

交流群

ASP.NET MVC交流群:

QQ群: 1215279(满)

2群: 1214648(满)

3群: 47788243

(加的时候请注明)



总访问量 151158

本月访问量 3593

昨日访问量 429

今日访问量 1

在线人数 5

[我的主页](#) [个人资料](#)[我的闪存](#) [发短消息](#)

<	2008年10月						>
日	一	二	三	四	五	六	
28	29	30	1	2	<a href="#">3</a>	<a href="#">4</a>	
<a href="#">5</a>	6	7	<a href="#">8</a>	<a href="#">9</a>	10	11	
12	<a href="#">13</a>	14	15	16	17	18	

## ASP.NET MVC 入门8、ModelState与数据验证

ViewData有一个ModelState的属性,这是一个类型为ModelStateDictionary的ModelState类型的字典集合。在进行数据验证的时候这个属性是比较有用的。在使用Html.ValidationMessage()的时候,就是从ViewData.ModelState中检测是否有指定的KEY,如果存在,就提示错误信息。例如在前一篇文章[ASP.NET MVC 入门7、Helper与数据的提交与绑定](#)中使用到的UpdateModel方法:

```
try
{
    UpdateModel(BlogSettings.Instance, new[] { "Name", "Description", "PostsPerPage" });
}
catch
{
    return View(BlogSettings.Instance);
}
```

我们在View中使用Html.ValidationMessage(string modelName)来对指定的属性进行验证:

```
<p>
    <label for="Name">Blog的名称</label>
    <%=Html.TextBox("Name", new { id = "Name", @class =
    <%=Html.ValidationMessage("Name")%>
</p>
<p>
    <label for="Description">Blog的简单描述</label>
    <%=Html.TextArea("Description", new { id = "Descri
    <%=Html.ValidationMessage("Description")%>
</p>
<p>
    <label for="PostsPerPage">每页显示的日志数</label>
    <%=Html.TextBox("PostsPerPage", new { id = "Descri
    <%=Html.ValidationMessage("PostsPerPage")%>
</p>
```

Html.ValidationMessage()有几个重载:

```
ValidationMessage(string modelName);
ValidationMessage(string modelName, IDictionary<string, object> htmlAttributes);
ValidationMessage(string modelName, object htmlAttributes);
ValidationMessage(string modelName, string validationMessage);
ValidationMessage(string modelName, string validationMessage, IDictionary<string, object> htmlAttributes);
ValidationMessage(string modelName, string validationMessage, object htmlAttributes);
ValidationSummary();
ValidationSummary(IDictionary<string, object> htmlAttributes);
ValidationSummary(object htmlAttributes);
```

其中ValidationSummary()是用于显示全部的验证信息的。跟ASP.NET里面的ValidationSummary验证控件差不多。

我们测试一下/Admin/Setting页面:

[19](#) [20](#) [21](#) [22](#) [23](#) [24](#) [25](#)  
[26](#) [27](#) [28](#) [29](#) [30](#) [31](#) [1](#)  
[2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#)

## 最新闪存

python为啥就是要用缩排而不用大括号来标记代码段呢？难道用C#写代码就不缩排的、不美观的？当你在文章中看到一段python的格式乱了的代码，那个郁闷啊~~ 10-26 15:31  
Python? Python! 09-20 16:32  
更多闪存...

## 搜索

## 常用链接

[我的随笔](#)  
[我的空间](#)  
[我的短信](#)  
[我的评论](#)  
[更多链接](#)

## 我参加的小组

[软件开发家园](#)  
[jQuery](#)  
[ASP.NET](#)  
[创业交流](#)  
[设计模式](#)  
[Asp.Net MVC](#)  
[读书\(Books\)](#)  
[.NET 3.x](#)  
[AJAX](#)  
[WEB程序安全小组](#)

## 我参与的团队

[大学生联盟\(0/0\)](#)  
[ASP.NET MVC团队\(0/0\)](#)

## 我的标签

[aspnetmvc\(30\)](#)  
[ASP.NET MVC\(21\)](#)  
[MVC\(16\)](#)  
[javascript\(13\)](#)

### 每页显示的日志数

10d

The value '10d' is invalid for property 'PostsPerPage'.

在用UpdateModel方法更新BlogSettings.Instance.PostsPerPage的时候，当我们如图所示填写“10d”的时候，由于PostsPerPage为整型的，所以UpdateModel方法就会出错，同时会往ViewData.ModelState添加相应的错误信息，从而Html.ValidationMessage()方法就可以从ViewData.ModelState中检测到错误并提示。同时Html.ValidationMessage()方法会为出错的属性的输入框添加一个名为“input-validation-error”的CSS类，同时后面的提示信息的CSS类名为“field-validation-error”：

```
<label for="PostsPerPage">每页显示的日志数</label>
<input class="input-validation-error" required id="Description" name="PostsPerPage" type="text"
<span class="field-validation-error">The value '10d' is invalid for property 'PostsPerPage'.</span>
```

CSS类的样式是可以由我们自己自由定义的。如上图的红色高亮显示。

好，下面我们来实现发表新随笔的功能。我们先写一个提供用户输入随笔内容的表单页面：

```
<p>
  <label for="Title">标题</label>
  <%=Html.TextBox("Title", new { id = "Title", @class = "required" })%>
  <%=Html.ValidationMessage("Title")%>
</p>
<p>
  <label for="Content">内容</label>
  <%=Html.TextArea("Content")%>
  <%=Html.ValidationMessage("Content")%>
</p>
<p>
  <label for="Slug">URL地址别名(如果为空则和标题同名)</label>
  <%=Html.TextBox("Slug", new { id = "Slug", @class = "required" })%>
  <%=Html.ValidationMessage("Slug")%>
</p>
```

然后我们对用户提交过来的数据进行保存：

```
[AcceptVerbs("POST"), ActionName("NewPost")]
public ActionResult SaveNewPost(FormCollection form)
{
    Post post = new Post();

    try
    {
        UpdateModel(post, new[] { "Title", "Content", "Slug" });
    }
    catch
    {
        return View(post);
    }
}
```

[jQuery\(11\)](#)  
[CSS\(7\)](#)  
[asp.net mvc 教程\(4\)](#)  
[Oxite\(4\)](#)  
[python\(4\)](#)  
[翻译\(3\)](#)  
[更多](#)

随笔分类

[.NET\(7\)](#) [HTML](#)  
[CMS\(2\)](#) [HTML](#)  
[DB4o\(1\)](#) [HTML](#)  
[Django\(1\)](#) [HTML](#)  
[Enterprise Library 3.1\(1\)](#) [HTML](#)  
[FLEX\(1\)](#) [HTML](#)  
[JavaScript\(28\)](#) [HTML](#)  
[jQuery\(19\)](#) [HTML](#)  
[LINQ\(4\)](#) [HTML](#)  
[MVC\(40\)](#) [HTML](#)  
[Python\(8\)](#) [HTML](#)  
[Silverlight\(2\)](#) [HTML](#)  
[Unity\(1\)](#) [HTML](#)  
[VS2008 & .NET 3.5\(3\)](#) [HTML](#)  
[WCF](#) [HTML](#)  
[WEB标准 & CSS\(10\)](#) [HTML](#)  
[WPF](#) [HTML](#)  
[翻译\(10\)](#) [HTML](#)  
[敏捷开发](#) [HTML](#)  
[心情笔记\(3\)](#) [HTML](#)

随笔档案

[2009年12月 \(1\)](#)  
[2009年11月 \(3\)](#)  
[2009年9月 \(3\)](#)  
[2009年7月 \(1\)](#)  
[2009年6月 \(1\)](#)  
[2009年5月 \(2\)](#)  
[2009年4月 \(1\)](#)  
[2009年3月 \(5\)](#)  
[2009年2月 \(2\)](#)

```
post.Save();  
return ShowMsg(new List<string>() { "发表新随笔成功" });  
}
```

由于这三个值都是字符串类型，所以如果值为空的话，UpdateModel也是不会出错的，而我们的Title和Content是不允许为空的，或者我们想我们的Slug的长度不能超过100，也就是需要我们有自己的业务规则。这时候我们或许会这样写：

```
try  
{  
    UpdateModel(post, new[] { "Title", "Content", "Slug" });  
}  
catch  
{  
    return View(post);  
}  
  
if (string.IsNullOrEmpty(post.Title))  
{  
    ViewData.ModelState.AddModelError("Title", post.Title, "标题不能为空");  
}  
if (string.IsNullOrEmpty(post.Content))  
{  
    ViewData.ModelState.AddModelError("Content", post.Content, "内容不能为空");  
}  
  
if (!ViewData.ModelState.IsValid)  
{  
    return View(post);  
}
```

ViewData.ModelState提供了一个AddModelError的方法，方便我们添加验证失败的信息。我们可以如上代码这样进行对象的业务规则验证，但是一旦业务规则多了，这样的代码是非常壮观的，而且不好控制。那么我们该怎么更好的进行业务规则的验证呢？得益于BlogEngine.Net的良好架构，我们可以很轻松的完成这一点。

首先，让我们修改一下BlogEngine.Core里面BusinessBase的代码。我们前面说过，BusinessBase实现了IDataErrorInfo接口，该接口有个索引器，导致ViewData.Eval()方法调用时搜索索引器的值时返回String.Empty而使ViewData.Eval()认为是找到值了，从而失效。

- 2008年12月 (4)
- 2008年11月 (11)
- 2008年10月 (13)
- 2008年9月 (2)
- 2008年8月 (4)
- 2008年7月 (5)
- 2008年6月 (1)
- 2008年5月 (4)
- 2008年4月 (15)
- 2008年3月 (22)
- 2008年2月 (4)
- 2008年1月 (6)
- 2007年12月 (1)
- 2007年10月 (1)

.NET 资源

我的好友

- <扫地僧>[XML](#)
- Jannock's Blog[XML](#)
- leven(神鱼)[XML](#)
- MK2[XML](#)
- TNT2(SZW)[XML](#)
- thxnz(V9)[XML](#)
- 太阳的博客[XML](#)
- 重典 In cnBlog[XML](#)
- 紫色永恒(老f)[XML](#)

积分与排名

积分 - 385277

```
#region IDataErrorInfo Members
/// <summary>
/// Gets an error message indicating what is wrong
/// </summary>
/// <returns>An error message indicating what is w
public string Error
{
    get { return validationMessage; }
}

/// <summary>
/// Gets the <see cref="System.String"/> with the
/// </summary>
public string this[string columnName]
{
    get
    {
        if (_BrokenRules.Containskey(columnName))
            return _BrokenRules[columnName];

        return string.Empty;
    }
}
```

我们可以将return string.Empty修改为return null。但我们这里并不需要用到这个接口，所以我们把该接口去掉,并把相应的代码注释了。然后我们再暴露一个BrokenRules的属性，用于返回当前的所有破坏性业务规则(红框部分代码为我们添加的)：

```
private StringDictionary _BrokenRules = new StringDictionary();
/// <summary>
/// 获取所有的破坏性规则。
/// 在获取前请用IsValid进行判断。
/// </summary>
public StringDictionary BrokenRules
{
    get
    {
        return _BrokenRules;
    }
}
```

BusinessBase提供了一个抽象的ValidationRules方法，用于在业务类重写这个方法往里面添加验证规则(具体请看BusinessBase的Validation节)。

Validation

我们在Post类中重写这个方法来添加验证规则：

```
protected override void validationRules()
{
    AddRule("Title", "标题不能为空", string.IsNullOrEmpty(Title));
    AddRule("Content", "内容不能为空", string.IsNullOrEmpty(Content));
    //更多....
}
```

然后我们可以在Controller的Action中很优雅的书写我们的代码来进行业务规则的验证：

```
[AcceptVerbs("POST"), ActionName("NewPost")]
public ActionResult SaveNewPost(FormCollection form)
```

## 最新评论 XML

## 1. Re:jQuery Ajax 全解析

写的不错，相当不错，核心的东西都点到了。

--啊.辉

## 2. Re:关于ASP.NET MVC的业务逻辑验证(validation)

看了一下，还不错，只是目前似乎只支持castle跟nhibernate，不支持Entityframework。

--深山老林

## 3. Re:使用XML文件来动态配置ASP.NET MVC的Route规则

谢谢，看了，很实用。

--DsJian

## 4. Re:GoogleAppEngineLauncher不能启动的解决方法

不错 我也是这样的问题

--雨中漫步的太阳

## 5. Re:GoogleAppEngineLauncher不能启动的解决方法

好的 我也出现这个问题 按照你的方法解决点 真的感谢！

--http://www.sc-seo.com

## 阅读排行榜

## 1. jQuery Ajax 全解析(67381)

## 2. 广州.NET俱乐部的活动(29526)

## 3. jQuery实例：图片展示效果(25005)

## 4. ASP.NET MVC 入门系列教程(21318)

## 5. ASP.NET MVC 入门1、简介(11245)

## 评论排行榜

## 1. ASP.NET MVC 1.0 发布了(90)

## 2. jQuery Ajax 全解析(77)

## 3. ASP.NET MVC 入门1、简介(62)

## 4. jQuery实例：图片展示效果(54)

## 5. 变态的VS(48)

## 60天内阅读排行

## 1. AJAX的跨域与JSONP(另送一个为文章自动添加短址的功能)(1459)

```
{
    Post post = new Post();

    try
    {
        UpdateModel(post, new[] { "Title", "Content", "Slug" });
    }
    catch
    {
        return View(post);
    }

    if (!post.IsValid)
    {
        foreach (string key in post.BrokenRules.Keys)
        {
            ViewData.ModelState.AddModelError(key, form[key], post.BrokenRules[key]);
        }
        return View(post);
    }

    post.Save();
    return ShowMsg(new List<string>() { "发表新随笔成功" });
}
```

我们注意到上面的Action中用到了一个FormCollection 的参数，这个参数系统会自动将Form提交过来的全部表单值(Request.Form)赋给它的。客户端验证可以用jQuery的验证插件来，这里就不罗嗦了。

暂时就写这么多吧，想到什么再补充。Enjoy! Post by Q.Lee.lulu。

本文的Blog程序示例代码： [4mvcBlog\\_8.rar](#)

更多内容，请关注<http://blog.51mvc.com/> 和 <http://bbs.51mvc.com/>



作者：QLeelulu

出处： <http://QLeelulu.cnblogs.com/>

本文版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章页面明显位置给出原文连接，否则保留追究法律责任的权利

Tag标签： [ASP.NET MVC](#), [aspnetmvc](#)

0

0

(请您对文章做出评价)

« 上一篇： [ASP.NET MVC 入门系列教程](#)

- 2. 访问需要HTTP Basic Authentication认证的资源的各种语言的实现(213)
- 3. GoogleAppEngineLauncher不能启动的解决方法(124)
- 4. SVN导出两个版本之间的差异文件(70)

» 下一篇: [ASP.NET MVC 入门9、Action Filter 与 内置的Filter实现\(介绍\)](#)

posted on 2008-10-08 08:25 [Q.Lee.lulu](#) 阅读(3685) 评论(4) 编辑 收藏 网摘 所属分类: MVC

评论:

- #1楼 2009-05-02 14:57 | [一人行](#)

博主技术不错!接触ASP.NET MVC多久了??

[回复](#) [引用](#) [查看](#)
- #2楼 2009-05-11 20:42 | [王昊\\_](#)

俺还是来学习的。

就是学习一下

[回复](#) [引用](#) [查看](#)
- #3楼 2009-09-12 16:55 | [破孩](#)

不错

[回复](#) [引用](#) [查看](#)
- #4楼 2009-09-17 14:27 | [混世者](#)

越往后越难，从评论的人数就能看出来。博主好像不是写教程吧，跳跃性忒大了，跟不上呀

[回复](#) [引用](#) [查看](#)

[刷新评论列表](#) [刷新页面](#) [返回页首](#)

[发表评论](#) [程序员招聘](#) [求职](#) [IT新闻](#)

昵称:  [\[登录\]](#) [\[注册\]](#)

主页:

邮箱:  (仅博主可见)

评论内容:

[登录](#) [注册](#)

[使用Ctrl+Enter键快速提交评论]

网站导航: [博客园首页](#) | [个人主页](#) | [闪存](#) | [IT新闻](#) | [程序员招聘](#) | [社区](#) | [博问](#) | [网摘](#)



IT新闻:

- [VeryCD](#)网站挂出公告, 提示明日中午可恢复
- 云计算潮流来袭 微软、IBM等携手共进退
- 飞信开通异网? 移动回复不知情
- [jQuery](#)荣获《.NET Magazine》09年度开源应用大奖
- 微软: Windows磁盘加密可放心使用

[China-pub](#) 计算机图书网上专卖店! 6.5万品种2-8折!

[China-Pub](#) 计算机绝版图书按需印刷服务

链接: [切换模板](#)

相关搜索:

[ASP.NET MVC](#) [aspnetmvc](#) [MVC](#)

在知识库中查看:

[ASP.NET MVC 入门8、ModelState与数据验证](#)

Powered by: [博客园](#) 模板提供: [沪江博客](#) Copyright ©2009 Q.Lee.lulu