

《ASP.NET动态网页设计项目教程(第三版)》

科学出版社

主编：郭建东

项目3 页面导航与视图切换





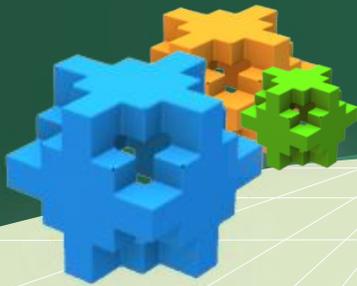
教学目标

❖ 知识教学目标

- 使用CSS+DIV设置一级横向导航和竖向导航
- 使用Menu控件建立一级横向导航和竖向导航
- 使用Menu控件建立二级横向导航
- 使用TreeView控件建立树状导航
- 使用MultiView控件实现视图切换

❖ 技能培养目标

- 根据页面设计需求，使用多种方式制作横向一级和多级导航
- 根据页面设计需求，使用多种方式制作竖向一级和多级导航
- 根据页面设计需要，制作多视图切换模块



目录

1

任务3.1 使用CSS+DIV创建导航

2

任务3.2 使用Menu控件建立网站导航

3

任务3.3 使用TreeView建立树状导航

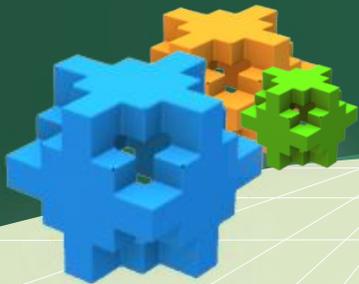
4

任务3.4 使用MultiView实现多视图切换

5

任务3.5 站点地图应用

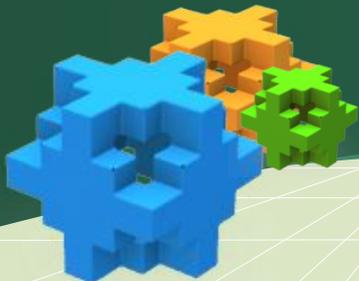
6



任务3.1 使用CSS+DIV创建导航

任务描述:

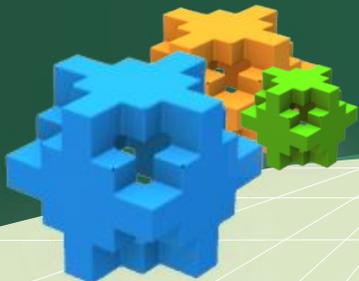
本任务基于CSS+DIV完成横向导航和竖向导航的制作，导航块内容可以是列表项，也可以采用链接文字，通过外部样式使得导航块内容按需要的方式横向或竖向显示。



任务3.1 使用CSS+DIV创建导航

关键知识： 导航样式

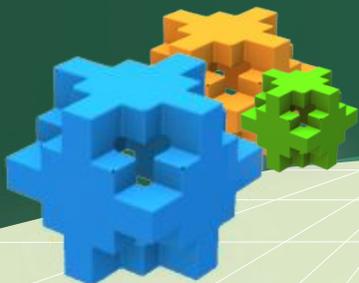
- ❖ 浮动设置：采用ul-li列表项的形式制作横向导航时，关键点是需要设置每一个li列表项由默认的竖向排列转成横向排列，通过CSS样式控制这种转向，需要设置li样式向左或向右浮动。
- ❖ 复合链接样式的设置：文字链接有默认的连接样式、鼠标经过样式和访问过后的样式，如果需要改变默认的样式，则需要对链接样式进行复合样式定义，两个样式如果一样，可用逗号隔开。
- ❖ 外部样式的应用：本项目所有样式均保存在css文件夹下的样式文件中，需要在网页文件源代码视图的<head>标签内添加对外部样式文件的引用。



任务3.1 使用CSS+DIV创建导航

任务实施：使用ul列表制作横向导航

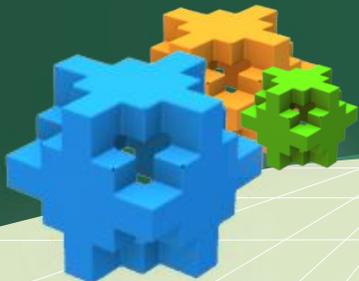
```
<div class="nav1">+  
  <ul>+  
    <li><a href="task1.aspx">首页<h5>Home</h5></a></li>+  
    <li><a href="#">产品展示<h5>products</h5></a></li>+  
    <li><a href="#">客户案例<h5>Solutions</h5></a></li>+  
    <li><a href="#">新闻中心<h5>News Center</h5></a></li>+  
    <li><a href="#">服务支持<h5>Services</h5></a></li>+  
    <li><a href="#">联系我们<h5>Contact Us</h5></a></li>+  
  </ul>+  
</div>+
```



任务3.1 使用CSS+DIV创建导航

任务实施：使用ul列表制作横向导航

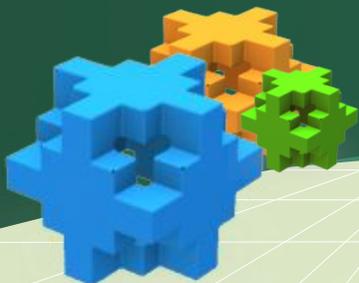
```
.nav1{ background:url(../images/headerBg.gif) repeat-x;width:
600px; height: 84px;}+
.nav1 ul{padding-top:40px;}+
.nav1 ul li {list-style-type:none;float:left; }+
.nav1 ul li a:link,.nav1 ul li a:visited{text-decoration:none;
display:block; padding:0 16px; font-size:14px;color:#2D79A5;
background:url(../images/navBg.gif)no-repeat top right;
font-weight:bold; font-family:"宋体"; }+
.nav1 a h5{ color:#C8C8C8; font-size:11px; float:none !important;
float:left;margin-top:0px;}+
.nav1 a:hover{ color:#292929; }+
```



任务3.1 使用CSS+DIV创建导航

任务实施：使用超链接制作横向导航

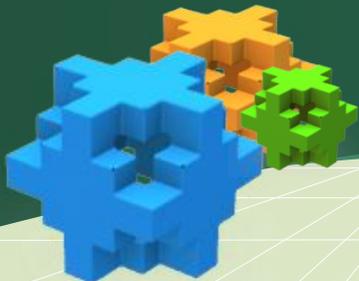
```
<div >  
  <a href="#">首页<h5>Home</h5></a>  
  <a href="#">产品展示<h5>products</h5></a>  
  <a href="#">客户案例<h5>Solutions</h5></a>  
  <a href="#">新闻中心<h5>News Center</h5></a>  
  <a href="#">服务支持<h5>Services</h5></a>  
  <a href="#">联系我们<h5>Contact Us</h5></a>  
</div>
```



任务3.1 使用CSS+DIV创建导航

任务实施：使用超链接制作横向导航

```
.nav { width: 600px; height: 40px; }  
.nav a:link ,.nav a:visited{ height:40px; background:url(../images/navBg.gif) no-repeat right  
top; display:block; padding:0 16px; float:left; font-size:14px; font-weight:bold;  
color:#2D79A5; font-family:"宋体"; }  
.nav a h5{ color:#C8C8C8; font-size:11px; float:none !important; float:left; }  
.nav a:hover{ color:#292929; text-decoration:none; }
```

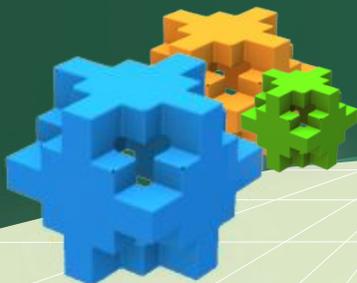


任务3.1 使用CSS+DIV创建导航

任务实施：制作竖向导航

竖状导航块内容采用链接文字方式实现，在窗体页面的源视图添加代码如下：

```
<div class="menu">  
  <a href="#" class="first">ASP.NET 中文网站学习</a>  
  <a href="#">ASP.NET 资源共享中心</a>  
  <a href="#">PHP 资源学习中心</a>  
  <a href="#">JSP 网站制作</a>  
  <a href="#">安卓移动编程</a>  
</div>
```



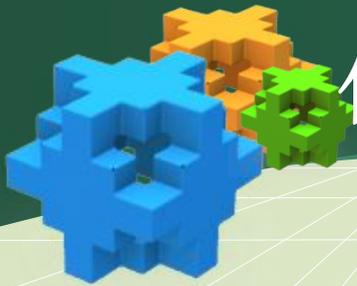
任务3.1 使用CSS+DIV创建导航

任务实施：制作竖向导航

```
.menu{ width:226px; height:auto; overflow:hidden;}  
.menu a:link{ background:url(..images/nav.gif) left top no-repeat; text-align:left; width:176px;  
             height:24px; float:left; display:inline; padding:8px 0 0 50px; color:#124A6D;  
             font-size:14px; font-weight:bold;}  
.menu a.first{ background:url(..images/nav01.gif) no-repeat;}  
.menu a:hover{ background:url(..images/navHover.gif) no-repeat; text-decoration:none;  
             color:#FFFFFFD; }
```



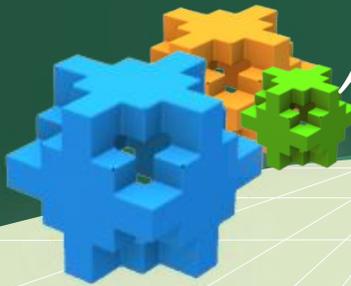
图 3-2 竖向导航的制作



任务3.2 使用Menu控件建立网站导航

任务描述:

本任务主要是了解menu控件的基本属性，并能够使用menu控件建立一级横向导航、二级横向导航、一级竖向导航和二级竖向导航。

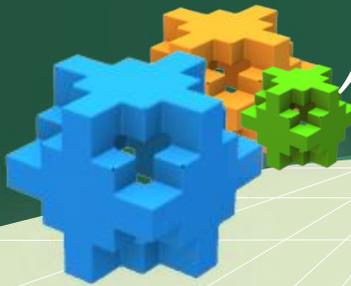


任务3.2 使用Menu控件建立网站导航

关键知识： Menu控件

Menu控件是一个菜单控件，由一个或多个菜单子项（MenuItem）组成，并且菜单子项之间可以形成一定的层次关系。菜单控件包含两种菜单：静态菜单（StaticMenu）和动态菜单（DynamicMenu）。静态菜单总是一直显示在网页上充当一级菜单的作用，而动态菜单由菜单的父菜单弹出，充当二级菜单作用。

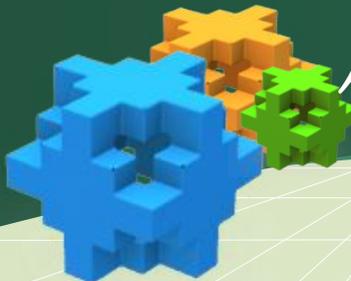
Menu控件的Items属性表示Menu控件的所有菜单项，可以在设计视图中以静态的方式添加Menu控件的菜单。



任务3.2 使用Menu控件建立网站导航

关键知识：Menu常用属性

- 1) Orientation属性：用于设置控件的呈现方向（Vertical、Horizontal。
- 2) StaticDisplayLevels属性：用于设置静态菜单显示级别数。默认值是1。
- 3) MaximumDynamicDisplayLevels属性：指定在静态显示层后应显示的动态显示菜单节点层数。
- 4) DisappearAfter属性：设置菜单动态显示部分从显示到消失所需的时间长度。默认值为 500 毫秒。
- 5) CssClass属性：可引用类样式。添加静态菜单项后可通过引用链接类样式实现导航的效果。
- 6) StaticEnableDefaultPopOutImage属性，该属性默认为True，有多级导航菜单时，菜单项之间会出现的间隔符号。设为False，则不出现间隔符号。
- 7) StaticMenuStyle样式：可设置一级静态菜单项的样式
- 8) DynamicMenuStyle样式：可设置除一级静态菜单项外的其它动态菜单的样式。



任务3.2 使用Menu控件建立网站导航

任务实施：建立横向导航

选择 Menu 控件，设置如下属性，运行可看到如图 3-6 所示的页面效果。

- 1) 属性 Orientation 值为 Horizontal，为横向显示模式。
- 2) 属性 CssClass 引用类样式名为 nav。

步骤见教材任务

首页

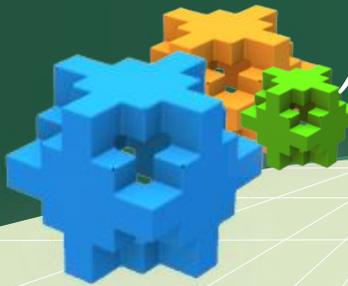
产品展示

客户案例

新闻中心

服务支持

联系我们



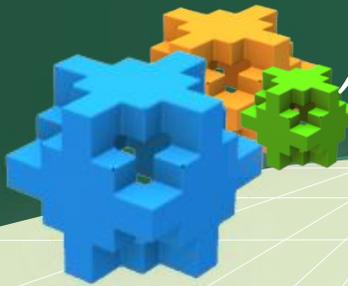
任务3.2 使用Menu控件建立网站导航

任务实施：建立二级横向导航

步骤见教材任务



图 3-7 二级横向导航



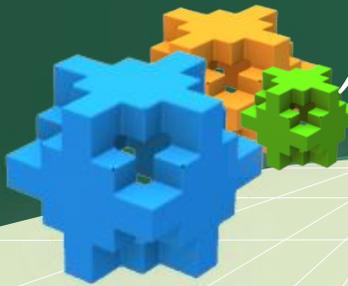
任务3.2 使用Menu控件建立网站导航

任务实施：建立一级竖向导航

步骤见教材任务



图 3-9 竖向导航效果图



任务3.2 使用Menu控件建立网站导航

任务实施：建立二级竖向导航

步骤见教材任务

▶ ASP.NET中文网站学习

▶ MSDN学习

▶ ASP.NET资源共享中心

▶ asp.net技术乐园

▶ PHP资源学习中心

▶ JSP网站制作

▶ 移动编程



任务3.3 使用TreeView建立树状导航

任务描述:

网站的后台管理主页面通常需要设置一个导航，通过该导航，使得管理员可链接到各个页面，从而实现数据库的插入、删除、更新和修改用户信息等任务，本任务使用 **TreeView** 控件建立一个管理员后台主页面的树状导航。



任务3.3 使用TreeView建立树状导航

关键知识：TreeView控件

TreeView控件由一个或多个节点构成，通过树状视图形式展示。可以将控件绑定到SiteMapDataSource控件上、绑定到XMLDataSource控件上和绑定到数据库上，树中的每个项都被称为一个节点，由TreeNode对象表示。每个节点都具有一个Text属性和一个Value属性。Text属性的值显示在TreeView控件中，而Value属性则用于存储有关该节点的任何附加数据。



任务3.3 使用TreeView建立树状导航

关键知识：TreeView控件

- ❖ Expanded属性：指定默认情况下节点是否打开。
- ❖ NavigateUrl属性：菜单被选中是被定位到的页面的Url。
- ❖ SelectAction属性：指定节点被选中时执行的操作，例如展开和折叠。
- ❖ Text属性：节点项显示的文本。
- ❖ Value属性：节点项的值。
- ❖ Target属性：设置为_blank，从空白页打开跳转的页面。
- ❖ CssClass属性：引用CSS类样式。
- ❖ NodeIndent属性：设置节点的缩进量。
- ❖ NodeStyle样式：设置节点的样式，包括类样式的引用、水平间距、字体等样式的设置。
- ❖ HoverNodeStyle样式：鼠标经过节点样式的设置。
- ❖ ExpandDepth：默认为FullExpand，当有多级菜单时可设置为只显示前面的级数。

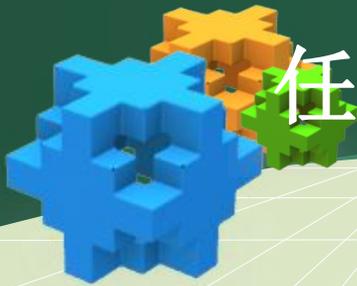


任务3.3 使用TreeView建立树状导航

任务实现:

步骤见教材任务

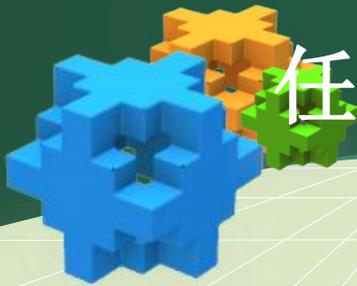




任务3.4 使用MultiView实现多视图切换

任务描述:

在各类门户网站中通常会设计一个多视图切换层，如在一个布局位置，用户可通过单击切换按钮实现同一位置查看新闻或通知或公告等信息，如图3-14所示效果。本任务使用MultiView和View控件结合，实现多视图的切换。



任务3.4 使用MultiView实现多视图切换

关键知识：MultiView和View

在ASP.NET中，使用MultiView和View控件可以做出选项卡的效果，一个MultiView控件可以包含多个View控件，一个View控件相当于一个选项卡。MultiView控件一次只能显示一个View视图，与按钮、链接按钮或图片按钮控件结合使用，可通过单击按钮，切换到不同的View控件，从而看到不同的内容。

主要属性：

Views和**ActiveViewIndex**

按钮属性设置：

CommandName属性和**CommandArgument**属性

任务3.4 使用MultiView实现多视图切换

任务实施：

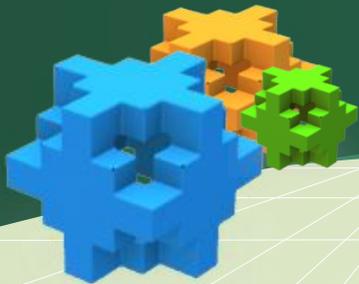
步骤如下：

- ❖ 添加样式文件和布局文件
- ❖ 定位并设置链接按钮
- ❖ 设置链接样式
- ❖ 添加MultiView和View控件
- ❖ 设计链接标题样式
- ❖ 编写事件代码
- ❖ 效果图如下：



图 3-14 点击链接按钮实现视图切换

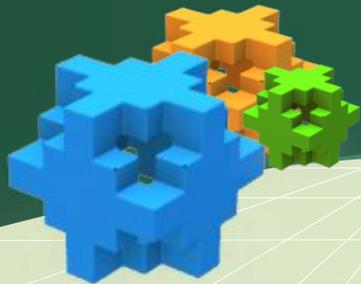
任务3.5 站点地图应用



任务描述:

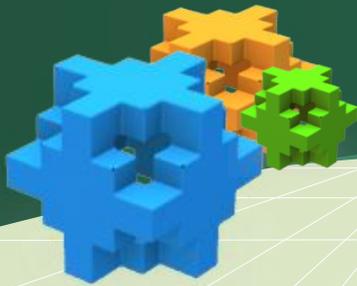
根据网站的文件结构创建站点地图，并依据站点地图与SiteMapPath控件结合，实现站点路径导航；与Menu控件和TreeView控件结合，实现页面导航。

任务3.5 站点地图应用



关键知识： 站点地图控件

站点地图文件按站点的分层形式组织页面，用于描述站点的逻辑结构，根节点是SiteMap。可以为网站中的每个页面添加一个SiteMapNode元素，并通过嵌入SiteMapNode元素来创建层次结构。

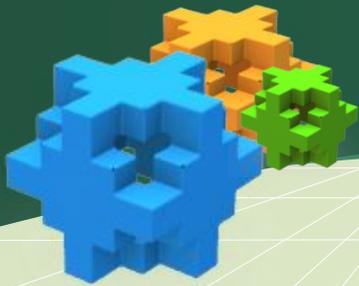


任务3.5 站点地图应用

关键知识： 站点地图控件

表 3-1 SiteMapNode 元素的常用属性

属 性	说 明
<u>url</u>	设置 <u>SiteMapNode</u> 节点所代表的 URL
title	设置 <u>SiteMapNode</u> 节点的标题，这个属性经常由导航控件用于显示 URL 的标题
description	指定关于这个页面的描述
roles	通过使用安全整修（后面将讨论），这个属性指定允许存取这个页面的角色



任务3.5 站点地图应用

关键知识： SiteMapPath

SiteMapPath控件自动从站点地图 Web.sitemap 文件中读取数据，提供了从当前位置沿页层次结构向上的跳转功能。

SiteMapPath 控件的主要属性如下。↵

- PathDirection 属性：可以反转 SiteMapPath 控件所显示的路径的方向，分为根节点指向当前节点方向，也可以设置为从当前节点指向根节点方向。↵
- PathSeparator 属性：设置路径之间的间隔符号。↵



任务3.5 站点地图应用

任务实施：建立网站地图

```
<?xml version="1.0" encoding="utf-8" ?>
<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >
  <siteMapNode url="Default.aspx" title="首页" description="导航站点介绍">
    <siteMapNode url="~/Admin/AdminDefault.aspx" title="管理员首页">
      <siteMapNode url="~/Admin/adminlogin.aspx" title="管理员登录" />
      <siteMapNode url="~/Admin/deletefiles.aspx" title="删除文件" />
      <siteMapNode url="~/Admin/insertfiles.aspx" title="插入文件" />
    </siteMapNode>
    <siteMapNode url="~/Customer/Default.aspx" title="客户端首页" >
      <siteMapNode url="~/Customer/cart.aspx" title="购物车"/>
      <siteMapNode url="~/Customer/order.aspx" title="订单页面"/>
    </siteMapNode>
    <siteMapNode url="task1.aspx" title="产品查询" />
    <siteMapNode url="task2.aspx" title="产品结构" />
    <siteMapNode url="task3.aspx" title="项目案例" />
    <siteMapNode url="task4.aspx" title="人才招聘" />
    <siteMapNode url="task5.aspx" title="联系我们" />
  </siteMapNode>
</siteMap>
```

任务3.5 站点地图应用



任务实施：

- ❖ 设置SiteMapPath控件
- ❖ 设置Menu控件

调整导航控件显示效果，设置 Menu 控件属性如下。↵

- Orientation 属性：值为 Horizontal，水平方向展示导航。↵
- StaticDisplayLevels 属性：值为 2。↵
- StaticEnableDefaultPopOutImage 属性：值为 False。↵
- StaticSubMenuIndent 属性：设置菜单之间的水平间隔，值为 40px。↵
- DynamicMenuStyle 动态样式属性：垂直间隔 VerticalPadding 值为 5px。↵
- DynamicMenuItemStyle 动态子菜单样式属性：设置二级菜单水平间隔 HorizontalPadding 值为 35px，垂直间隔 VerticalPadding 值为 5px。↵

- ❖ 设置TreeView控件

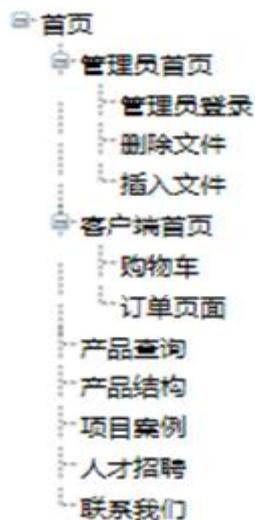
任务3.5 站点地图应用

任务实施:



A screenshot of a web application's menu control. The menu items are displayed in a horizontal list with a light blue background. The items are: 首页 (Home), 管理员首页 (Admin Home), 客户端首页 (Client Home), 产品查询 (Product Search), 产品结构 (Product Structure), 项目案例 (Project Cases), 人才招聘 (Recruitment), and 联系我们 (Contact Us). Under 管理员首页, there are three sub-items: 管理员登录 (Admin Login), 删除文件 (Delete File), and 插入文件 (Insert File).

图 3-21 menu 控件显示站点地图文件



A screenshot of a TreeView control showing a hierarchical tree structure of site map files. The root node is 首页 (Home). Under 首页, there are three main branches: 管理员首页 (Admin Home), 客户端首页 (Client Home), and 产品查询 (Product Search). Under 管理员首页, there are three sub-items: 管理员登录 (Admin Login), 删除文件 (Delete File), and 插入文件 (Insert File). Under 客户端首页, there are two sub-items: 购物车 (Shopping Cart) and 订单页面 (Order Page). Under 产品查询, there are three sub-items: 产品结构 (Product Structure), 项目案例 (Project Cases), and 人才招聘 (Recruitment). Under 联系我们 (Contact Us), there is one sub-item: 联系我们 (Contact Us).

图 3-23 TreeView 控件运行效果图