

● 明月创作室 编 著

ASP.NET

编程 精彩 百例



人民邮电出版社
www.pptph.com.cn



附光盘
CD-ROM

ASP.NET 编程精彩百例

ASP.NET

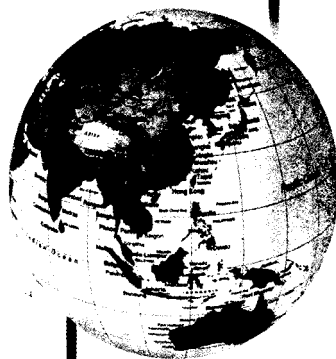
编程
精彩
百例



● 明月创作室 编 著

ASP.NET

编程 精彩 百例



本书附盘可从本馆主页 <http://lib.szu.edu.cn/>
上由“馆藏检索”该书详细信息后下载，
也可到视听部复制

人民邮电出版社

图书在版编目 (CIP) 数据

ASP.NET 编程精彩百例 / 明月创作室编著. —北京: 人民邮电出版社, 2002.1

ISBN 7-115-09822-0

I.A... II.明... III.主页制作—程序设计 IV.TP393.092

中国版本图书馆 CIP 数据核字 (2001) 第 088146 号

内 容 提 要

ASP.NET 是微软 .NET 战略中的一个重要成员,除了可以使用 Visual C#、VB.NET、Visual C++.NET、JScript.NET 等语言编写外,还可以使用第三方的 .NET 接口 (如 COBOL.NET、Perl.NET 等) 进行编写。

本书通过 100 个精彩实例,由浅入深介绍了 ASP.NET 网络程序设计的编程方法与实用技巧。为了方便读者学习与深入理解 ASP.NET,本书将这 100 个实例分为 4 篇,即基础编程实例、中级编程实例、高级编程实例与混合编程实例。在每个实例中,除了介绍 ASP.NET 中所提供的基础知识点及编程技巧外,还介绍了 ASP.NET 在网络编程、图形编程、多媒体编程、Mobile (移动通信) 等方面的重要应用及开发技术。

本书附赠一张光盘,其中包括全部 100 个实例源代码及可执行文件 (详细内容请参见光盘中的“本书配套光盘说明.txt”)。

本书体系结构清晰,解释说明详尽,开发实例典型,适合于计算机编程用户、网络用户及对 ASP.NET 感兴趣的用户阅读,也可作为 ASP.NET 培训参考书。

ASP.NET 编程精彩百例

-
- ◆ 编 著 明月创作室
责任编辑 魏雪萍
执行编辑 胡芳颖
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@pptph.com.cn
网址 <http://www.pptph.com.cn>
读者热线:010-67129212 010-67129211(传真)
北京汉魂图文设计有限公司制作
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本:787×1092 1/16
印张:26
字数:636 千字 2002 年 1 月第 1 版
印数:1-5 000 册 2002 年 1 月北京第 1 次印刷

ISBN 7-115-09822-0/TP·2571

定价:45.00 元(附光盘)

本书如有印装质量问题,请与本社联系 电话:(010)67129223

前 言

ASP.NET 可以使用多种语言（如 C#、VB.NET、JScript.NET、COBOL.NET、Perl.NET 等）来编写，可以运行在基于 Microsoft.NET 框架的任何机器上（包括 Mobile 设备）。ASP.NET 的这种功能为计算机爱好者提供了多种选择，即只要懂一种编程语言就可以编写十分复杂而有效的 Web 应用程序。另外，如果已有了 DLL 组件（可用 C#、VB.NET 等），则在 ASP.NET 中还可以直接调用它们，运行在 IE 等浏览器或者远程计算机上。

为了方便读者阅读书中的 100 个实例，本书分为 4 大篇：基础编程实例、中级编程实例、高级编程实例及混合编程实例。主要采用以下的结构介绍每个实例：

- 实例说明；
- 编程思路；
- 编程步骤；
- 运行结果；
- 程序拓展。

在“编程思路”中，除了介绍本实例程序设计的方法，还介绍一些知识难点，帮助读者快速掌握 ASP.NET 基础内容或者新增内容。

在“编程步骤”中，除了使用记事本直接编写 ASP.NET 网络程序外，还提供了 Visual Studio.NET 的 Web 应用程序的编程环境，同时还提供了十分重要并很常用的编程调试方法，如通过 Web 页面上的错误信息来跟踪错误之处等。

在“程序拓展”中给出实例程序进一步开发的思路，针对某些实例本书还给出扩展实例，以帮助读者进一步地提高 ASP.NET 的开发应用水平。

本书所提供的 100 个开发实例无论大小都是针对某一个或者某一类应用问题的，因而具有较强的针对性。如果读者已熟悉了这些实例中的开发方法与技巧，则要开发大型复杂的 Web 应用程序就比较容易了。只要将本书所提供的实例重新组合或者粘贴在一起即可，也可以将某些已编写的 C#或 VB.NET 做成 DLL 组件，然后在 ASP.NET 主程序中直接调用即可。这样就可以重复使用已有程序代码，这也正是 Microsoft.NET 框架的目的。

我们在本书的附录中还提供了 ASP.NET 热点问答，帮助读者进一步地了解与熟悉 ASP.NET。

本书由明月创作室集体编著，主要由田啸、林虎、王志珍、王博、曹美芳、王康、李美、王东、东海、曹国华、李珊珊、李明兰、王华、冬美、闵东东、晓花、王任、何中人、何达、许堂、任东等同志编写。

读者在使用本书时，有什么意见或者建议，请与我们联系。E-mail（电子函件）：
caogjwj@sina.com、zhiyin101@163.net。

明月创作室
2001.12

目 录

第 1 篇 ASP.NET 基础编程实例

| | | |
|-------|-------------------------------------|----|
| 实例 1 | 一个 HTML 化的 ASP.NET 程序 | 1 |
| 实例 2 | ASP+WebForms 广告轮换控制过程 | 4 |
| 实例 3 | 驱动器列表应用 | 7 |
| 实例 4 | 列出当前盘上目录和文件 | 9 |
| 实例 5 | 使用“文件预览”显示文件夹信息 | 11 |
| 实例 6 | 使用递归循环以显示目录列表 | 13 |
| 实例 7 | 在 ASP.NET 中随意创建图形信息 | 14 |
| 实例 8 | 在 ASP.NET 中显示事件日志记录 (1) | 19 |
| 实例 9 | 在 ASP.NET 中显示事件日志记录 (2) | 22 |
| 实例 10 | 在 ASP.NET 中使用 DataGrid 分页 | 25 |
| 实例 11 | 在 ASP.NET 访问 Excel 2000/XP 文件 | 28 |
| 实例 12 | 显示当前浏览器头信息 | 31 |
| 实例 13 | 利用 ASP.NET 获取主机名和 IP 地址 | 33 |
| 实例 14 | 由域名获取其 IP 地址 | 34 |
| 实例 15 | 由 IP 地址获得其域名 | 36 |
| 实例 16 | ASP.NET 中的正则表达式应用 | 38 |
| 实例 17 | 主机远程控制与探测应用 | 40 |
| 实例 18 | ASP.NET 控件——AdRotator 应用 | 42 |
| 实例 19 | ASP.NET 日程序 | 44 |
| 实例 20 | 查看主机内存情况 | 51 |
| 实例 21 | ASP.NET 简易应用程序“Hello, World!” | 53 |
| 实例 22 | 读取 Config.Web 配置文件 | 56 |
| 实例 23 | 显示 BrowseCaps 配置段内容 | 59 |
| 实例 24 | 写入文件操作与应用 | 61 |
| 实例 25 | 文件读操作及应用 | 63 |
| 实例 26 | 文件删除与应用 | 64 |
| 实例 27 | ASP.NET 中异常处理及其应用 | 65 |
| 实例 28 | AutoPostBack 技术及其应用 | 73 |
| 实例 29 | 用户合法性验证与应用 | 76 |
| 实例 30 | ASP.NET 列表框应用 | 79 |
| 实例 31 | Calendar 控件操作与应用 | 81 |
| 实例 32 | 用户操作验证与应用 | 83 |
| 实例 33 | CompareValidator 服务器比较控件操作与应用 | 86 |
| 实例 34 | CustomValidator 控件操作与应用 | 89 |

| | | |
|-------|-------------------------------------|-----|
| 实例 35 | 服务器端控件 ValidationSummary 应用操作 | 92 |
| 实例 36 | HtmlTable 控件操作与应用 | 97 |
| 实例 37 | 在 HTML 控件中运用样式 | 100 |
| 实例 38 | 在 Web 服务器控件中运用样式 | 101 |

第 2 篇 ASP.NET 中级编程实例

| | | |
|-------|---------------------------------------|-----|
| 实例 39 | ASP.NET 界面设计经典程序“Hello, World!” | 105 |
| 实例 40 | 使用 C#与 ASP.NET 创建简易 Web 站点 | 118 |
| 实例 41 | 在 ASP.NET 中检查表单填写的正确性 | 123 |
| 实例 42 | 表格填写校验应用 | 127 |
| 实例 43 | 电脑图书数据库修改与应用 | 135 |
| 实例 44 | 电脑图书订购系统 | 141 |
| 实例 45 | ASP.NET 数据绑定 DataBind 简单应用 | 146 |
| 实例 46 | 数组绑定及其列表演示 | 148 |
| 实例 47 | 使用 DataReader 显示数据库中数据 | 151 |
| 实例 48 | 多按钮事件处理与应用 | 155 |
| 实例 49 | 利用 ASP.NET 编写 FTP 上传程序 | 160 |
| 实例 50 | 使用 ASP.NET 实现简易注册系统 | 164 |
| 实例 51 | 使用 ASP.NET 与 XML.NET 实现新闻更新系统 | 169 |
| 实例 52 | 动态生成页面控件 | 172 |
| 实例 53 | 使用 ASP.NET 快速生成 XML 文档 | 174 |
| 实例 54 | 表示代码与逻辑代码分离应用 | 178 |
| 实例 55 | 在 ASP.NET 中使用应用程序配置 | 182 |
| 实例 56 | 轻松地创建 GUID | 186 |
| 实例 57 | 使用 ASP.NET 编写域名查询程序 | 189 |
| 实例 58 | 使用 C#及其继承实现 ASP.NET 数据库编程 | 192 |
| 实例 59 | C#类多态性在 ASP.NET 编程中的应用 | 197 |
| 实例 60 | ASP.NET 安全身份验证技术与应用 | 204 |
| 实例 61 | 网络追捕与应用 | 209 |
| 实例 62 | 使用 ASP.NET 实现 POP3 功能 | 213 |

第 3 篇 ASP.NET 高级编程实例

| | | |
|-------|---|-----|
| 实例 63 | 深入 Web Form 编程与应用 | 219 |
| 实例 64 | 使用 C#和 ASP.NET 实现网上 Football Pool | 222 |
| 实例 65 | 使用 ASP.NET 编写网络追捕程序 | 228 |
| 实例 66 | Beta2 版的 ASP.NET 追捕程序 | 247 |
| 实例 67 | 用 ASP.NET 构建完整电子邮件发送系统 | 254 |
| 实例 68 | 使用 ASP.NET 编写电子邮件发送程序 | 259 |
| 实例 69 | 用 JScript.NET 开发 ASP.NET 应用程序 (1) | 261 |
| 实例 70 | 用 JScript.NET 开发 ASP.NET 应用程序 (2) | 264 |

| | | |
|-------|--|-----|
| 实例 71 | 使用 JScript.NET/ASP.NET 深入编程与应用 | 267 |
| 实例 72 | 在 JScript.NET 中访问 SQL Server 数据库 | 273 |
| 实例 73 | JScript.NET 的 Web Service 编程与应用 | 276 |
| 实例 74 | ASP.NET 中的错误处理支持技术与应用 | 279 |
| 实例 75 | 用 ASP.NET 加密网络口令 | 283 |
| 实例 76 | 在 ASP.NET 中获取文件属性信息 | 286 |
| 实例 77 | 使用 ASP.NET 在 Mobile 机器上显示文件信息 | 290 |
| 实例 78 | 使用 ASP.NET 在 Mobile 机器上显示菜单信息 | 292 |
| 实例 79 | 在一个 Mobile ASP.NET 页面上放置多个 Forms | 295 |
| 实例 80 | 在 Mobile 页面上响应控件事件 | 297 |
| 实例 81 | 在 Mobile Web 上显示图像 | 299 |
| 实例 82 | 在 Mobile 页面上允许用户使用电话呼叫 | 302 |
| 实例 83 | 在 Mobile 页面上显示万年日历 | 303 |
| 实例 84 | Mobile 自定义控件编写与应用 | 305 |
| 实例 85 | 编写 Mobile 继承型用户控件 | 308 |
| 实例 86 | 编写 Mobile 复合型自定义控件 | 312 |
| 实例 87 | 使用 ASP.NET 动态地生成图形与图像 | 319 |
| 实例 88 | 创建完整的 ASP.NET 框架的 Web Forms 应用程序 | 334 |

第 4 篇 ASP.NET 与 COBOL.NET/Perl.NET 混合编程实例

| | | |
|--------|---|-----|
| 实例 89 | ASP.NET/COBOL.NET 集成基础与应用 | 359 |
| 实例 90 | 使用 COBOL.NET 编写 ASP.NET 的“Hello!”程序 | 364 |
| 实例 91 | 使用 COBOL.NET 编写字体放大程序 | 365 |
| 实例 92 | 使用 COBOL.NET 编写矩形面积的 Web Service | 367 |
| 实例 93 | 使用 COBOL.NET 编写事件代码 | 369 |
| 实例 94 | 使用 COBOL.NET 类接口编写 Fibonacci 序列 | 373 |
| 实例 95 | 在 COBOL.NET 中验证 ASP.NET 代码生成功能 | 385 |
| 实例 96 | COBOL.NET 中的 Web Forms 控件编写与应用 | 391 |
| 实例 97 | 使用 COBOL.NET 编写 ASP.NET 单击事件 | 394 |
| 实例 98 | 使用 COBOL.NET 编写两数相加 Web Service | 397 |
| 实例 99 | 使用 ASP.NET 与 Perl.NET 编写 Web 应用程序 | 399 |
| 实例 100 | 使用 Perl.NET 实现 Web Service 编程与应用 | 403 |
| 附录 | ASP.NET 热点问题精粹 | 405 |
| 参考文献 | | 407 |

第 1 篇 ASP.NET 基础编程实例

本篇主要通过 ASP.NET 基础编程实例详细地介绍了 ASP.NET 编程基础知识，如 ASP+WebForms 广告轮换控制过程、驱动器列表应用、文件操作、绘制图形、显示事件日志、页面分页、域名与 IP 地址转换、ASP.NET 控件应用、配置文件使用、异常情况处理等。

实例 1 一个 HTML 化的 ASP.NET 程序

1.1 实例说明

此实例程序使用一个 HTML 化的文件演示了 ASP.NET 程序的运行及显示结果。

1.2 编程思路

实际上，ASP.NET 程序与标准 HTML 文件并没有什么区别。对于普通的 HTML 文件，只要将它保存为后缀为.aspx 的文件，然后在浏览器里运行此文件，则它将以 ASP.NET 解释方式运行，同时显示出相关结果来。

注意：某些电脑图书或者杂志，将 ASP.NET 说成是 ASP+，它主要用于区别 Microsoft 原先的 ASP 程序。

这个 ASP.NET 程序比较简单。ASP.NET (ASP+) 向前兼容 ASP，所以任何一个 ASP 文件也可以作为 ASP.NET 文件来运行。在后面的某些实例中将看到此类现象。

1.3 编程步骤

在记事本中输入如下程序代码，且取名为 intro1.aspx。

```
<! 程序:intro1.aspx>
<html>
  <head>
    <link rel="stylesheet" href="intro.css">
  </head>
  <body>
    <center>
      <form action="intro1.aspx" method="post">
```

```

<h3> Name: <input id="Name" type="text">
Category: <select id="Category" size=1>
    <option>psychology</option>
    <option>business</option>
    <option>popular_comp</option>
</select>
<input type="submit" value="Lookup">
</form>
</center>
</body>
</html>

```

1.4 运行结果

在 IE 浏览器中输入（假设 intro1.aspx 程序放在 D:\Shu\Asp.net 文件夹中）
http://localhost/asp.net/intro1.aspx，则出现如图 1-1 所示的运行结果。

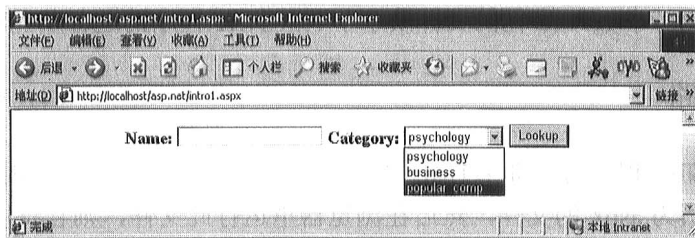


图 1-1

1.5 程序拓展

下面把上边这个文件稍微改动一下，即把两个 input 前面的 Name 和 Category 分别改为“姓名”和“分类”，修改后的文件命名为 intro2.aspx。

```

<! 程序: intro2.aspx>
<html>
<head>
    <link rel="stylesheet" href="intro.css">
</head>
<body>
<center>
    <form action="intro1.aspx" method="post">
        <h3> 名字: <input id="Name" type="text">
        分类: <select id="Category" size=1>
            <option>psychology</option>
            <option>business</option>
            <option>popular_comp</option>

```

```

        </select>
        <input type=submit value="Lookup">
    </form>
</center>
</body>
</html>

```

如果在 Beta1 中运行此文件，则发现所有汉字都变成“????”符号了。这说明 Beta1 版不支持汉字，解决方法如下：

(1) 首先找到 Winnt/Microsoft.NET/Framework/[version]/config.web 这个文件，注意 [version]是指 NGWS 的版本号，它的形式可能如 Winnt/complus/1812.9/config.web。

(2) 用记事本打开这个文件，找到如下几行：

```

<globalization
    requestencoding="us-ascii"
    responseencoding="iso-8859-1"
/>

```

可以看到上面语句行中的“iso-8859-1”，正是此字符串造成了 Windows 操作系统不能显示汉字，把它改成“gb2312”就可以了，如图 1-2 所示。



图 1-2

再运行，则显示正常，如图 1-3 所示。

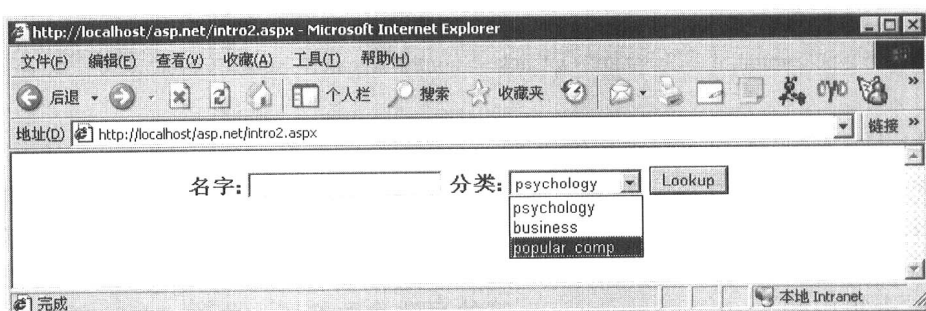


图 1-3

实例 2 ASP+WebForms 广告轮换控制过程

2.1 实例说明

此实例程序演示了一个广告轮换控制过程，这是一个真正意义上的 ASP.NET Web Forms 应用程序。

2.2 编程思路

现在来看一个更为复杂的 Web Forms，它带有一个广告轮换控制，一个文本输入框，一个选择框，一个按钮和一个文本标签。

在此 ASP.NET 程序中，原来 ASP 的<% %>脚本定界符不见了，取而代之的是<script language=...>。当然，您还是可以使用<% %>，毕竟它是向前兼容 ASP，不过还是尽量少用为好。

```
<script language="C#" runat=server>
void SubmitBtn_Click(Object sender, EventArgs e) {
    Message.Text = "Hi " + Name.Text + ", you selected: " + Category.SelectedItem;
}
</script>
```

用过 C++ 的读者一定很熟悉上面的程序段。以上 ASP.NET 程序段是用 Visual C# 编写的一个事件处理函数：void SubmitBtn_Click(Object sender, EventArgs e)。其中“void”代表该函数没有返回值，该函数带有两个参数，此 C# 代码只有一行，但是这行代码中的 Message、Name、Category 并没有在程序中定义，那么它们从哪里来的呢？请看下面的代码：

```
<form action="intro3.aspx" method="post" runat="server">
<asp:adrotator AdvertisementFile="ads.xml" BorderColor="black" BorderWidth=1 runat="server"/>
<h3> Name: <asp:textbox id="Name" runat="server"/>
Category: <asp:dropdownlist id="Category" runat=server>
    <asp:listitem>psychology</asp:listitem>
    <asp:listitem>business</asp:listitem>
    <asp:listitem>popular_comp</asp:listitem>
</asp:dropdownlist>
<asp:button type=submit text="Lookup" OnClick="SubmitBtn_Click" runat="server"/>
<p>
<asp:label id="Message" runat="server"/>
</form>
```

以上是 ASP.NET 表单，其写法和 HTML 表单完全不同。首先，所有的表单项（包括表单本身）后面都加上了“runat=server”，其意思是说这是一个服务器端控制项。另外，传统的 HTML 表单中的“<input type=text>”等的写法也改变了，例如，原来的文本框变为

“<asp:textbox>”，选择框变为“<asp:dropdownlist>”，选择框选项变为“<asp:listitem>”，而 submit 按钮变为“<asp:button>”（实际上，这个按钮对应的控制函数就是上面所提到的 SubmitBtn_Click()函数，它运行在服务器端）。

还有一个服务器端控制项为“<asp:label id="Message" runat="server"/>”。asp:label 是一个 ASP.NET 标签控制项语句，在传统 HTML 表单中是没有的。这样就存在一个问题：如果传统的 HTML 表单中没有这个元素，那么 ASP.NET 是怎么接收的呢？运行一下上面的 ASP.NET 程序，然后看一下所生成的 HTML 源代码，会发现此代码中有下面一行：

```
<INPUT type="hidden" name="__VIEWSTATE" value="a0z-1715863018_x">
```

这说明了 ASP.NET 就是通过这个隐藏表单的形式传递到 HTML 中的。在<form>标记下有这么一行代码：

```
<asp:adrotator AdvertisementFile="ads.xml" BorderColor="black" BorderWidth=1 runat="server"/>
```

asp:adrotator 也是一个服务器端控制项。看一下它的名字就能知道它的含义：这是一个广告轮换控制项，不过它现在不是内置组件，而是作为一个服务器端控制，控制它显示的就是 ads.xml，下面为其程序片段。

```
<Advertisements>
  <Ad>
    <ImageUrl>/quickstart/asplus/images/banner1.gif</ImageUrl>
    <TargetUrl>http://www.microsoft.com</TargetUrl>
    <AlternateText>Alt Text</AlternateText>
    <Keyword>Computers</Keyword>
    <Impressions>80</Impressions>
  </Ad>
```

2.3 编程步骤

(1) 在记事本中输入如下程序代码，命名为 intro3.aspx。

```
<! 程序:intro3.aspx>
<html>
  <head>
    <link rel="stylesheet" href="intro.css">
  </head>
  <script language="C#" runat=server>
    void SubmitBtn_Click(Object sender, EventArgs e)
    {
      Message.Text = "Hi " + Name.Text + ", you selected: " +Category.SelectedItem;
    }
  </script>
  <body>
    <center>
      <form action="intro3.aspx" method="post" runat="server">
```

```

<asp:adrotator AdvertisementFile="ads.xml" BorderColor="black" BorderWidth=1 runat="server"/>
<h3> Name: <asp:textbox id="Name" runat="server"/>
Category: <asp:dropdownlist id="Category" runat="server">
    <asp:listitem>psychology</asp:listitem>
    <asp:listitem>business</asp:listitem>
    <asp:listitem>popular_comp</asp:listitem>
</asp:dropdownlist>
<asp:button type="submit" text="Lookup" OnClick="SubmitBtn_Click" runat="server"/>
<p>
<asp:label id="Message" runat="server"/>
</form>
</center>
</body>
</html>

```

(2) 在记事本中输入如下程序代码，命名为 ads.xml。

```

<! 程序 ads.xml>
<Advertisements>
  <Ad>
    <ImageUrl>/quickstart/aspplus/images/banner1.gif</ImageUrl>
    <TargetUrl>http://www.microsoft.com</TargetUrl>
    <AlternateText>Alt Text</AlternateText>
    <Keyword>Computers</Keyword>
    <Impressions>80</Impressions>
  </Ad>
  <Ad>
    <ImageUrl>/quickstart/aspplus/images/banner2.gif</ImageUrl>
    <TargetUrl>http://www.microsoft.com</TargetUrl>
    <AlternateText>Alt Text</AlternateText>
    <Keyword>Computers</Keyword>
    <Impressions>80</Impressions>
  </Ad>
  <Ad>
    <ImageUrl>/quickstart/aspplus/images/banner3.gif</ImageUrl>
    <TargetUrl>http://www.microsoft.com</TargetUrl>
    <AlternateText>Alt Text</AlternateText>
    <Keyword>Computers</Keyword>
    <Impressions>80</Impressions>
  </Ad>
</Advertisements>

```

2.4 运行结果

在 IE 浏览器中输入 `http://localhost/asp.net/intro3.aspx`，则出现如图 2-1 所示的操作界面。

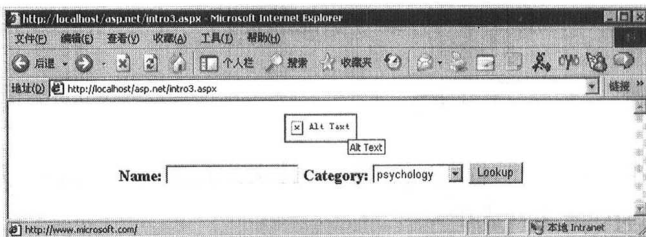


图 2-1

在此操作界面的上方将出现一个 Microsoft 公司的动态广告图形。如果您未上网，则显示一个“Alt Text”图形框。

2.5 程序拓展

您可以修改 `ads.xml` 中的如下语句：

```
<Ad>
  <ImageUrl>/quickstart/aspplus/images/banner1.gif</ImageUrl>
  <TargetUrl>http://www.microsoft.com</TargetUrl>
  <AlternateText>Alt Text</AlternateText>
  <Keyword>Computers</Keyword>
  <Impressions>80</Impressions>
</Ad>
```

这样就可以修改广告所出现的图形。

实例 3 驱动器列表应用

3.1 实例说明

此实例程序可以列出本地硬盘上所有驱动器。

3.2 编程思路

在使用 ASP 3.0 语言编程时，我们应该不会忘记 `FileSystemObject` 这个文件操作系统对象，它允许对文件系统进行读、写以及目录浏览。但是这个对象模型并不十分完善，比如对于二进制文件来说，还没有开始操作就已经到达范围的结尾了。

另外，还有一些希望的函数和属性在 ASP 3.0 中也没有具备。

自从微软公司发布了 ASP.NET 后，以上这些愿望就都能在 `System.IO` 名字空间中被 `Microsoft.NET Framework` 类所实现。您会发现，在 ASP.NET 中文件处理从没有这么容易过，

而且最重要的是它比以前更为强大。

在此实例程序中，根据@Import 名字空间标志，在 System.IO 名字空间中有一个类叫做“Directory”，它包含了可用于目录上操作的功能。这个类中还包含静态（不需要对象实例就能调用的）方法 GetLogicalDrives()。此方法能够提供包含驱动器字母的一个字符串数列。

3.3 编程步骤

在记事本中输入如下程序代码，且以 ListDrives.aspx 为文件名保存到 C:\inetpub\wwwroot\asp.net 文件夹中。

```
<% @Page Language="C#" %>
<% @Import Namespace="System.IO" %>
<%
string[] achDrives = Directory.GetLogicalDrives();
int nNumOfDrives = achDrives.Length;
    Response.Write("<ul>");
for (int i=0; i < nNumOfDrives; i++)
{
    Response.Write("<li><a href='\"listdir.aspx?dir=}\"");
    Response.Write(Server.UrlEncode(achDrives[i]));
    Response.Write(">\" + achDrives[i]);
    Response.Write("</a><br>");
}
    Response.Write("</ul>");
%>
```

3.4 运行结果

在 IE 浏览器中输入 <http://localhost/asp.net/listdrives.aspx>，则出现如图 3-1 所示的运行结果。

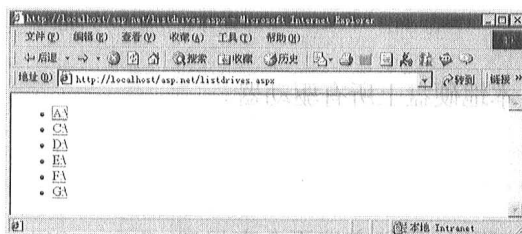


图 3-1

3.5 程序拓展

此实例程序中的 C#部分代码可以被其他脚本程序所使用。

实例4 列出当前盘上目录和文件

4.1 实例说明

此实例程序在前面实例的基础上列出当前驱动器（当前盘）上的目录及文件。

4.2 编程思路

很显然，不能只有一个从 `GetLogicalDrives()` 获取的驱动器简单列表，还需要增加一个 ASP.NET 页面，在这个页面中提供有关驱动器根目录的信息，因此，有了目录结构后，就可以进一步探索子目录、文件以及目录本身的属性（如创建的日期等）了。

此实例程序将进一步地演示如何执行这些功能。首先使用 `GetDirectories` 方法实现目录信息的读取，此方法返回一个目录对象数组，可以用这个数组来建立想要的功能，比如说建立一个更深入的链接。这同样适用于 `GetFiles()` 方法，它返回一个文件对象数组。

您可能在此实例程序的代码中注意到这里使用了“try-catch”语句。这是为了防止用户试图访问一些不允许其访问的内容等异常情况的发生。

4.3 编程步骤

在记事本中输入如下程序代码，且以 `listdir.aspx` 为文件名保存到 `C:\inetpub\wwwroot\asp.net` 文件夹中。

```
<% @Page Language="C#" %>
<% @Import Namespace="System.IO" %>
<%
string strDir2List = Request.QueryString.Get("dir");
Directory thisOne = null;
try
{
thisOne = new Directory(strDir2List);
// Reading the directory properties
Response.Write("<p>Creation: " +
thisOne.CreationTime.ToString() + "</p>");
Directory[] subDirectories = thisOne.GetDirectories();
Response.Write("<ul>");
for (int i=0; i < subDirectories.Length; i++)
{
Response.Write("<li><a href='\"listdir.aspx?dir=}\"");
Response.Write(Server.UrlEncode(subDirectories[i].FullName));
Response.Write(">\" + subDirectories[i].Name);
```