

配盘

# 亲密接触 ASP.NET

杜亮 编著

302...48412...

302...48412...964834



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



# 亲密接触 ASP.Net

杜 亮 编著

清华大学出版社

(京)新登字 158 号

## 内 容 简 介

本书结合大量的具体实例，以图文并茂的方式详细介绍了 ASP.Net 带来的编程新思想和新方法。

全书共 12 章。第 1 章、第 2 章简要介绍 ASP.Net 的基本概念；第 3 章介绍 ASP.Net 与 C# 的语法；第 4 章介绍常用对象的应用；第 5 章介绍 Server 控件的用法；第 6 章介绍数据库的访问；第 7 章介绍 ASP.Net Application 的有关知识；第 8 章介绍文件的操作；第 9 章介绍 XML 的基本用法；第 10 章介绍 WebService 的编程方法；第 11 章是 Asp.Net 的高级应用；第 12 章为高级应用实例。

本书以讲解 ASP.Net 基础为主，后面部分介绍了 ASP.Net 的高级应用，实用性强，适用于初学 ASP.Net 编程的 Web 程序员，对高级 Web 程序员也有一定参考价值。

版权所有，翻印必究。

本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

书 名：亲密接触 ASP. Net

作 者：杜 亮 编著

责任编辑：欧振旭

出 版 者：清华大学出版社(北京清华大学学研大厦, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者：清华大学印刷厂

发 行 者：新华书店总店北京发行所

开 本：787 × 1092 1/16 印张：24.25 字数：568 千字

版 次：2002 年 1 月第 1 版 2002 年 1 月第 1 次印刷

书 号：ISBN 7-302-05003-1/TP · 2951

印 数：0001 ~ 5000

定 价：39.00 元(附软盘)

# 序 言

自从 2000 年 6 月 22 日，微软公司宣布其称之为“公司命脉”的“Windows.Net”计划以来，“.Net”就成为了 IT 界的热门话题。“Net”给我们带来了崭新的思维，也给我们带来了崭新的技术。ASP.Net 就是“.Net”给我们带来的新技术之一。

看到“ASP.Net”这个名字，很多人马上就会想到这是一门 WEB 技术，因为微软的 ASP（Active Server Pages 动态服务页面）技术现在已经是家喻户晓，它的使用范围之广，也是微软公司始料未及的。而正是由于 ASP 技术的巨大成功，才促使微软公司一步步地推出其 ASP 系列产品，从 ASP 1.0 到 ASP 3.0，其功能越来越强大。

ASP 成功了，但是它的缺点也不少：

(1) 安全性差。自从 ASP 问世以来，它的安全性就一直倍受质疑，比如查看 ASP 源代码的漏洞，有一段时间达到了 20 余种，微软提供的文件组件可以任意查看和修改服务器下的任意文件。因此 ASP 一时间成为“不安全”的代名词。

(2) 内置组件少。微软虽然提供了功能强大的 ADO 组件，但是其他的方面提供的组件太少，以至于如文件上传这样常用的功能都需要使用第三方组件。

(3) 结构性差。由于使用的是 VBScript 和 JavaScript 作为常用的语言，使得程序的结构性很差。当 ASP 程序很长时，HTML 与 ASP 代码混杂在一起，很难让人看懂，修改起来也很麻烦。

对于用户的这些抱怨，微软公司当然不会置之不理，因此在 ASP 3.0 推出不到两个月，微软公司就推出了新一代的 ASP 网络技术，称之为 ASP+。2000 年 11 月正式更名为 ASP.Net，标明其为“.Net”计划的一部分。

ASP.Net 对 ASP 做了重大的改进，很好地解决了上面所列举的 ASP 的缺陷。并且提供了更强大的功能。ASP.Net 相对于 ASP 走出了不是一小步，而是真正的一大步。它的强大功能，本书会详细讲述。

阅读本书所需要的知识：

在写这本书之前，最令笔者烦恼的是，阅读本书到底需要掌握哪些知识？笔者总希望本书能让各个层次的人都能接受。但是要做到这一点是很不容易的。本书涉及的网络知识很广，不可能对每个知识都作详细的说明，所以笔者希望读者对 HTTP 和 HTML 语言有一定的了解，这样才有利于您更好地阅读本书。

编 者

2001 年 10 月

# 目 录

第 1 章 初识 ASP.Net .....	1
1.1 ASP.Net 能做哪些事情.....	1
1.2 ASP.Net 到底有哪些优越性.....	2
1.3 ASP.Net 与众不同的功能.....	3
1.3.1 ADO.Net.....	3
1.3.2 DataGrid.....	6
1.3.3 丰富的 Web 控件 .....	6
1.3.4 支持事件处理(Event Handle) .....	8
1.3.5 程序结构与代码的分离.....	9
第 2 章 第一个 ASP.Net 程序 .....	10
2.1 ASP.Net 的运行环境.....	10
2.1.1 安装 Internet Information Server(IIS).....	10
2.1.2 安装 Internet Explorer 6.0 .....	11
2.1.3 安装 .Net Framework SDK Beta 2 .....	12
2.1.4 中文显示的问题.....	14
2.2 Hello World——第一个 ASP.Net 程序.....	15
2.2.1 设置虚拟目录.....	15
2.2.2 Hello World .....	16
2.3 ASP.Net 执行原理.....	19
2.3.1 ASP.Net 的组成.....	19
2.3.2 ASP.Net 的执行原理.....	20
2.3.3 JIT 编译器 .....	21
第 3 章 ASP.Net、C# 语法 .....	22
3.1 ASP.Net 语法.....	22
3.1.1 <% %> .....	22
3.1.2 <Script Language="..." Runat="server">...</Script> .....	24
3.1.3 定义 Server 控件 .....	25
3.1.4 <object runat="server" /> .....	26
3.1.5 <%--注释--%> .....	28
3.1.6 <% @ Page ... %> 指令 .....	28
3.1.7 <% @ Import %> 指令 .....	29

3.2	C#	29
3.2.1	值(Values)类型	29
3.2.2	引用(References)类型	32
3.2.3	修饰符	37
3.2.4	控制语句	38
3.3	ASP.Net 编辑器	50
<b>第 4 章</b>	<b>常用对象的应用</b>	<b>52</b>
4.1	Request 对象	52
4.1.1	QueryString 集合	53
4.1.2	Form 集合	55
4.1.3	Cookie 集合	57
4.1.4	ServerVariable 集合	57
4.1.5	Browser 对象	61
4.1.6	其他 Request 对象、方法	65
4.2	Response 对象	66
4.3	Server 对象	67
<b>第 5 章</b>	<b>Server 控件</b>	<b>70</b>
5.1	Html 控件	70
5.1.1	HtmlImage 控件	73
5.1.2	HtmlInput 类控件	73
5.1.3	HtmlForm 控件	78
5.1.4	HtmlTable、HtmlTableRow、HtmlTableCell 控件	79
5.1.5	HtmlSelect 控件	81
5.1.6	其他 HTML 控件	84
5.1.7	Style 与 Attributes 属性	85
5.2	Web 控件	87
5.2.1	AdRotator 控件	87
5.2.2	Calendar 控件	90
5.2.3	Button 控件	97
5.2.4	CheckBox 与 CheckBoxList 控件	99
5.2.5	RadioButton 与 RadioButtonList 控件	107
5.2.6	Image 与 ImageButton 控件	107
5.2.7	HyperLink 与 LinkButton 控件	112
5.2.8	DropDownList 与 ListBox 控件	113
5.2.9	Label 控件	120
5.2.10	Panel 控件	120

---

5.2.11	PlaceHolder 控件.....	123
5.2.12	验证控件.....	124
5.2.13	Web 控件的样式控制.....	146
5.3	用户控件(User Control).....	147
5.3.1	最简单的用户控件.....	148
5.3.2	在用户控件中定义属性.....	149
5.3.3	用户控件的事件处理.....	151
5.3.4	在程序代码中创建用户控件.....	152
<b>第 6 章</b>	<b>数据库访问.....</b>	<b>155</b>
6.1	DataBind 技术.....	155
6.1.1	绑定属性.....	156
6.1.2	绑定集合.....	158
6.1.3	绑定表达式或者方法.....	159
6.2	SQL 语言.....	161
6.2.1	数据定义语言.....	161
6.2.2	数据操作语言.....	162
6.3	ADO.Net.....	163
6.3.1	Connection 对象.....	164
6.3.2	Command 与 DataReader 对象.....	166
6.3.3	DataSet 与 DataAdapter.....	173
6.4	数据 Web 控件.....	181
6.4.1	DataGrid 控件.....	181
6.4.2	Repeater 控件.....	209
6.4.3	DataList 控件.....	214
<b>第 7 章</b>	<b>ASP.Net Application.....</b>	<b>229</b>
7.1	Application 对象.....	229
7.2	Session.....	231
7.3	Cookie.....	233
7.4	ViewState.....	237
7.5	Global.asax 文件.....	240
7.5.1	Application 对象事件.....	240
7.5.2	Session 对象事件.....	242
<b>第 8 章</b>	<b>文件的操作.....</b>	<b>245</b>
8.1	File 类.....	245
8.1.1	操作文本文件.....	246

8.1.2 删除、拷贝、移动文件.....	250
8.2 FileStream 类.....	258
8.2.1 操作文本文件.....	259
8.2.2 操作二进制文件.....	264
8.3 DirectoryInfo 与 FileInfo.....	269
<b>第 9 章 XML.....</b>	<b>274</b>
9.1 XML 基础.....	274
9.1.1 扩展性.....	274
9.1.2 XML 与 HTML.....	276
9.1.3 XML 的基本概念.....	277
9.1.4 XML 语法.....	279
9.2 操作 XML.....	280
9.2.1 XML 控件.....	280
9.2.2 XmlTextReader.....	284
9.2.3 XmlTextWriter.....	287
9.2.4 DataSet 与 XML.....	290
<b>第 10 章 WebService.....</b>	<b>299</b>
10.1 WebService 简述.....	299
10.1.1 第一个 WebService 程序.....	300
10.1.2 WebService 解析的类型.....	308
10.2 WebService 中传递参数.....	308
10.3 使用 Session 与 Application 对象.....	310
<b>第 11 章 高级应用.....</b>	<b>313</b>
11.1 网络应用.....	313
11.1.1 WebRequest 与 WebResponse.....	313
11.1.2 TcpClient.....	316
11.2 编写组件.....	320
11.2.1 编写第一个组件.....	321
11.2.2 使用组件.....	324
11.3 使用追踪(Trace).....	327
11.3.1 页面追踪.....	328
11.3.2 网络应用程序整体的追踪.....	331
11.4 Code-Behind.....	334
<b>第 12 章 高级应用实例.....</b>	<b>341</b>
12.1 电子邮件的发送.....	341



---

12.1.1	简单的电子邮件发送.....	341
12.1.2	使用 MailMessage 对象发送复杂的电子邮件.....	345
12.1.3	发送带有附件的电子邮件.....	350
12.2	制作图形计数器.....	355
12.3	域名查询.....	361
附录 1	从 ASP 到 ASP.Net.....	367
附录 2	常见问题及解答.....	372

# 第 1 章 初识 ASP.Net

本章要点:

- ⊙ ASP.Net 概述
- ⊙ ASP.Net 的优点
- ⊙ ASP.Net 的功能

ASP.Net 的前身 ASP(Active Server Pages), 以其简单的语法、嵌入 HTML 的编辑方法, 在很短的时间内获得了巨大的成功, 成为了 NT 平台上 Web 技术的霸主, 这种地位至今无人能动摇。

但是随着 PHP, JSP 等技术的出现, ASP 的王者地位受到了严峻的挑战。Web 技术的新贵——JSP, 它是由微软公司的竞争对手 Sun 公司出品, 在执行效率、安全性等特性上已经完全超过了 ASP, 它更有着 ASP 无法比拟的跨平台性, 使得 JSP 在 Window、Unix、Linux 主机上均能使用。它的诞生, 使得越来越多的人选择了能够跨平台使用的 JSP。ASP 的用户群在日益减少, ASP 面临着前所未有的危机。

在这种情况下, 如果微软任由 JSP 取代 ASP 而坐视不理, 那么就意味着微软 Windows NT 操作系统的市场也将被别人占领。微软公司自然不会答应, 这就迫使微软公司拿出更加适合自己操作系统的 Web 技术来应对别人的挑战。因此, 微软公司提出了 “.Net” 构想, 如何改进 ASP, 也就理所当然成为了微软跨向 “.Net” 梦想的重要一步了。

2000 年 6 月 22 日, 微软公司首次向外界公布了其宏伟的 “Window .Net” 计划, 同时发布了下一代网络服务框架(Next Generation Web Services Framework 简称 NGWS)。在 NGWS 中人们首次看到了 ASP+ 的身影。2000 年 11 月, 微软公司推出了 Visual Studio .Net 7.0 beta1, 同时扩展了 NGWS 的功能并将其更名为 .Net Framework。此时, ASP+ 也更名成为现在的 ASP.Net。

到此, ASP.Net 正式推出了, 但它还是测试版本, 而且还是 Beta 1。不过从来还没有有一个 Beta 版的技术会获得如此多的好评, 这在微软产品的历史上是绝无仅有的。

ASP.Net 到底有多大的魔力会赢得如此多的赞扬, 它到底有哪些先进的特性, 这正是本书要向大家介绍的。

## 1.1 ASP.Net 能做哪些事情

ASP.Net 作为 Web 设计技术的一种, 它的主要功能还是产生动态网页, 以满足不同用户的需要。在网上见得最多的是 HTML 页面(文件的名缀的名为 .htm 或 .html), 它只是将别人编辑好的 Web 页面展现给大家, 这种页面如果不手动去修改, 是不会发生任何变化的。如果想在网页中访问数据库, 这种纯 HTML 页面就无能为力了。这时就得依赖各种 Web

技术。聊天室，论坛等都是由这些技术产生的。

ASP.Net 作为一种新的 Web 技术，它给了设计者一种全新的 Web 设计概念。它将软件设计和 Web 设计融为一个整体。由于它和 VC.Net, VB.Net, VC# 这些程序设计语言使用同一个 .Net Framework 对象开发库，可以想象 ASP.Net 所能实现的功能是多么强大，几乎 VC.Net 能做到的事情，ASP.Net 也能做到。

## 1.2 ASP.Net 到底有哪些优越性

ASP.Net 是一种建立在 CLR(Common Language Runtime 通用语言运行机)基础之上的程序开发构架，它主要用于在服务器上开发功能强大的 Web 应用。ASP.Net 提供了下面一些超越以前 Web 技术的重要特性：

- ◇ 优良的性能：ASP.Net 在服务器上运行的是经过编译的 CLR 代码。与以前的技术相比，ASP.Net 能够充分利用 Binding(绑定)，及时编译，本地优化，缓冲服务来提高程序的性能。
- ◇ 大量工具箱的支持：在 Visual Studio 的开发环境中，ASP.Net 因其丰富的工具箱和设计器而受到很高的赞扬。比如所见即所得的编辑方式，支持拖放的服务器控件，以及动态的配制管理，这些都是被大受赞扬的。而这一些还只是其提供的强大特性的冰山一角，还有很多值得在实践中发掘。
- ◇ 强大的功能以及良好的机动性：因为 ASP.Net 是建立在 CLR 基础上的，所以整个平台的功能和适用性更加适合网络应用程序的开发。CLR 中基本的类库、消息、数据访问解决方案几乎全部来源于 Web 思想。ASP.Net 可以使用很多种语言来编写，可以选择最适合的那一种，甚至可以选择使用多种语言来编写。此外，CLR 的协同工作能力保证以前开发的基于 COM 的程序，在移植到 ASP.Net 后，依然能正常使用。这就意味着，以前的 ASP 程序依然可以正常运行。
- ◇ 易用性：使用 ASP.Net 来完成一些普通的任务是很简单的事情。比如，ASP.Net 页面构架允许建立用户接口，这样就能够把代码和程序结构分离。写程序只需要考虑程序的结构，而不需要考虑怎样去书写代码，因为它们都事先已经写好了。
- ◇ 崭新的配制管理方式：ASP.Net 使用的是一种基于文本格式，等级式的配制系统(这里有些像 Linux)，这种方式使得应用服务器环境和 Web 应用的设置更加简单，因为配制信息是以文本格式储存的，新设置将不需要本地管理工具的帮助，也可使之生效。一个 ASP.Net 应用程序只需要将必须的程序拷贝到服务器上就可以使用了。更令人惊奇的是，对于正在运行中的被编译的代码，也同样可以对其配制进行更改，甚至是替换其本身，而不需要重启服务器。一个最明显的例子就是服务器上的控件的替换和修改再也不需要重启服务器了，这个本是写 DLL 控件最烦人的事情(作者的 AspncUP 上传组件就是在不停地重启 IIS 中写出的)，现在这种烦恼已经不存在了。
- ◇ 可移植性和良好的适用性：ASP.Net 对多线程有着自己独特的一套操作方法，当一

条线程出现异常情况后(比如溢出, 锁死), 一个新的进程会在它的地方重新建立, 继续担负原进程所处理的任务。这就使得应用程序能够不中断地处理客户请求。

- ◇ 超强的扩展性: ASP.Net 崭新的结构体系允许开发者开发自己的插件。实际上, ASP.Net 下任何一个组件都能够被用户自己开发的组件扩展或替换。

## 1.3 ASP.Net 与众不同的功能

使用一种新的 Web 技术, 除了考虑效率外, 最重要的是关心它到底提供了多少新功能, 使编写程序更加方便。下面通过实例来看看 ASP.Net 到底有多少新的功能。

### 1.3.1 ADO.Net

ASP 之所以成功, 在很大程度上是得益于微软提供的 ADO(ActiveX Data Object 数据对象)控件。ADO 提供了很大的便利去存取数据库、XML 数据。但是 ADO 也有它的局限性, 比如一个 RecordSet 只能存放一个数据表的内容, 如果需要同时操作多个表, 那就不得不打开多个 RecordSet, 这显得有些麻烦, 其他 Web 技术同样存在这种烦恼。

这个问题在 ASP.Net 可以获得很好的解决。在 ADO.Net 中, 微软提供了一个新的操作对象 DataSet, 这是一个功能十分强大的对象, 它的用法本书将在后面的章节进行讲解, 这里仅仅对其给个大概的了解, 看看如何在一个 DataSet 中储存多个数据表(DataTable)。

#### **DataSet1.aspx**

```
<% @ Page Language="C#" %>
<% @ Import Namespace="System.Data" %>
<% @ Import Namespace="System.Data.OleDb" %>
<Script Language="C#" Runat="Server">
public void Page_Load(Object src,EventArgs e)
{
    //各种 String 的设置
    string MyConnString = "Provider=Microsoft.Jet.OLEDB.4.0; Data
Source="+Server.MapPath("../DataBase\\db1.mdb")+";";
    string strSel1 = "select * from UserList";
    string strSel2 = "select * from Score";
    string strSel3 = "select * from BookMark";

    DataSet MyDataSet = new DataSet();
    OleDbConnection MyConnection = new OleDbConnection(MyConnString);
    //打开 DataAdapter
    OleDbDataAdapter MyAdapter1 = new
OleDbDataAdapter(strSel1,MyConnection);
    OleDbDataAdapter MyAdapter2 = new
```

```
OleDbDataAdapter(strSel2,MyConnection);
    OleDbDataAdapter MyAdapter3 = new
OleDbDataAdapter(strSel3,MyConnection);

    //将各表的数据,存入 DataSet
    MyAdapter1.Fill(MyDataSet,"UserList");
    MyAdapter2.Fill(MyDataSet,"Score");
    MyAdapter3.Fill(MyDataSet,"BookMark");
    //交给 DataGrid 去显示
    DataGrid1.DataSource = MyDataSet.Tables["UserList"].DefaultView;
    DataGrid2.DataSource = MyDataSet.Tables["Score"].DefaultView;
    DataGrid3.DataSource = MyDataSet.Tables["BookMark"].DefaultView;
    DataGrid1.DataBind();
    DataGrid2.DataBind();
    DataGrid3.DataBind();
}
</script>
<html>
<head>
<title></title>
</head>
<body>
<form runat="server">
<ASP:DataGrid id="DataGrid1" runat="server"
AllowPaging="True"
PageSize="5"
PagerStyle-Mode="NumericPages"
BorderColor="black"
BorderWidth="1"
GridLines="Both"
CellPadding="3"
CellSpacing="0"
Font-Name="Verdana"
Font-Size="8pt"
HeaderStyle-BackColor="#aaaadd"
AlternatingItemStyle-BackColor="#eeeeee"
/>
<hr>
<ASP:DataGrid id="DataGrid2" runat="server"
AllowPaging="True"
PageSize="4"
PagerStyle-Mode="NumericPages"
```

```

BorderColor="black"
BorderWidth="1"
GridLines="Both"
CellPadding="3"
CellSpacing="0"
Font-Name="Verdana"
Font-Size="8pt"
HeaderStyle-BackColor="#aaaadd"
AlternatingItemStyle-BackColor="#eeeeee"
/>
<hr>
<ASP:DataGrid id="DataGrid3" runat="server"
AllowPaging="True"
PageSize="4"
PagerStyle-Mode="NumericPages"
BorderColor="black"
BorderWidth="1"
GridLines="Both"
CellPadding="3"
CellSpacing="0"
Font-Name="Verdana"
Font-Size="8pt"
HeaderStyle-BackColor="#aaaadd"
AlternatingItemStyle-BackColor="#eeeeee"
/>
</form>
</body>
</html>

```

例程序显示如图 1-1 所示。

id	Username	Age	HomePage	Address
1	飞刀	20	http://www.aspcn.com	中国 湖南
2	大风	23	http://www.aspcn.com	中国 重庆
3	Zsir	26	http://www.aspcn.com	中国 福建
4	望豪	25	http://www.aspcn.com	中国 北京
5	柳丁	25	http://www.aspcn.com	中国 广西

id	Name	Class	Chinese	Math	English	Physics	History
1	王五	207	70	90	100	100	90
2	飞刀	201	70	99	100	100	100
3	李四	201	40	60	90	100	100
4	露露贝贝	202	100	100	100	100	100

id	SiteName	URL	ChineseURL	Comment
1	ASP中华网	http://www.aspcn.com	http://www.ASP中华网.com	ASP.Net,JSP,ASP教程
2	新浪网	http://www.sina.com.cn	新浪网	中国门户网站
3	搜狐	http://www.sohu.com	搜狐	搜索引擎站点
4	网易	http://www.163.com	网易	中文信息站点

图 1-1 DataSet 例

暂时不需要看懂上面的例子，只需要对 DataSet 的强大功能有个初步了解。从上面的例子可以看到，一个 DataSet 储存了三个数据表的数据，可以从容地在一个 DataSet 中操作数据。

DataSet 的功能远不止如此，将在第 6 章详细学习。

### 1.3.2 DataGrid

在上面的程序中除了看到储存功能强大的 DataSet 以外，也看到了一个新控件——DataGrid。DataGrid 控件主要用来显示数据表中的数据，同类的控件还有 Repeater 和 DataList。DataGrid 是显示数据控件中最常用，功能最强大的。在上面的例程中，可以看出 DataGrid 的强大功能。

下面代码就是对一个 DataGrid 控件的定义。

```
<ASP:DataGrid id="DataGrid3" runat="server" />
```

在其他的 Web 技术中，对于数据显示，最令人头疼的就是实现数据的分页显示。经常是一大堆长长的代码才能实现分页。而且每次分页都要重复同样的工作，这显得非常麻烦。

在 ASP.Net 中，使用 DataGrid 的控件可以把分页带来的烦恼统统抛开，只需要对 DataGrid 控件进行一些简单的设置就可以实现分页，而且还可以改变分页显示的样式。在 ASP.Net 中，分页的烦恼再也不会出现。具体方法，参见第 6 章。

### 1.3.3 丰富的 Web 控件

ASP.Net 区别于其他的 Web 技术的最明显特征就是提供了丰富 Web 控件。

在 VB 中，可以使用很多控件来完成程序。很多年来人们都向往着在 Web 设计上也能使用这些控件，现在这些梦想已经成真。现在也能在 ASP.Net 中使用这些与 VB 控件拥有相同功能的各种 Web 控件。从这里也可以看出微软公司试图将 Web 设计与软件设计融为一体。设想。

Web 控件的出现给编写程序提供了极大的便利。当熟悉 Web 控件以后，程序只是对各种控件的属性进行修改，可以将精力集中在程序的逻辑结构上，而不是物理结构上。彻底改变了其他技术，当程序很长时 HTML 与程序代码混杂而不容易分不清的毛病。

先来看 DropDownList 控件的使用例子。

#### DropDownList1.aspx

```
<% @ Page Language="C#" %>  
<% @ Import Namespace="System.Data" %>  
<Script Language="C#" Runat="Server">  
void Button1_Click(Object src,EventArgs e)  
{
```

```
string DownListValue=DownList1.SelectedItem.Text;
//在这里修改名为 Label1 的 Label 控件的 Text 值
Label1.Text=DownListValue;
}
</script>
<html>
<head>
<title></title>
</head>
<body>
<form runat="server">
<table border="1" cellpadding="0" cellspacing="0" style="border-collapse:
collapse" bordercolor="#111111" width="100%">
<tr>
<td width="100%" align="center" bgcolor="#FFFF00">DropDownList 控件演示例子
</td>
</tr>
</table>
请您选择一个青春组合:
<p>
<asp:DropDownList id="DownList1" runat="server" >
<asp:ListItem>--请选择一个青春组合--</asp:ListItem>
<asp:ListItem>中国力量</asp:ListItem>
<asp:ListItem>漂亮宝贝</asp:ListItem>
<asp:ListItem>青春美少女</asp:ListItem>
<asp:ListItem>动感男孩</asp:ListItem>
</asp:DropDownList>
</p>
<asp:Button id="Button1" Text="选择" OnClick="Button1_Click" runat="server"
/><br><br>
您选择的组合是:
<asp:Label id="Label1" Text="暂时没有选择" ForeColor="red" runat="server" />
</form>
</body>
</html>
```

这是一个极其简单的例子，运行此程序，看到如图 1-2 所示的画面。在其中选取一个青春组合，这里选择了“青春美少女”，如图 1-3 所示。

从图 1-3 可以看出，选择“青春美少女”后，DropDownList 控件依然显示“青春美少女”，这就是 Web 控件一大特征，即能完整地保持 Web 控件中数据。这也是 ASP.Net 引以自豪的地方。以前编写程序，当一个页面被提交后，不可能在<select>这种标签中保留被



选择的数据，如果想在其中显示被选择的数据，还必须逐一判断 Option 项的值是否与提交的值相等，虽然这并不是什么复杂的事情，但是重复的比较也是件麻烦的事情。

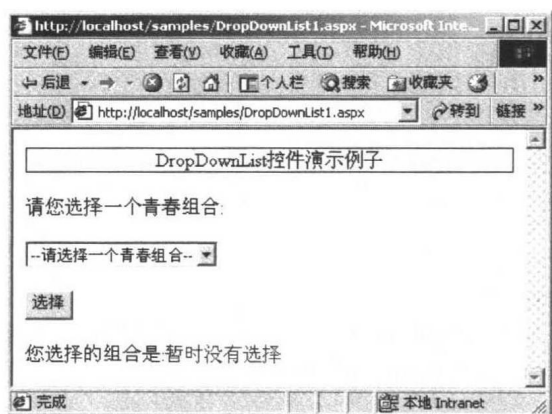


图 1-2 DropDownList 控件被选择之前

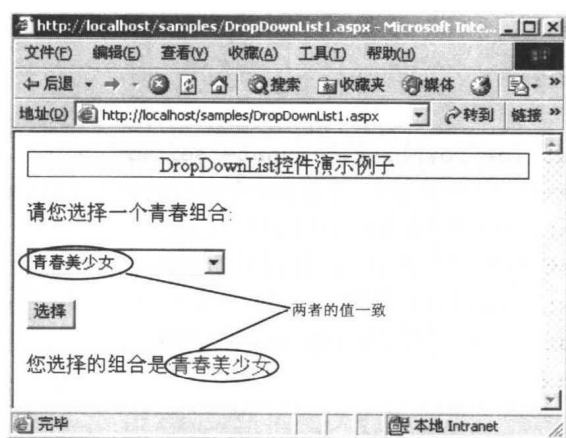


图 1-3 DropDownList 控件被选取后

这里仅仅是介绍了一个简单的 DropDownList 控件，但已经能看到 Web 控件提供的一些新功能，当然这还只是 Web 控件的一小部分，更多的控件将在后面章节继续学习。

### 1.3.4 支持事件处理(Event Handle)

既然微软公司试图将 Web 设计与软件设计结合为一个整体，那么软件设计中事件(Event)处理的思想在 ASP.Net 一定也会得到体现。

在 ASP.Net 中，会有两个最基本的事件 Page\_Load 与 Page\_UnLoad，它们分别在页面被调用与页面被释放时发生(当然这些都是发生在服务器上的)。可以在 Page\_Load 事件中初始化一些数据，比如部分控件的属性、对数据库的访问等，在 Page\_UnLoad 事件中关闭在程序中打开的数据库或者释放所占的内存空间。它们的定义如下(语言为 C#):