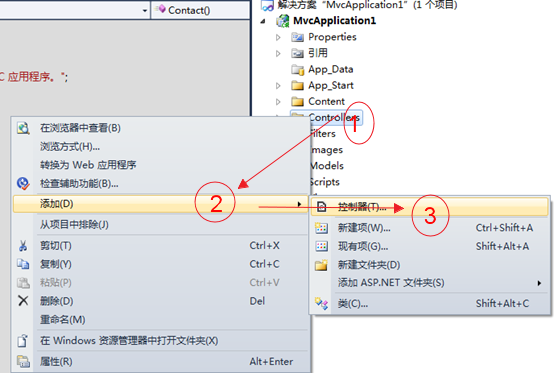
[**学习ASP.NET MVC(二)——我的第一个ASP.NET MVC 控制器**](http://www.cnblogs.com/chillsrc/p/3708995.html)

MVC全称是Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，是一种软件设计典范，用一种业务逻辑和数据显示分离的方法组织代码，将业务逻辑被聚集到一个部件里面，在界面和用户围绕数据的交互能被改进和个性化定制的同时而不需要重新编写业务逻辑。MVC是一种具有可测试性和易于维护的应用程序开发模式。

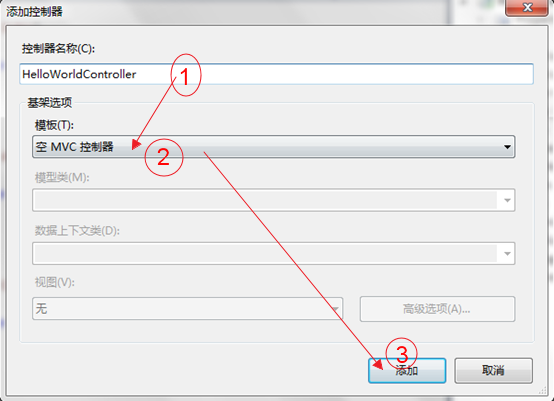
**基于MVC的应用程序应该包含以下内容：**•模型：表示应用程序核心，是该应用程序中用来进行校验、执行业务逻辑的数据类，业务实体对象。   
•视图：表示显示数据，是应用程序中处理数据显示的部分，是该应用程序动态生成HTML响应的模板文件。   
•控制器：表示处理输入，是应用程序中处理用户交互的部分，是该应用程序处理浏览器传入的请求，从数据库中读取数据，然后指定视图模板返回给浏览器的响应。

**创建我的第一个控制器类**

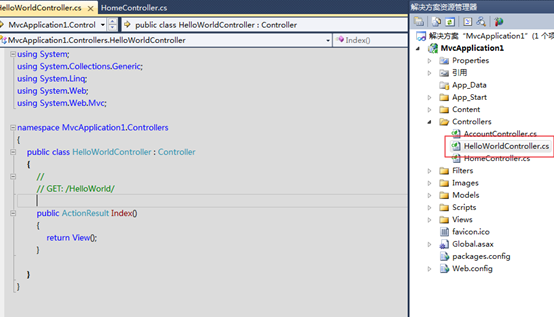
首先，在Visual stdio中的“解决方案资源管理器”中，选中Controllers”文件夹，单击右键，在弹出菜单中选择“添加-->控制器”。如下图。



其次，Visual Studio会弹出的“添加控制器”对话框，在对话框的“控制器名称”文本框中输入“HelloWorldController”，在“模板”下拉框中选择默认模板“空MVC控制器”，然后单击“添加”。如下图。



第三，当鼠标点击“添加”之后，Visual Studio会创建一个.cs文件 ，文件名为HelloWorldController.cs。如下图。



第四，用下面的代码替换文件中的默认代码。

[复制代码](javascript:void(0);)

public class HelloWorldController : Controller

{

//

// GET: /HelloWorld/

public string Index()

{

return "这是我的一个欢迎您来到ASP.NET MVC世界的 <b>控制器</b>的默认方法 ！！！";

}

//

// GET: /HelloWorld/Welcome/

public string Welcome()

{

return "这是一个欢迎您来到ASP.NET MVC世界的控制器中的指定方法...";

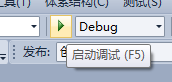
}

}

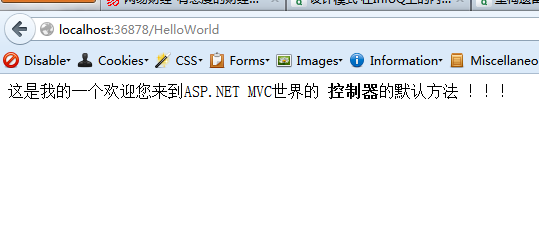
[复制代码](javascript:void(0);)

 我的这个控制器中的Index与Welcom方法都只是返回一个字符串。

第五，在填写完代码之后，运行应用程序（按F5或Ctrl+ F5）。如下图。



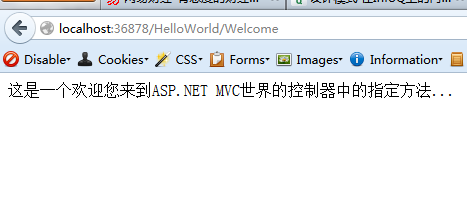
Visual Studio会启动IIS Express，同时运行Visual Studio中Web应用程序，然后Visual Studio会启动默认浏览器并打开应用程序的首页。在已启动的浏览器的地址栏中的地址后面加上“HelloWorld” （例如，如下图中，它的http://localhost:36878/HelloWorld）。浏览器就会去调用新的页面，页面中显示的内容如下图。由于我在Index方法中，直接返回的是字符串，所以浏览器中显示的就是我在代码中所写的一串中文。



ASP.NET MVC根据传入的不同的URL去调用不同的控制器类以及控制器类中的不同的方法。ASP.NET MVC默认的URL路由格式如下：

/[Controller]/[ActionName]/[Parameters]

以上格式的URL，其中第一部分决定了由什么控制器类来执行，就本例来说HelloWorld映射到HelloWorldController类。 URL的第二部分决定了要执行的控制器类中的哪个方法，就本例来说HelloWorld/Index会让HelloWorldController类的Index方法执行。请注意，Index是ASP.NET MVC控制器类的默认方法，如果在没有指定具体方法的情况下使用，将调用控制器上的默认方法（Index）。   
在浏览器地址栏中输入http://localhost:36878/HelloWorld/Welcome，浏览器会返回以下字符串“这是一个欢迎你来到ASP.NET MVC世界的控制器的指定方法......”。如下图。



第六，我要通过URL传递一些参数给控制器，来看我是如何修改的（例如，/HelloWorld/Welcome?name=dotnet菜园&numtimes=9）。下面的方法是用参数来改变应用程序的输出内容。代码如下所示。请注意，该代码使用C＃可选参数的功能，当numTimes没有参数传递过来时，numTimes参数默认为1。

[复制代码](javascript:void(0);)

public string Welcome(string name, int numTimes = 1)

{

return HttpUtility.HtmlEncode("欢迎 " + name + string.Format("，第{0}次来到ASP.NET MVC的世界！ " , numTimes));

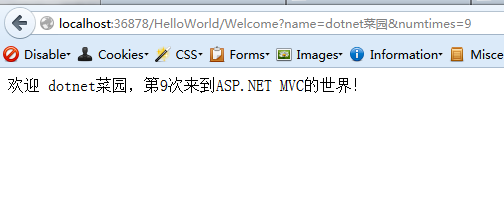
}

[复制代码](javascript:void(0);)

直接运行程序，会报一个以下错误信息。如下图。原来在ASP.NET MVC的控制器类中默认是不支持方法重载的。如果希望方法重载要使用httpverbs或是重写actionexcuting方法



我们把Welcome方法注释掉，然后按F5来运行应用程序并在浏览器中输入http://localhost:36878//HelloWorld/Welcome?name=dotnet菜园&numtimes=9。您可以输入不同的name和numtimes值。ASP.NET MVC会自动从URL请求中获取相应的参数值。



在这个例子中，控制器一直在做MVC中的“VC”部分 - 也就是，视图和控制器的工作。本示例中的控制器直接返回一个HTML字符串。按照MVC的理论来说，控制器不应该直接返回HTML字符串，因为这样做会让代码变得非常繁琐，具体的呈现方式应该由视图去做，而不应该由控制器去做，在下一次的文章中再来介绍视图。