

高等院校信息技术规划教材

C#程序开发 习题解析及实验教程

高国江 高凯 主 编
王向 李媚 张坤 副主编

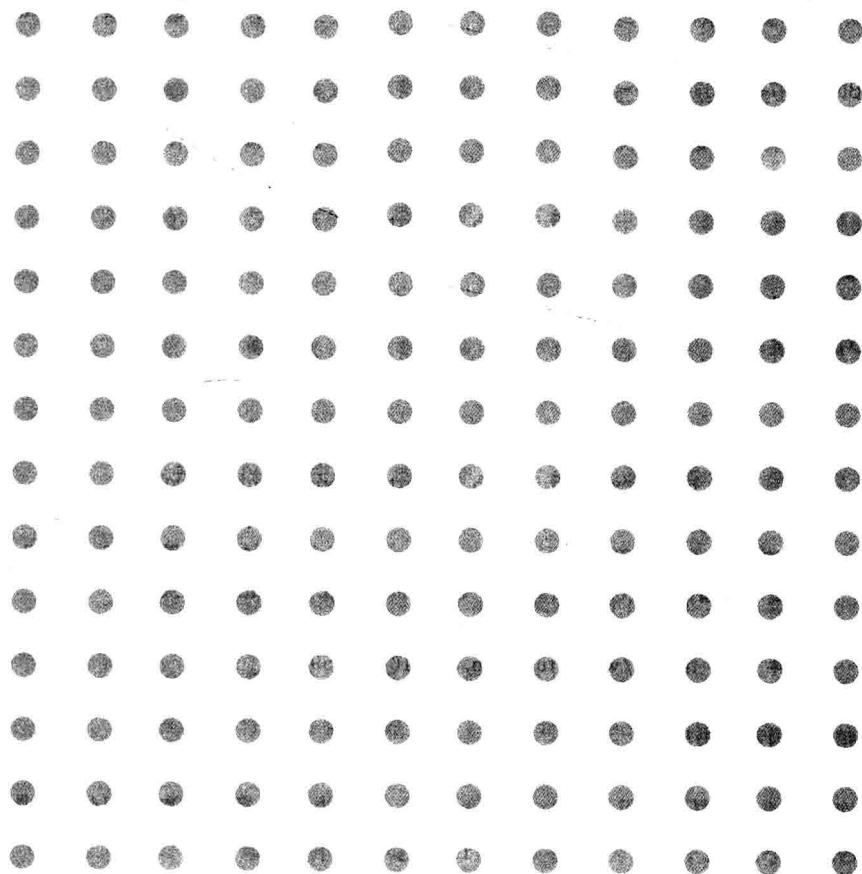


清华大学出版社

高等院校信息技术规划教材

C#程序开发 习题解析及实验教程

高国江 高凯 主 编
王向 李媚 张坤 副主编



清华大学出版社
北京

内 容 简 介

本书以基础知识讲解为先导,以习题分析和案例实践为特色,以实用性为方针,以提高工程实践水平和动手能力为目的,基于.NET Framework 4.0 和 Visual Studio C# 2010,深入浅出地讲解 C# 程序设计的方法、技巧。全书按照要点解析、习题分析以及实验指导的方式讲解了有关面向对象程序设计、异常处理与代码调试、集合索引器与泛型、字符处理和正则表达式、窗体应用程序设计、流与文件操作、基于 ADO.NET 的数据库应用程序开发、多线程技术等知识。

本书读者对象包括高等学校计算机、软件、网络、通信、电子信息类的学生,以及从事软件开发与维护的工程技术人员。

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# 程序开发习题解析及实验教程/高国江,高凯主编.--北京:清华大学出版社,2013

高等院校信息技术规划教材

ISBN 978-7-302-33447-7

I. ①C… II. ①高… ②高… 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(2013)第 188489 号

责任编辑:焦 虹

封面设计:常雪影

责任校对:时翠兰

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

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

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:10.5

字 数:236 千字

版 次:2014 年 1 月第 1 版

印 次:2014 年 1 月第 1 次印刷

印 数:1~2000

定 价:22.00 元

产品编号:047669-01

前言

foreword

本书以基础知识讲解为先导,以习题分析和案例实践为特色,以实用性为方针,以提高工程实践水平和动手能力为目的,基于 .NET Framework 4.0 和 Visual Studio C# 2010,深入浅出地讲解 C# 程序设计的方法、技巧。全书按照要点解析、习题分析以及实验指导的方式讲解了有关面向对象程序设计、异常处理与代码调试、集合索引器与泛型、字符处理和正则表达式、窗体应用程序设计、流与文件操作、基于 ADO.NET 的数据库应用程序的开发、多线程技术等知识。全书分为内容概述与习题和实验指导两部分。

第 1 部分的内容安排如下:第 1 章概述了 Microsoft .NET 和 IDE 使用,分析了解决方案的组织方式,介绍了不同类型 C# 程序的建立与执行。第 2 章简述了 C# 程序设计语言基础,并对程序的基本构成、命名空间、数据类型、流程控制、类型转换、函数等进行了说明。第 3 章介绍了面向对象程序设计基础,包括类及其构成、构造函数和析构函数、类的数据成员、类的方法成员、方法和运算符的重载、类的属性成员等,着重讲解了面向对象的三大特征。第 4 章是异常处理与代码调试部分。第 5 章对集合、索引器与泛型进行了介绍,并对 .NET 中常见的集合类、索引器、泛型类和泛型方法等的概念及其操作方法进行了说明。第 6 章是在信息处理中能发挥重要作用的有关字符处理和正则表达式的内容,主要对静态串 string 及动态串 StringBuilder 相关方法的使用,给出了正则表达式的相关内容和应用实例。第 7 章中对常见窗体相关控件的使用进行了说明,并给出了多窗体创建及窗体间数据传递的方法。第 8 章讲述 C# 中对流和文件进行操作的方法,并简介 Directory 和 DirectoryInfo 类、File 和 FileInfo 类、StreamReader 和 StreamWriter 类、FileStream 类、Path 类等的使用。第 9 章是基于 ADO.NET 的数据库应用程序的开发部分,并以 SQL Server 数据源为例,介绍基于 ADO.NET 的数据库应用程序开发中常用 ADO.NET 类的使用以及和窗体中控件的绑定。第 10 章介绍有关多线程编程方法,涉及创建线程、控制线程和线程同步。



第2部分分成10个实验,内容安排如下:实验1是 Visual Studio 2010 编程环境,通过创建项目了解 IDE 的使用。实验2是 C# 语言编程基础,对 C# 的基本语法进行练习。实验3是 C# 面向对象编程基础,组织了几个小程序对面向对象的三大特征进行验证。实验4是异常处理的应用。实验5是 Windows 窗体程序,对文档程序和菜单进行实战操作。实验6是控件应用,演示了常用控件的使用方法。实验7是简易记事本程序,着重文件和对话框的应用。实验8是 ADO.NET 编程(1),着重实现数据库应用程序中查询的操作。实验9是 ADO.NET 编程(2),着重实现数据库应用程序中更新的操作。实验10是多线程下载,实现了一个多线程下载程序。

本书注重对读者实践能力的培养,力求为读者进一步的程序设计与软件开发奠定坚实的基础。通过知识点概述,可使读者掌握实际的 C# 编程基础;通过习题及分析,可提高对知识的运用;通过案例实验,可掌握分析问题和使用 C# 解决问题的手段。无论是刚接触 .NET Framework 和 C# 的初学者,还是有一定基础的程序员,本书都是适合的。本书既可帮助初学者快速入门,也可对有一定基础的工程技术人员提供参考。全书理论联系实际,知识体系完整,讲述力求清晰,示例尽量丰富,程序有代表性,案例有针对性和实用性。本书可以单独使用,亦可作为清华大学出版社出版的《C# 程序开发与应用教程》的配套教学用书。

本书提供源代码下载,相关源代码均在课程网站上发布。各章相应工程的完整代码可以在相应章节中的对应工程中找到,方便读者自学和实践。所有练习均在 Visual Studio 2010 环境下通过调试,保证了程序的可用性。

全书由高国江、高凯主编,王向、李媚、张坤副主编。高凯提出写作大纲,第1~10章由高国江编写,实验1~实验4由王向编写,实验5~实验8由李媚编写,实验9和实验10由张坤编写,最后由高凯完成全书的统稿与审校工作。在本书的写作与相关科研课题的研究工作中,作者得到了多方面的支持与帮助。有关 C# 开发的相关资料和网站,如 MSDN、CSDN 等,亦为本书提供了良好的基础。我们还参考了相关文献和互联网上众多热心网友提供的素材,本书的顺利完成也得益于大量的相关工作及研究成果。在此谨向这些文献的作者、热心网友以及为本书提供帮助的老师致以诚挚的谢意和崇高的敬意。

由于我们的学识、水平均有限,书中不妥之处恳请广大读者批评指正。

作者

目录

Contents

第1部分 内容概述与习题

第1章	Microsoft.NET 概述	3
1.1	内容概述	3
1.2	习题	4
1.3	答案及解析	5
第2章	C# 程序设计语言基础	7
2.1	内容概述	7
2.2	习题	12
2.3	答案及解析	13
第3章	面向对象程序设计基础	22
3.1	内容概述	22
3.2	习题	26
3.3	答案及解析	27
第4章	异常处理	36
4.1	内容概述	36
4.2	习题	38
4.3	答案及解析	39
第5章	集合、索引器与泛型	43
5.1	内容概述	43
5.2	习题	44
5.3	答案及解析	45



第 6 章 字符处理和正则表达式	50
6.1 内容概述	50
6.2 习题	52
6.3 答案及解析	52
第 7 章 窗体应用程序设计	54
7.1 内容概述	54
7.2 习题	59
7.3 答案及解析	60
第 8 章 流与文件	66
8.1 内容概述	66
8.2 习题	69
8.3 答案及解析	69
第 9 章 基于 ADO.NET 的数据库	74
9.1 内容概述	74
9.2 习题	80
9.3 答案及解析	80
第 10 章 多线程技术	89
10.1 内容概述	89
10.2 习题	92
10.3 答案及解析	92

第 2 部分 实验指导

实验 1 Visual Studio 2010 编程环境	99
实验 2 C# 语言编程基础	105
实验 3 C# 面向对象编程基础	110
实验 4 异常处理的应用	116
实验 5 Windows 窗体程序	120

实验 6 控件应用	126
实验 7 简易记事本程序	132
实验 8 ADO.NET 编程(1)	136
实验 9 ADO.NET 编程(2)	145
实验 10 多线程下载	152
参考文献	157

第 1 部分 内容概述与习题

Microsoft.NET 概述

本章是 Microsoft.NET 概述。本章重点内容如下：

- .NET Framework 4.0 的组成；
- Visual Studio 版本；
- Visual Studio 集成化开发环境。

1.1 内容概述

1. .NET Framework 4.0 的组成

(1) 公共语言运行时 Common Language Runtime(CLR)环境。它提供了一个可靠而完善的多语言运行环境,简化了应用程序的开发、部署和管理,实现了在多语言环境下的跨平台工作。

(2) .NET 的基础类库(Basic Class Library, BCL)或称统一编程类。它是.NET 框架使用的一个大的类库,提供了几乎所有应用程序都需要的公共代码,解决了由于开发语言的差异而造成的代码共享和迁移问题。

(3) ADO.NET、ASP.NET 以及 XML Web Services 等构件。

2. Visual Studio 2010

Visual Studio 2010 的开发环境分为以下 4 个版本。

Professional 版本适用于个人执行基本的开发任务,可简化在各种平台(包括 SharePoint 和云)上创建、调试和开发应用程序的过程,自带对测试驱动开发的集成支持以及调试工具,以确保提供高质量的解决方案。

Premium 版本适用于个人和团队开发可扩展的高质量应用程序,无论编写代码、构建数据库、测试还是调试,它都提供强大的支持。

Ultimate 版本提供了一个综合性的应用程序生命周期管理工具套件,无论是创建新的解决方案,还是改进现有的应用程序,它可确保从设计到部署的整个过程都能取得高质量的结果。

Test Professional 是质量保障团队的专用工具集,可简化测试规划和手动测试执行过程。Test Professional 与开发人员的 Visual Studio 软件配合运行,可在整个应用程序开发生命周期内实现开发人员和测试人员之间的高效协作。

3. Visual Studio 2010 IDE

VS 2010 开发环境包括项目与解决方案资源管理器、代码编辑器和设计器、代码生成与调试、代码部署。

IDE 的视图包括：

- 项目与解决方案资源管理器：提供项目及其文件的有组织的视图，并且提供对项目 and 文件相关命令的便捷访问。
- 属性窗口：用户查看和更改位于编辑器或设计器中选定对象的设计时属性及事件，也可以使用“属性”窗口编辑和查看文件、项目和解决方案的属性。
- 工具箱：实现对 IDE 控件的组织管理，可以通过拖曳或双击控件的方式将控件添加到窗体上。
- 类视图窗口：以树型视图的方式显示当前项目中包含的所有类、方法和特性等信息。
- 代码定义窗口：实现对程序中代码编辑的窗口。
- 体系结构资源管理器：使用体系结构资源管理器来选择要进行可视化的代码的垂直切面。
- 服务器资源管理器：在服务器资源管理器中可以添加和管理对应的服务器，可以在 IDE 中集中监控和管理一个应用程序方案所涉及的所有服务器和数据连接。

4. 应用程序创建

通过依次选择菜单中的【文件】、【新建】、【项目】可以打开“新建项目”对话框，在对话框中可以选择不同的模板创建不同的新项目。在默认情况下，解决方案中只有一个项目。一般来说，每个项目会包含 Properties、该项目对应命名空间的引用（命名空间是一种把相关的类型声明分组并命名的方法）、类文件列表等三部分。

常用程序包括控制台应用程序、Windows 窗体应用程序、ASP.NET 应用程序等。

1.2 习 题

1. Microsoft .NET 由哪些主要部分所构成？
2. 了解你目前使用的 .NET 版本。
3. C# 的主要特点有哪些？
4. 什么是解决方案？
5. 什么是项目？
6. 简述项目与解决方案的关系。
7. .NET Framework 是一种（ ）。

A. 编程语言 B. 运行平台 C. 操作系统 D. 数据库管理系统

1.3 答案及解析

1. 试题答案及解析

Microsoft .NET 主要由以下几部分构成:

公共语言运行时 Common Language Runtime(CLR)环境:它提供了一个可靠而完善的多语言运行环境,简化了应用程序的开发、部署和管理,实现了组建能在多语言环境下的跨平台工作。

.NET 的基础类库(Basic Class Library, BCL)或称统一编程类:它是 .NET 框架使用的一个大的类库,提供了几乎所有应用程序都需要的公共代码,解决了由于诸如 C++ 和 Java 等开发语言的差异而造成了代码共享和迁移问题。

ADO.NET、ASP.NET 以及 XML Web Services 等构件。

2. 试题答案及解析

目前常见的 .NET 版本有 .NET Framework 1.1、.NET Framework 2.0、.NET Framework 3.0、.NET Framework 3.5、.NET Framework 4.0。不同的 .NET 版本随着不同的 Visual Studio 版本发布。目前使用的 Visual Studio 2010 使用的 .NET Framework 版本是 4.0 版,而 Visual Studio 2008 使用 .NET Framework 3.5。

3. 试题答案及解析

C# 的主要特点有:

(1) C# 的代码必须要放在类中。

(2) C# 简化了 C++ 在类、命名空间、方法重载和异常处理等方面的操作,借鉴了 Java 的许多特性。

(3) 基本不需指针,相应的指针功能用 delegates 来模拟。

(4) 支持接口编程、泛型、面向对象。

(5) 去掉了全局函数、全局变量和全局常量,并将它们都封装在一个类中(或者作为一个实例成员),即全局的概念退化成类的静态成员(需要通过类型访问),这就减少了命名冲突的可能性。

(6) 采用无用单元收集程序自动管理程序内存。代码工作在受托管的环境中,垃圾收集器能自动管理内存,不需程序员操心。

(7) C# 使用统一类型系统,无须再对 char、unsigned char、signed char 等进行区别。

(8) 许多情况下,C# 的特定语言功能取决于 .NET,或依赖于 .NET 基类,因此须使用 using 把 .NET Framework 类库相对应的命名空间引入到基于 C# 的项目中来。如在设计窗体应用程序时需要 using System. Windows. Forms;而在设计 Web 应用程序时,则需要引用 using System. Web. UI. WebControls 等。

4. 试题答案及解析

解决方案提供了一种管理 Visual Studio 配置、生成和部署相关项目集的方式,使得集成开发环境(IDE)能够应用它的各种工具、设计器、模板和设置。Visual Studio 实现了概念上的容器,可以将一个或多个项目(如 Windows 应用程序、Web 应用程序和 Web 服

务)包含在解决方案中。创建一个新的项目而没有说明将它添加到当前解决方案中时, Visual Studio. NET 会创建一个新的解决方案文件。虽然项目文件的扩展名由所包含项目的类型来决定,但解决方案的扩展名总是 sln。此外, Visual Studio 还提供了解决方案文件夹,用于将相关的项目组织成组,然后对这些项目组执行操作。

5. 试题答案及解析

Visual Studio 项目用作解决方案内的容器,它通过逻辑方式管理、生成和调试构成应用程序的各个项。项目的输出通常是可执行程序(. exe)、动态链接库(. dll)文件或模块等。

此外,所有 Visual Studio 开发产品都提供了许多预定义的项目模板。可以使用项目模板创建基本项目容器以及开发应用程序、类、控件或库等可能需要的预备项。例如,如果选择创建 Windows 应用程序,则项目会提供可自定义的 Windows 窗体项;同样,如果选择创建一个 Web 应用程序,则项目将提供一个 Web 窗体项。

6. 试题答案及解析

在 Visual Studio 中项目和解决方案都是一种容器,为所解决的问题提供了一种逻辑管理的方式。不同之处在于解决方案提供了配置生成和部署项目集的方式,而项目提供了管理生成和调试构成各个应用程序项的管理。例如:对于一个需解决的问题创建一个解决方案,整个方案可能有人机交互部分、业务逻辑部分、数据管理部分,对于每个部分可以创建一个项目进行管理。

7. 试题答案及解析

试题答案: B

.NET Framework 是提供了一种运行平台,在这个平台上可以支持多种编程语言,例如 C#、C++ 等。

第2章

chapter 2

C# 程序设计语言基础

本章简介 C# 语言基础知识,包括程序的基本构成,常量、变量、表达式和运算符,值类型和引用类型,类型转换,流程控制语句,函数和方法等内容。本章重点内容如下:

- 了解 C# 程序的基本构成,掌握命名空间的使用方法;
- 了解数据类型、表达式、运算符、常量与变量等知识;
- 掌握 C# 流程控制语句;
- 掌握常见的数据类型转换方法;
- 掌握函数的使用。

2.1 内容概述

1. 命名空间

C# 使用命名空间来组织和管理代码,类似于 Java 的 jar 包。命名空间的使用分为两种。NET Framework 使用命名空间来管理它众多的类,开发人员可以通过声明命名空间可以控制类和方法的范围。

自定义的命名空间可以采用创建类库项目的方式来实现,类库编译后生成 Dll 文件。

命名空间的使用分为:用 using 语句在程序开头包括行,在使用时用完整的命名空间。在使用自定义类库需要将创建的 DLL 项目添加到项目的引用中。如果在一个解决方案中,则可以通过添加引用项目的方式添加引用。

2. 数据类型

常用的数据类型见图 1.2.1。

3. 标识符

标识符是适用于变量、常量、类、方法和其他各种用户定义对象命名的一般术语。命名标识符时应当遵守以下规则:

- (1) 标识符不能以数字开头,也不能包含空格。
- (2) 标识符可以包含大小写字母、数字、下划线和@字符。
- (3) 标识符必须区分大小写。大写字母和小写字母被认为是不同的字母。
- (4) @字符只能是标识符的第一个字符。带@前缀的标识符称为逐字标识符。
- (5) 不能使用 C# 中的关键字。虽然@字符加关键字可以成为合法的标识符,但建

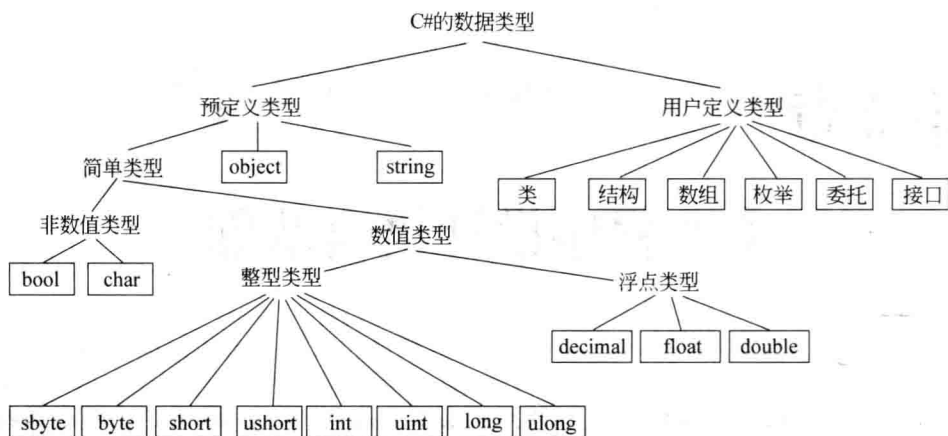


图 1.2.1 常用数据类型

议不要这样做。

(6) 不能与 C# 的类库名称相同。

 需要注意的是标识符的命名尽量能反映出该标识符所代表内容的含义,例如 `int sum`; 声明一个求和的变量。

4. 常量、变量

- 常量: 在程序的生命周期中保持不变的量。在 C# 中常量的声明通过关键字 `const` 来声明。通常情况下常量名用大写。
- 变量: 在程序的生命周期中可以改变的量。如果在一个语句中声明和初始化了多个变量,那么所有的变量都具有相同的数据类型。同时 C# 作为强类型语言,要求变量先声明后使用。

在命名变量和常量的过程中,常量名或变量名的命名应遵循标识符的命名规则。

5. 运算符

运算符包括算数运算符、赋值运算符、关系运算符、逻辑运算符和三元运算符。

- 算数运算符: `+`、`-`、`*`、`/`、`++`、`--`
- 赋值运算符: `=`
- 关系运算符: `==`、`!=`、`>`、`<`、`>=`、`<=`
- 逻辑运算符: `&`、`&&`、`||`、`!`
- 三元运算符: `?:`

6. 表达式

表达式是变量或常量通过运算符连接起来的组合。

7. 数据存储类型

数据存储类型按照在内存中存储方式的不同,分为值类型和引用类型。值类型直接存储其值,引用类型存储对其值的引用。

8. 数组

在 C# 中把拥有相同名字、通过不同下标来访问的一组变量称为数组。

数组的声明格式如下所示。

- 一维数组：

类型[] 数组名；

- 多维数组：

类型[,] 数组名；

- 数组的数组(交错的)：

类型[][] 数组名；

在 C# 中,数组是对象,声明数组并不实际创建它们,必须要进行实例化。下面的示例说明如何创建数组：

- 一维数组：

```
int[] numbers=new int[5];
```

- 多维数组：

```
string[,] names=new string[5,4] ;
```

- 数组的数组(交错的)：

```
byte[][] scores=new byte[5][] ;
for (int x=0; x<scores.Length; x++)
{
    scores[x]=new byte[4] ;
}
```

 在 C# 语言中,下标是从 0 开始。对于上面定义的一维数组: int[] numbers=new int[5]; 正确的访问是 numbers[0]—numbers[4]。

9. 流程控制

程序的流程控制包括顺序、分支、循环三种结构。在顺序结构中,程序按顺序编写;分支结构包括 if 语句、switch…case 语句;循环语句包括 for 语句、while 语句、do…while 语句和 foreach 语句。

(1) if 语句

语法格式：

```
if (布尔表达式)
    if 子句
else
    else 子句
```

说明：根据布尔表达式的值选择要执行的语句。如果表达式值为真,执行 if 子句;否则,执行 else 子句。