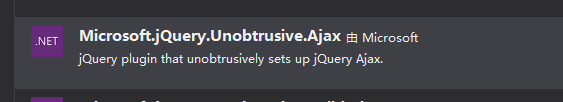
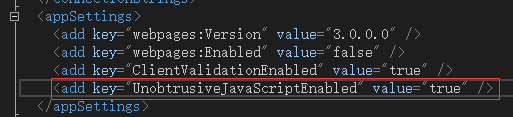
2017-12-18日更新，添加了表单异步提交支持，以下是用法简介。

## 准备工作（已完成，写出来是让大家了解一下过程方便以后自己搞）

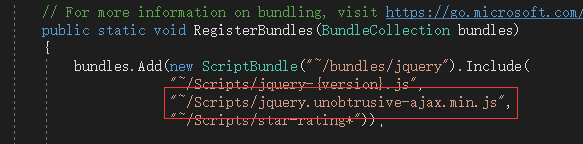
1. 在nuget中添加Unobtrusive-Ajax功能：



1. 在web.config中启用AJAX支持：



1. 然后在BundleConfig.cs中添加脚本文件引用：



准备工作完成。

## 客户端代码写法

1. 第一种，直接写HTML：

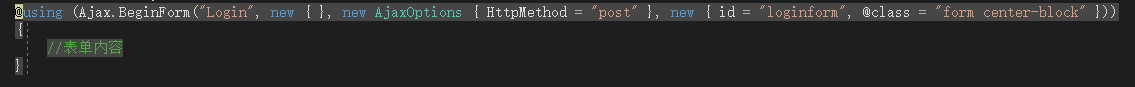


<form id="loginform" class="form center-block" action="/Account/Login" method="post" data-ajax="true" data-ajax-method="post">

</form>

重点在于method="post" data-ajax="true" data-ajax-method="post" 这三个特性，一定要正确。

1. 第二种，用Razor语法：



@using (Ajax.BeginForm("Login", new { }, new AjaxOptions { HttpMethod = "post" }, new { id = "loginform", @class = "form center-block" }))

{

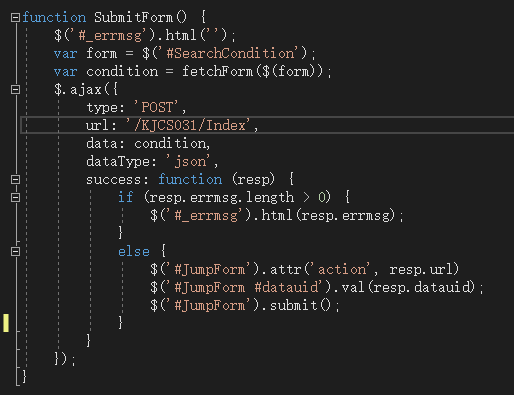
//表单内容

}

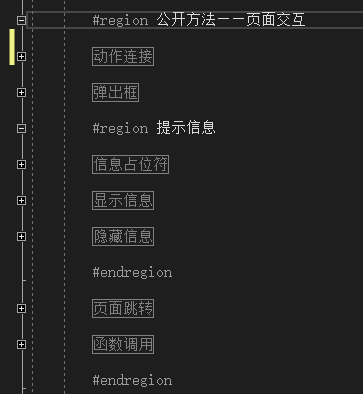
注意写上HttpMethod = "post" ，还有注意@class的写法就行了。

## 重点：控制器写法

首先表单都是默认通过<input type=”subbmit”>或者<button type=”subbmit”>按钮来提交，提交的目标是form的action属性，这部分可以自定义，此处就不详谈了。现在因为使用了ajax，所以subbmit实际上相当于被重写成了类似这样（这是我另一个项目里用到的写法，就是因为这种写法太令人讨厌我才决定不在我们项目里用了）：



现在由于使用了nuget包，所以上面这种代码就会由系统自动替我们生成，我们要做的就是修改Controller的返回数据。这里我打包了以下几个类型，都在Global里：



除了弹出框有待完善外，其余都测试可用。下面以我自己的AccountController为例讲一下用法。

先放示意图：



首先是提示信息的用法，我们要在需要动态显示提示信息的位置，使用Razor语法添加一个消息占位符：





意义见注释，已经写的很清楚了。

然后在表单Post回Controller里时，经过判断如果需要显示消息

注意一定要写this，正常情况下，写过this之后可以直接将ShowMessage点出来（这是因为我函数声明时是用的扩展函数声明法，我觉得这部分也比较符合“对扩展开放，对修改关闭”的原则，以后课上可以讲一讲，然而我不想讲）。

ShowMessage有很多重载，注释都有写，自己看。

如果在一次post时要显示多条消息，可以将多个ShowMessage的返回值用函数连接起来。这个函数也有两个重载，可以写成这种，也可以把参数做成一个IEumerable<JavaScriptResult>（我就是这么用的）：。

除了消息提示外，Global还提供了别的常用函数，比如当用户密码正确时，会让页面跳转：

。

因为我用了<iframe>的网页嵌套，所以要使用父页面跳转。正常的话用this.RedirectTo()就行了。



还有些特殊情况，需要在返回时（不是跳转）执行页面上已经有的js函数，则可以调用。用法见注释，注意大小写。