

国家示范性高职建设教材·电子商务专业



# 智慧商场

## 电商平台开发

主编 高洁 王喜胜



国家示范性高职建设教材·电子商务专业

# 智慧商场电商平台开发

主 编 高 洁 王喜胜  
副主编 王 可 李桂芹



南京大学出版社

## 图书在版编目(CIP)数据

智慧商场电商平台开发 / 高洁, 王喜胜主编. — 南京: 南京大学出版社, 2016. 1

ISBN 978-7-305-14595-7

I. ①智… II. ①高… ②王… III. ①电子商务—研究 IV. ①F713.36

中国版本图书馆 CIP 数据核字(2015)第 004468 号

出版发行 南京大学出版社

社 址 南京市汉口路 22 号 邮编 210093

出 版 人 金鑫荣

书 名 智慧商场电商平台开发

主 编 高 洁 王喜胜

责任编辑 耿士祥 王抗战 编辑热线 025-83596997

照 排 南京理工大学资产经营有限公司

印 刷 常州市武进第三印刷有限公司

开 本 787×1092 1/16 印张 13 字数 308 千

版 次 2016 年 1 月第 1 版 2016 年 1 月第 1 次印刷

ISBN 978-7-305-14595-7

定 价 30.00 元

网 址: <http://www.njupco.com>

官方微博: <http://weibo.com/njupco>

官方微信号: njupress

销售咨询热线: (025)83594756

---

\* 版权所有, 侵权必究

\* 凡购买南大版图书, 如有印装质量问题, 请与所购  
图书销售部门联系调换

# 前 言

本书是与北京京胜世纪科技有限公司合作的教材,指导学生循序渐进地完成智慧商城应用的开发,掌握物联网设备以及相关接口的操作原理及编程技巧。

本书共分为四个大的项目:项目一:C#基础知识;项目二:Windows 窗体应用程序的基本控件使用;项目三:物联网设备接口使用和开发;项目四:智慧商城功能开发。

项目一是C#基础部分,通过一系列的小案例引导学生掌握智慧商城项目开发需要的技能,特别是数据库连接方面的技能。

项目二是Windows窗体应用程序的基本控件使用,为更深入地使用C#进行编程做一些铺垫。

项目三是物联网设备接口使用和开发,介绍了15693高频读写器和超高频(ISO/18000—6C)读写器及无线传感网网关等相关知识编程技能。

项目四是智慧商城功能开发,把智慧商城整个案例按功能模块分成较小的任务,主要包括:任务一:商品登记;任务二:商品入库;任务三:商品出库;任务四:商品上架;任务五:发送价格;任务六:人流量监控;任务七:广告推送;任务八:购物引导;任务九:发会员卡;任务十:会员卡充值;任务十一:商品结算;任务十二:商品试穿;任务十三:温湿度检测预警;任务十四:烟雾监控预警;任务十五:红外监控预警;任务十六:震动监控预警。

本书由高洁、王喜胜主编,王可、李桂芹任副主编,参加编写工作的还有冯叶发等。主要分工为:李桂芹编写项目一,王可编写项目二,高洁编写项目三,王喜胜、冯叶发编写项目四。

由于时间仓促,加之水平有限,书中难免存在错误和不妥之处,敬请读者批评指正。

编者

2015年10月

# 目 录

前 言	1
项目一 C#基础知识	1
拟实现的能力目标	1
须掌握的知识内容	1
任务1 SQL Server 数据库连接	1
1.1.1 任务引导训练	1
1.1.2 支撑知识	1
1.1.3 任务同步训练	2
1.1.4 自主任务学习	2
任务2 数据库增删改查	3
1.2.1 任务引导训练	3
1.2.2 支撑知识	3
1.2.3 任务同步训练	4
1.2.4 自主任务学习	4
任务3 IO处理	5
1.3.1 任务引导训练	5
1.3.2 支撑知识	5
1.3.3 任务同步训练	6
1.3.4 自主任务学习	7
任务4 XML应用	7
1.4.1 任务引导训练	7
1.4.2 支撑知识	8
1.4.3 任务同步训练	8
1.4.4 自主任务学习	10
任务5 多线程应用	10
1.5.1 任务引导训练	10
1.5.2 支撑知识	11
1.5.3 任务同步训练	11
1.5.4 自主任务学习	12

任务 6 Socket 应用 .....	13
1.6.1 任务引导训练 .....	13
1.6.2 支撑知识 .....	13
1.6.3 任务同步训练 .....	13
1.6.4 自主任务学习 .....	15
<b>项目二 Windows 窗体应用程序的基本控件使用 .....</b>	<b>17</b>
拟实现的能力目标 .....	17
须掌握的知识内容 .....	17
任务 1 Button 控件使用 .....	17
2.1.1 任务引导训练 .....	17
2.1.2 支撑知识 .....	17
2.1.3 任务同步训练 .....	18
2.1.4 自主任务学习 .....	18
任务 2 CheckBox 控件使用 .....	19
2.2.1 任务引导训练 .....	19
2.2.2 支撑知识 .....	19
2.2.3 任务同步训练 .....	20
2.2.4 自主任务学习 .....	21
任务 3 ContextMenuStrip 控件使用 .....	21
2.3.1 任务引导训练 .....	21
2.3.2 支撑知识 .....	22
2.3.3 任务同步训练 .....	22
2.3.4 自主任务学习 .....	23
任务 4 DateTimePicker 控件使用 .....	24
2.4.1 任务引导训练 .....	24
2.4.2 支撑知识 .....	24
2.4.3 任务同步训练 .....	24
2.4.4 自主任务学习 .....	25
任务 5 FlowLayoutPanel 控件使用 .....	26
2.5.1 任务引导训练 .....	26
2.5.2 支撑知识 .....	26
2.5.3 任务同步训练 .....	27
2.5.4 自主任务学习 .....	28
任务 6 ListView 控件使用 .....	28
2.6.1 任务引导训练 .....	28

2.6.2	支撑知识	29
2.6.3	任务同步训练	29
2.6.4	自主任务学习	32
任务 7	MenuStrip 控件使用	33
2.7.1	任务引导训练	33
2.7.2	支撑知识	34
2.7.3	任务同步训练	34
2.7.4	自主任务学习	34
任务 8	NumericUpDown 控件使用	35
2.8.1	任务引导训练	35
2.8.2	支撑知识	36
2.8.3	任务同步训练	36
2.8.4	自主任务学习	36
任务 9	PictureBox 控件使用	37
2.9.1	任务引导训练	37
2.9.2	支撑知识	38
2.9.3	任务同步训练	38
2.9.4	自主任务学习	38
任务 10	RadioButton 控件使用	39
2.10.1	任务引导训练	39
2.10.2	支撑知识	39
2.10.3	任务同步训练	40
2.10.4	自主任务学习	40
任务 11	SplitContainer 控件使用	41
2.11.1	任务引导训练	41
2.11.2	支撑知识	42
2.11.3	任务同步训练	42
2.11.4	自主任务学习	43
任务 12	TabControl 控件使用	43
2.12.1	任务引导训练	43
2.12.2	支撑知识	44
2.12.3	任务同步训练	44
2.12.4	自主任务学习	45
任务 13	Timer 控件使用	46
2.13.1	任务引导训练	46
2.13.2	支撑知识	46

2.13.3	任务同步训练	47
2.13.4	自主任务学习	48
任务 14	ToolStrip 控件使用	48
2.14.1	任务引导训练	48
2.14.2	支撑知识	49
2.14.3	任务同步训练	49
2.14.4	自主任务学习	49
任务 15	TreeView 控件使用	50
2.15.1	任务引导训练	50
2.15.2	支撑知识	50
2.15.3	任务同步训练	51
2.15.4	自主任务学习	53
任务 16	ComboBox 控件使用	53
2.16.1	任务引导训练	53
2.16.2	支撑知识	54
2.16.3	任务同步训练	54
2.16.4	自主任务学习	55
任务 17	SerialPort 控件使用	56
2.17.1	任务引导训练	56
2.17.2	支撑知识	56
2.17.3	任务同步训练	56
2.17.4	自主任务学习	57
项目三	物联网设备接口使用和开发	58
子项目一	15693 高频读写器	58
拟实现的能力目标		58
须掌握的知识内容		58
任务 1	串口打开(关闭)	58
3.1.1.1	任务引导训练	58
3.1.1.2	支撑知识	59
3.1.1.3	任务同步训练	59
3.1.1.4	自主任务学习	60
任务 2	标签读取	60
3.1.2.1	任务引导训练	60
3.1.2.2	支撑知识	61
3.1.2.3	任务同步训练	61



# 项目一 C# 基础知识

## 拟实现的能力目标

- 能够实现对 SQL 数据库的连接操作；
- 能够实现 SQL 数据库的增删改查等操作；
- 能够使用 Windows 窗体应用程序进行 IO 操作；
- 能够使用 Windows 窗体应用程序对 XML 文件进行读取与写入操作；
- 能够使用线程操作；
- 能够使用 Socket 操作。

## 须掌握的知识内容

完成该实验需要掌握的知识点如下：

- 了解数据库 ODBC 连接；
- 了解 SQL 数据库的语句使用；
- 了解 IO 操作的基本概念；
- 掌握 XML 文件的特性及优缺点；
- 掌握线程的优缺点；
- 掌握 Socket 连接过程及方式。

## 任务 1 SQL Server 数据库连接

### 1.1.1 任务引导训练

#### 【引导任务】

Windows 窗体程序设计中需要连接数据库，将数据写入数据库中。

#### 【训练任务分析】

为实现本任务，需掌握以下知识：

- 掌握数据库的基本概念；
- 使用数据库的必要性。

### 1.1.2 支撑知识

- 了解 ODBC 数据库连接；
- 了解 ASP.NET 数据库连接知识。

### 1.1.3 任务同步训练

#### 【任务描述】

数据库连接是在整个 C# 基础中最常用也是最基础的功能之一。数据库连接是通过 Windows 窗体应用程序语句与 SQL 数据库连接。

#### 【同步训练任务】

(1) 打开 Visual Studio 2010 开发环境,新建一个 Windows 窗体应用程序,并将其命名为 DatabaseConnection。

(2) 在该窗体中添加 1 个 Button 控件。如下图 1-1 所示:



图 1-1

(3) 在提供的光盘中找到 db\_BookManage 文件夹下的数据库文件,附加到数据库即可。

(4) 程序的主要代码如下:

//建立与数据库的连接,其中 server = .; 中的 "." 代表本地服务器,db\_BookManage 代表需要连接的数据库名,Trusted\_Connection = true; 中的 "true" 代表打开数据库连接,为 "fause" 则表示关闭。

```
SqlConnection conn=new SqlConnection("server=.;database=db_UserInfo;  
Trusted_Connection = true;");
```

```
//打开数据库
```

```
conn.Open();
```

```
//判断数据库是否连接成功
```

```
if (conn.State == ConnectionState.Open)
```

### 1.1.4 自任务学习

#### 【课后自主学习】

了解 Windows 窗体应用程序在连接数据库时的注意事项。

## 每次课程意见反馈表

日期： 年 月 日

我喜欢的：☺
我不喜欢的：☹
我不理解的：(?)
我的建议：★
今天学到的最重要的内容：

## 任务 2 数据库增删改查

### 1.2.1 任务引导训练

#### 【引导任务】

Windows 窗体程序设计中需要连接数据库,对数据库中的表数据进行添加、删除、修改或者查询。

#### 【训练任务分析】

为实现本任务,需掌握以下知识:

- 如何连接数据库,详见项目一任务 1;
- SQL Server 基础知识。

### 1.2.2 支撑知识

- 了解数据库增删改查语句的使用。

### 1.2.3 任务同步训练

#### 【任务描述】

数据库增删改查是在整个 C# 基础中最常用也是最基础的功能之一。数据库增删改查是通过 Windows 窗体程序语句与数据库交互。

#### 【同步训练任务】

(1) 打开 Visual Studio 2010 开发环境,新建一个 Windows 窗体应用程序,并将其命名为 DatabaseInsertDeleteUpdateSelect。

(2) 在该窗体中添加 4 个 Button 控件。如下图 1-2 所示:

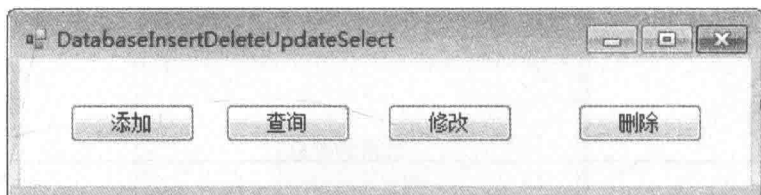


图 1-2

(3) 在提供的光盘中找到 db\_BookManage 文件夹下的数据库文件,附加到数据库即可(注:如在任务 1 中附加了数据库则在此不需要再附加数据库)。

(4) 程序的主要代码如下:

```
//向数据库添加信息
sh.insert();
//查询数据库中的信息
sh.Select();
//修改数据库中的信息
sh.Change();
//删除数据库中的信息
sh.Select();
//获得受影响的行数
sh.IsSuccess();
//获得查询结果
sh.IsSucced()
```

### 1.2.4 自主任务学习

#### 【课后自主学习】

利用 SQL Server 数据库语句将其他数据写入数据库。

## 每次课程意见反馈表

日期： 年 月 日

我喜欢的：😊
我不喜欢的：😞
我不理解的：❓
我的建议：★
今天学到的最重要的内容：

## 任务 3 IO 处理

## 1.3.1 任务引导训练

## 【引导任务】

利用 IO 处理通过字节流读取及写入内容到 txt 文档。

## 【训练任务分析】

为实现本任务,需掌握以下知识:

- IO 操作的基本概念。

## 1.3.2 支撑知识

- 了解 IO 操作的基本原理。

### 1.3.3 任务同步训练

#### 【任务描述】

IO 处理是在整个 C# 基础中最常用也是最基础的功能之一。IO 处理是利用 Windows 窗体程序设计代码读取本机中的文本图片或将输入的内容保存到文本中。

#### 【同步任务训练】

(1) 打开 Visual Studio 2010 开发环境,新建一个 Windows 窗体应用程序,并将其命名为 IO。

(2) 在该窗体中添加 3 个 Button 控件、1 个 Label 标签、2 个 TextBox 文本框和 1 个 GroupBox 控件。如下图 1-3 所示:

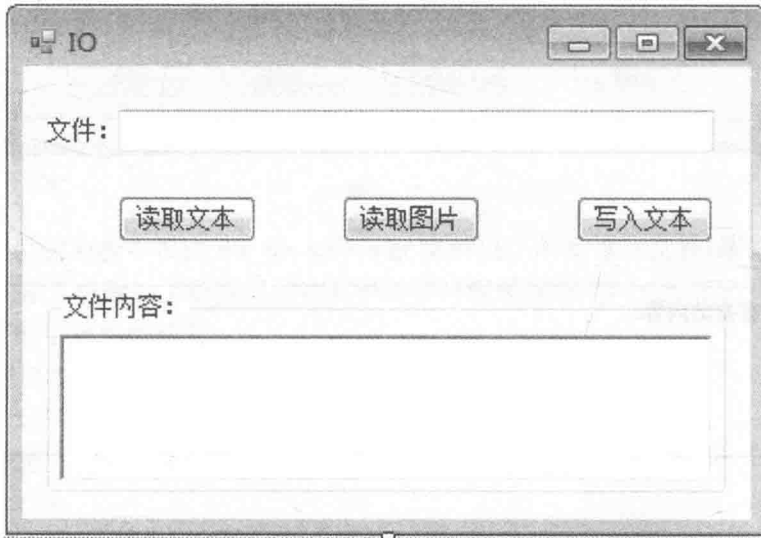


图 1-3

(3) 程序的主要代码如下:

```
//设置打开文件的类型
ofd.Filter = "文本文件(*.txt)|*.txt";
//显示文件路径选择对话框
ofd.ShowDialog();
//设置打开的文件名称
textBox1.Text = ofd.FileName;
//打开现有文件进行读取
FileStream fs = File.OpenRead(textBox1.Text);
//把字符串写入文本流
sw.WriteLine(rtxtbox_Content.Text);
//移除剪切板中所有数据
```

```
Clipboard.Clear();  
//将选择的图片添加到剪切板中  
Clipboard.SetImage bmp);  
//用剪贴板内容替换文本框中当前选定内容  
rtxtbox_Content.Paste();
```

### 1.3.4 自主任务学习

#### 【课后自主学习】

利用 IO 处理获取其他格式的文件并进行显示。

每次课程意见反馈表

日期： 年 月 日

我喜欢的：😊
我不喜欢的：😞
我不理解的：❓
我的建议：★
今天学到的最重要的内容：

## 任务 4 XML 应用

### 1.4.1 任务引导训练

#### 【引导任务】

通过读写 XML 文件内容配置设备信息。

## 【训练任务分析】

为实现本任务,需掌握以下知识:

- XML 文件的特性及优缺点。

### 1.4.2 支撑知识

- XML 文件的基本概念。

### 1.4.3 任务同步训练

## 【任务描述】

XML 应用是在整个 C# 基础中最常用也是最基础的功能之一。XML 可以用来标记数据、定义数据类型,是一种允许用户对自己的标记语言进行定义的源语言。通过读写 XML 文件可以配置设备信息。

## 【同步任务训练】

(1) 打开 Visual Studio 2010 开发环境,新建一个 Windows 窗体应用程序,并将其命名为 XML。

(2) 在该窗体中添加 4 个 Button 控件、13 个 Label 标签、12 个 TextBox 文本框、2 个 GroupBox 控件、1 个 DataGridView 和 1 个 ComboBox 控件。如下图 1-4 所示:

图 1-4



(3) 程序的主要代码如下:

```
// 显示 XML 文件中的所有信息
getXmlInfo();
// 创建 XML 文档
XDocument doc = new XDocument(
    // 创建 XML 文档对象
    new XDeclaration("1.0", "utf-8", "yew"),
    // 添加 XML 文件声明
    new XElement(textBox1.Text, // 创建 XML 元素
    new XElement(textBox2.Text, new XAttribute(textBox3.Text,
    textBox10.Text), // 为 XML 元素添加属性
    new XElement(textBox4.Text, textBox5.Text),
    new XElement(textBox6.Text, textBox7.Text),
    new XElement(textBox8.Text, textBox9.Text))
    )
);

// 保存 XML 文档
doc.Save(strPath);
// 将 XML 文件内容绑定到 DataGridView 控件
getXmlInfo()
// 加载 XML 文档
XElement xe = XElement.Load(strPath);
// 创建 IEnumerable 泛型接口
IEnumerable<XElement> elements1 = from element in xe.Elements("People")
    select element;

// 生成新的编号
string str = (Convert.ToInt32(elements1.Max(element => element.Attribute("ID").Value)) + 1)
    .ToString("000");
// 根据编号查找信息
IEnumerable<XElement> elements = from element in xe.Elements("People")
    where element.Attribute("ID").Value
    == strID
    select element;

// 删除找到的 XML 元素信息
elements.First().Remove();
// 遍历查找到的所有信息
foreach (XElement element in elements)
```