



普通高等教育“十一五”国家级规划教材

孙践知 张迎新 肖媛媛 编著

C# 程序设计

21世纪计算机科学与技术实践型教程

丛书主编 陈明



清华大学出版社



普通高等教育“十一五”国家级规划教材

孙践知 张迎新 肖媛媛 编著

C#程序设计

21世纪计算机科学与技术实践型教程

丛书主编 陈明

清华大学出版社
北京

内 容 简 介

本书将 C# 程序设计必须掌握的知识归纳为若干案例,每个案例解决一个问题。初学者只需模仿案例,获得直接体验,然后再学习和案例直接相关的知识。通过一个单元接着一个单元的模仿、学习,逐步地构成完整的知识体系。

全书共分为 12 章,主要包括 C# 概述及开发环境,C# 语言基础,分支结构,循环结构,面向对象的基本概念,多态、事件和接口,数组和集合,.NET 类库,文件和流,用户界面设计,ADO.NET 和数据库,以及异常处理等内容。

本书在选用软件版本时进行了折中,目前最新版本是 Visual Studio .NET 2010 Beta 版,考虑到 Beta 版软件可能存在问题,本书采用了 Visual Studio .NET 2008 作为开发环境。

本书适合作为高等院校 C# 程序设计课程教材,书中涉及了大量的最新 C# 程序设计技术,也可作为程序开发人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C# 程序设计/孙践知,张迎新,肖媛媛编著. —北京:清华大学出版社,2010.11
(21 世纪计算机科学与技术实践型教程)

ISBN 978-7-302-23751-8

I. ①C… II. ①孙… ②张… ③肖… III. ①C 语言—程序设计—高等学校—教材
IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 168103 号

责任编辑:谢 琛 薛 阳

责任校对:时翠兰

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954,jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市漂源装订厂

经 销:全国新华书店

开 本:185×260

印 张:19.5

字 数:442 千字

版 次:2010 年 11 月第 1 版

印 次:2010 年 11 月第 1 次印刷

印 数:1~4000

定 价:29.50 元

产品编号:038405-01

《21 世纪计算机科学与技术实践型教程》

编辑委员会

主 任：陈 明

委 员：	毛国君	白中英	叶新铭	刘淑芬	刘书家
	汤 庸	何炎祥	陈永义	罗四维	段友祥
	高维东	郭 禾	姚 琳	崔武子	曹元大
	谢树煜	焦金生	韩江洪		

策划编辑：谢 琛

《21 世纪计算机科学与技术实践型教程》

序

21 世纪影响世界的三大关键技术：以计算机和网络为代表的信息技术；以基因工程为代表的生命科学和生物技术；以纳米技术为代表的新型材料技术。信息技术居三大关键技术之首。国民经济的发展采取信息化带动现代化的方针，要求在所有领域中迅速推广信息技术，导致需要大量的计算机科学与技术领域的优秀人才。

计算机科学与技术的广泛应用是计算机学科发展的原动力，计算机科学是一门应用科学。因此，计算机学科的优秀人才不仅应具有坚实的科学理论基础，而且更重要的是能将理论与实践相结合，并具有解决实际问题的能力。培养计算机科学与技术的优秀人才是社会的需要、国民经济发展的需要。

制定科学的教学计划对于培养计算机科学与技术人才十分重要，而教材的选择是实施教学计划的一个重要组成部分，《21 世纪计算机科学与技术实践型教程》主要考虑了下述两方面。

一方面，高等学校的计算机科学与技术专业的学生，在学习了基本的必修课和部分选修课程之后，立刻进行计算机应用系统的软件和硬件开发与应用尚存在一些困难，而《21 世纪计算机科学与技术实践型教程》就是为了填补这部分空白。将理论与实际联系起来，使学生不仅学会了计算机科学理论，而且也学会应用这些理论解决实际问题。

另一方面，计算机科学与技术专业的课程内容需要经过实践练习，才能深刻理解和掌握。因此，本套教材增强了实践性、应用性和可理解性，并在体例上做了改进——使用案例说明。

实践型教学占有重要的位置，不仅体现了理论和实践紧密结合的学科特征，而且对于提高学生的综合素质，培养学生的创新精神与实践能力有特殊的作用。因此，研究和撰写实践型教材是必需的，也是十分重要的任务。优秀的教材是保证高水平教学的重要因素，选择水平高、内容新、实践性强的教材可以促进课堂教学质量的快速提升。在教学中，应用实践型教材可以增强学生的认知能力、创新能力、实践能力以及团队协作和交流表达能力。

实践型教材应由教学经验丰富、实际应用经验丰富的教师撰写。此系列教材的作者不但从事多年的计算机教学，而且参加并完成了多项计算机类的科研项目，他们把积累的经验、知识、智慧、素质融合于教材中，奉献给计算机科学与技术的教学。

我们在组织本系列教材过程中，虽然经过了详细的思考和讨论，但毕竟是初步的尝试，不完善甚至缺陷不可避免，敬请读者指正。

本系列教材主编 陈明

2005 年 1 月于北京

前 言

随着.NET开发平台的日益流行,作为.NET平台首选的程序设计语言C#以其功能强大、面向对象、易于学习等显著优点,成为主流的程序设计语言之一。

C#有功能强大的但稍显复杂的集成开发环境,有完整的面向对象的概念,有丰富的类库,还有对数据库、网络、安全、图形处理等应用领域广泛的支持。上述这些特点使C#功能非常强大,但同时也给初学者学习和使用C#带来了困难,使初学者感到知识体系繁杂,一时间难以理清思路、分清重点。

如何将C#中最重要、最常用的知识和技能从C#庞大的体系中提取出来,用什么方式传授给学生是C#老师首先要面对的问题。

为给初学者提供一个事半功倍的学习方法,本书在编写过程中摒弃了传统的“先系统学习理论知识,获得完整的知识体系后再完成实践操作”的学习模式,采用了“先操作、再学习,边模仿、边思考”的模式。该模式经过了3年的教学试点,教学实践表明,该模式大大降低了学习难度,使学生容易感受到学习的乐趣,从而大幅度地提高教学质量。

本书对C#的知识点进行了充分的筛选、梳理,将必须掌握的知识分解并构建为一个个小的自成体系的案例,读者仅需模仿案例,学习案例涉及的内容即可掌握最常用、最核心的知识和技能,当模仿完成所有案例后稍加总结即可以建立起完整的知识体系,而无须再考虑学什么、怎么学的问题。

本书所涉及的C#程序设计知识体系完整,讲述的基础知识适度,适合作为高等院校C#程序设计课程教材。本书还涉及大量的最新技术,对在程序设计领域工作的工程技术人员也有很好的参考作用。

本书是集体智慧的结晶,另外,负冰、杨东燕、张媛也参加了编写工作。

在本书的编写过程中力求准确体现C#核心特点、贴近实际教学需要、叙述简明清楚。但由于时间仓促,以及编者水平所限,书中难免存在疏漏和不妥之处,请读者批评指正。

编 者

2010年6月

目 录

第 1 章 C# 概述及开发环境	1
1.1 C# 的历史和特点	1
1.1.1 C# 历史	1
1.1.2 C# 特点	2
1.1.2.1 语法简洁	2
1.1.2.2 面向对象	2
1.1.2.3 可视化集成开发环境	2
1.1.2.4 自动内存管理	2
1.1.2.5 功能强大的类库	2
1.2 .NET 架构	3
1.3 任务 1: 安装 Visual Studio 2008	4
1.3.1 要求和目的	4
1.3.2 操作步骤	4
1.3.3 相关知识	5
1.3.3.1 C# 开发工具	5
1.3.3.2 Visual Studio 简介	6
1.4 任务 2: 建立最简单的 C# 程序	7
1.4.1 要求和目的	7
1.4.2 操作步骤	7
1.4.3 相关知识	9
1.4.3.1 C# 解决方案	9
1.4.3.2 C# 文件类型	9
1.4.3.3 Visual Studio 2008 主要功能	10
1.5 任务 3: 建立可交互的 C# 程序	14
1.5.1 要求和目的	14
1.5.2 操作步骤	15
1.5.3 相关知识	16
1.5.3.1 C# 程序结构	16
1.5.3.2 类和命名空间	17
1.5.3.3 控件、属性和方法	18
1.5.3.4 事件和事件驱动	18
1.6 任务 4: 更改文本框背景颜色	19
1.6.1 要求和目的	19
1.6.2 操作步骤	20
1.6.3 相关知识	21
1.6.3.1 窗体	21
1.6.3.2 命令按钮	21
1.6.3.3 标签	23
1.6.3.4 文本框	23
1.7 任务 5: 使用控制台方式创建 C# 程序	24
1.7.1 要求和目的	24
1.7.2 操作步骤	24
1.7.3 相关知识	25
1.7.3.1 C# 基本语法规则	25
1.7.3.2 C# 注释	25
1.8 小结	25
1.9 作业	25
第 2 章 C# 语言基础	27
2.1 任务 1: 计算圆的周长和面积	27
2.1.1 要求和目的	27
2.1.2 操作步骤	28
2.1.3 相关知识	29
2.1.3.1 变量	29
2.1.3.2 常量	30
2.1.3.3 文本	31
2.1.3.4 赋值语句	32
2.2 任务 2: 简单乘方运算	33
2.2.1 要求和目的	33
2.2.2 操作步骤	33
2.2.3 相关知识	34

2.2.3.1 数据类型	34	3.3 任务 3: 判断字符类型	56
2.2.3.2 简单类型	34	3.3.1 要求和目的	56
2.2.3.3 枚举类型	35	3.3.2 操作步骤	56
2.2.3.4 object 类型	37	3.3.3 相关知识	57
2.2.3.5 string 类型	37	3.3.3.1 多分支 if 语句格式	57
2.3 任务 3: 计算平均成绩	38	3.3.3.2 多分支 if 语句执行 过程	58
2.3.1 要求和目的	38	3.4 任务 4: 用户信息验证	58
2.3.2 操作步骤	38	3.4.1 要求和目的	58
2.3.3 相关知识	39	3.4.2 操作步骤	58
2.3.3.1 类型转换	39	3.4.3 相关知识	60
2.3.3.2 装箱和拆箱	42	3.5 任务 5: 成绩转换	60
2.3.3.3 MessageBox 类	42	3.5.1 要求和目的	60
2.4 任务 4: 温度转换	43	3.5.2 操作步骤	60
2.4.1 要求和目的	43	3.5.3 相关知识	62
2.4.2 操作步骤	44	3.5.3.1 switch 语句格式	62
2.4.3 相关知识	45	3.5.3.2 switch 语句执行过程	63
2.4.3.1 运算符和表达式	45	3.5.3.3 switch 语句的嵌套	63
2.4.3.2 算术运算符和表达式	45	3.6 任务 6: 判断正负数	63
2.4.3.3 关系和类型测试运算符 和表达式	46	3.6.1 要求和目的	63
2.4.3.4 逻辑运算符和表达式	48	3.6.2 操作步骤	63
2.4.3.5 位运算符和表达式	48	3.6.3 相关知识	64
2.4.3.6 赋值运算符	49	3.6.3.1 条件运算符格式	64
2.4.3.7 运算符的优先级	49	3.6.3.2 条件运算符执行过程	64
2.5 小结	50	3.7 小结	65
2.6 作业	51	3.8 作业	65
第 3 章 分支结构	52	第 4 章 循环结构	66
3.1 任务 1: 求最大数	52	4.1 任务 1: 计算奇数和及偶数和	66
3.1.1 要求和目的	52	4.1.1 要求和目的	66
3.1.2 操作步骤	53	4.1.2 操作步骤	67
3.1.3 相关知识	54	4.1.3 相关知识	67
3.1.3.1 单分支 if 语句格式	54	4.2 任务 2: 求自然对数 e 的近似值	68
3.1.3.2 单分支 if 语句执行 过程	54	4.2.1 要求和目的	68
3.2 任务 2: 判断奇偶数	54	4.2.2 操作步骤	69
3.2.1 要求和目的	54	4.2.3 相关知识	70
3.2.2 操作步骤	54	4.3 任务 3: 求最大公约数和最小公倍数	70
3.2.3 相关知识	55	4.3.1 要求和目的	70
3.2.3.1 双分支 if 语句格式	55	4.3.2 操作步骤	71
3.2.3.2 双分支 if 语句执行 过程	56	4.3.3 相关知识	72
		4.4 任务 4: 满足一定条件的最大的自然 数 n	73

4.4.1 要求和目的	73	5.3.3 相关知识	95
4.4.2 操作步骤	73	5.3.3.1 形参与实参	95
4.4.3 相关知识	74	5.3.3.2 通过值传递参数	95
4.5 任务 5: 有趣的 3 位数	76	5.3.3.3 使用 ref 方式传递参数	96
4.5.1 要求和目的	76	5.3.3.4 使用 out 方式传递参数	97
4.5.2 操作步骤	76	5.3.3.5 使用 params 方式传递参数	97
4.5.3 相关知识	77	5.4 任务 4: 封装示例	98
4.5.3.1 循环嵌套	77	5.4.1 要求和目的	98
4.5.3.2 获取自然数中的每位数字	77	5.4.2 操作步骤	98
4.5.3.3 其他相关知识	78	5.4.3 相关知识	99
4.6 任务 6: 求素数	79	5.4.3.1 问题分析	99
4.6.1 要求和目的	79	5.4.3.2 对象的封装	100
4.6.2 操作步骤	79	5.4.3.3 字段和属性	101
4.6.3 相关知识	80	5.5 任务 5: 继承示例	102
4.7 小结	81	5.5.1 要求和目的	102
4.8 作业	81	5.5.2 操作步骤	102
第 5 章 面向对象的基本概念	82	5.5.3 相关知识	104
5.1 任务 1: 类定义示例	83	5.5.3.1 问题分析	104
5.1.1 要求和目的	83	5.5.3.2 继承的概念	104
5.1.2 操作步骤	83	5.5.3.3 方法的改写	104
5.1.3 相关知识	84	5.5.3.4 密封类和密封方法	105
5.1.3.1 对象的概念	84	5.5.3.5 抽象类和抽象方法	106
5.1.3.2 类的概念	84	5.5.3.6 虚方法	107
5.1.3.3 定义类	84	5.5.3.7 base 关键字	108
5.1.3.4 访问修饰符	85	5.6 小结	109
5.1.3.5 构造函数和析构函数	85	5.7 作业	109
5.1.3.6 创建、使用对象	86	第 6 章 多态、事件和接口	111
5.1.3.7 静态成员	87	6.1 任务 1: 说 Hello	111
5.1.3.8 结构	87	6.1.1 要求和目的	111
5.2 任务 2: 兔子生子	89	6.1.2 操作步骤	112
5.2.1 要求和目的	89	6.1.3 相关知识	112
5.2.2 操作步骤	89	6.1.3.1 方法的重载	112
5.2.3 相关知识	90	6.1.3.2 构造函数重载	113
5.2.3.1 问题分析	90	6.2 任务 2: 两学生 PK 成绩	113
5.2.3.2 方法定义	91	6.2.1 要求和目的	113
5.2.3.3 方法调用	91	6.2.2 操作步骤	114
5.2.3.4 递归	91	6.2.3 相关知识	115
5.3 任务 3: 聪明猪	92	6.2.3.1 操作符重载的概念	115
5.3.1 要求和目的	92		
5.3.2 操作步骤	93		

6.2.3.2 操作符重载格式	116	7.2.2 操作步骤	133
6.3 任务3: 师生秀	116	7.2.3 相关知识	134
6.3.1 要求和目的	116	7.2.3.1 二维数组的声明和 初始化	134
6.3.2 操作步骤	116	7.2.3.2 二维数组遍历	134
6.3.3 相关知识	118	7.3 任务3: 用交错数组存储数据	135
6.3.3.1 多态的概念	118	7.3.1 要求和目的	135
6.3.3.2 多态的作用	118	7.3.2 操作步骤	135
6.4 任务4: 说汉语说英语	118	7.3.3 相关知识	136
6.4.1 要求和目的	118	7.3.3.1 交错数组的声明和 初始化	136
6.4.2 操作步骤	119	7.3.3.2 交错数组的遍历	137
6.4.3 相关知识	120	7.4 任务4: 数组的排序(1)	138
6.4.3.1 委托的概念	120	7.4.1 要求和目的	138
6.4.3.2 定义委托	120	7.4.2 操作步骤	138
6.4.3.3 调用委托	120	7.4.3 相关知识	139
6.5 任务5: 清晨鸡叫	120	7.4.3.1 Array类的常用属性	139
6.5.1 要求和目的	120	7.4.3.2 Array类的常用方法	140
6.5.2 操作步骤	121	7.5 任务5: 数组的排序(2)	140
6.5.3 相关知识	122	7.5.1 要求和目的	140
6.5.3.1 事件的概念	122	7.5.2 操作步骤	141
6.5.3.2 事件的定义	122	7.5.3 相关知识	142
6.5.3.3 事件的订阅	123	7.5.3.1 声明定制类型的数组	142
6.6 任务6: 老鹰和直升机	123	7.5.3.2 使用 IComparable 接口 对数组排序	142
6.6.1 要求和目的	123	7.6 任务6: 用集合类 ArrayList 存储 数据	144
6.6.2 操作步骤	124	7.6.1 要求和目的	144
6.6.3 相关知识	126	7.6.2 操作步骤	144
6.6.3.1 接口的概念	126	7.6.3 相关知识	148
6.6.3.2 接口定义	126	7.6.3.1 ArrayList 集合类	148
6.6.3.3 接口和抽象类	127	7.6.3.2 ArrayList 的方法	149
6.7 小结	127	7.6.3.3 关于 InputBox() 函数	149
6.8 作业	127	7.7 任务7: 用集合类 List<T> 存储 数据	150
第7章 数组和集合	129	7.7.1 要求和目的	150
7.1 任务1: 用一维数组存储数据	129	7.7.2 操作步骤	150
7.1.1 要求和目的	129	7.7.3 相关知识	153
7.1.2 操作步骤	130	7.7.3.1 List<T> 集合类	153
7.1.3 相关知识	130	7.7.3.2 List<T> 的方法	153
7.1.3.1 数组的概念	130	7.8 小结	154
7.1.3.2 一维数组的声明和 初始化	131		
7.1.3.3 一维数组的遍历	132		
7.2 任务2: 用二维数组存储数据	132		
7.2.1 要求和目的	132		

7.9 作业	154	8.5.3.2 MailMessage 类	178
第 8 章 .NET 类库	156	8.5.3.3 Attachment 类	178
8.1 .NET 类库概述	156	8.5.3.4 SmtplibClient 类	180
8.2 任务 1: 小学生算术测验	158	8.6 任务 5: 多线程应用	181
8.2.1 要求和目的	158	8.6.1 要求和目的	181
8.2.2 操作步骤	159	8.6.2 操作步骤	181
8.2.3 相关知识	160	8.6.3 相关知识	182
8.2.3.1 System 命名空间	160	8.6.3.1 多线程的基本概念	182
8.2.3.2 Math 类	161	8.6.3.2 System.Threading 命名空间	183
8.2.3.3 Random 类	162	8.6.3.3 Thread 类	184
8.2.3.4 DateTime 结构	162	8.6.3.4 线程的状态	184
8.2.3.5 String 类	163	8.6.3.5 线程的优先级	186
8.2.3.6 Convert 类	164	8.7 任务 6: 文件加密	186
8.2.3.7 DayOfWeek 枚举	164	8.7.1 要求和目的	186
8.3 任务 2: 绘制花瓣图案	165	8.7.2 操作步骤	187
8.3.1 要求和目的	165	8.7.3 相关知识	188
8.3.2 操作步骤	165	8.7.3.1 数据加密的基本概念	188
8.3.3 相关知识	166	8.7.3.2 主要加密算法	189
8.3.3.1 System.Drawing 命名 空间	166	8.7.3.3 System.Security.Cryptography 命名空间	190
8.3.3.2 Point 结构	166	8.7.3.4 DES 类	190
8.3.3.3 Color 结构	167	8.7.3.5 CryptoStream 类	190
8.3.3.4 Pen 类	168	8.8 小结	191
8.3.3.5 Bitmap 类	169	8.9 作业	191
8.3.3.6 Graphics 类	170	第 9 章 文件和流	193
8.4 任务 3: 简单端口扫描器	172	9.1 基本概念	193
8.4.1 要求和目的	172	9.1.1 文件	193
8.4.2 操作步骤	172	9.1.2 文件分类	193
8.4.3 相关知识	174	9.1.2.1 程序文件和数据文件	193
8.4.3.1 System.NET 命名 空间	174	9.1.2.2 ASCII 文件和二进制 文件	193
8.4.3.2 System.NET.Sockets 命名空间	175	9.1.3 流与 System.IO 命名空间	194
8.4.3.3 TcpClient 类	175	9.2 任务 1: 文本文件读写	194
8.4.3.4 Dns 类	176	9.2.1 要求和目的	194
8.5 任务 4: 简单邮件发送系统	177	9.2.2 操作步骤	195
8.5.1 要求和目的	177	9.2.3 相关知识	197
8.5.2 操作步骤	177	9.2.3.1 FileStream 类	197
8.5.3 相关知识	178	9.2.3.2 StreamReader 类	199
8.5.3.1 System.NET.Mail 命名空间	178	9.2.3.3 StreamWriter 类	201
		9.3 任务 2: 二进制文件读写	202

9.3.1 要求和目的	202	10.4.3.1 控件常用属性	230
9.3.2 操作步骤	203	10.4.3.2 控件常用事件	230
9.3.3 相关知识	204	10.5 任务 5: 简易文本编辑器	231
9.3.3.1 BinaryReader 类	204	10.5.1 要求和目的	231
9.3.3.2 BinaryWriter 类	205	10.5.2 操作步骤	231
9.4 任务 3: 简单文件管理器	206	10.5.3 相关知识	233
9.4.1 要求和目的	206	10.5.3.1 文件对话框	
9.4.2 操作步骤	207	OpenFileDialog	235
9.4.3 相关知识	210	10.5.3.2 保存文件对话框	
9.4.3.1 文件管理控件	210	SaveFileDialog	235
9.4.3.2 File 类	210	10.5.3.3 字体对话框	
9.5 小结	214	FontDialog	235
9.6 作业	214	10.5.3.4 颜色对话框	
第 10 章 用户界面设计	216	ColorDialog	235
10.1 任务 1: 图片浏览	216	10.5.3.5 RichTextBox 控件	235
10.1.1 要求和目的	216	10.6 任务 6: 菜单、工具栏和状态栏的	
10.1.2 操作步骤	216	设计	236
10.1.3 相关知识	218	10.6.1 要求和目的	236
10.1.3.1 PictureBox 控件	218	10.6.2 操作步骤	237
10.1.3.2 ImageList 控件	219	10.6.3 相关知识	240
10.2 任务 2: 字体格式设置	220	10.6.3.1 菜单的设计	240
10.2.1 要求和目的	220	10.6.3.2 工具栏和状态栏的	
10.2.2 操作步骤	220	设计	240
10.2.3 相关知识	222	10.7 任务 7: 多重窗体	241
10.2.3.1 分组控件	222	10.7.1 要求和目的	241
10.2.3.2 单选按钮 RadioButton		10.7.2 操作步骤	241
和复选框 CheckBox		10.7.3 相关知识	243
.....	222	10.7.3.1 多重窗体的操作	243
10.2.3.3 New Font 构造函数	223	10.7.3.2 多文档的界面	244
10.3 任务 3: 学生信息录入界面	223	10.8 任务 8: 打字小游戏	245
10.3.1 要求和目的	223	10.8.1 要求和目的	245
10.3.2 操作步骤	224	10.8.2 操作步骤	246
10.3.3 相关知识	226	10.8.3 相关知识	248
10.3.3.1 列表框 ListBox	226	10.8.3.1 键盘事件 KeyPress	249
10.3.3.2 复选列表框		10.8.3.2 键盘事件 KeyDown	
CheckedListBox	227	和 KeyUp 事件	249
10.3.3.3 组合框 ComboBox	228	10.9 任务 9: 小小画笔	250
10.4 任务 4: 设置闹钟	228	10.9.1 要求和目的	250
10.4.1 要求和目的	228	10.9.2 操作步骤	250
10.4.2 操作步骤	229	10.9.3 相关知识	251
10.4.3 相关知识	230	10.10 小结	252
		10.11 作业	252

第 11 章 ADO.NET 和数据库	255	11.5.1 要求和目的	277
11.1 任务 1: 创建数据库	255	11.5.2 操作步骤	277
11.1.1 要求和目的	255	11.5.3 相关知识	282
11.1.2 操作步骤	256	11.5.3.1 DataGridView 控件	282
11.1.3 相关知识	259	11.5.3.2 更新数据源	283
11.1.3.1 数据库的基本概念	259	11.6 小结	283
11.1.3.2 SQL 查询	261	11.7 作业	283
11.2 任务 2: 连接数据库	262	第 12 章 异常处理	284
11.2.1 要求和目的	262	12.1 任务 1: 异常处理示例 I	284
11.2.2 操作步骤	262	12.1.1 要求和目的	284
11.2.3 相关知识	263	12.1.2 操作步骤	285
11.2.3.1 ADO.NET	263	12.1.3 相关知识	285
11.2.3.2 System.Data.OleDb		12.1.3.1 错误和异常	285
命名空间中的对象	263	12.1.3.2 try-catch 语句	285
11.2.3.3 System.Data.SqlClient		12.2 任务 2: 异常处理示例 II	287
命名空间中的对象	264	12.2.1 要求和目的	287
11.3 任务 3: 操作数据库	265	12.2.2 操作步骤	288
11.3.1 要求和目的	265	12.2.3 相关知识	289
11.3.2 操作步骤	266	12.2.3.1 Exception 类	289
11.3.3 相关知识	271	12.2.3.2 常用异常类	289
11.3.3.1 创建 OleDbDataAdapter		12.2.3.3 各异常类之间的	
.....	271	继承关系	289
11.3.3.2 创建和填充 DataTable		12.3 任务 3: 主动抛出异常示例	290
.....	271	12.3.1 要求和目的	290
11.3.3.3 创建 DataRow 对象	271	12.3.2 操作步骤	291
11.4 任务 4: BindingSource 组件的应用(1)		12.3.3 相关知识	291
.....	272	12.3.3.1 throw 语句	291
11.4.1 要求和目的	272	12.3.3.2 if 语句和抛出异常	292
11.4.2 操作步骤	272	12.4 小结	293
11.4.3 相关知识	276	12.5 作业	293
11.4.3.1 BindingSource 组件	276	参考文献	294
11.4.3.2 数据绑定	276		
11.5 任务 5: BindingSource 组件的应用(2)			
.....	277		

第 1 章 C# 概述及开发环境

学习提示

本节以 5 个任务为线索介绍了 C# 所涉及的基本概念、程序结构和开发环境,涉及的问题、概念非常多,在本节中,建议读者重点掌握简单 C# 程序的结构和基本开发方法,其他细节问题可留待后面的章节解决。

在本书的叙述中会大量涉及具体操作,在描述具体操作过程时,做如下约定:将涉及的操作对象用【 】符号括住,如单击【开始】打开 Windows 开始菜单,若在符号【 】前没有特别说明表示单击,即【开始】等同于单击【开始】菜单。用符号→表示下一步,如【开始】→【程序】。

C# 是微软公司(Microsoft)推出的新一代软件开发平台.NET 提供的 4 种默认的程序设计语言之一,它集中体现了微软公司软件产品一贯的功能强大、用户界面友好、学习方便、相关资源丰富的特点。

1.1 C# 的历史和特点

1.1.1 C# 历史

C# 作为 .NET 开发平台的一部分,是和 .NET Framework 及其开发环境 Visual Studio 一同成长起来的。

C# 的历史可以追溯到 1998 年微软公司的 COOL 语言项目,2000 年 7 月微软公司发布了 C# 语言的第一个预览版;

2002 年 2 月,微软公司推出 .NET Framework 1.0 版和 .NET 开发环境 Visual Studio .NET 2002,同时推出 C# 1.0 版,该版本是 C# 的第一个正式版本;

2003 年 5 月,微软公司推出了 .NET Framework 1.1 和 Visual Studio .NET 2003,同时也发布了 C# 1.1 版;

2005 年 10 月,微软公司推出了 .NET Framework 2.0 和 C# 2.0 版;

2005 年 11 月,微软公司发布 Visual Studio 2005 正式版;

2006 年 11 月,微软公司发布 .NET Framework 3.0;

2007 年 8 月,微软公司发布 C# 3.0 版;

2007年11月,微软公司发布.NET Framework 3.5和Visual Studio 2008;目前最新的版本是C# 4.0版,对应的开发平台是Visual Studio 2010 Beta版。

1.1.2 C#特点

C#是由C/C++衍生出来的,在继承了C/C++强大功能的同时,抛弃了C/C++一些复杂的特性,如宏、模板、多重继承等,在发展过程中又广泛借鉴了Java的主要优点,是一种安全的、稳定的、简单的、优雅的程序设计语言,很好地体现了当今软件开发的发展趋势及最新的研究成果。

C#具有功能强大、语法简洁、面向对象、提供了完整的安全性、支持组件技术、具有自动内存管理、良好的版本控制能力以及与Web紧密结合等一系列优点。C#最重要的特点如下所述。

1.1.2.1 语法简洁

C#是由C/C++衍生而来的,其语法非常类似于C++和Java,许多关键字都是相同的。语法中的冗余是C++中常见的问题之一,C#对此进行了简化,只保留了常见的形式。

1.1.2.2 面向对象

C#采用了面向对象设计思想,它将复杂的问题分解为一个能够完成独立功能的相对简单的对象的集合。在程序设计领域中,对象可视为一个可操作实体,每个对象具有属性、方法,也可以响应特定的事件。

C#具有面向对象程序设计语言的所有特征,可很好地支持抽象、封装、继承、重载、多态等特性。

1.1.2.3 可视化集成开发环境

C#采用了可视化编程方式,用户界面良好。程序设计者可以像搭积木一样,根据界面设计要求,通过直接拖放控件来设计界面,所见即所得,非常方便、高效。

Visual Studio是.NET平台默认的集成开发环境,在这个环境中,C#程序设计者可以完成程序设计的所有工作,如设计界面、编写代码、调试、编译成为可在Windows中运行的可执行文件、生成安装程序等。

1.1.2.4 自动内存管理

C#具有自动内存管理机制,该机制也被称为垃圾收集机制,系统会根据一定算法自动回收不再被使用对象所占用的内存,而无须程序员手工释放内存,把程序员从繁重的内存管理上解放出来。

1.1.2.5 功能强大的类库

.NET类库内容非常丰富,提供了包罗万象的处理功能,通过引用.NET类库可以方便、高效地完成各种程序设计工作,对.NET类库的学习是C#的重点内容之一。

1.2 .NET 架构

.NET 架构(.NET Framework)是当前程序设计的主流体系之一,代表了程序设计技术发展的方向。按照微软总裁兼首席执行官 Steve Ballmer 的说法,.NET 是个集合,一个环境,一个可以作为平台支持下一代 Internet 的可编程结构。.NET 架构如图 1-1 所示,微软公司还为.NET 提供了一个强有力的开发工具 Visual Studio。

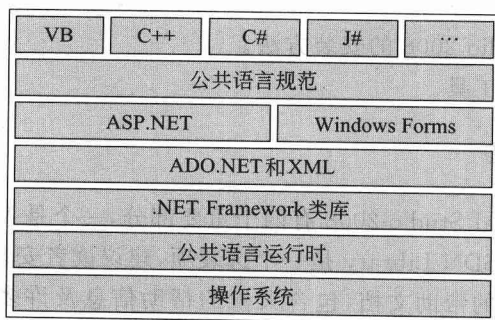


图 1-1 .NET 架构

- .NET 语言(.NET Language): 符合公共语言规范要求,能被编译成 MSIL 的程序设计语言。.NET 提供了 4 种程序设计语言 Visual Basic .NET、Visual C++ .NET、Visual C# .NET 和 Visual J# .NET。此外还有大量的符合公共语言规范的第三程序设计语言可供选择。
- 公共语言规范(Common Language Specification): 描述了.NET 框架下各种程序设计语言必须遵守的共同的特征。
- ASP.NET 和 Windows Forms: 是.NET 主要的界面技术,ASP.NET 是面向 Web 服务的,主要有 Web 应用程序和 Web 服务,Windows Forms 是面向传统 Windows 应用程序的。
- ADO.NET 和 XML: 为.NET 提供了统一的数据访问技术。
- .NET Framework 类库(.NET Framework Class Libraries): 包含大量常用功能的代码库,用户可以通过继承来使用代码库中已有的代码。
- 公共语言运行时(Common Language Runtime,CLR): 提供所有核心服务,如内存管理、调试支持、执行 MSIL 代码以及与 Windows 和 IIS 交换时所涉及的所有核心任务。

微软中间语言(Microsoft Intermediate Language,MSIL)是.NET 的通用语言,用户编写的所有程序在执行前都会被优化并编译为 MSIL 代码。MSIL 与机器无关,可以在任何装有 CLR 的计算机上运行。

当 MSIL 被执行时,CLR 通过 JIT(Just-In-Time)编译器将 MSIL 代码进行最后的、与机器匹配的优化,然后将其编译为真正的机器语言。

1.3 任务 1: 安装 Visual Studio 2008

1.3.1 要求和目的

要求

在 Windows 平台上安装 Visual Studio 2008。

目的

- 掌握 Visual Studio 2008 的安装方法。
- 了解 C# 的开发工具。

1.3.2 操作步骤

如图 1-2 所示, Visual Studio 2008 有两个重要部分, 一个是 Visual Studio 2008, 另一个是 MSDN Library, MSDN Library 是一个选装项, 建议读者安装 MSDN。

MSDN 是厂商提供的帮助文档, 包含大量的帮助信息及许多使用案例, 无论是对熟练的 .NET 程序员还是对初学者都有很大的帮助。



图 1-2 Visual Studio 2008 安装窗口

选择【安装 Visual Studio 2008】后即进入如图 1-3 所示的安装向导窗口。

单击【下一步】按钮后即进入图 1-4 所示的窗口, 在该窗口中用户需要输入软件的产品密钥、名称等信息。

Visual Studio 2008 是一个庞大的集成开发环境, 用户可以根据需要选择安装部分组件, 在图 1-5 所示窗口中, 可选择典型安装、完全安装或自定义安装。

在自定义安装方式下, 可以按照用户需要选择相应组件。