

第12章 ASP.NET MVC

本章讲述的主要内容

12.1 ASP.NET MVC概述

12.2 ASP.NET MVC路由

12.3 ASP.NET MVC控制器

12.4 ASP.NET MVC视图和模型

12.5 实训 设计一个用户管理程序

12.1 ASP.NET MVC概述

ASP.NET MVC是Microsoft推出的新型ASP.NET应用程序开发模式，MVC是模型（Model）、视图（View）和控制器（Controller）3个单词的缩写，它意味着这种开发模式将一个应用程序分为模型、视图和控制器3个组成部分。Visual Studio 2015中内置的MVC版本为ASP.NET MVC 5。

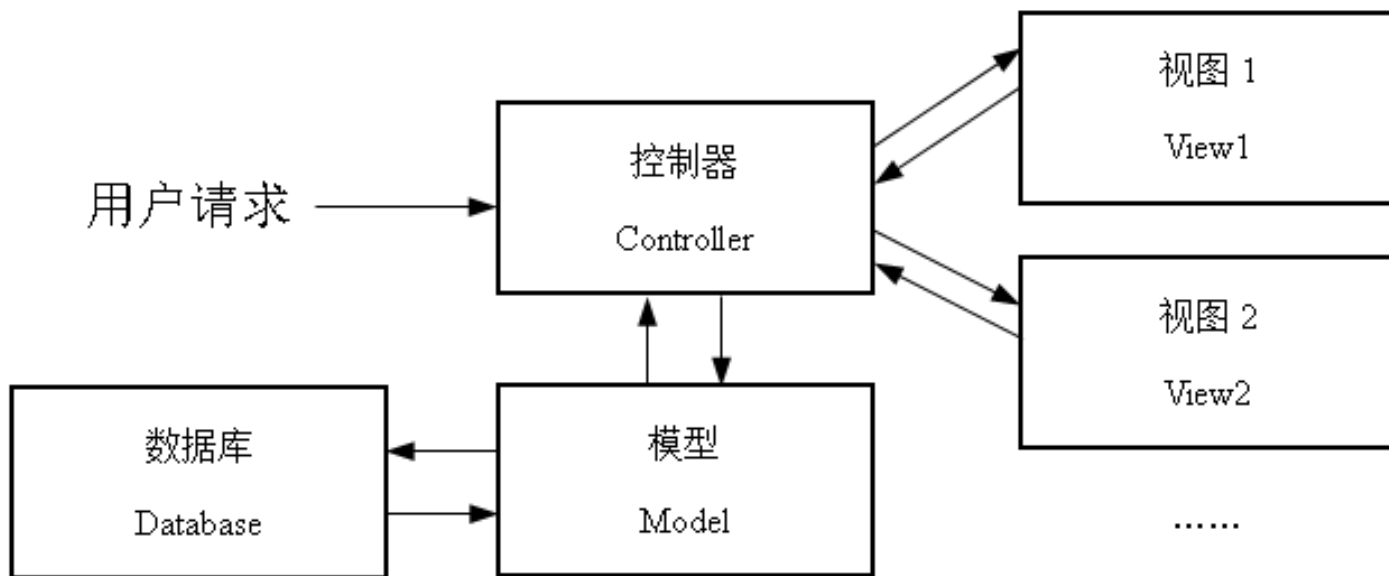
12.1.1 MVC的构成

1. 模型（Model）
2. 视图（View）
3. 控制器（Controller）

12.1 ASP.NET MVC概述

12.1.1 MVC的构成

4. MVC的工作原理



12.1 ASP.NET MVC概述

12.1.2 Razor语法

ASP.NET MVC 5默认使用Razor语法来设计动态网页。由于Razor是通过System.Web.Mvc下的RazorViewEngine类来实现的，所以也称为“Razor视图引擎”。在ASP.NET MVC 5项目中使用的Razor视图引擎为基于Razor语法的Web网页第3版（也称为ASP.NET Web Pages 3）。

1. Razor的特点

- ① 利用Razor可以在视图（.cshtml）文件中混合使用C#和HTML、JavaScript、jQuery等服务器端和客户端语言。
- ② Razor有非常简洁的语法格式。在视图文件中只需要使用一个“@”符号就可以表示C#语句块或内联表达式。
- ③ Razor能自动对网页中输入的字符串进行HTML编码，可以有效地防范客户端脚本攻击。

12.1 ASP.NET MVC概述

12.1.2 Razor语法

2. 使用Razor

(1) 使用@标记

@标记表示其后跟随的是C#代码的一个内联表达式、单行语句或一个语句块。

(2) 使用@Html.Raw()方法

对于HTML代码中出现的@符号本身或双引号、单引号等特殊符号，可以通过@Html.Raw()方法进行转义。

12.1 ASP.NET MVC概述

12.1.3 创建MVC应用程序

参阅源代码： ex12-1

12.1.4 ASP.NET MVC项目的组成

1. 应用程序信息文件

① Properties文件夹：项目属性文件夹，其中包含有一个名为“AssemblyInfo.cs”，用于保存程序集信息（如，名称、版本、版权等）的文件。该文件由项目属性窗口中各选项生成，一般不需要手工编辑。

② 引用文件夹（bin）：用于存放系统运行所必须的外部引用（各类.dll文件）。

12.1 ASP.NET MVC概述

12.1.4 ASP.NET MVC项目的组成

2. 项目文件夹

① App_Data文件夹：用于存放项目中使用的数据文件或数据库文件，作用及功能与Web Forms项目中App_Data文件夹相同。

② App_Start文件夹：用于存放配置类的代码文件，默认包含有一个名为“RouteConfig.cs”的路由配置文件，也就是ASP.NET MVC项目的“路由表”。

③ Controllers文件夹：用于存放项目中所有控制器文件的文件夹。

④ Models文件夹：用于存放项目中所有模型文件的文件夹。

⑤ Views文件夹：用于存放项目中所有视图的文件夹。每个视图以对应控制器名为子文件夹名，视图文件的扩展名为.cshtml。

12.1 ASP.NET MVC概述

12.1.4 ASP.NET MVC项目的组成

3. 配置文件

① ApplicationInsights.config: 监视配置文件, 用于设置如何监视应用程序的运行情况。

② Global.asax: 全局配置文件, 用于设置全局URL路由的默认值, 在应用程序启动时可通过该文件执行一些特殊操作。

③ packages.config: 项目中附加的软件包配置文件, 由系统自动生成和管理, 一般不需要程序员编写和修改。

④ Web.config: XML格式的网站或文件夹的配置文件。其作用与Web Forms项目中的web.config文件相同。

12.2 ASP.NET MVC路由

MVC路由是指用来将用户请求与控制器对应的处理方法进行关联的设置，它也是除了模型、视图和控制器外最为重要的一个概念，它分为入站和出站路由两种类型。在前面几章介绍过的Web Forms开发模式中所有的用户请求都会指向一个物理存在的文件（如.aspx、html等），而在MVC中用户请求则一律指向控制器中某个操作方法，并由该方法决定下一步的操作（打开视图、获取数据等）。

12.2.1 MVC路由

1. 注册路由

所谓“注册路由”就是通过存放在App_Start文件夹下的RouteConfig.cs所表述的RouteConfig类中定义的RegisterRoutes()方法，将URL、控制器及对应的Action关联起来

12.2 ASP.NET MVC路由

12.2.1 MVC路由

2. 从URL向控制器传递数据

MVC程序中可以通过Controller类的RouteData属性或查询字符串Request.QueryString[]进行页面间的数据传递。

参阅源代码：ex12-2

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/578114103034006051>