set
Appendix C Number Systems
INDEX
CREDITS

《Python语言程序设计》

编辑推荐

梁勇编著的《Python语言程序设计(英文版)》保持了Liang博士系列丛书中一贯的、标志性的教与学的哲学：以实例教，由实践学。书中采用了他所提出的已经经过实践检验的“基础先行”的方法，即在定义类之前，首先使用清晰简明的语言介绍基本程序设计概念，如选择语句、循环和函数；在介绍面向对象程序设计和GUI编程之前，首先介绍基本逻辑和程序设计概念。书中除了给出一些以游戏和数学为主的典型实例外，还在每章的开始使用简单的图形给出一两个例子，以激发学生的学习兴趣。

精彩短评

- 1、例子比较多，对于gui讲解比较好...
对我这种动手低分学生估计很适合.....
- 2、美国谭浩强。每次都是浅尝辄止。
- 3、书写的挺好的，自己英文完全可以对付，programming
- 4、一本大杂烩，什么都讲，什么都讲不深，既然想谈程序设计，并没有提供一些有价值的python /oo的设计哲学，基本的type,descriptor,mro都不说，更不用提设计模式，我很怀疑封底说的该作者写java教材大学使用率最高这一说法，这本书更像小白看的，看了能怎样呢，也只是会用tk画几个演示图，对工程应用并卵。不如The Hacker's Guide to Python
- 5、内容很赞,就是纸张太烂了,有盗版的嫌疑
- 6、书名说了是introduction,真的是introduction！虽然看完这本书现在在看廖雪峰的教程也是一脸蒙逼，但是对于初学者来说这本书真的还不错。
- 7、很不错，非常满意，很实惠
- 8、梁教授虽然早就不搞科研了，但编写基础教材还是可以的。
- 9、通俗易懂
- 10、第一个好处是确实是基于3.0，py3和py2的区别还是很大的，这一点起码让你专注于3，对初学者，并且选择3入手的比较有用。第二个好处就是用图形设计做例子，图形给人的效果肯定是胜过字符的，这让你更有动力完成后面的习题。趣味性更足。第三个好处就是按作者自己的理论体系组织了章节，针对各章在教授入门的前提下都分别有所偏重，逐步引导初学者登堂入室。这个要赞一下。第四个好处就是如果阅读英文难度不大的话，起码避免了不良翻译给人带来的困惑，国内译文的质量实在让人不能恭维。建议具有一定英文的初学者选择此书，确实不错的。

《Python语言程序设计》

精彩书评

- 1、pdf版本的只有1-15章，后面的几章让你去培生教育网站上注册，才能看到。虽然前15章的内容不错，实例讲解和很多的练习题。但是为什么不一次性都写在到书里呢？更坑的是没有学习码，无法完成注册。好吧，其实我看的是网上下来的PDF。不过就算是真的买了，感觉也不值。
- 2、自己本专业和计算机有比较强的联系，但学生时代在瞎玩，读研时也没好好学习，上班后醒悟，重新开始学习，纯属小白，本书通俗，初学编程最怕的就是有大量各种枯燥的概念与专业术语，本书却配合大量的例子，非常适合入门。

章节试读

1、《Python语言程序设计》的笔记-第1页

`** python语言 **`
得当初专业填的。就是应用开发

学习编程的意愿由来已久，记

《Python语言程序设计》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu111.com