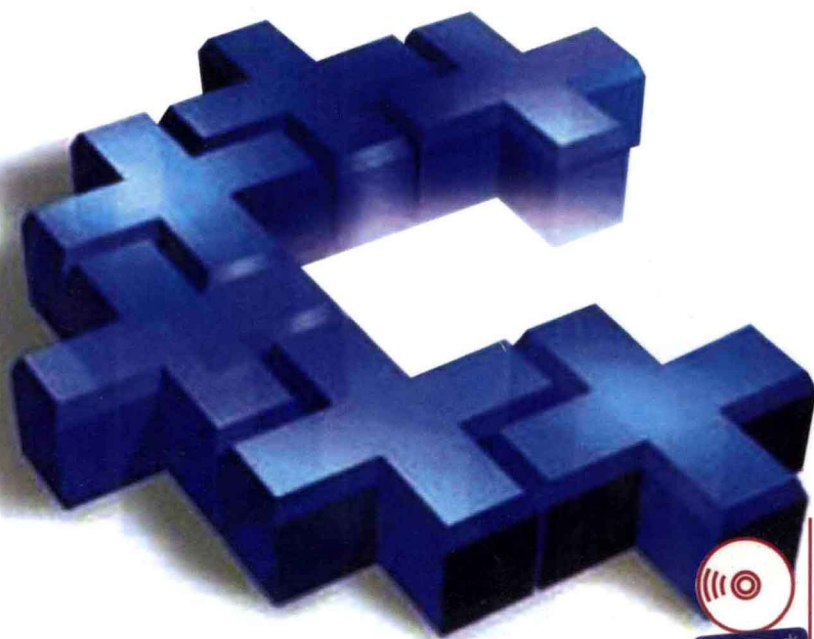


C#

程序设计

基础教程与实验指导

孙晓非 冉晓旻 冯冠 等编著



附
光
盘

- 完整实例源代码
- 12 段教学视频文件

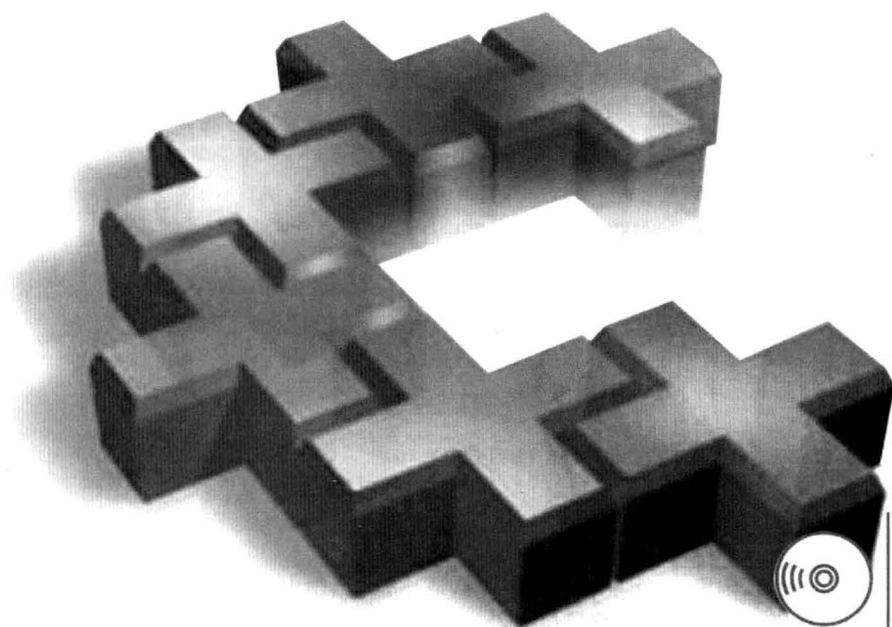


清华大学出版社

C#

程序设计 基础教程与实验指导

孙晓非 冉晓旻 冯冠 等编著



附
光
盘

- 完整实例源代码
- 12 段教学视频文件

清华大学出版社
北 京

内 容 简 介

本书由浅入深地介绍了 C# 开发知识。全书共分上下两篇。上篇为 11 章, 分别介绍了 C# 基础知识, 变量、数据类型、运算符与表达式, 控制语句, 集合, 面向对象编程, String 类, try/catch 块、throw 子句、异常涉及的类以及创建用户自定义异常, 基本控件, 路径、目录、文件及其相关类, ADO.NET 数据库编程, GDI+ 知识等内容。第 11 章是综合案例, 对工资管理系统需求和功能进行分析, 并使用 C# 来实现工资管理系统。本书附有配套光盘, 提供了书中实例的源代码和视频教学文件。下篇是实验指导, 有 10 个单元, 选择了一批实用的上机练习, 以巩固和提高读者的动手技能, 加深对基础理论的理解。

本书体现了作者在软件技术教学改革过程中形成的“项目驱动、案例教学、理论实践一体化”教学方法, 可以作为 C# 职业培训教材和各级院校 C# 授课培训教程, 也适合作为 C# 自学资料和参考资料。

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# 程序设计基础教程与实验指导/孙晓非, 冉晓旻, 冯冠等编著. —北京: 清华大学出版社, 2008.11

ISBN 978-7-302-17586-5

I. C… II. ①孙…②冉…③冯… III. C 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 066074 号

责任编辑: 夏兆彦

责任校对: 徐俊伟

责任印制: 孟凡玉

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: 010-62776969, c-service@tup.tsinghua.edu.cn

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

印 刷 者: 北京市清华园胶印厂

装 订 者: 三河市溧源装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 21.75 字 数: 513 千字

附光盘 1 张

版 次: 2008 年 11 月第 1 版

印 次: 2008 年 11 月第 1 次印刷

印 数: 1~5000

定 价: 39.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 028261-01

Visual C#是 Microsoft 公司开发的一种使用简单、功能强大、面向组件、表达力丰富的语言。它结合了 C++强大灵活和 Java 语言简洁等特性,还吸取了 Delphi 和 Visual Basic 所具有的易用性。C#在 Microsoft .NET Framework 中扮演着重要角色。它是 Microsoft 公司面向下一代互联网软件和服务战略的重要技术。

.NET Framework 的基础是公共语言运行时。它是执行时管理代码的代理,提供内存管理、线程管理和远程处理等核心服务。本书以 Visual Studio.NET 2005 为基础,介绍 C#语言的知识以及窗体应用程序的开发技巧,并配有完整的开发实例讲解。

本书定位与特色

□ 面向职业技术教学

本书是作者在总结了多年开发经验与成果的基础上编写的,以实际项目为中心,全面、翔实地介绍了 C#开发所需的各种知识和技能。通过对本书的学习,读者可以快速、全面地掌握使用 C#进行面向对象开发的方法。本书体现了作者“项目驱动、案例教学、理论实践一体化”教学理念,是一本真正面向职业技术教学的教材。

□ 合理的知识结构

面向程序员职业培训市场,结合程序开发实践介绍 C#编程知识,突出了职业实用性。全书各章都有实例分析,带领读者经历程序开发全过程,是一本真正的实训性案例教程。

□ 真实的案例教学

针对每个知识点,本书设计了针对性强的教学案例,这些案例既相对独立,又具有一定的联系,是综合性开发实例的组成部分。学生在制作这些案例的过程中可以掌握每个知识点。本书有效地避免了其他 C#教程抽象、空洞,或者实例不丰富等缺点。

□ 理论实践一体化

在每个案例中有机地融合了知识点讲解和技能训练目标,融“教、学、练”于一体。每个案例的讲解都先提出功能目标,然后是实例制作演示和学生模仿练习,让学生掌握案例的完成过程,体现“在练中学,学以致用”的教学理念。

□ 阶梯式实践环节

本书精心设置了 3 个教学环节:课堂练习、扩展练习、综合实训。让学生通过不断练习实践,实现编程技能的逐步推进,最终实现与职业能力的接轨。

本书主要内容

本书由浅入深地介绍了 C#开发知识。全书共分为上下篇。上篇 11 章,介绍了 .NET Framework 及其类库、公共语言运行时、程序集、命名空间等 C#基础知识;变量、数据类型、运算符与表达式、结构类型、枚举类型、控制语句等;集合知识;面向对象的类、

对象、类成员变量、类的继承、抽象类与接口等知识；String 类和 StringBuilder 类及其方法、try/catch 块、throw 子句、异常涉及的类、finally 块以及创建用户自定义异常等知识；基本控件、图形和图像控件、按钮类控件、列表类控件以及容器类控件；路径、目录、文件及其相关类的知识；ADO.NET 数据库编程知识；GDI+知识。下篇是实验指导，分为 10 个单元，精选了一批实用的上机练习，以巩固和提高读者的动手技能，又加深对基础理论的理解和应用

第 11 章为综合案例，对工资管理系统的需求和功能进行分析，介绍了系统总体结构和用例图，最后介绍了数据库设计，并使用 C#来实现工资管理系统。每章都详细剖析了实现这些模块的 C#代码。

下篇是实验指导，分为 10 个单元，精选了一批实用的上机练习，以巩固和提高读者的动手技能，又加深对基础理论的理解和应用。

本书附有配套光盘。光盘提供了书中实例的源代码，全部经过精心调试，在 Windows XP/Windows 2000/Windows 2003 Server 下全部通过，能够保证正常运行。

读者对象

本书体现了作者在软件技术教学改革过程中形成的“项目驱动、案例教学、理论实践一体化”教学方法，读者通过本书可以快速、全面地掌握使用 C#应用于 Windows 编程的开发经验和技能。本书可以作为 C#职业培训教材和各级院校 C#授课培训教程，也适合作为 C#自学资料和参考资料。

除了封面署名人员之外，参与本书编写的人员还有李乃文、孙岩、马海军、张仕禹、夏小军、赵振江、李振山、李文采、吴越胜、李海庆、何永国、李海峰、陶丽、吴俊海、安征、张巍屹、崔群法、王咏梅、康显丽、辛爱军、牛小平、贾栓稳、王立新、苏静、赵元庆、郭磊、徐铭、李大庆、王蕾、张勇、郝安林等。

因时间仓促，在编写过程中难免会有漏洞，欢迎读者登录清华大学出版社网站 www.tup.tsinghua.edu.cn 与我们联系，以帮助我们改正提高。

编 者

2008 年 4 月

上篇 基础知识.....1

第 1 章 C#基础知识1

- 1.1 .NET Framework 概述1
- 1.2 公共语言运行时4
 - 1.2.1 公共类型系统4
 - 1.2.2 公共语言规范5
 - 1.2.3 中间语言5
 - 1.2.4 执行管理6
 - 1.2.5 垃圾回收机制7
- 1.3 .NET Framework 类库8
- 1.4 命名空间9
 - 1.4.1 命名空间的组织方式9
 - 1.4.2 定义命名空间11
 - 1.4.3 使用.NET Framework
类库13
- 1.5 程序集14
- 1.6 配置 C#环境15
 - 1.6.1 安装 Visual Studio 200515
 - 1.6.2 熟悉 Visual Studio 2005
开发环境17

第 2 章 C#编程基础19

- 2.1 变量和数据类型19
 - 2.1.1 使用变量和数据类型19
 - 2.1.2 声明和初始化变量21
 - 2.1.3 改变数据类型22
 - 2.1.4 使用引用变量25
- 2.2 运算符与表达式29
 - 2.2.1 运算符29
 - 2.2.2 表达式30
- 2.3 控制语句31
 - 2.3.1 条件语句31

2.3.2 循环语句34

2.3.3 跳转语句36

2.4 结构和枚举37

2.4.1 结构38

2.4.2 枚举38

2.5 扩展练习39

第 3 章 使用集合40

3.1 引用集合项40

3.2 System.Collections 命名空间41

3.3 使用集合43

3.3.1 使用堆栈43

3.3.2 使用队列44

3.3.3 使用散列表45

3.3.4 使用排序列表47

3.3.5 使用位数组48

3.3.6 使用数组列表49

3.4 创建对象集合50

3.4.1 管理员对象集合50

3.4.2 声明集合51

3.4.3 向集合中添加对象53

3.4.4 从集合中删除元素54

3.4.5 在集合中检索元素55

3.5 扩展练习56

第 4 章 面向对象58

4.1 面向对象概述58

4.2 类的结构60

4.2.1 定义类60

4.2.2 定义成员方法61

4.2.3 方法的返回值63

4.2.4 成员方法重载64

4.2.5 构造方法66

目录

4.3 创建对象.....	67	5.5 扩展练习.....	104
4.3.1 对象的创建.....	67	第6章 结构化的异常处理.....	106
4.3.2 对象的使用.....	68	6.1 结构化异常处理基本知识.....	106
4.3.3 对象的生存周期.....	69	6.1.1 抛出和捕获异常.....	106
4.3.4 对象的清理.....	70	6.1.2 嵌套 try 语句.....	108
4.4 类成员变量.....	72	6.2 异常类.....	110
4.4.1 访问修饰符.....	72	6.2.1 基于类型的筛选异常.....	110
4.4.2 static 静态成员.....	73	6.2.2 System.Exception 类的 成员.....	111
4.4.3 sealed 关键字.....	74	6.2.3 内部异常.....	112
4.5 继承.....	74	6.2.4 抛出预定义异常的实例.....	113
4.5.1 继承的意义.....	74	6.3 用户自定义的异常类.....	115
4.5.2 如何定义派生类.....	75	6.4 finally 块.....	116
4.5.3 覆盖基类成员方法.....	76	6.5 扩展练习.....	118
4.5.4 this 关键字的使用.....	78	第7章 Windows 窗体控件.....	119
4.5.5 base 关键字的使用.....	78	7.1 Windows 窗体控件概述.....	119
4.6 抽象类与接口.....	79	7.2 基本控件.....	122
4.6.1 抽象类.....	79	7.3 显示信息的控件.....	124
4.6.2 接口.....	80	7.4 图形和图像类控件.....	126
4.7 扩展练习.....	82	7.4.1 ImageList 控件.....	126
第5章 字符串.....	83	7.4.2 PictureBox 控件.....	127
5.1 字符串基础.....	83	7.5 按钮类控件.....	127
5.2 String 类.....	84	7.5.1 Button 控件.....	127
5.2.1 String 类的构造函数.....	84	7.5.2 RadioButton 控件.....	128
5.2.2 String 类的属性和索引器.....	86	7.5.3 CheckBox 控件.....	129
5.2.3 比较字符串.....	87	7.5.4 ToolBar 控件.....	129
5.2.4 定位与查找字符串.....	88	7.5.5 NotifyIcon 控件.....	131
5.2.5 提取字符串.....	89	7.6 列表类控件.....	132
5.2.6 拆分字符串.....	90	7.6.1 ListBox 控件.....	132
5.2.7 String 类中的其他 常用方法.....	91	7.6.2 ComboBox 控件.....	133
5.3 StringBuilder 类.....	94	7.6.3 ListView 控件.....	134
5.3.1 追加字符串.....	94	7.7 容器类控件.....	136
5.3.2 使用 AppendFormat.....	97	7.8 扩展练习.....	140
5.3.3 插入字符串.....	98	第8章 文件存取 I/O.....	141
5.3.4 替换字符串.....	100	8.1 System.IO 简介.....	141
5.3.5 StringBuilder 类的其他 常用成员.....	100	8.2 路径、目录和文.....	142
5.4 格式化.....	102	8.2.1 Directory 类.....	143

8.2.2	File 类	144
8.2.3	Path 类	147
8.2.4	DirectoryInfo 类和 FileInfo 类	148
8.3	目录和文件对话框	150
8.3.1	OpenFileDialog 类	150
8.3.2	SaveFileDialog 类	152
8.4	流和顺序存取方法	153
8.5	二进制文件的读写	154
8.5.1	FileStream 类	154
8.5.2	BinaryWriter 类和 BinaryReader 类	156
8.5.3	二进制文件读写的示例	157
8.6	扩展练习	159
第 9 章	ADO.NET 数据库编程	161
9.1	引言	161
9.2	ADO.NET 基础	163
9.3	定义一个数据库连接	165
9.3.1	定义一个连接字符串	165
9.3.2	存储连接字符串	168
9.3.3	读取连接字符串	169
9.3.4	测试连接	170
9.4	操作数据库	172
9.4.1	使用 DataAdapter 填充 DataSet 对象	172
9.4.2	使用 DataReader 类	173
9.4.3	使用 DataTable 和 DataView 类	175
9.4.4	定义数据库关系	177
9.4.5	使用 Command 对象	178
9.4.6	使用 Command Builder 生成 SQL 语句	179
9.5	DataSet 应用	181
9.5.1	更新 DataSet	181
9.5.2	向 DataSet 添加数据	183
9.5.3	对 DataSet 排序和筛选	185
9.6	扩展练习	187

第 10 章	GDI+简介	188
10.1	图形绘制概述	188
10.1.1	了解 GDI+	188
10.1.2	Graphics 类	190
10.1.3	GDI+的坐标系统	191
10.1.4	颜色	192
10.2	画笔与绘制基本图形	193
10.3	画刷与区域填充	196
10.3.1	使用 SolidBrush 填充 图形	196
10.3.2	使用 HatchBrush 填充图形	198
10.3.3	使用 TextureBrush 填充图形	199
10.3.4	使用 LinearGradientBrush 填充图形	200
10.4	字体与绘制文本	202
10.5	绘制复杂的形状	204
10.6	打印图形	205
10.6.1	PrintDocument 类	205
10.6.2	PrintDialog 控件	207
10.6.3	PrintPreview 控件	208
10.7	扩展练习	210
第 11 章	工资管理系统	212
11.1	系统分析	212
11.1.1	需求分析	212
11.1.2	功能分析	213
11.2	系统概述	214
11.2.1	系统总体结构	214
11.2.2	系统用例图	214
11.3	系统模块设计	218
11.3.1	登录管理	218
11.3.2	角色信息管理	219
11.3.3	员工信息管理	220
11.3.4	工资类别管理	221
11.3.5	工资支出管理	221
11.3.6	信息浏览与查看	222
11.4	数据库设计	222

目录

11.5 登录模块.....	224	练习 5-3 将当前日期与时间 格式化为标准格式.....	291
11.5.1 用户登录.....	224	练习 5-4 获取指定字符在 字符串中的位置.....	293
11.5.2 密码修改.....	232	第 6 单元.....	295
11.6 角色信息管理模块.....	235	练习 6-1 处理除数为 0 的异常.....	295
11.7 员工信息管理模块.....	239	练习 6-2 处理空字符转换数字 异常.....	296
11.7.1 员工信息添加.....	239	练习 6-3 处理溢出异常.....	298
11.7.2 管理员工信息.....	242	第 7 单元.....	300
11.8 工资支出模块.....	245	练习 7-1 制作管理员登录窗体.....	300
11.8.1 添加支出.....	245	练习 7-2 制作添加员工窗体.....	302
11.8.2 支出查询.....	245	练习 7-3 制作支出类别窗体.....	303
练习 7-4 制作工资支出窗体.....	305	第 8 单元.....	306
下篇 实验指导.....	249	练习 8-1 列表显示文件.....	306
第 1 单元.....	249	练习 8-2 列表显示驱动器信息.....	308
练习 1-1 自定义 Visual Studio 2005 环境.....	249	练习 8-3 模拟【我的电脑】.....	311
练习 1-2 设置应用程序属性.....	253	练习 8-4 读取.ini 文件.....	312
练习 1-3 创建 C#控制台程序.....	254	练习 8-5 制作记事本.....	316
练习 1-4 创建 C#窗体程序.....	258	第 9 单元.....	322
第 2 单元.....	262	练习 9-1 数据源连接 Access.....	322
练习 2-1 创建汽车实例变量.....	262	练习 9-2 显示管理员表数据.....	324
练习 2-2 计算器.....	263	练习 9-3 对 DataView 筛选和 排序.....	325
练习 2-3 制作九九乘法表.....	268	练习 9-4 通过 DataTable 对列 过滤和排序.....	328
练习 2-4 创建管理员结构.....	269	第 10 单元.....	329
练习 2-5 枚举管理员权限.....	270	练习 10-1 绘制汽车标志.....	329
第 3 单元.....	272	练习 10-2 绘制图像.....	330
练习 3-1 冒泡排序.....	272	练习 10-3 对图像进行坐标 变换.....	331
练习 3-2 求平均值.....	273	练习 10-4 翻转、缩放和移 动图形.....	332
练习 3-3 模拟资源管理器.....	275	练习 10-5 绘制条形图.....	335
第 4 单元.....	279	练习 10-6 绘制饼形图.....	337
练习 4-1 创建员工类和 管理员类.....	279		
练习 4-2 模拟员工的操作.....	282		
练习 4-3 序列化输出.....	284		
练习 4-4 树与节点类.....	286		
第 5 单元.....	288		
练习 5-1 转换字符串大小写.....	288		
练习 5-2 过滤字符.....	290		

上篇 基础知识.....1

第 1 章 C#基础知识.....1

- 1.1 .NET Framework 概述.....1
- 1.2 公共语言运行时.....4
 - 1.2.1 公共类型系统.....4
 - 1.2.2 公共语言规范.....5
 - 1.2.3 中间语言.....5
 - 1.2.4 执行管理.....6
 - 1.2.5 垃圾回收机制.....7
- 1.3 .NET Framework 类库.....8
- 1.4 命名空间.....9
 - 1.4.1 命名空间的组织方式.....9
 - 1.4.2 定义命名空间.....11
 - 1.4.3 使用.NET Framework
类库.....13
- 1.5 程序集.....14
- 1.6 配置 C#环境.....15
 - 1.6.1 安装 Visual Studio 2005.....15
 - 1.6.2 熟悉 Visual Studio 2005
开发环境.....17

第 2 章 C#编程基础.....19

- 2.1 变量和数据类型.....19
 - 2.1.1 使用变量和数据类型.....19
 - 2.1.2 声明和初始化变量.....21
 - 2.1.3 改变数据类型.....22
 - 2.1.4 使用引用变量.....25
- 2.2 运算符与表达式.....29
 - 2.2.1 运算符.....29
 - 2.2.2 表达式.....30
- 2.3 控制语句.....31
 - 2.3.1 条件语句.....31

2.3.2 循环语句.....34

2.3.3 跳转语句.....36

2.4 结构和枚举.....37

2.4.1 结构.....38

2.4.2 枚举.....38

2.5 扩展练习.....39

第 3 章 使用集合.....40

3.1 引用集合项.....40

3.2 System.Collections 命名空间.....41

3.3 使用集合.....43

3.3.1 使用堆栈.....43

3.3.2 使用队列.....44

3.3.3 使用散列表.....45

3.3.4 使用排序列表.....47

3.3.5 使用位数组.....48

3.3.6 使用数组列表.....49

3.4 创建对象集合.....50

3.4.1 管理员对象集合.....50

3.4.2 声明集合.....51

3.4.3 向集合中添加对象.....53

3.4.4 从集合中删除元素.....54

3.4.5 在集合中检索元素.....55

3.5 扩展练习.....56

第 4 章 面向对象.....58

4.1 面向对象概述.....58

4.2 类的结构.....60

4.2.1 定义类.....60

4.2.2 定义成员方法.....61

4.2.3 方法的返回值.....63

4.2.4 成员方法重载.....64

4.2.5 构造方法.....66

4.3	创建对象	67
4.3.1	对象的创建	67
4.3.2	对象的使用	68
4.3.3	对象的生存周期	69
4.3.4	对象的清理	70
4.4	类成员变量	72
4.4.1	访问修饰符	72
4.4.2	static 静态成员	73
4.4.3	sealed 关键字	74
4.5	继承	74
4.5.1	继承的意义	74
4.5.2	如何定义派生类	75
4.5.3	覆盖基类成员方法	76
4.5.4	this 关键字的使用	78
4.5.5	base 关键字的使用	78
4.6	抽象类与接口	79
4.6.1	抽象类	79
4.6.2	接口	80
4.7	扩展练习	82

第5章 字符串 83

5.1	字符串基础	83
5.2	String 类	84
5.2.1	String 类的构造函数	84
5.2.2	String 类的属性和索引器	86
5.2.3	比较字符串	87
5.2.4	定位与查找字符串	88
5.2.5	提取字符串	89
5.2.6	拆分字符串	90
5.2.7	String 类中的其他 常用方法	91
5.3	StringBuilder 类	94
5.3.1	追加字符串	94
5.3.2	使用 AppendFormat	97
5.3.3	插入字符串	98
5.3.4	替换字符串	100
5.3.5	StringBuilder 类的其他 常用成员	100
5.4	格式化	102

5.5	扩展练习	104
-----	------	-----

第6章 结构化的异常处理 106

6.1	结构化异常处理基本知识	106
6.1.1	抛出和捕获异常	106
6.1.2	嵌套 try 语句	108
6.2	异常类	110
6.2.1	基于类型的筛选异常	110
6.2.2	System.Exception 类的 成员	111
6.2.3	内部异常	112
6.2.4	抛出预定义异常的实例	113
6.3	用户自定义的异常类	115
6.4	finally 块	116
6.5	扩展练习	118

第7章 Windows 窗体控件 119

7.1	Windows 窗体控件概述	119
7.2	基本控件	122
7.3	显示信息的控件	124
7.4	图形和图像类控件	126
7.4.1	ImageList 控件	126
7.4.2	PictureBox 控件	127
7.5	按钮类控件	127
7.5.1	Button 控件	127
7.5.2	RadioButton 控件	128
7.5.3	CheckBox 控件	129
7.5.4	ToolBar 控件	129
7.5.5	NotifyIcon 控件	131
7.6	列表类控件	132
7.6.1	ListBox 控件	132
7.6.2	ComboBox 控件	133
7.6.3	ListView 控件	134
7.7	容器类控件	136
7.8	扩展练习	140

第8章 文件存取 I/O 141

8.1	System.IO 简介	141
8.2	路径、目录和文	142
8.2.1	Directory 类	143

8.2.2	File 类	144
8.2.3	Path 类	147
8.2.4	DirectoryInfo 类和 FileInfo 类	148
8.3	目录和文件对话框	150
8.3.1	OpenFileDialog 类	150
8.3.2	SaveFileDialog 类	152
8.4	流和顺序存取方法	153
8.5	二进制文件的读写	154
8.5.1	FileStream 类	154
8.5.2	BinaryWriter 类和 BinaryReader 类	156
8.5.3	二进制文件读写的示例	157
8.6	扩展练习	159
第 9 章	ADO.NET 数据库编程	161
9.1	引言	161
9.2	ADO.NET 基础	163
9.3	定义一个数据库连接	165
9.3.1	定义一个连接字符串	165
9.3.2	存储连接字符串	168
9.3.3	读取连接字符串	169
9.3.4	测试连接	170
9.4	操作数据库	172
9.4.1	使用 DataAdapter 填充 DataSet 对象	172
9.4.2	使用 DataReader 类	173
9.4.3	使用 DataTable 和 DataView 类	175
9.4.4	定义数据库关系	177
9.4.5	使用 Command 对象	178
9.4.6	使用 Command Builder 生成 SQL 语句	179
9.5	DataSet 应用	181
9.5.1	更新 DataSet	181
9.5.2	向 DataSet 添加数据	183
9.5.3	对 DataSet 排序和筛选	185
9.6	扩展练习	187

第 10 章	GDI+简介	188
10.1	图形绘制概述	188
10.1.1	了解 GDI+	188
10.1.2	Graphics 类	190
10.1.3	GDI+的坐标系统	191
10.1.4	颜色	192
10.2	画笔与绘制基本图形	193
10.3	画刷与区域填充	196
10.3.1	使用 SolidBrush 填充 图形	196
10.3.2	使用 HatchBrush 填充图形	198
10.3.3	使用 TextureBrush 填充图形	199
10.3.4	使用 LinearGradientBrush 填充图形	200
10.4	字体与绘制文本	202
10.5	绘制复杂的形状	204
10.6	打印图形	205
10.6.1	PrintDocument 类	205
10.6.2	PrintDialog 控件	207
10.6.3	PrintPreview 控件	208
10.7	扩展练习	210
第 11 章	工资管理系统	212
11.1	系统分析	212
11.1.1	需求分析	212
11.1.2	功能分析	213
11.2	系统概述	214
11.2.1	系统总体结构	214
11.2.2	系统用例图	214
11.3	系统模块设计	218
11.3.1	登录管理	218
11.3.2	角色信息管理	219
11.3.3	员工信息管理	220
11.3.4	工资类别管理	221
11.3.5	工资支出管理	221
11.3.6	信息浏览与查看	222
11.4	数据库设计	222

目录

11.5 登录模块.....	224	练习 5-3 将当前日期与时间 格式化为标准格式	291
11.5.1 用户登录.....	224	练习 5-4 获取指定字符在 字符串中的位置	293
11.5.2 密码修改.....	232	第 6 单元.....	295
11.6 角色信息管理模块.....	235	练习 6-1 处理除数为 0 的异常	295
11.7 员工信息管理模块.....	239	练习 6-2 处理空字符转换数字 异常	296
11.7.1 员工信息添加.....	239	练习 6-3 处理溢出异常	298
11.7.2 管理员工信息.....	242	第 7 单元.....	300
11.8 工资支出模块.....	245	练习 7-1 制作管理员登录窗体	300
11.8.1 添加支出.....	245	练习 7-2 制作添加员工窗体	302
11.8.2 支出查询.....	245	练习 7-3 制作支出类别窗体	303
下篇 实验指导.....	249	练习 7-4 制作工资支出窗体	305
第 1 单元.....	249	第 8 单元.....	306
练习 1-1 自定义 Visual Studio 2005 环境	249	练习 8-1 列表显示文件	306
练习 1-2 设置应用程序属性	253	练习 8-2 列表显示驱动器信息	308
练习 1-3 创建 C#控制台程序	254	练习 8-3 模拟【我的电脑】	311
练习 1-4 创建 C#窗体程序	258	练习 8-4 读取.ini 文件	312
第 2 单元.....	262	练习 8-5 制作记事本	316
练习 2-1 创建汽车实例变量	262	第 9 单元.....	322
练习 2-2 计算器	263	练习 9-1 数据源连接 Access	322
练习 2-3 制作九九乘法表	268	练习 9-2 显示管理员表数据	324
练习 2-4 创建管理员结构	269	练习 9-3 对 DataView 筛选和 排序	325
练习 2-5 枚举管理员权限	270	练习 9-4 通过 DataTable 对列 过滤和排序	328
第 3 单元.....	272	第 10 单元.....	329
练习 3-1 冒泡排序	272	练习 10-1 绘制汽车标志	329
练习 3-2 求平均值	273	练习 10-2 绘制图像	330
练习 3-3 模拟资源管理器	275	练习 10-3 对图像进行坐标 变换	331
第 4 单元.....	279	练习 10-4 翻转、缩放和移 动图形	332
练习 4-1 创建员工类和 管理员类	279	练习 10-5 绘制条形图	335
练习 4-2 模拟员工的操作	282	练习 10-6 绘制饼形图	337
练习 4-3 序列化输出	284		
练习 4-4 树与节点类	286		
第 5 单元.....	288		
练习 5-1 转换字符串大小写	288		
练习 5-2 过滤字符	290		

上篇 基础知识

第1章 C#基础知识



内容摘要 | Abstract

Microsoft Visual C#是 Microsoft 公司开发的一种使用简单、功能强大、面向对象、表达力丰富的语言。它不但结合了 C++强大灵活和 Java 语言简洁等特性,而且还吸取了 Delphi 和 Visual Basic 所具有的易用性。C#在 Microsoft .NET Framework 构架中扮演着重要角色。说它是 Microsoft 公司面向下一代互联网软件和服务战略的重要内容。本章将向大家介绍有关 C#的基础知识,如什么是 .NET Framework 及其类库、公共语言运行时、程序集、命名空间等。最后简单介绍如何配置 C#开发环境。



学习目标 | Objective

- 了解 .NET Framework
- 理解公共语言运行时
- 掌握理解 .NET Framework 类库
- 理解什么是程序集
- 安装 Visual Studio 2005
- 熟悉 Visual Studio 2005 的开发环境
- 自定义 Visual Studio 2005 的开发环境
- 理解应用程序属性的设置
- 理解如何创建 C#控制台应用程序
- 理解如何创建 C#窗体应用程序

1.1 .NET Framework 概述

Visual Studio.NET 应用程序开发环境的核心就是 .NET Framework。它定义了语言之间相互操作的规则,以及如何把应用程序编辑为可执行代码,同时还负责管理任何 Visual Studio.NET 语言创建的应用程序的执行。 .NET Framework 还是支持生成、运行下一代应用程序和 XML Web 服务的内部 Windows 组件。

如果希望开发和运行 .NET 应用程序,就必须安装 .NET Framework。 .NET Framework 包含把 .NET 应用程序转换为可执行文件所需要的所有编译器。开发者可以在文本编译器

中创建应用程序，而不必使用 Visual Studio.NET。

Visual Studio.NET 包含了创建窗体和管理 XML 文档的可视化设计器，以及管理大型多文件工程，并把这些工程编译为可执行文件的工具。Visual Studio.NET 还包含部署应用程序以及把 .NET Framework 部署到其他计算机的工具。尽管不使用 Visual Studio.NET 也可以开发出功能强大的应用程序，但是使用 Visual Studio.NET 无疑会更高效。.NET Framework 和 Visual Studio.NET 之间的关系如图 1-1 所示。

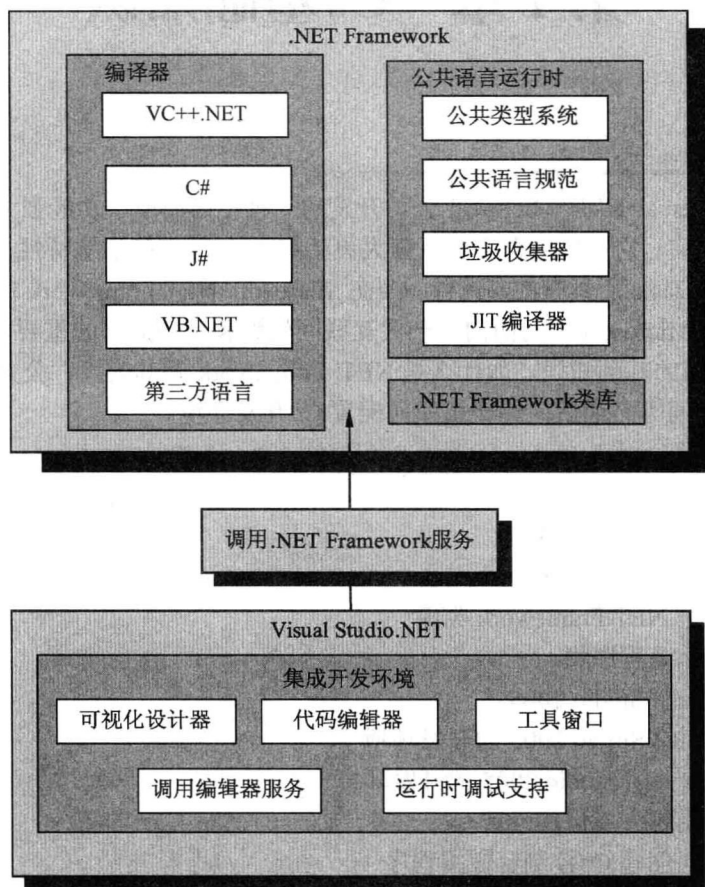


图 1-1 .NET Framework 和 Visual Studio.NET 关系图

从图 1-1 中可以看出，Visual Studio.NET 依赖于 .NET Framework 提供的服务，这些服务包括 Microsoft 公司或者第三方提供的语言编译器，这些语言编译器是 .NET Framework 自身的组成部分，而不属于 Visual Studio.NET。Visual Studio.NET 提供了大量的工具来调用某一种安装的编译器。

用户在执行由任何 .NET Framework 语言开发的应用程序时，必须安装 .NET Framework。 .NET Framework 在安装 Visual Studio.NET 时自动安装。可以从 Microsoft 公司的站点下载免费的 .NET Framework。最新版本的 Windows，包括 Windows XP Service Pack2，已经安装好了 .NET Framework。 .NET Framework 能够实现下列目标：

- 提供一致的面向对象的编程环境，而无论对象代码在本地存储和执行，还是在

本地执行但在 Internet 上分布，或者在远程执行。

- ❑ 提供将软件部署和版本控制冲突最小化的代码执行环境。
- ❑ 提供可提高代码（包括由未知的或不完全受任的第三方创建的代码）执行安全性的代码执行环境。
- ❑ 提供可消除脚本环境或解释环境性能问题的代码执行环境。
- ❑ 使开发人员的经验在面对类型大不相同的应用程序（如基于 Windows 的应用程序和基于 Web 的应用程序）时保持一致。
- ❑ 按照工业标准生成所有通信，以确保基于 .NET Framework 的代码可与任何其他代码集成。

.NET 开发平台是为简化在第三代 Internet 分布式环境下的应用程序开发，基于开发互联网标准和协议之上，实现异构语言和平台高度交互性，而构建的新一代的通信平台。

.NET 开发平台使得开发者创建运行在 IIS（Internet Information Server）Web 服务器上的 Web 应用程序更为容易，也使创建稳定、可靠而又安全的 Windows 桌面应用程序更加容易。.NET 开发平台如图 1-2 所示。

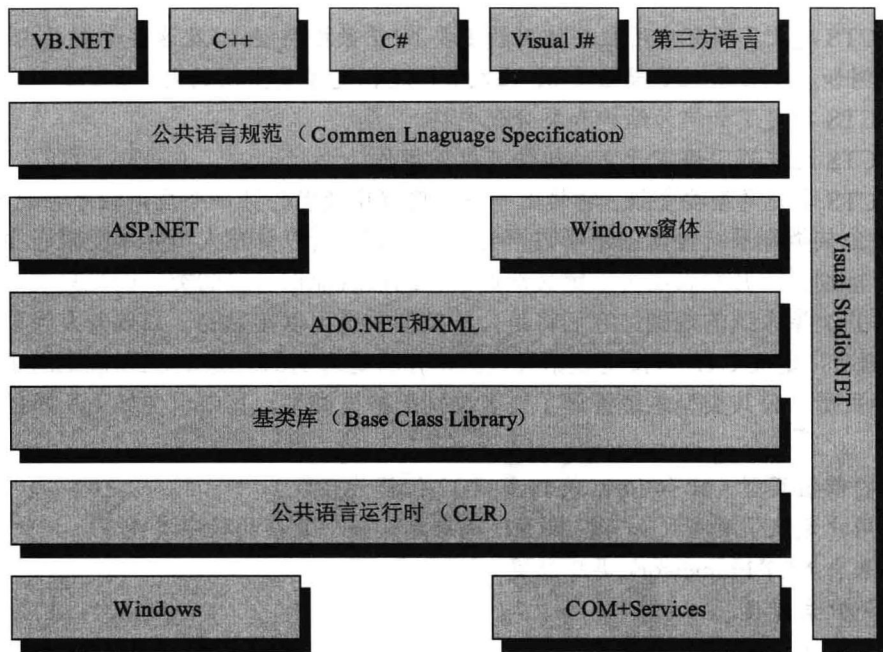


图 1-2 .NET 开发平台

.NET 开发平台包括 .NET Framework 和 .NET 开发者工具等组成部分。.NET Framework 是整个开发平台的基础，包括两个主要组件：公共语言运行时（Common Language Runtime, CLR）和 .NET Framework 类库（FCL）。.NET 开发者工具包括 Visual Studio.NET 集成开发环境和 .NET 编程语言。其中 Visual Studio.NET 集成开发环境用来开发、测试和部署应用程序。.NET 编程语言包括 VB.NET、Visual C++ 和 Visual C# 等用来创建运行在 CLR 下的应用程序。

1.2 公共语言运行时

公共语言运行时 (Common Language Runtime, CLR) 是 .NET Framework 的基础。它是执行时管理代码的代理, 它提供内存管理、线程管理和远程处理等核心服务。首先, 它分别通过公共类型系统 (Common Type System, CTS) 和公共语言规范 (Common Language Specification, CLS) 定义了标准数据类型和语言间互操作性的规则。Just-In-Time 编辑器在运行应用程序之前把中间语言 (Intermediate Language, IL) 代码转换为可执行代码。CLR 还管理应用程序, 在应用程序运行时为其分配内存和解除分配内存。

1.2.1 公共类型系统

公共类型系统 (CTS) 定义了声明和使用类型的标准, 使得 CLR 可以在不同语言开发的应用程序之间管理这些标准化的类型, 并且在不同计算机之间以标准化的格式进行数据通信。它具有以下功能:

- ❑ CTS 定义了所有应用程序使用的主要 .NET 数据类型, 以及这些类型的内部格式。例如, CTS 定义了整型是 32 位大小, 还指定了整型值的内部格式。
- ❑ CTS 指定了如何为结构和类分配内存。
- ❑ CTS 允许不同语言开发的组件可以互操作。
- ❑ CTS 实施类型安全性, 它禁止一个应用程序使用为另一个应用程序分配的内存。

由于公共类型系统包含创建新数据类型的规则, 所以开发人员并不受限于少量的数据类型。特别是, 开发人员可以定义他们自己的值或他们自己的类, 而且只要这些值或类符合公共类型系统的规则, 它们将是公共类型系统可以接受的。只要开发人员正在使用 .NET 语言, 定义 CTS 可接受数据类型的操作将是透明的。遵守 CTS 规则的事宜将由编译程序负责。公共类型系统管理了许多类别的数据类型, 其中包含如下所示的这些类别:

- ❑ 内部值类型, 比如 Byte 或 Int32 (32 位符号整数)
- ❑ 用户定义值类型 (如可以编写代码来定义一个复杂的数字类型)
- ❑ 来自 .NET Framework 类库的类
- ❑ 用户自定义类
- ❑ 代表 (指向函数的指针)
- ❑ 指针
- ❑ 枚举
- ❑ 数组
- ❑ 接口

公共类型系统不仅定义了所有的数据类型, 而且提供了面向对象的模型以及各种语言需要遵守的标准。CTS 可以分为两个大类: 值类型和引用类型, 同时这两种类型之间还可以进行强制转换, 这种转换被称为装箱 (Boxing) 和拆箱 (UnBoxing)。CTS 的每一种类型都是对象, 并继承自一个基类——System.Object。