

WANG LUO SHU JU KU XI TONG KAI FA
JI SHU YING YONG

网络数据库系统开发 技术应用

芦丽萍 编著

南开大学出版社

网络数据库系统开发技术应用

芦丽萍 编著

南开大学出版社

天 津

图书在版编目(CIP)数据

网络数据库系统开发技术应用 / 芦丽萍编著. —天津:
南开大学出版社, 2016.3

ISBN 978-7-310-05070-3

I. ①网… II. ①芦… III. ①关系数据库系统—系统
开发 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2016)第 033170 号

版权所有 侵权必究

南开大学出版社出版发行

出版人:孙克强

地址:天津市南开区卫津路 94 号 邮政编码:300071

营销部电话:(022)23508339 23500755

营销部传真:(022)23508542 邮购部电话:(022)23502200

*

唐山新苑印务有限公司印刷

全国各地新华书店经销

*

2016 年 3 月第 1 版 2016 年 3 月第 1 次印刷

260×185 毫米 16 开本 17 印张 396 千字

定价:34.00 元

如遇图书印装质量问题,请与本社营销部联系调换,电话:(022)23507125

南开大学出版社

李 大

前 言

ASP.NET 是目前最流行的 Web 开发工具之一。它为用户提供了完整的可视化开发环境,结合 SQL Server 数据库技术,快速开发网络数据库应用程序。从 ASP.NET 1.0 版本至 4.0 版本的发布过程中,微软推出了一系列新技术,如:站点导航、数据控件、主题与外观、Ajax 等技术,在一定程度上降低了开发难度。ASP.NET+SQL Server 作为网络数据库开发技术得到了越来越广泛的应用。

在教学及教学管理领域,网络数据库开发技术的应用需求呈现不断扩大趋势。如网络教学平台、资源共享课程、无纸考试系统、智能教务系统、论文管理系统,等等,这些系统的开发毫无疑问都应归属于网络数据库开发技术的应用。作为高校教育技术专业,是培养应用型人才的专业,其专业目标是培养教育信息化的建设者、管理者、研究者,培养能够在新技术教育领域从事教学资源、教学媒体和教学系统的设计、开发、运用、管理和评价等的高级专门人才。

本书以教育技术专业本科生为对象,网络数据库开发技术应用的单一功能模块和完整系统开发为主线,实际操作为手段,技术应用为目标。其特点是内容新颖、实用性强、案例完整。从简单应用入手,逐步提升难度,使读者从网络数据库开发技术应用的案例模块和完整系统设计的应用过程中,全面掌握网络数据库开发技术的应用。

全书分为基础篇和应用篇。其中基础篇包括 4 章,应用篇包括 7 章。基础篇的第 1 章和第 2 章主要介绍 C#语言的方法应用以及必要的面向对象的程序设计思想;第 3 章和第 4 章以实例介绍了如何使用 ASP.NET 进行 Windows 应用程序设计的方法以及 Windows 高级界面与多媒体程序的设计方法。应用篇的第 5 章到第 11 章,按照单一功能模块和完整系统应用开发为顺序,分别是:单用户登录模块、注册模块、信息查询模块、留言板模块、多用户登录模块、网上论坛模块、在线考试系统。本书所提供的案例简明易懂、上手快,每章后均附有实践活动,所有程序都在 Visual Studio 和 SQL Server 环境下调试通过。

本书可作为教育技术专业网络数据库开发技术应用的教科书,也可作为具有一定 Web 应用程序开发基础的读者参考。

本书由芦丽萍编写。由于网络数据库开发技术发展迅速、应用广泛,加之作者水平有限,书中错误和不足之处在所难免,敬请读者批评指正。

编者

2015 年 11 月

目 录

基础篇

第 1 章 C#程序中的方法.....	3
1.1 方法的概念	3
1.2 方法的定义	4
1.3 方法的调用	4
1.4 方法的参数传递	5
1.5 方法应用实例	8
1.6 实践活动	12
第 2 章 面向对象的程序设计.....	15
2.1 面向对象程序设计概述.....	15
2.2 命名空间	16
2.3 类和对象的声明	17
2.4 类的构造函数和析构函数.....	19
2.5 类的方法重载	24
2.6 this 关键	25
2.7 实践活动	26
第 3 章 Windows 应用程序设计	27
3.1 C#可视化编程	27
3.2 Windows 窗体.....	28
3.3 基本控件的使用	29
3.4 实践活动.....	42
第 4 章 Windows 高级界面与多媒体程序设计	43
4.1 Windows 高级界面设计	43
4.2 多媒体控件应用	57
4.3 实践活动	66

应用篇

第5章 单用户登录模块	69
5.1 系统开发工具和运行环境	69
5.2 模块实现目标	69
5.3 数据库设计	69
5.4 模块实现过程	70
5.5 本章小结	75
5.6 实践活动	76
第6章 注册模块	78
6.1 模块实现目标	78
6.2 数据库设计	78
6.3 模块实现过程	79
6.4 本章小结	88
6.5 实践活动	90
第7章 信息查询模块	92
7.1 模块实现目标	92
7.2 数据库设计	92
7.3 模块实现过程	94
7.4 本章小结	101
7.5 实践活动	109
第8章 留言板模块	110
8.1 模块功能	110
8.2 模块体系结构	110
8.3 数据库设计	113
8.4 数据访问层	114
8.5 业务逻辑层	117
8.6 页面显示层	118
8.7 模块关键技术	126
8.8 本章小结	127
8.9 实践活动	130
第9章 多用户登录模块	131
9.1 模块功能	131

9.2 模块体系结构	131
9.3 数据库设计	133
9.4 数据访问层	134
9.5 业务逻辑层	136
9.6 页面显示层	139
9.7 本章小结	143
9.8 实践活动	145
第 10 章 网上论坛模块.....	146
10.1 模块功能	146
10.2 模块体系结构	146
10.3 数据库设计	148
10.4 数据访问层	151
10.5 业务逻辑层	155
10.6 页面显示层	166
10.7 模块关键技术	190
10.8 本章小结	192
10.9 实践活动	194
第 11 章 在线考试系统.....	195
11.1 系统功能	195
11.2 系统体系结构.....	196
11.3 数据库设计.....	198
11.4 创建类文件.....	204
11.5 登录模块	207
11.6 学生模块	214
11.7 教师模块	227
11.8 管理员模块.....	244
11.9 系统关键技术.....	262
11.10 实践活动	263

基础篇

第 1 章 C#程序中的方法

1.1 方法的概念

1. 知识回顾

C 语言程序的基本结构是函数。函数可以被其他函数调用，函数可以嵌套调用，不可嵌套定义。函数定义的语法格式：

函数类型名 函数名(参量表) //函数首部

{

函数体;

}

例如：

long fac1(int n)

{

int i;

long p;

for(p=1,i=2;i<=n;i++)

p*=i;

return p;

}

2. C#语言程序的方法

C#语言中的方法类似于 C 语言中的函数，可以作为一个功能模块出现在 C#程序中，完成某种特定的功能。

方法按来源分为用户自定义方法、系统方法；按有无参数分为无参方法和有参方法；按方法的调用关系分为主调方法和被调方法等。

3. C#语言源程序文件结构

C#语言的源程序文件是.cs 文件，其结构如下：

using System; //引用系统命名空间

{

class 类名 1

{

方法 1;

.....

```
static void Main() //Main()方法
{
    .....//在此处添加代码
}
.....
方法 n;
}
.....
class 类名 n
{
    .....
}
```

1.2 方法的定义

方法在使用之前必须先定义。方法的定义是对方法功能的描述，要执行方法需要调用它。方法一般放在类中，一个类可以包含多个方法。方法定义的语法格式如下：

```
方法修饰符 数据类型名 方法名([形参表]) //方法头
{
    方法体;
}
```

例如：

```
public class BaseClass
{
    public static SqlConnection DBCon() //建立数据库连接
    {
        SqlConnection conn= new SqlConnection();
        conn.ConnectionString="server=.;database=db_ExamOnline;user.id=sa;pwd=123
456";
        return conn;
    }
}
```

1.3 方法的调用

在 C#中，调用方法有三种格式。

【格式 1】方法名([实参表])

调用本类中定义的方法，即方法的调用和定义在同一个类中。例如：

```
display(); //直接调用本类中的方法
```

【格式2】对象名.方法名([实参表])

一般情况下是先定义一个类对象，再调用该类的方法。例如：

```
SqlConnection conn= new SqlConnection();    //定义 SqlConnection 类对象 conn
conn.ConnectionString="server=.;database=db_ExamOnline;user id=sa;pwd=123456";
//创建连接
conn.Open();    //调用 conn 对象的 Open()方法
```

【格式3】类名.方法名([实参表])

调用类的静态方法。例如：

```
SqlConnection conn = BaseClass.DBCon();    //调用类的 DBCon()方法，是 static 方法
```

1.4 方法的参数传递

参数传递是指实参把数据传给形参的方式。C#中方法的参数传递可分为值传递、引用传递、输出参数和参数数组。

1.4.1 值传递

实参值传给形参，是单向传递。形参接收了实参的值后与实参不再有任何关系。在方法中对形参的修改不会影响到对应的实参。

【例 1-1】用“值传递”传递参数。在方法中交换形参变量的值，不影响实参值。代码如下：

```
using System;
class Program
{
    static void exch(int a,int b)
    {
        int t;
        t=a;a=b;b=t;
    }
    static void Main()
    {
        int x,y;
        Console.WriteLine("请输入 x 和 y 的值: ");
        x=Convert.ToInt32(Console.ReadLine());
        y=Convert.ToInt32(Console.ReadLine());
        exch(x,y); //调用本类方法
        Console.WriteLine("x={0},y={1}",x,y);
    }
}
```

程序运行结果如图 1-1 所示。



图 1-1 【例 1-1】运行结果

1.4.2 引用传递

使用引用方式传递数据，主调方法赋予被调用方法的权利是直接访问主调方法中的数据，并且可修改主调方法中的数据，是双向传递。调用方法时，在实参前面和形参的数据类型前面加上关键字 `ref`。

优点：避免了对大型数据的复制（例如对象）。

缺点：削弱了数据的安全性。

【例 1-2】用“引用传递”传递参数。代码如下：

```
using System;
class Program
{
    static void UseRef(ref int i)
    {
        i+=100;
        Console.WriteLine("i = {0}", i);
    }
    static void Main()
    {
        int i = 10;
        Console.WriteLine("Before the method calling: i = {0}", i);
        UseRef(ref i);
        Console.WriteLine("After the method calling: i = {0}", i);
        Console.Read();
    }
}
```

程序运行结果如图 1-2 所示。

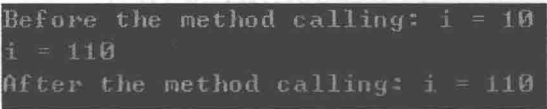


图 1-2 【例 1-2】运行结果

1.4.3 输出参数

若将引用传递中的关键字 `ref` 用 `out` 替换, 则参数传递就为输出参数。输出参数通常用来指定由被调用方法对参数进行初始化。通常在方法中接收到了一个未初始化的数值时, 编译器会产生错误, 但使用带关键字 `out` 的参数, 指定了被调用方法将对变量进行初始化, 该错误将不会发生。

【例 1-3】用“输出参数”传递参数。代码如下:

```
using System;
class Program
{
    static void UseRef(out int i)
    {
        i = 100;
        Console.WriteLine("i = {0}", i);
    }
    static void Main()
    {
        int i; //i 未赋初值, 但系统不会报错
        UseRef(out i);
        Console.WriteLine("After the method calling: i = {0}", i);
        Console.Read();
    }
}
```

程序运行结果如图 1-3 所示。



```
i = 100
After the method calling: i = 100
```

图 1-3 【例 1-3】运行结果

1.4.4 参数数组

使用关键字 `params`。若参数数目可变, 则采用参数数组方法传递参数。调用方法时, 在形参前面加上关键字 `params`。

【例 1-4】用“参数数组”传递参数。代码如下:

```
using System;
class Program
{
    static void UseParams(params object[] list)
```

```
程序运行结果如图 1-4 所示。

for (int i = 0; i < list.Length; i++)
{
    Console.WriteLine(list[i]);
}

static void Main()
{
    object[] arr = new object[3] { 100, 'a', "keywords" };
    UseParams(arr);    //3 个参数
    UseParams(200);    //1 个参数
    Console.Read();
}
}
```

程序运行结果如图 1-4 所示。



图 1-4 【例 1-4】运行结果

1.5 方法应用实例

1.5.1 Web 项目

在 Microsoft Visual Studio 中，可以通过新建项目和新建网站来创建 Web 项目。Web 项目是指在服务器上运行的项目，页面是通过运行服务器上的程序所得到的结果。常见的 Web 项目按计算机语言分，有 Java、ASP.NET、PHP 等项目。新建项目主要可以创建控制台应用程序和 Windows 应用程序。

新建 Web 网站与新建项目的区别参见本章后面的学习资料。

1.5.2 创建控制台应用程序实例

【例 1-5】创建一个项目，项目名称 ASimpleProject。在 Program.cs 中编写方法 static long fun(int n){ }，用 Main（）方法调用 fun()，输出从 n 个数据中取出 m 个数共有几种方法。

操作步骤：

(1) 创建新项目

启动 Microsoft Visual Studio，执行“文件”→“新建”→“项目”，选择项目位置 Projects，

输入项目名称 ASimpleProject。项目类型选择“Visual C#”，模板选“控制台应用程序”，如图 1-5 所示。



图 1-5 新建项目 ASimpleProject

(2) 在 Program.cs 中添加如下代码：

```
class Program
```

```
{
```

```
    static long fun(int n)
```

```
{
```

```
        int p = 1;
```

```
        for(int i=2;i<=n;i++)
```

```
            p*=i;
```

```
        return p;
```

```
}
```

```
    static void Main()
```

```
{
```

```
        string str1,str2;
```

```
        Console.WriteLine("请输入 n 的值: ");
```

```
        str1 = Console.ReadLine();
```

```
        Console.WriteLine("请输入 m 的值: ");
```

```
        str2 = Console.ReadLine();
```

```
        int n = Convert.ToInt32(str1);
```

```
        int m = Convert.ToInt32(str2);
```

```
        long p=fun(n)/fun(m)/fun(n-m);
```

```
        Console.WriteLine("p={0}", p);
```

```
        Console.ReadLine();
```

```
}
```

```
}
```

(3) 运行程序，如图 1-6 所示。

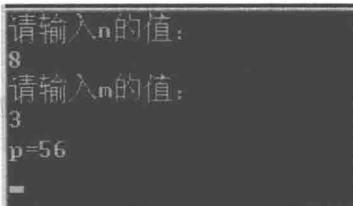


图 1-6 【例 1-5】运行结果

注释：

- 1. 如何显示代码行号？“工具”→“选项”→“文本编辑器”→C#（行号复选框）。
- 2. Response.Write 与 Console.Write 有什么区别？

Response 是 ASP.NET 内置对象，是在浏览器上输出信息；Console.Write 则只能在控制台（黑屏幕）输出。

- 3. 打开该项目文件时，选择“ASimpleProject.sln”解决方案文件，执行“打开”。
- 4. 项目文件夹中的所有文件都不要删除。

1. 5. 3 创建 Windows 应用程序实例

【例 1-6】 创建一个项目，项目名称 BSimpleProject。在 Program.cs 中编写一个给杨辉三角赋值的方法 void fun (int n){ }，用 Main()方法调用 fun()，其中参数 n 接收杨辉三角的显示行数。

操作步骤：

(1) 创建新项目

启动 Microsoft Visual Studio，执行“文件”→“新建”→“项目”，选择项目位置 Projects，输入项目名称 BSimpleProject。项目类型选择“Visual C#”，模板选“Windows 应用程序”，如图 1-7 所示。

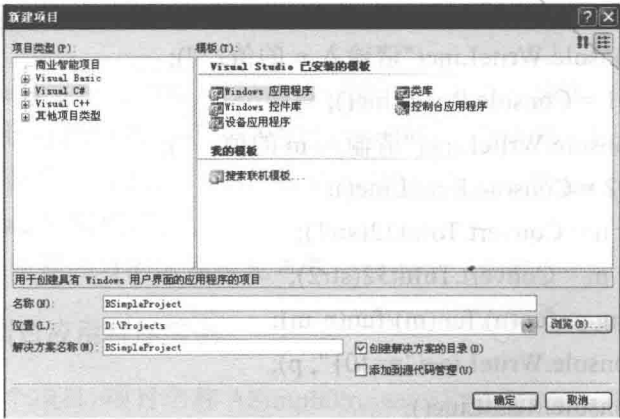


图 1-7 新建项目 BSimpleProject

(2) 设计窗体界面

选择 Form1.cs 设计视图，进行窗体布局如图 1-8 所示。