

<<C#程序设计>>

图书基本信息

书名：<<C#程序设计>>

13位ISBN编号：9787300104034

10位ISBN编号：7300104037

出版时间：2009-4

出版时间：中国人民大学出版社

作者：韩小祥，刘志军 主编

页数：276

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：<http://www.tushu007.com>

<<C#程序设计>>

前言

随着计算机技术的快速发展，计算机之间互连互通、数据共享的需求越来越突出，而目前计算机硬件设备的不兼容性严重束缚了互联网的发展，由此引发了新一轮的跨平台软件开发浪潮。

软件开发商纷纷推出新的战略规划和解决方案，Microsoft公司提出的.NET战略就是其中的经典之作。

C#作为.NET的核心编程语言，是Microsoft公司极力推荐的新一代高级开发语言。

它囊括了所有流行语言的优点。

C#一发布便受到了程序员的热捧，他们纷纷从原有的开发环境中跳出来，转而学习C#，因为C#代表了高级开发语言的发展趋势。

本书共分11章，主要内容包括：第1章为编程语言的发展与概述，介绍了程序设计语言的发展、C#语言特性及程序设计方法等内容。

第2章为.NET与C#集成开发环境，首先介绍了.NET Framework组件的结构和运行原理；其次介绍了Visual Studio 2008的安装及其简单的界面布局和基本菜单、工具的使用；最后通过一个简单的C#程序作为入门程序，使读者了解到了Visual Studio代码的基本格式，也起到了引导下文的作用。

第3章为C#语言基础，介绍了C#的数据类型、常量、变量、运算符、表达式、语句、数组和命名空间，本章是学习用C#编写程序的基础。

第4章为C#面向对象编程基础，介绍了面向对象的基本概念、类、对象、构造函数和析构函数、方法、域、属性。

第5章为C#面向对象编程进阶，介绍了OOP技术特性——继承、多态、抽象和密封、重载，以及类型转换、接口与结构、集合与索引器、委托和事件、异常处理、预处理指令、组件与程序集等。

第6章为文件，详细介绍了用于文件操作的类。

第7章为Windows环境下应用程序开发，介绍了如何创建Windows应用程序，如何使用控件和绘制图形，以及应用控制台程序。

第8章为数据访问，介绍了ADO.NET的体系结构和实现数据操作的过程，并分别介绍了ADO.NET是如何访问关系数据库和XML数据的。

第9章为Windows Service设计，介绍了什么是Windows Service，如何在Visual Studio中创建并运行、调试Windows Service。

第10章为Web应用设计。

第11章为在C#中使用XML。

<<C#程序设计>>

内容概要

C#是一种面向对象的、运行于.NET Framework之上的高级程序设计语言。

为了适应读者对C#语言的学习需要，本书系统讲解了以下内容：C#语言特性；C#集成开发环境；C#的数据类型、语句；类、对象以及面向对象的特性；文件的操作；ADO.NET访问数据库的原理及过程；Visual Studio中如何创建Windows、Web、Web Service应用程序；最后介绍了在C#中如何使用XML。

本书由浅入深，循序渐进，重点突出，对C#从基础的数据类型到高级应用都做了详细讲解。

本书既可作为高等院校、高职高专院校相关课程的教材，也可作为高级语言学习者和程序员的参考用书。

<<C#程序设计>>

书籍目录

第1章 概述 1.1 程序设计语言发展史与C#特性和前景 1.2 程序设计方法与软件开发流程 1.3 本章小结 1.4 练习题第2章 .NET与C#集成开发环境 2.1 .NET Framework组件 2.2 C#集成开发环境 2.2.1 VS 2008的安装 2.2.2 VS 2008简介 2.3 一个简单的C#程序 2.4 本章小结 2.5 练习题第3章 C#语言基础 3.1 简单数据类型 3.2 变量和常量 3.2.1 变量 3.2.2 常量 3.3 运算符与表达式 3.3.1 运算符 3.3.2 表达式 3.4 语句 3.4.1 选择语句 3.4.2 循环语句 3.4.3 跳转语句 3.4.4 异常处理语句 3.5 数组 3.5.1 声明数组 3.5.2 初始化数组 3.5.3 访问数组元素 3.5.4 System Array 3.5.5 foreach遍历数组元素 3.6 命名空间 3.7 本章小结 3.8 练习题第4章 C#面向对象编程基础 4.1 面向对象的基本概念 4.1.1 什么是面向对象编程 4.1.2 面向对象编程的特点 4.2 类 4.2.1 类的概念 4.2.2 类的声明与实例化 4.2.3 类成员的访问 4.3 对象 4.3.1 对象的定义、实例化及访问 4.3.2 类与对象的关系 4.4 构造函数和析构函数 4.4.1 构造函数的概念及使用 4.4.2 析构函数的概念及使用 4.5 方法 4.5.1 方法的声明 4.5.2 方法的参数 4.5.3 静态和非静态方法的使用 4.5.4 方法的重载 4.5.5 方法的覆盖 4.6 域和属性 4.6.1 域的概念及用途 4.6.2 域的声明 4.6.3 属性的概念及用途 4.6.4 属性的声明及使用 4.7 本章小结 4.8 练习题第5章 C#面向对象编程进阶 5.1 OOP技术特性 5.1.1 继承 5.1.2 多态 5.1.3 抽象和密封 5.1.4 操作符重载 5.2 类型转换 5.2.1 隐式类型转换 5.2.2 显式类型转换 5.2.3 装箱和拆箱 5.3 结构和接口 5.3.1 结构 5.3.2 接口 5.4 集合与索引器 5.4.1 集合 5.4.2 索引器 5.5 委托和事件 5.5.1 委托 5.5.2 事件 5.6 异常处理 5.6.1 异常类 5.6.2 抛出和捕获异常 5.7 预处理指令 5.8 组件与程序集 5.8.1 组件 5.8.2 程序集 5.9 this关键字 5.10 程序风格 5.10.1 程序布局 5.10.2 注释 5.10.3 使用常量 5.10.4 类 5.10.5 嵌套if 5.10.6 嵌套循环 5.10.7 复合条件 5.10.8 文档 5.11 本章小结 5.12 练习题第6章 文件 6.1 用于文件操作的类 6.1.1 文件与流 6.1.2 文件操作类 6.2 File类和Directory类 6.3 FileInfo类和DirectoryInfo类 6.4 FileStream类、StreamWriter类和StreamReader类 6.4.1 FileStream类 6.4.2 StreamWriter类 6.4.3 StreamReader类 6.5 本章小结 6.6 练习题第7章 Windows环境下应用程序开发 7.1 建立Windows应用程序 7.2 使用WindowsForms控件 7.3 使用菜单 7.3.1 创建主菜单 7.3.2 创建快捷菜单 7.4 对话框 7.4.1 模态和非模态对话框 7.4.2 通用对话框 7.5 单文档界面和多文档界面程序 7.6 图形简介 7.6.1 对象、方法、属性和类的类比 7.6.2 第一次绘图 7.6.3 创建程序 7.6.4 图形坐标系统 7.6.5 解释程序 7.6.6 绘图方法 7.6.7 颜色 7.7 控制台程序 7.7.1 第一个控制台程序 7.7.2 命令提示符cd和dir 7.7.3 运行程序的方式 7.7.4 控制台应用程序中的类 7.7.5 脚本和输出重定向 7.7.6 脚本和批处理文件 7.8 本章小结 7.9 练习题第8章 数据访问 8.1 ADO.NET引言 8.1.1 ADO.NET与ADO的差异 8.1.2 ADO.NET的对象体系 8.1.3 数据访问的3个典型阶段 8.2 ADO.NET访问关系数据库 8.2.1 引入操作数据库的命名空间 8.2.2 数据库连接 8.2.3 数据库访问 8.3 DataSet操作XML 8.4 本章小结 8.5 练习题第9章 Windows Service设计 9.1 Windows Service简介 9.1.1 系统中现有的Windows Service 9.1.2 Windows Service的独特之处 9.2 使用Visual studio.NET创建Windows Service 9.3 本章小结 9.4 练习题第10章 Web应用设计 10.1 Web应用程序基础 10.1.1 HTML页面 10.1.2 动态Web页面 10.2 ASP.NET简介 10.2.1 对编译语言的支持 10.2.2 程序代码与页面内容的分离 10.2.3 简单的配置 10.3 ASP.NET应用程序的建立 10.4 Web窗体和控件 10.4.1 Web窗体 10.4.2 Web控件 10.4.3 HTML控件 10.4.4 服务器控件 10.5 Web服务 10.5.1 Web Service结构体系 10.5.2 Visual Studio.NET对Web服务的支持 10.5.3 将Web引用添加到项目 10.5.4 从项目中移除“Web引用” 10.5.5 创建一个Web服务 10.5.6 应用一个Web服务 10.6 ASP.NET Web应用程序模板 10.6.1 要求 10.6.2 由模板添加的文件和引用 10.6.3 Web应用程序模板与Web服务模板的比较 10.7 本章小结 10.8 练习题第11章 在C#中使用XML 11.1 XML概述 11.2 读取XML 11.3 编写XML 11.4 实现DOM 11.5 本章小结 11.6 练习题

<<C#程序设计>>

章节摘录

插图：(4) 编码。

软件编码是指把软件设计转换成计算机可以接受的程序，即写成以某一程序设计语言表示的“源程序清单”。

充分了解软件开发语言、工具的特性和编程风格，有助于开发工具的选择以及保证软件产品的开发质量。

当前软件开发中，除在专用场合，已经很少使用20世纪80年代的高级语言了，取而代之的是面向对象的开发语言，而且面向对象的开发语言和开发环境大都合为一体，大大提高了开发的速度。

(5) 测试。

软件测试的目的是以较小的代价发现尽可能多的错误。

要实现这个目标，关键在于设计一套出色的测试用例（测试数据和预期的输出结果组成了测试用例）

如何才能设计出一套出色的测试用例，关键在于理解测试方法。

不同的测试方法有不同的测试用例。

两种常用的测试方法是白盒测试法和黑盒测试法。

白盒测试法测试对象是源程序，依据的是程序内部的逻辑结构来发现软件的编程错误、结构错误和数据错误，其中结构错误包括逻辑、数据流、初始化等错误，用例设计的关键是以较少的用例覆盖尽可能多的内部程序逻辑结构。

黑盒测试法依据的是软件的功能或软件行为描述，发现软件的接口、功能和结构错误，其中接口错误包括内部/外部接口、资源管理、集成化以及系统错误，用例设计的关键同样也是以较少的用例覆盖模块输出和输入接口。

(6) 维护。

维护是指在已完成对软件的研制（分析、设计、编码和测试）工作并交付使用以后，对软件产品所进行的一些软件工程的活动。

即根据软件运行的情况，对软件进行适当修改，以适应新的要求，以及纠正运行中发现的错误。

编写软件问题报告、软件修改报告。

一个中等规模的软件，如果研制阶段需要一年至二年的时间，在它投入使用以后，其运行或工作时间可能持续五年至十年，那么它的维护阶段也是运行的这五年至十年期间。

在这段时间，人们几乎需要着手解决研制阶段所遇到的各种问题，同时还要解决某些维护工作本身特有的问题。

做好软件维护工作，不仅能排除障碍，使软件能正常工作，而且还可以扩展它的功能，提高性能，为用户带来明显的经济效益。

然而，遗憾的是对软件维护工作的重视往往远不如对软件研制工作的重视。

事实上，和软件研制工作相比，软件维护的工作量和成本都要大得多。

<<C#程序设计>>

编辑推荐

《C#程序设计》为21世纪大学计算机系列教材之一

<<C#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:<http://www.tushu007.com>