

第 3 章

数据验证控件

在 Web 交互环境中,经常需要对用户输入的数据进行有效性验证,而本章要介绍的验证控件就为开发人员提供了一种简单有效的验证途径。验证控件主要包括 RequiredFieldValidator(必需项验证)控件、RangeValidator(范围验证)控件、RegularExpressionValidator(格式验证)控件、CompareValidator(比较验证)控件、CustomValidator(自定义验证)控件和 ValidationSummary(验证总结)控件。

验证控件都位于工具箱的“验证”组内,如图 3.1 所示。



图 3.1 验证控件

3.1 必需项验证控件

必需项验证控件正常情况下在页面中是不显示的,只有在出现了输入错误时,才会在控件中显示出错误提示信息。在页面布局时,一般可将必需项验证控件放置在被验证控件的旁边。

相关知识

(1) 描述

必需项验证控件,用于文本框的非空验证,图标为  RequiredFieldValidator。

(2) 属性

① ControlToValidate 被验证控件的 ID 属性的值,该属性指定验证控件对哪一个控件的输入值进行验证。例如:

```
this.RequiredFieldValidator1.ControlToValidate = txtName;
```

② ErrorMessage 验证不合法时出现的错误信息。

③ IsValid 获取一个布尔值(True 或 False),用于表示验证是否通过。

④ Display 设置错误信息的显示方式(Static/Dynamic/None),默认为 Static。

Static 即静态占用固定空间, 无论是否显示验证控件的出错信息, 验证控件都会占有其文本宽度的位置空间; Dynamic 即动态占用空间, 当验证通过不显示验证信息时不占用空间; None 即不显示错误信息。

⑤ InitialValue 设置或获取验证控件的初始值。

注意: InitialValue 仅指示不希望用户在输入控件中输入的值。当验证执行时, 如果输入控件包含该值, 则其验证失败。



能力目标

能够检验文本框等控件中的数据是否为空。



具体要求

- (1) 能够添加必需项验证控件。
- (2) 会设置必需项验证控件的 ControlToValidate、ErrorMessage、InitialValue 等属性。



实训任务

创建一个模拟的用户登录页面, 要求用户名、身份证号两项内容必须填写, 同时要求在专业下拉列表框中必须选择一个专业, 否则出现错误提示, 如图 3.2 所示; 当页面通过验证时的运行效果如图 3.3 所示。

RequiredFieldValidator控件的应用示例	
用户名: <input type="text"/>	用户名不能为空
身份证号: <input type="text"/>	身份证号不能为空
选择专业: <input type="text" value="--请选择专业--"/>	请选择一个专业
<input type="button" value="提交"/>	

图 3.2 验证未通过时的页面效果图

RequiredFieldValidator控件的应用示例	
用户名: <input type="text" value="张三"/>	
身份证号: <input type="text" value="220604198901131324"/>	
选择专业: <input type="text" value="软件技术专业"/>	
<input type="button" value="提交"/>	
张三你的身份证号是: 220604198901131324 : 你的专业是软件技术专业, 请确认	

图 3.3 验证通过时的页面效果图

【操作步骤】

- (1) 创建并设置页面

创建站点, 并在网站中添加登录页面, 页面设计如图 3.2 所示。

页面各控件的属性设置：用户名文本框 ID 为 txtName；身份证号文本框 ID 为 txtIDCard；专业下拉列表框 DropDownList1 的 ID 为 ddlSpec，专业列表值如图 3.4 所示；按钮 Button1 的 ID 为 btnOk，Text 为“提交”；标签 Label1 的 ID 为 lblInfo。



图 3.4 DropDownList 的列表值

(2) 添加必需项验证控件

从工具箱的“验证”选项卡中找到 RequiredFieldValidator，并将其拖曳到用户名文本框的右侧单元格中，同样的方法为身份证号文本框 (txtIDCard) 和选择专业下拉列表框 (ddlSpec) 添加必需项验证控件，如图 3.5 所示。

RequiredFieldValidator 控件的应用示例		
用户名:	<input type="text"/>	用户名不能为空
身份证号:	<input type="text"/>	身份证号不能为空
选择专业:	--请选择专业--	请选择一个专业
	<input type="button" value="提交"/>	
	[lblInfo]	

图 3.5 页面设计效果

(3) 设置必需项验证控件的属性

必需验证控件的属性设置如表 3.1 所示。

表 3.1 必需项验证控件的属性

控 件	属 性	值	说 明
RequiredFieldValidator1	ControlToValidate	txtName	检验用户名文本框中输入的内容，确保其不能为空
	ErrorMessage	用户名不能为空	
RequiredFieldValidator2	ControlToValidate	txtIDCard	检验身份证号文本框中输入的内容不能为空
	ErrorMessage	身份证号不能为空	
RequiredFieldValidator3	ControlToValidate	ddlSpec	当未选择专业或选择“--请选择专业--”时显示错误信息“请选择一个专业”
	InitialValue	--请选择专业--	
	ErrorMessage	请选择一个专业	

(4) 在浏览器中查看页面

当执行验证时,若验证控件的初始值(InitialValue 属性值)没有改变,则导致验证失败。InitialValue 的初始默认值为空字符串。本程序中,当对下拉列表框控件进行验证时,如果 InitialValue 属性值与下拉列表框的提示项值相同时,则未通过验证;反之则通过验证。



能力测试

1. 具体要求

创建一个虚拟的用户注册页面,页面的用户名、密码、确认密码和电子邮箱不能为空,用户名、密码和确认密码为空时的错误信息分别为“用户名不能为空”、“密码不能为空”、“确认密码不能为空”;当电子邮箱为空时,错误信息要求为一个红色的“*”符号提示,程序设计页面如图 3.6 所示。

说明:重点设置 RequiredFieldValidator 控件的 ControlToValidate 和 ErrorMessage 属性。

用户注册页面	
用户名:	<input type="text"/> 用户名不能为空
密码:	<input type="password"/> 密码不能为空
确认密码:	<input type="password"/> 确认密码不能为空
电子邮箱:	<input type="text"/> *
出生日期:	<input type="text"/>
<input type="button" value="注册"/>	

图 3.6 程序设计页面

2. 自测题

验证输入控件的值不能为空,使用_____验证控件。



知识扩展

验证控件的 Text 属性和 ErrorMessage 属性都用于表示验证失败时显示的出错信息。如果一个验证控件同时设置了这两个属性,那么在验证失败时会以 Text 的值为主,当 Text 属性值为空时,才会显示 ErrorMessage 中的信息。

ErrorMessage 中的值可以在验证总结控件(ValidationSummary)中显示,而 Text 中的值不可以。

3.2 比较验证控件



相关知识

(1) 描述

将输入控件的值与常数或其他输入控件中的值进行比较,以确定这两个值是否由比较运算符(=、!=、<、>等)指定的关系匹配,图标为  CompareValidator。

(2) 属性

- ① ControlToValidate 被验证控件的 ID 值。
- ② ControlToCompare 要比较控件的 ID 值。
- ③ ErrorMessage 验证不合法时出现的错误信息。
- ④ Type 设置比较数据类型,默认值为 string。只有是同一数据类型的数据才能进行比较。
- ⑤ ValueToCompare 设置要比较的值。
- ⑥ Operator 指定要比较的关系。

Operator 即设置验证中使用的比较操作运算符,如 NotEqual 为不等于, GreaterThan 为大于, GreaterThanEqual 为大于或等于,DataTypeCheck 为只对数据类型进行比较,默认值为 Equal 即相等。

能力目标

会使用比较验证控件进行相应数据或相应控件间指定关系的比较。

具体要求

- (1) 添加并设置比较验证控件。
- (2) 能够利用验证控件进行指定控件之间的比较,即会设置 ControlToValidate、ControlToCompare、ErrorMessage 等相关属性。
- (3) 能够利用验证控件进行指定值之间的比较,即会设置 ControlToValidate、ValueToCompare、Operator、Type、ErrorMessage 等相关属性。

实训任务

设计一个模拟的用户注册页面,要求用户名、密码都不能为空,密码必须与确认密码相同,出生日期的格式必须是日期格式,作品数量大于 5。当页面验证通过时,程序运行效果如图 3.7 所示;当页面验证未通过时,程序运行效果如图 3.8 所示。

【操作步骤】

(1) 设计页面

新建一个 ASP.NET 网站,创建会员注册页面,并切换到“设计”视图,向页面插入一个 7 行 3 列的表格,添加 5 个文本框,控件 ID 分别为 txtName、txtPwd、txtRePwd、txtBirthDay、txtNum,1 个 btnOk 按钮,1 个 lblInfo 标签控件,3 个必需项验证控件,3 个比较验证控件,程序页面设计效果如图 3.9 所示。

会员注册	
用户名:	<input type="text" value="李明"/>
密码:	<input type="password"/>
确认密码:	<input type="password"/>
出生日期:	<input type="text" value="1987-3-3"/>
作品数量:	<input type="text" value="6"/>
<input type="button" value="确认"/>	
本页面已经通过了验证!	

图 3.7 通过验证的运行效果

图 3.8 未通过验证的运行效果

图 3.9 注册页面的设计效果

(2) 设置对象属性

页面中必需项验证控件的设置方法与 3.1 节所讲的方式一致,这里不再重复。以下主要介绍比较验证控件属性的设置,如表 3.2 所示。

表 3.2 比较验证控件的属性设置

控 件	属 性	值	说 明
CompareValidator1	ControlToValidate	txtRepwd	设置要验证的控件 ID
	ControlToCompare	txtPwd	设置要进行比较的控件的 ID
	ErrorMessage	密码不一致	显示的出错信息
CompareValidator2	ControlToValidate	txtBirthDay	设置要验证的出生日期文件框的 ID
	ErrorMessage	日期格式应为“1981-3-3”	显示的出错信息
	Operator	DataTypeCheck	对值进行数据类型验证
	Type	Date	进行日期比较
CompareValidator3	ControlToValidate	txtNum	设置要验证的作品总数文本框的 ID
	Operator	GreaterThan	大于
	Type	Integer	整型
	ValueToCompare	5	要比较的值是 5

(3) 编写事件代码

编写的具体代码如下。

```
protected void Page_Load(object sender, EventArgs e)
{
    txtName.Focus();           //页面加载时,用户名文本框得到焦点
}
protected void btnOk_Click(object sender, EventArgs e)
{
    if(IsValid == True)       //页面所有验证都通过
    {
        lblInfo.Text = "本页面已经通过了验证!"; //通过验证后在标签中显示的信息
    }
}
```



能力测试

1. 具体要求

将 3.1 节中的“能力测试”的测试用例打开,利用比较验证控件,设置确认密码文本框中的值必须与密码文本框中的值相同,页面设计效果如图 3.10 所示。

用户注册页面	
用户名:	<input type="text"/> 用户名不能为空
密码:	<input type="password"/> 密码不能为空
确认密码:	<input type="password"/> 确认密码不能为空 确认密码必须与密码保持一致
电子邮箱:	<input type="text"/> *
出生日期:	<input type="text"/>
<input type="button" value="注册"/>	

图 3.10 注册页面的设计效果

2. 自测题

CompareValidator 的 Operator 属性可以提供哪几种比较方式?



知识扩展

将验证控件的 Operator 属性设置为 DataTypeCheck 运算符时,将指定用户输入数据与 Type 属性指定的数据进行比较,若无法将该值转换为 Type 指定的类型,则验证失败。使用 DataTypeCheck 运算符时,将忽略 ControlToCompare 和 ValueToCompare 属性设置。

比较控件,若同时设置了 ControlToCompare 属性和 ValueToCompare 两个属性,则 ControlToCompare 属性优先。

3.3 格式验证控件



相关知识

(1) 描述

格式验证控件用于验证输入的数据格式是否匹配某种特定的规则(正则表达式),例如可以对邮编、电子邮箱、网址等进行格式验证,控件图标为  RegularExpressionValidator。

(2) 属性

- ① ControlToValidate 被验证控件的 ID 属性的值。
- ② ErrorMessage 验证不合法时出现的错误信息。
- ③ ValidationExpression 正则表达式,指定输入内容所要遵循的格式规则。



能力目标

会利用格式验证控件进行数据格式的检验。



具体要求

- (1) 能够添加格式验证控件,并能设置其 ControlToValidate、ValidationExpression、ErrorMessage 等属性。
- (2) 能够利用格式验证控件对电子邮箱、URL、身份证号、邮编等进行格式验证。



实训任务

设计一个模拟的企业注册页面,要求使用格式验证控件(RegularExpressionValidator)对公司的 URL、公司邮箱、公司邮编进行格式验证,使用必需项验证控件对公司名称、公司邮箱进行必需项验证。

【操作步骤】

(1) 设计 Web 页面

新建一个 ASP.NET 网站,将由系统自动创建的 Default.aspx 页面重命名为“CompanyRegister.aspx”,将页面切换到“设计”视图,并向页面中添加 HTML 表格进行页面的布局,适当调整行列数;向表格中添加必要的控件和说明文字;添加 5 个文本框、2 个按钮、4 个 Label、2 个必需项控件、3 个格式验证控件,页面设计效果如图 3.11 所示。

(2) 设置对象属性

参照表 3.3 进行页面各控件属性的设置。

企业注册		
公司名称:	<input type="text"/>	公司名称不能为空
公司URL:	<input type="text"/>	公司URL格式不正确
公司邮箱:	<input type="text"/>	请留下邮箱账号 公司邮箱格式不正确
公司地址:	<input type="text"/>	
公司邮编:	<input type="text"/>	公司邮编格式不正确
	<input type="button" value="确定"/> <input type="button" value="清空"/>	
	[!blComName]	
	[!blComUrl]	
	[!blComEmail]	
	[!blComAddress]	

图 3.11 页面设计效果

表 3.3 页面各控件的属性设置

控 件	属 性	值	说 明
TextBox1~TextBox5	ID	txtComName	公司名称文本框 ID
		txtComUrl	公司 URL 文本框 ID
		txtComEmail	公司邮箱文本框 ID
		txtComAddress	公司地址文本框 ID
		txtPostalcode	公司邮编文本框 ID
Button1、Button2	ID	btnOk btnClear	按钮 1 的 ID 按钮 2 的 ID
	Text	确定 清空	按钮 1 的文本 按钮 2 的文本
RequiredFieldValidator1、RequiredFieldValidator2	ErrorMessage	公司名称必需填写 请留下邮箱账号	必需项验证控件 1、2 的出错信息
	ControlToValidate	txtComName txtComEmail	检验公司名称文本框、公司邮箱文本框是否为空
RegularExpressionValidator1~RegularExpressionValidator3	ErrorMessage	公司 URL 格式不正确 公司邮箱格式不正确 公司邮编格式不正确	格式验证控件中的出错信息
	ControlToValidate	txtComUrl txtComEmail txtPostalCode	格式验证控件 1~3 分别对公司 URL、公司邮箱、公司邮编进行检验
	ValidationExpression	选择“Internet Url” 选择“Internet 电子邮件地址” 选择“中华人民共和国邮政编码”	选择格式验证控件的验证格式即正则表达式
Label1~Label4	ID	lblComName lblComUrl lblComEmail lblComAddress	4 个 Label 各自的 ID

