

C#程序设计经典 教程



第九章 Windows程序的界面设计

1. 总体要求

- 理解基于Windows的应用程序的基本概念
- 掌握Windows窗体和控件的常用属性。
- 掌握一些常用的Windows窗体控件的使用。
- 学会创建较为复杂的Windows应用程序。。

2. 学习重点

- Windows 应用程序中窗体、控件、事件的基本概念。
- Windows 窗体控件的共同特性。
- 常用的Windows 窗体控件。

第九章 Windows程序的界面设计

主要内容

- 9.1 窗体与控件概述
- 9.2 按钮与文本显示、编辑控件
- 9.3 列表与选择控件
- 9.4 图形显示控件
- 9.5 容器控件
- 9.6 对话框
- 9.7 菜单、工具栏和状态栏
- 9.8 SDI和MDI应用程序

9.1 窗体与控件概述

- 9.1.1 Windows窗体
- 9.1.2 窗体的控件

返回

9.1.1 Windows窗体

- 窗体是控件的容器，而控件是窗体的成员对象，用于显示信息并且接收用户输入的信息。
- **Windows窗体的基类是Form，位于System.Windows.Forms命名空间中。**
- 表9-1列出了**Windows窗体的主要属性（见教材）**
- 窗体和控件的属性可以在**Visual Studio .NET的属性窗口**进行设计，也可以通过编写代码来完成。

例如:

- `this.ShowInTaskbar = true;`
- `this.StartPosition =
FormStartPosition.CenterScreen;`
- `this.Text = "窗体属性设置";`
- `this.TopMost = true;`
- `this.WindowState =
FormWindowState.Maximized;`
- `this.FormBorderStyle =
FormBorderStyle.Fixed3D;`

9.1.2 窗体的控件

1. .NET Framework中的窗体控件

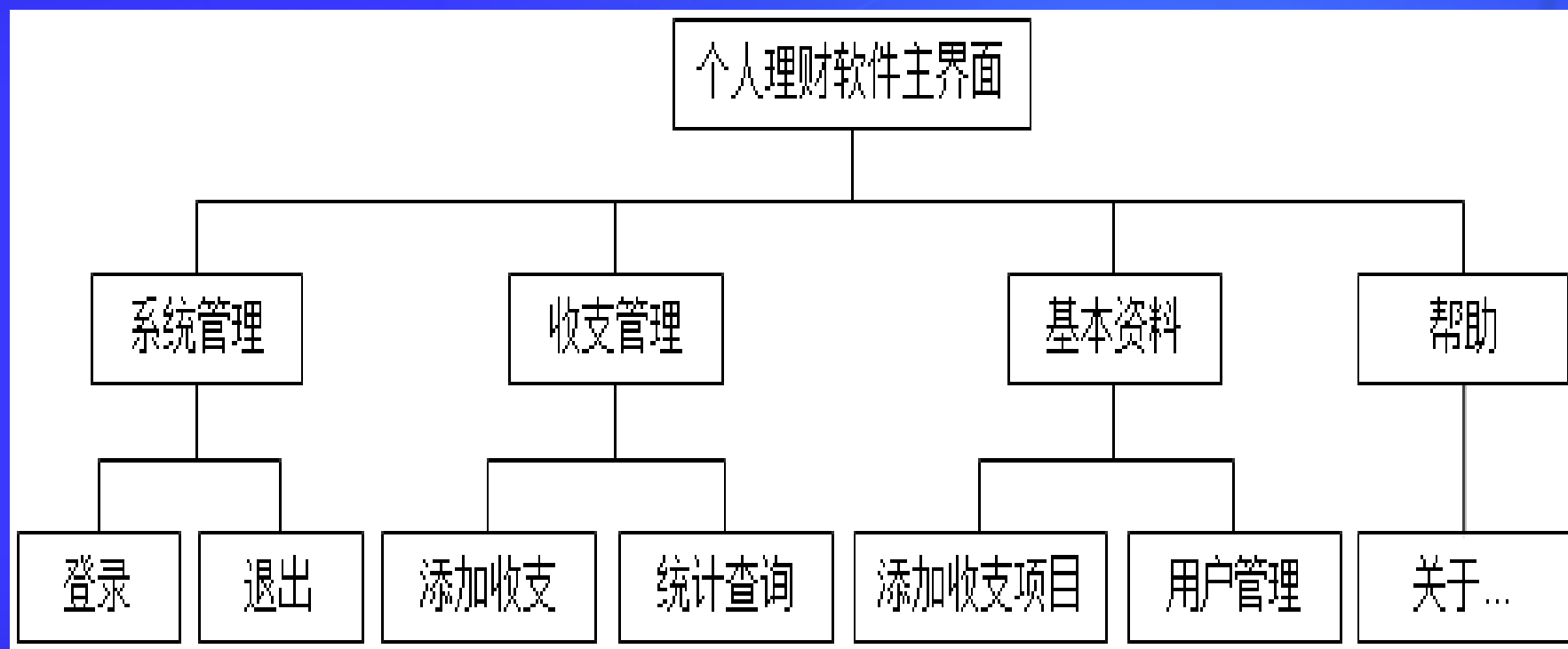
- 控件是用来完成特定的输入/输出功能的。例如，按钮控件响应用户的单击事件，文本框控件接收用户的输入等。
- 在.NET Framework中，控件派生于 **System.Windows.Forms.Control** 类，该类定义了控件的基本功能。
- 表9-2列出了一些常见的Windows控件和组件。
- 控件的用法：静态引用和动态引用。其中，静态引用就是在设计时把控件从工具箱拖放到窗体设计区中。动态引用就是在源程序代码中通过控件类来创建控件对象，在完成对象属性设置再将其添加到窗体中。

2. 控件的属性

- **Anchor** 获取或设置控件绑定到的容器的边缘并确定控件如何随其父级一起调整大小
- **BackColor** 控件的背景色
- **Bottom** 控件下边缘与其容器的工作区上边缘之间的距离（以像素为单位）
- **Dock** 获取或设置控件边框停靠到其父控件并确定控件如何随其父级一起调整大小
- **Enabled** 控件是否可以对用户交互做出响应
- **ForeColor** 控件的前景色
- **Height** 控件的高度
- **Left** 控件左边缘与其容器的工作区左边缘之间的距离（以像素为单位）

- **Location** 控件的左上角相对于其容器的左上角的坐标
- **Name** 控件的名称，可以在代码中用于引用该控件
- **Parent** 控件的父容器
- **Right** 获取控件右边缘与其容器的工作区左边缘之间的距离（以像素为单位）
- **Size** 控件的高度和宽度
- **TabIndex** 控件的 Tab 键顺序
- **TabStop** 指示用户能否使用 Tab 键将焦点放到该控件上
- **Text** 与此控件关联的文本
- **Top** 控件上边缘与其容器的工作区上边缘之间的距离（以像素为单位）
- **Visible** 指示是否显示该控件
- **Width** 控件的宽度

个人理财软件的功能模块图



9.2 按钮与文本显示、编辑控件

- 9.2.1 按钮控件
- 9.2.2 文本显示控件
- 9.2.3 文本编辑控件
- 9.2.4 应用实例

返回

9.2.1 按钮控件

- **Button**（按钮）控件用来接收用户的操作命令，激发相应的事件，例如，“确定”或者“取消”。
- 在设计时，通常先添加控件，再为其**Click**事件编写代码；在程序运行时，单击该按钮就会执行**Click**事件方法。
- 常用的属性有**Name**、**Text**、**Visible**、**Enabled**、**FlatStyle**、**Image**和**ImageAlign**。
- 其中，**Text**属性指定要显示的标题文字，在字母前添加“&”字符“可设置快捷键。
- **Image**属性可以指定要显示的图像；
- **ImageAlign**、**TextAign**和**TextImageRelation**属性用来设置图像与文本的排列方式。



9.2.2 文本显示控件

- .NET Framework包含两个标签控件：**Label**和**LinkLabel**。其中，**Label**是标准的Windows标签；**LinkLabel**类似于**Label**，但以超链接方式显示。
- **Label**控件的常用的属性：
 - **AutoSize**：默认值为true。
 - **BorderStyle**：决定控件边框的样式，其值是**BorderStyle**枚举值。**=None**时，无边框、**=FixedSingle**，单行边框，**=Fixed3D**，三维边框。默认为None。
- **LinkLabel**其常用的属性如表9-4所示（见教材）

9.2.3 文本编辑控件

- 1. **Textbox**控件
- 其用途是让用户输入文本，用户可以在其中输入任何字符或指定类型的字符。默认情况下，最多可输入**2048**个字符，但如果将**Multiline**属性设置为**true**，则最多可输入**32KB**的文本。用户输入的文本保存在**Text**属性中，在程序中引用**Text**属性即可获得用户输入的文本。
- **TextBox**控件的常见属性如表9-5所示。
- **TextBox**控件的常用事件如表9-6所示。

2. RichTextBox控件

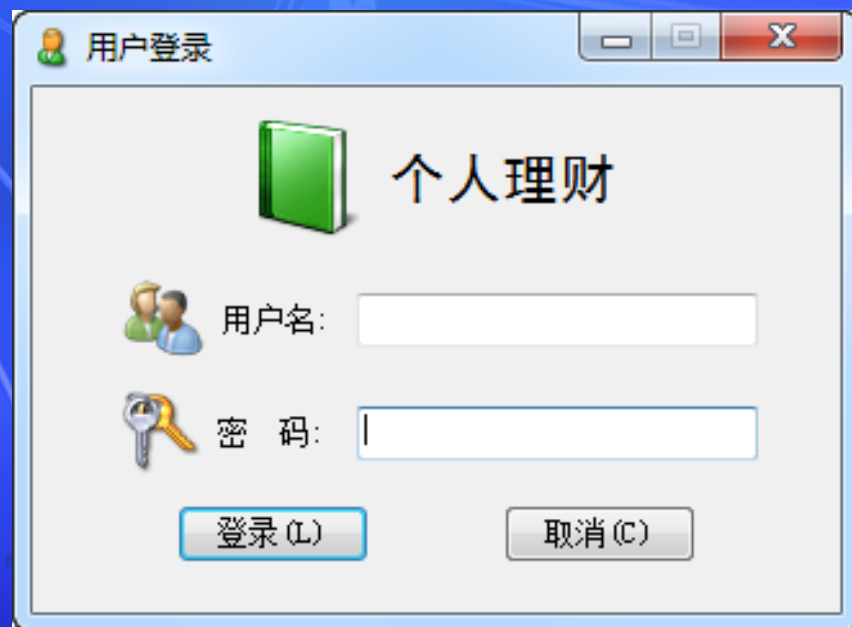
- **RichTextBox**用于显示和输入格式化的文本（又称富文本格式**Rich Text Format, RTF**），可以显示字体、颜色和链接，从文件加载文本和加载嵌入的图像，以及查找指定的字符，因此**RichTextBox**常常称为增强的文本框。
- **RichTextBox** 控件通常用于提供类似字处理应用程序（如**Microsoft Word**）的文本操作和显示功能。**RichTextBox** 控件的默认设置是水平和垂直滚动条均根据需要显示，并且拥有更多的滚动条设置。
- **RichTextBox**常见的属性见表9-7。

9.2.4 应用实例

【实例9-1】设计一个简单的个人记账软件的用户登录界面，当输入正确的用户名和密码时，系统将给出正确的提示，否则给出错误提示。注意，由于实际的身份验证需要与数据库建立连接，所以在这里先将功能简化，下一章将进一步完善本程序。

【操作演示】

代码分析（见教材）



9.3 列表与选择控件

- 9.3.1 RadioButton控件
- 9.3.2 CheckBox控件
- 9.3.3 ListBox控件
- 9.3.4 ComboBox控件
- 9.3.5 其它常用控件
- 9.3.6 应用实例

返回

9.3.1 RadioButton控件

- **RadioButton**（单选按钮）控件为用户提供由两个或多个互斥选项组成的选项集。用户在一组单选按钮中只能选择一个。



男



女

9.3.1 RadioButton控件

- **RadioButton控件的常用属性**
 - **Checked属性:**
 - 获取或者设置一个值，其值为**True**，则控件处于选中状态，否则为未选中状态；
 - **Text属性:**
 - 控件上显示的文本；
 - **AutoCheck属性**
 - 为**true**时单击自动选定，为**false**时必须在其**Click**事件中通过修改**Checked**属性的值来选定；
- 一个容器的**RadioButton**为一组，通常的分组容器控件为**GroupBox**；

9.3.1 RadioButton控件

- **RadioButton控件的常用事件**
 - **Click事件：**
 - 用鼠标单击RadioButton时发生；
 - **CheckedChanged事件：**
 - 其Checked属性发生改变时发生；注意：某个RadioButton该属性的值发生改变可能会引起两个RadioButton控件的CheckedChanged事件先后发生。
- **见例9.3.1**

9.3.2 CheckBox控件

- **CheckBox**控件允许用户选择多个选项，而 **RadioButton** 控件只允许用户从互相排斥的选项中选择的一个。
- **CheckBox**控件的常用属性：
 - **Checked**属性:其值为**True**或者**False**;
 - **CheckState**: 其值可为**Checked**、**Unchecked**和**Indeterminate**。值为**Indeterminate**时，表示复选框的当前值无效。
 - **ThreeState**: 设置工作模式为“两种状态”或“三种状态”，前者使用 **Checked** 属性返回状态值，后者使用 **CheckState** 属性返回状态值。



Unchecked



Checked



Indeterminate

9.3.2 CheckBox控件

- CheckBox控件的常用事件:

- CheckChanged事件

- 复选框Checked属性发生改变时此事件发生;
- 通过代码将复选框从Checked状态改变到Indeterminate不会触发此事件, 但从Unchecked状态改变到Indeterminate则会触发此事件;
- 当复选框的ThreeState为false时, 用户单击复选框一定会引发此事件;
- 当复选框的ThreeState为True时, 用户单击复选框从状态Checked改变到Indeterminate 时也不会触发此事件;

9.3.3 ListBox控件

- **ListBox**（列表框）控件用于显示一组字符串，可以从中选择一个或多个选项。与复选框和单选按钮一样，列表框也提供了要求用户选择一个或多个选项的方式。
- **ListBox**控件的常用属性如表9-13所示。
- **ListBox**控件的常用方法如表9-14所示。
- **ListBox**控件的常用事件主要是：**SelectedIndexChanged**，表示选中项的索引被改变时触发的事件。

9.3.3 ListBox控件

- **Items集合属性**

- 访问第i项: `listBox1.Items[i]`
- 添加项目: `listBox1.Items.Add(string s)`
- 项目数: **Count**属性返回
- 插入项目: `listBox1.Items.Insert(位置,项)`
- 删除项目:
 - `listBox1.Items.Remove(listBox1.SelectedItem);`
 - `listBox1.Items.RemoveAt(listBox1.SelectedIndex);`

9.3.4 ComboBox控件

- **ComboBox**（组合框）把文本框控件和列表框组合在一起，使用户可以从列表中选择选项，也可以输入新文本。**ComboBox** 的默认行为是显示一个可编辑文本框，该文本框具有一个隐藏的下拉列表。
- 属性
 - **DropDownStyle**属性：确定要显示的组合框的样式，**=Simple**，简单的下拉列表，始终显示列表和下拉列表框；**=DropDownList**，文本部分不可编辑，并且必须单击下拉箭头才能查看下拉列表中的选项；**=DropDown**，默认下拉列表框，文本部分可编辑，并且用户必须单击下拉箭头才能查看列表中的选项。
- **Combo**具**TextBox**和**ListBox**的大部分属性和事件

9.3.5 其它常用控件

- **CheckedListBox**控件
 - 和CheckBox和ListBox的很多属性、事件比较接近;
 - **CheckOnClick**属性: 单击切换复选框的选中状态
 - **SelectionMode**属性只能None或One;
 - **GetItemChecked(i)**方法返回第i项是否被复选;
 - **GetSelected(i)**方法返回第i项是否被选中;
 - **SetItemChecked(i, true|false)**方法用于设置第i项是否被复选;
 - **CheckedItems**和**CheckedIndices**属性分别返回被复选的项的集合和项的索引的集合;

9.3.5 其它常用控件

- **ImageList控件**
 - 通过其**Images**属性存储一组图像;
- **ListView控件**
 - 可以显示带图标列表;
 - **View**属性定义视图;
 - **LargeImageList**和**SmallImageList**分别定义大图标和小图标对应的**ImageList**控件名称;
 - **Items**属性定义列表项;

9.3.5 其它常用控件

- **TreeView控件**
 - 通过树状结构来展示层次性数据；
 - **Nodes**集合属性返回或者设置所有节点；每个节点都有**Nodes**集合表示其子节点；
 - 节点的常用方法（**Expand**、**ExpandAll**、**Collapse**、**Add**、**Remove**）
 - **Index**属性返回每个节点在本层的序号；
 - **ImageList**设置节点图标所对应的**ImageList**控件；
 - **CheckBoxs**设置节点是否带复选框；

9.3.5 其它常用控件

- **DomainUpDown控件**
 - 功能相当于一个组合框，含一个用于输入的文本框和切换列表项的两个箭头；
 - 核心属性为**Text**和**Items**；
- **NumericUpDown控件**
 - 和**DomainUpDown**类似，用于输入一个数值
 - **Minimum**和**Maximum**属性控制值的范围；
 - **Value**属性返回或者设置控件的值；
 - **Increment**属性定义递增或者递减的幅度；
 - **DecimalPlaces**属性定义小数位数；

9.3.5 其它常用控件

- **dateTimePicker控件**

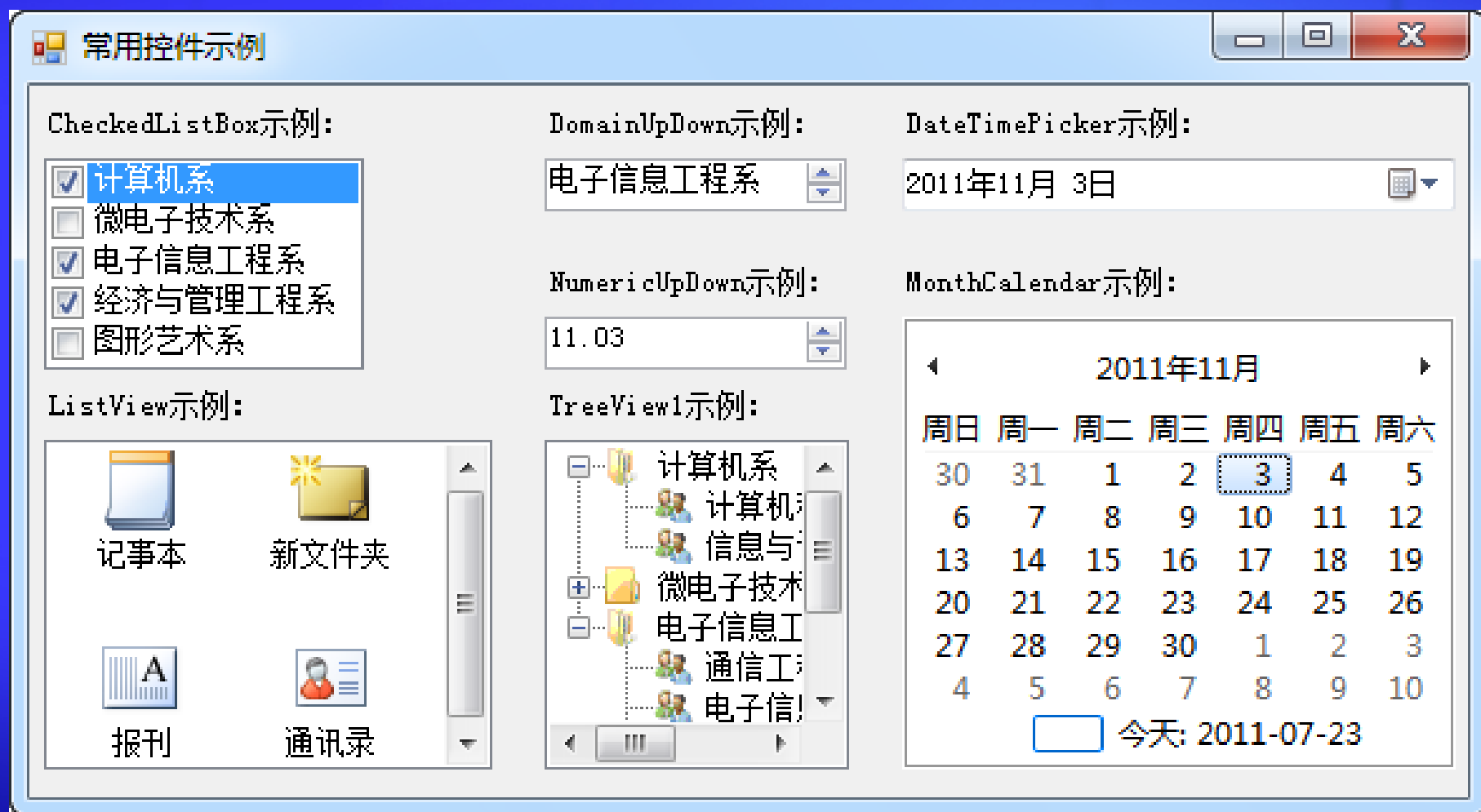
- 用于输入一个日期时间型数据；
- **Value**属性返回或者设置你显示的日期和时间；
- 自定义格式

```
dateTimePicker1.Format = DateTimePickerFormat.Custom  
dateTimePicker1.CustomFormat = "yyyy-MM-dd hh:mm:ss"
```

- **MonthCalendar**

- 以日历的方式显示，用于让用户选择一个日期范围；
- **MaxSelectionCount**属性设置选择的日期数；
- **SelectionStart**、**SelectionEnd**返回选择的日期的开始和终止值；

9.3.5 其它常用控件



9.3.6 应用实例

【实例9-2】在项目MyAccounting中添加一个窗体，实现如图9-12所示的效果，用于添加个人收支明细。

操作演示

代码分析

收支情况记录

收支类型: ☐ 收入 ☒ 支出

日期: 2011年11月3日

收支类别: 生活消费

说明: 外出就餐

收支项目: 餐饮
生活用品
交通费
水电气

收支人: ☒ 我 ☒ 家人 ☐ 亲戚
☐ 朋友 ☐ 同事 ☐ 其它

金额: 288.00

备注: 要保存的信息为:
生活消费-餐饮
日期: 2011年11月3日
说明: 外出就餐
收支人: 我、家人
金额: 288.00

保存(S) 取消(C)

9.4 图形显示控件

- 9.4.1 PictureBox控件
- 9.4.2 ImageList控件
- 9.4.3 应用实例

返回

9.4.1 PictureBox控件

- **PictureBox**控件用于显示位图、GIF、JPEG、图元文件或图标格式的图形。
- 常用属性有：
 - **Image**: 设置所要显示的图片
 - **SizeMode**: 控制图像显示方式:
 - **=Normal**（默认），图像置于**PictureBox** 的左上角，尺寸过大时自动被剪裁掉；
 - **=StretchImage**时，图像将拉伸显示；
 - **=AutoSize**时，自动调整大小以完整显示；
 - **=CenterImage**，图像居中显示。

9.4.2 ImageList控件

- **ImageList**控件只用于存储图像。将位图、图标添加到该控件，再由其他控件显示，这样只需更改某个控件的**ImageIndex**或**ImageKey**属性，就可改变显示的图像。
- 另外，还可在同一个**ImageList**控件与多个控件之间设置关联，例如用**ListView**控件和**TreeView**控件显示同一个文件列表，当更改图像列表中某个图标时，新图标将同时显示在两个视图中。
- **ImageList**控件的主要属性：
 - **Images**: 包含一个图像集合。
 - **ColorDepth**: 确定呈现图像时所使用的颜色数量。
 - **ImageSize**: 设置图像大小，所有图像都将以同样的大小显示。

9.4.3 应用实例

【实例9-3】在项目MyAccounting中添加一个窗体，实现如图9-13所示的效果，用于显示系统说明。

- 操作演示
- 代码分析



9.5 容器控件

- 9.5.1 GroupBox控件
- 9.5.2 Panel控件
- 9.5.3 TabControl控件
- 9.5.4 应用实例

返回

9.5.1 GroupBox控件

- **GroupBox**（分组框）控件用于为其他控件提供可识别的分组。通常，使用分组框按功能细分窗体，将一个窗体的各种功能进一步分类。
- 在窗体上创建**GroupBox**控件及其内部控件时，必须先建立**GroupBox**控件，然后在其内建立各种控件，
- 如果要将窗体上已经放好的控件进行分组，则应选中控件，然后将它们剪切并粘贴到**GroupBox**控件中，或者直接把控件拖到**GroupBox**中。

9.5.2 Panel控件

- **Panel**（面板）控件类似于**GroupBox** 控件。
- 主要区别是：只有**GroupBox** 控件可显示标题，只有**Panel**控件可以有滚动条。若要显示**Panel**控件中的滚动条，需将其**AutoScroll** 属性设置为**true**。也可以通过设置**BackColor**、**BackgroundImage**和**BorderStyle**属性自定义面板的外观。

9.5.3 TabControl控件

- 该控件用于显示多个选项卡，每个选项卡可包含图片和其他控件。
- 使用该控件可生成多页对话框，还可以创建用于设置一组相关属性的属性页。
- 其重要属性：**TabPage**，该属性包含单独的选项卡。每一个单独的选项卡都是一个**TabPage**对象。单击选项卡时，将引发该**TabPage**对象的**Click**事件。
- **TabControl**控件的常用属性如表9-17所示

9.5.4 应用实例

【实例9-4】在项目MyAccounting中添加一个窗体，实现如图9-15所示的效果，用于添加收支项目信息。

- 操作演示

- 代码分析

The screenshot shows a Windows-style dialog box titled "添加收支项目" (Add Income/Expense Item). It has a green plus icon in the title bar. The dialog contains two tabs: "收支项目信息" (Income/Expense Item Information) and "确认信息" (Confirmation Information). The "收支项目信息" tab is active. It contains the following fields and controls:

- "项目名称:" (Project Name): A text box containing the value "书籍" (Books).
- "所属类别:" (Category): A section containing two radio buttons: "收入" (Income) and "支出" (Expense). The "支出" radio button is selected.
- "类别:" (Category): A dropdown menu showing "教育培训" (Education and Training).
- "预览 (V)" (Preview): A button at the bottom.

9.6对话框

- 9.6.1 对话框概述
- 9.6.2 消息框
- 9.6.3 通用对话框
- 9.6.4 应用实例

返回

9.6.1 对话框概述

1. 模态对话框

- 所谓模态对话框，就是指当这个对话框弹出的时候，鼠标不能单击这个对话框之外的区域，这种对话框往往是在用户进行了某种操作后才出现的。

- 打开一个模态对话框：

- 窗体对象.ShowDialog();

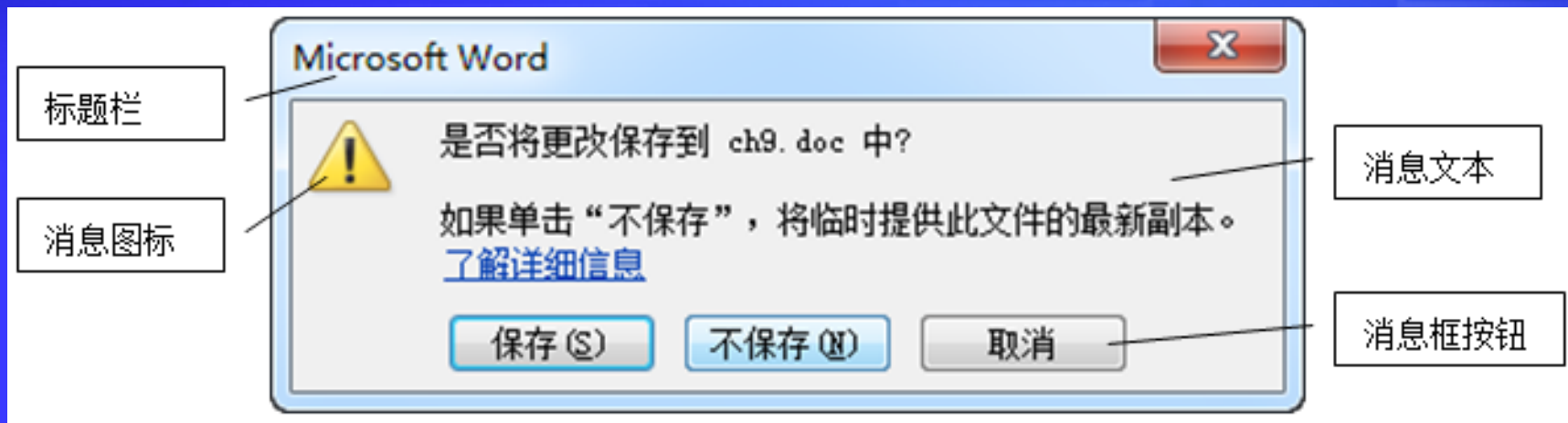
【实例9-5】创建一个新Windows应用程序，打开一个模态对话框。

- 操作演示

2. 非模态对话框

- 非模态对话框通常用于显示用户需要经常访问的控件和数据，并且在使用这个对话框的过程中需要访问其他窗体的情况。
- 打开非模态对话框使用**Show**方法。
【实例9-6】在实例9-5创建的Windows应用程序中，打开一个非模态对话框。
- 操作演示

9.6.2 消息框



1. 消息框按钮

- 消息框最多有3个按钮，但不能随意定义，只能从预定按钮组中选择，如表9-20所示。
- 单击某一按钮时，**Show**方法将返回一个**DialogResult**枚举值。表9-21显示了**DialogResult**的枚举成员。

2. 消息框图标

- **MessageBoxIcon**枚举定义了消息框的图标，只有4个，但是在**MessageBoxIcon**枚举中共有9个成员，如下表所示。

成 员	包含的图标	成 员	包含的图标
Asterisk		Information	
Error		Question	
Exclamation		Stop	
Hand		Warning	
None	不显示图标		

9.6.3 通用对话框

- NET提供了一组基于标准对话框，包括OpenFileDialog（文件打开）、SaveFileDialog（文件另存为）、FolderBrowserDialog（文件夹选择）、ColorDialog（颜色）以及FontDialog（字体）对话框等。
- 1. ColorDialog
- 重要属性：

AllowFullOpen

AnyColor

Color

FullOpen

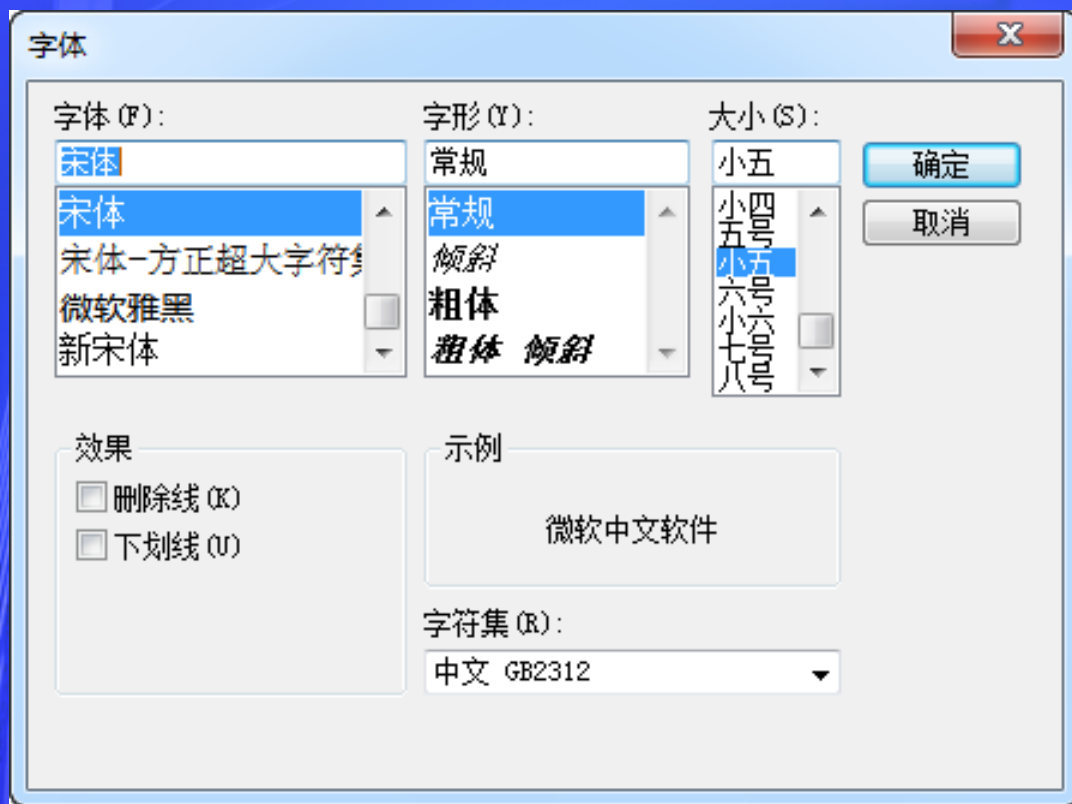
SolidColorOnly



2. FontDialog

重要属性:

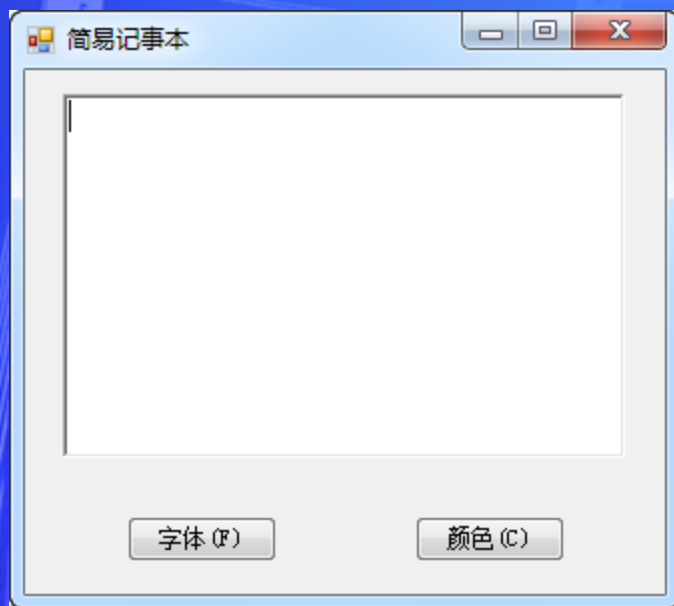
- AllowVectorFonts
- AllowVerticalFonts
- Color
- FixedPitchOnly
- Font
- MaxSize
- MinSize
- ShowApply
- ShowColor
- ShowEffects



9.6.4 应用实例

【实例9-6】制作一个简单的文本编辑器，可实现对文本内容进行编辑和修饰，包括更改文本的颜色和字体等。

- 操作演示
- 代码分析



9.7 菜单、工具栏和状态栏

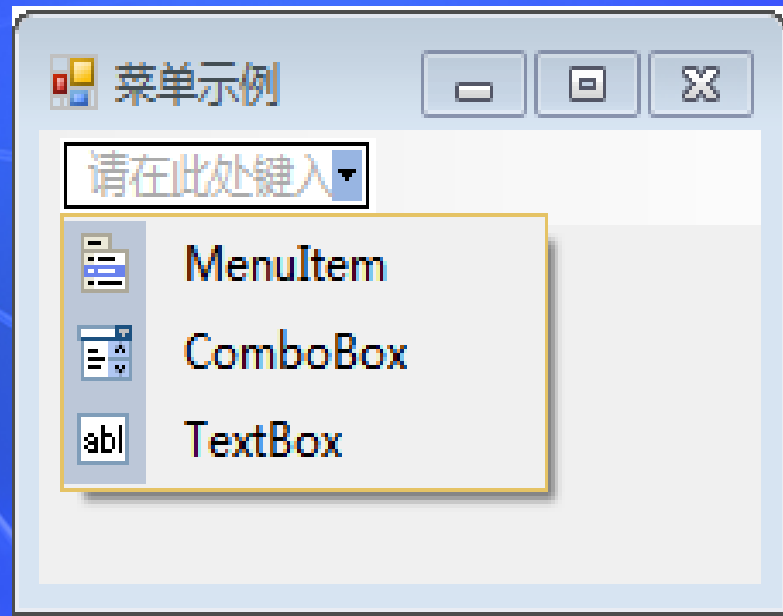
- 9.7.1 菜单
- 9.7.2 工具栏
- 9.7.3 状态栏
- 9.7.4 应用实例

返回

9.7.1 菜单

1. 下拉菜单MenuStrip控件

- 在窗体上生成一个MenuStrip控件;
- 菜单项可以直接输入, 并通过“&”符号引导快捷键;
- 菜单栏可以选择是右图所示三种情况, 即除了可以建立一个菜单项以外, 还可以建立文本框或者组合框;
- 如果输入“-”, 则生成菜单分割条;
- 菜单项、分割条、文本框和组合框分别是一个ToolStripMenuItem、ToolStripSeparator、ToolStripComboBox、ToolStripTextBox对象;



9.7.1 菜单

2. ToolStripMenuItem常用属性

- **Checked**: 菜单项是否处于选中状态;
- **CheckOnClick**: 单击是否切换选中状态;
- **DisplayStyle**: 显示文本还是图像还是两者都显示;
- **Text**: 显示的文本;
- **Image**: 显示的图像;
- **DropDownItems**: 定义下级菜单的集合;
- **ShortcutKeys**: 定义该菜单项快捷方式;
- **ShowShortcutKeys**: 是否在菜单后显示快捷键;
- **ToolTipText**: 定义上下文提示

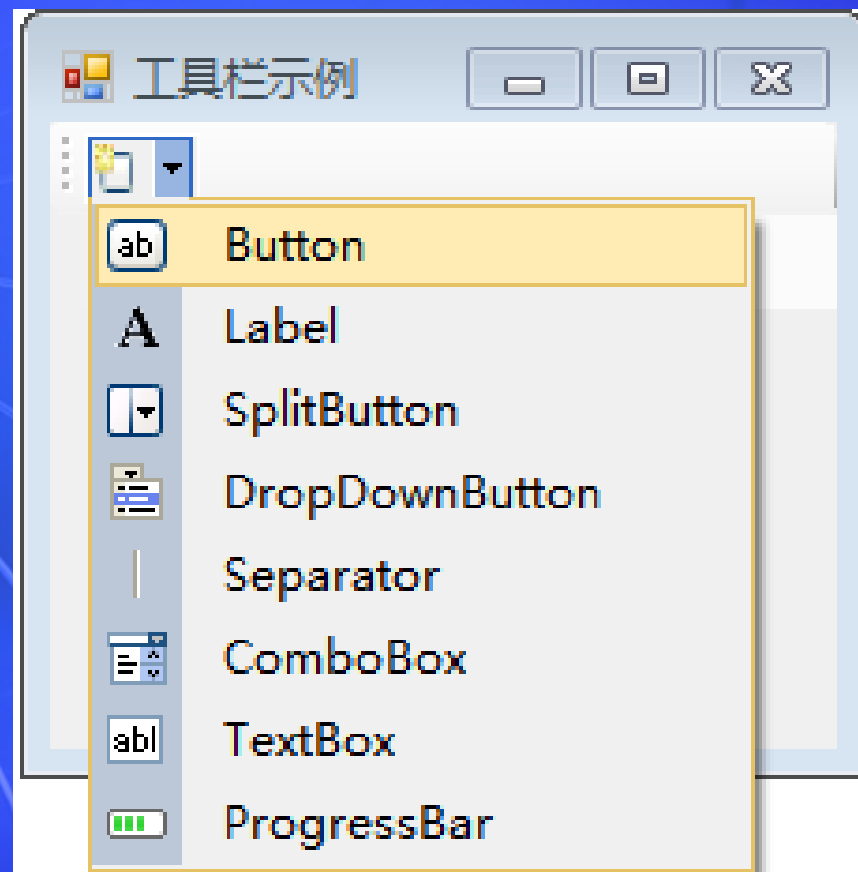
3.事件:Click

2. 上下文菜单，也称快捷菜单

- 设计步骤如下：
- (1) 在工具箱中双击**ContextMenuStrip**控件。
- (2) 为其设计菜单项，设计方法与**MenuStrip**控件相同，只是不必设计主菜单项，如图9-30所示。
- (3) 选中需要使用的快捷菜单的窗体或控件，在其“属性”窗口中，单击**ContextMenuStrip**选项，从弹出的下拉列表中选择所需的**ContextMenuStrip**控件，如图9-31所示。当运行程序时，在窗体中右击，即可弹出上下文菜单，如图9-32所示。

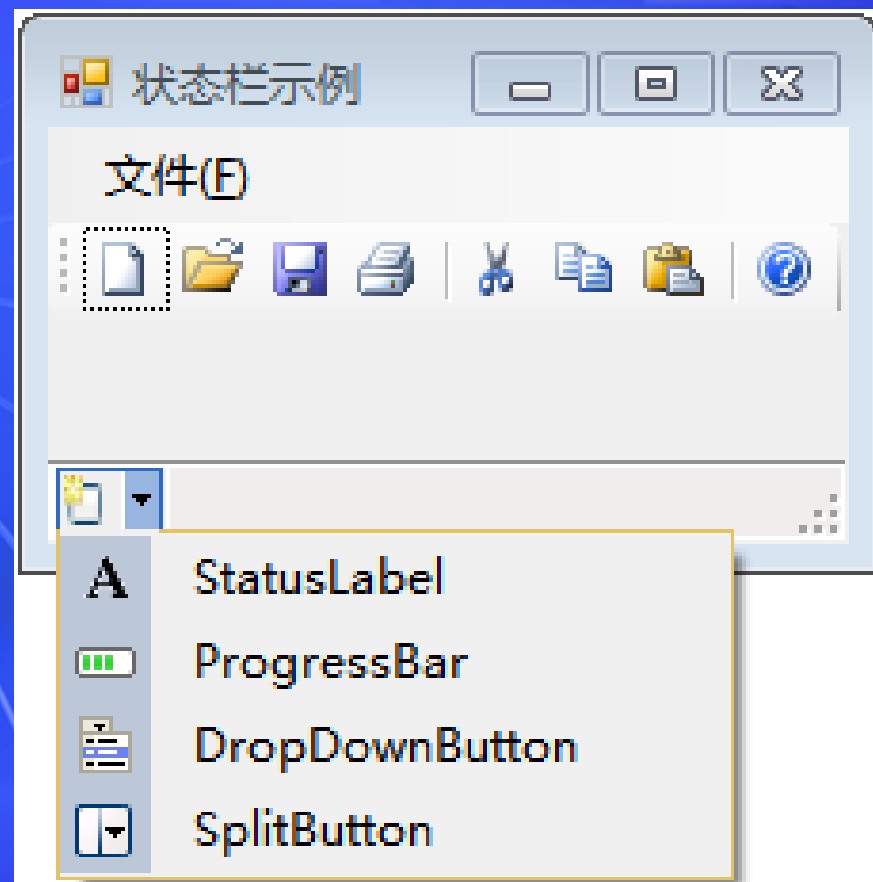
9.7.2 工具栏

- 工具栏实际上可以看成是菜单项的快捷方式，工具栏上的每一个工具项都应有对应的菜单项。工具栏提供了单击访问程序中功能的方式。
- 工具栏上的按钮通常包含一个图片，不包含文本，但它可以既包含图片又包含文本。



9.7.3 状态栏

- 状态栏**StatusStrip**一般位于窗体的底部，用来显示窗体的状态信息。
- 表9-28列出了该控件的常用属性。
- 在状态栏中可添加：**StatusLabel**、**ProgressBar**、**DropDownButton**、**SplitButton**等控件。
- 表9-30列出了这些控件的常用属性。

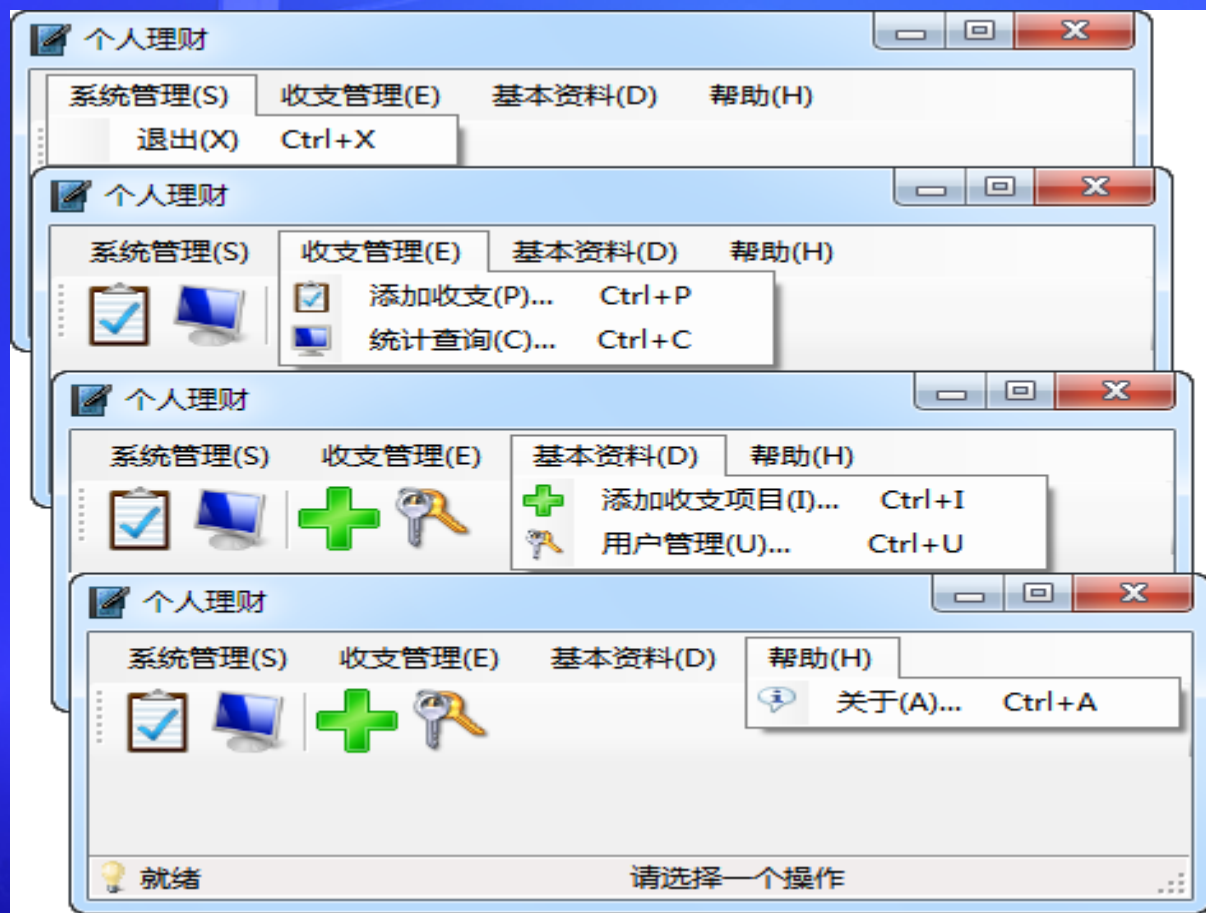


9.7.4 应用实例

【实例9-7】在项目MyAccounting中添加一个窗体，用做个人理财软件的主窗体。

●操作演示

●代码分析



9.8 SDI和MDI应用程序

- 9.8.1 创建SDI应用程序
- 9.8.2 创建MDI应用程序
- 9.8.3 应用实例

返回

9.8.1 创建SDI应用程序

- 单一文档界面（**SDI**）一次只能打开一个窗体，如**Windows**的记事本，一次只能处理一个文档，如果用户要打开第二个文档，就必须打开一个新的**SDI**应用程序实例，它与第一个实例没有关系，对一个实例的任何配置都不会影响第二个实例。
- 在默认情况下，我们创建的窗体都是**SDI**应用程序。

9.8.2 创建MDI应用程序

- 多文档应用程序（MDI）可以在不同的窗口中保存多个已打开的文档，用户可以在任一时间打开多个窗口，例如Word。
- MDI至少包含两个窗口，一个是MDI容器（“主窗口”），另一个在主窗口中显示，称为“子窗口”。

9.8.2 创建MDI应用程序

- 创建MDI应用程序

- 先创建一个Windows应用程序；
- 创建MDI容器窗体：向该应用程序添加一个Windows窗体，并将其**IsMdiContainer**属性设置为true。
- 创建一个MDI子窗体：添加一个Windows窗体，并在打开该窗体时指定其**MdiParent**属性。
- 例如：打开一个名为Form1的窗体作为MDI子窗体的代码为：

```
Form1 frm1 = new Form1();  
frm1.MdiParent = this;  
frm1.Show();
```


9.8.3 应用实例

【实例9-8】 设置个人理财软件为**MDI**应用程序，并在主窗体中打开子窗体。

- 操作演示
- 代码分析（见教材）

9.9 其他控件

●9.9.1 时钟控件 (Timer)

●常用属性

- Interval**: 一个大于1的整数, 用于设置Tick事件周期的大小; 其单位为ms;

- Enabled**: 设置计时器的有效性

●常用方法

- Start**: 启动时钟, 即将其Enabled设置为true

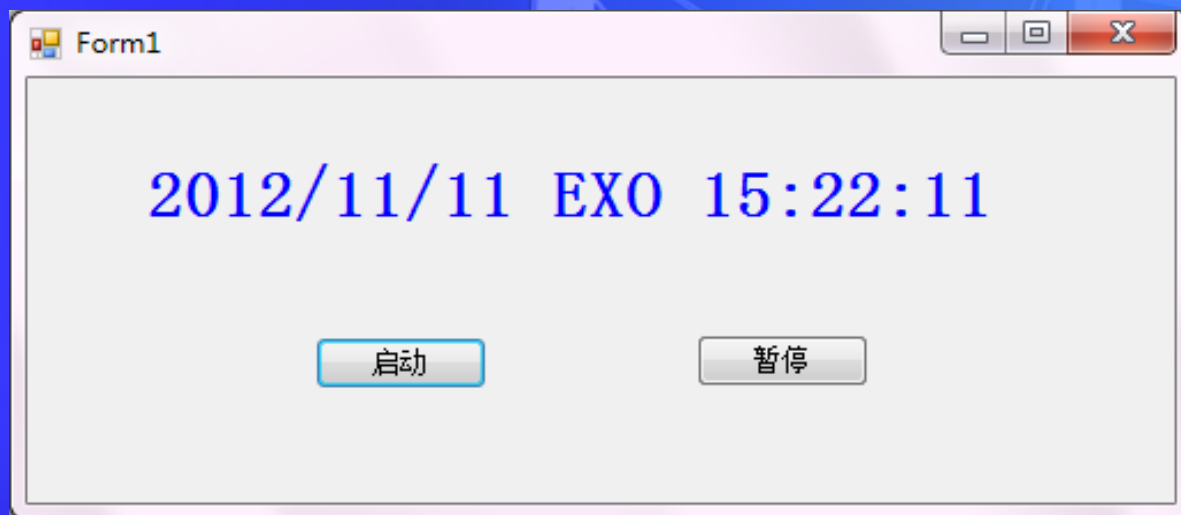
- Stop**: 停止时钟, 即将其Enabled设置为false

●事件: Tick

- 该事件每隔Interval时间执行一次;

9.9.1 时钟控件 (Timer)

- 例9.9.1 在窗体上显示当前系统时间，并使之从左向右移动



9.9.2 滚动条控件

- 水平滚动条、垂直滚动条、轨道条控件
 - 常用属性
 - **Minimum** :滚动范围的最小值;
 - **Maximum** :滚动范围的最小值;
 - **SmallChange**: 最小改变值;
 - **LargeChange** : 最大改变值;
 - **TickStyle** : 刻度位置(轨道条控件的属性)
 - **Value**: 获取设置当前滑块的位置所对应的值;
 - 常用事件: **Scroll**
 - 例见Test9_9的Form2

9.9.3 分割条控件(SplitContainer)

- 提供通过分割条分隔的两个面板，在运行阶段分割条可以移动。控件可以添加到这两个大小可调的面板中。
- 常用属性
 - Orientation**:指定分割条方向是水平或者垂直；默认方向为垂直方向；
 - FixedPanel**：指定某个面板不随容器一起调整大小；
 - Dock**：一般设置为 **DockStyle.Fill**，使该控件覆盖整个容器；
- 常用事件：无
- 例见Test9_9的Form3

习题

- 见教材 第1~5小题



上机实训

- 完成上机实验9





Class Over

Thank you!