**[ASP.NET状态管理](http://www.cnblogs.com/xieex/archive/2007/11/28/975480.html)**

Posted on 2007-11-28 12:58 [xieex](http://www.cnblogs.com/xieex/) 阅读(6982) 评论(10) [编辑](https://i.cnblogs.com/EditPosts.aspx?postid=975480) [收藏](http://www.cnblogs.com/xieex/archive/2007/11/28/975480.html)

同样，听了微软讲师邵志东的讲座“ASP.NET状态管理”的视频讲座，总结如下，希望得到大家的指点：  
**第一部分：页面状态介绍**  
Http协议————“无状态协议”  
Web服务器每分钟对上千个用户进行管理的一种方式就是执行所谓的“无状态”链接。只要有一个希望浏览器返回一个页面、图像或其他资源的请求，就发生以下事情：  
1、连接到服务器  
2、告诉服务器想要的页面、图像或者其他项  
3、服务器发送请求的资源  
4、服务器切断连接，把用户忘的干干净净。  
也就是页面之间在Http协议下是没有任何关系的，这样就需要有状态管理来传输页面之间的数据。

WEB页面处理过程  
1、页面的一次往返处理：用户对服务器控件的一次操作，就可能引起页面的一次往返处理：页面被提交到服务器端，执行响应的事件处理代码，重建页面，然后返回到客户端。  
2、页面重建：每一次页面被请求，或者页面事件被提交到服务器，ASP.NET运行环境将执行必要的代码，重建整个页面，把结果页面送到浏览器，然后抛弃页面的变量、控件的状态和属性等等页面信息。  
3、页面处理内部过程：  
   (1)、Page\_Load:IsPostBack属性判定页面是否为第一次被请求。  
   (2)、事件处理：这一阶段处理表单的事件  
   (3)、Page\_Unload:这个阶段页面已经处理完毕，需要做些清理工作，一般地，你可以在这个阶段关闭打开文件和数据库链路，或者释放对象。

ASP.NET Web Form框架的“连续”和“有状态”假象  
访问者打开页面的连续画面，这实际上一种假象，这种假象是由ASP.NET页框架、页及其控件实现的。

**第二部分：基于客户端的状态管理：**  
状态信息放在客户端上。  
**一、视图状态（ASP.NET特有）  
二、隐藏的窗体域  
三、Cookie  
四、查询字符串**

**一、视图状态**  
ASP.ENT使用了ViewState视图状态，是所有服务器控件的一个属性。如果你查看Web Form产生的HTML代码，可以看到一个名为\_ViewState的隐藏字段，ASP.NET将状态信息以Hash的方式存储在这里，通过它，可以在下一次回发时知道回发前各控件的状态。

1http://www.cnblogs.com/Images/OutliningIndicators/None.gif<input type="hidden" name="\_\_VIEWSTATE" id="\_\_VIEWSTATE" value="" />

ASP.NET服务器控件的生命周期：  
1、初始化——Init事件（OnInit方法）  
2、加载视图状态——LoadViewState方法（当页面回发时，首先从\_ViewSatae字段信息中加载该控件的状态）  
3、处理回发数据——LoadPostData方法  
4、加载——Load事件（OnLoad方法）  
5、发送回发更改通知——RaisePostDataChangedEvent方法  
6、处理回发事件——RaisePostBackEvent方法  
7、预呈现——PreRender事件（OnPreRender方法）  
8、保存视图状态——SaveViewState方法  
9、呈现——Render方法  
10、处置——Dispose方法  
11、卸载——UnLoad事件（OnUnLoad方法）  
  
  
视图状态的用法：  
1、启用视图状态,即EnableViewState = "true"，默认为true,如果为false,那么该控件和子控件的视图状态就不会被串行化。  
2、可以在视图状态中存储的类型：Int32,bool,string,color,array,arrayList,Unit以及以上类型的HashTable对象  
3、视图状态与安全  
视图状态串行化的字符串表达式作为明文来往返传送，这是不安全的，在视图状态中绝不能保存任何信息（例如口令、连接字符、文件路径等。）  
参考例子（ViewState.aspx和ViewState\_userinfo.aspx）

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class ViewState : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            if (!IsPostBack)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Label1.Text = "EnableViewState=True";  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                //EnableViewState设为True时，Lable1的状态保存在\_viewState隐藏字段中，当刷新页面时，不执行此语句，但是可以从\_ViewState中取到以前的状态信息，所以还是原值。  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Label2.Text = "EnableViewState=False";     
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                //EnableViewState设为False时，Lable2的状态没有保存在\_viewState隐藏字段中，当刷新页面时，不执行此语句，并且取不到以前的状态信息，所以为空。  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Button2\_Click(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            if (ViewState["count"] == null)　　//从ViewState中读数据  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {　  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                ViewState["count"] = 0;　　　　//向ViewState中保存数据（可以保存多种类型的数据）  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            int ncount = (Int32)ViewState["count"];  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            ncount += 1;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            ViewState["count"] = ncount;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Label3.Text = ViewState["count"].ToString();  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Button1\_Click(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**二、隐藏的窗体域**  
1、隐藏域不会显示在用户的浏览器中，但我们可以像设置标准控制的属性那样设置其属性。当一个网页被提交给服务器时，隐藏域的内容和其他控制的值一块儿被送到HTTP Form集合中。隐藏域可以是任何存储在网页中的与网页有关的信息的存储库，隐藏域在其value属性中存储一个变量，而且必须被显性地添加在网页上。  
2、ASP.NET中的HtmlInputHidden控制提供了隐藏域的功能。

简单的说，就是页面上有一个保存值的隐藏控件，页面上可以取到它的值，但是看不见它。  
例子见（hidden.aspx）

**三、Cookie**  
Cookie定义：  
1、由网络服务器发送出来以储存在网络浏览器上小量信息；  
2、Cookie是把与用户和网站相关的信息存储比会话时间还长的一种方式。  
3、Cookie存储在用户的硬盘上（一般存储在Web浏览器软件所在的文件夹上，称为Cookies）  
用途：  
1、用户的个人配置  
2、注册和“Remember me”  
3、弹出窗口  
如何工作：  
1、Cookie存储在C:\Documents and Settings\<Username>\Cookies下  
2、IE选项中的“隐私”选项下修改cookie设置，也可以在“常规”选项卡下选择“删除cookie”  
如何使用：  
1、使用Response对象设置Cookie状态  
   Response.Cookies["UserName"].Value = "张三";  
2、使用Request对象读取已有的Cookie  
   string strName = Request.Cookies["UserName"].Value;  
3、清除：  
   Response.Cookies["UserName"].Value = null;或  
   Response.Cookies["UserName"].Expires = new System.DateTime(1999,10,12);（过期）  
属性：  
1、Value:值，是string类型的  
2、Domain:设置这个属性后，只有在这个域下才能访问该Cookie  
  例如：  
  Response.Cookies["UserName"].Domain = ".Webcast.com.cn";  
  //只有指定以“.Webcast.com.cn”结尾的域可以访问本Cookie  
3、Path:该属性指定那些路径下的页面可以访问此Cookie  
4、Expires:指定Cookie过期的日期（清除Cookie）  
  Response.Cookies["UserName"].Expires = new System.DateTime(1999,10,12);（过期）

参见实例（Cookie.aspx）  
该例子是说在页面上的TextBox中输入一个用户名，然后点“注册”按钮，此时将用户名存入Cookie中，然后在页面加载时取Cookie值显示在页面中，同时10s中自动删除Cookie。

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class Cookie : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            if (Request.Cookies["UserName"] != null)　　//注意是Request  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Response.Write("亲爱的" + Request.Cookies["UserName"].Value + "，欢迎你光临！");  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            else  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Response.Write("欢迎光临！");  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif              
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void btnZC\_Click(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Cookies["UserName"].Value = txtUser.Text;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Cookies["UserName"].Expires = DateTime.Now.AddSeconds(10);  //10秒钟后失效，即Cookies["UserName"]的信息丢失；或者手工删除（IE中选项删除Cookie）  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**四、查询字符串**  
在讲查询字符串之前先了解一下GET 和 POST方法的不同：  
   通过HTTP从Web服务器请求页面或其他资源，有两个通用的方法（GET 和 POST方法）。可使用GET方法直接获得资源，也可使用POST把值传给相应资源。GET方法是缺省的  
   假如把一个或多个成对的名称/值附在请求页面的URL后，就变成请求的查询字符串，且在QueryString集合中提供给ASP页面。单击Web页面、Email消息或其它文档的超链接，或在浏览器的地址栏中输入地址并按回车，或单击浏览器中的Links或Favorites按钮，所有这些都要使用GET方法。  
   因此，对这些动作中传递值给ASP的唯一方法是通过QueryString集合，把值附在URL后。  
   http://mysite.com/process\_page.asp?FirstName=Priscilla&LastName=Descartes  
   可以采用如下方式访问在QueryString集合中提供的值（QueryString就是下面要讲的查询字符串）：  
   strFirstName = Request.QueryString("FirstName")   ''Return "Priscilla"  
   strLastName = Request.QueryString("LastName")     ''Return "Descartes"  
   strRaw = Request.QueryString    
   Return "FirstName=Priscilla&LastName=Descartes"  
在一个页面内使用<FORM>段时，可以设置打开的FORM标记的METHOD属性值为“GET”或“POST”，缺省值为“GET”。假如使用“GET”或省略其属性，浏览器将该值绑定在页面所有控件上，成为一个查询字符串，且附在被请求页面的URL上。当这个请求到达Web服务器时，其值由ASP的Request.QueryString集合提供。然而，假如设置METHOD属性为“POST”，浏览器将值包装进发送服务器的HTTP报头中，通过Request.Form集合提供给ASP。  
   通常来说，可以在所有的HTML窗体中使用GET方法。然而，浏览器或服务器的URL字符串长度存在一定的限制。因此，附有长的字符串可能会引起溢出和某些字符串的字符被截掉。同时，查询字符串出现在浏览器的地址栏和所有的保存的链接和收藏夹中。不仅如此，还显露了通过Web服务器时在HTTP请求中不想显示的值，它也可能出现你的服务器和其他路由服务器的日志文件中。在HTTP请求报头中的值很少是可见的，并且不出现在日志文件中。  
   使用POST方法需要注意的小问题是，当用户重新下载<FORM>时，窗体的值将不再保留，其值为空且必须重新输入。然而，当附在URL上时，其值被存储为一个链接，将被保留，因此将出现在所有的URL与字符串结合的请求中，这或许是个优点也可能是个缺点，这根据应用而定（一些浏览器在客户端上能够在一定范围内自动保留一个页面上的值）。  
  
先看看get方法（利用查询字符串）：  
1、查询字符串提供了一种简单而受限制的维护状态信息的方法，我们可以方便地给那个信息从一个网页传递给另一个网页。  
2、带有查询字符串的URL如下所示：  
http://localhost:1305/QueryString\_Show.aspx?username=xieex&password=1111  
3、使用：  
string sUserName,sPwd;  
sUserName = Request.Params["username"].ToString();  
sPwd = Request.Params["password"].ToString();  
或  
sUserName = Request.QueryString["username"].ToString();  
sPwd = Request.QueryString["password"].ToString();  
或  
sUserName = Request["username"].ToString();  
sPwd = Request["password"].ToString();  
都可以取到xieex和1111  
  
其实这样的传值方式很常见也很有用，例如有些系统中，Gird中有很多条数据，我可以对某条数据进行浏览或者编辑，当点“浏览”按钮时，我弹出对话框对该条记录进行浏览，此时不允许编辑，此时我就需要从主页面上传一个状态（state）到对话框页面上来，然后在对话框页面上取其状态（Request["state"].ToString()），此时根据其值（edit或browse）就可以控制是否可以编辑了。  
  
参见实例（QueryString.aspx和QueryString\_Show.aspx）  
该例子是说在QueryString.aspx页面上注册用户名和密码，然后跳转到另一个页面上，在另一个页面上取其用户名和密码。  
在该页面上注册用户名和密码

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class QueryString : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Button1\_Click(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Redirect("QueryString\_Show.aspx?username=" + txtUser.Text + "&password=" + txtPwd.Text);  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

在另一个页面上取用户名和密码:

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class QueryString\_Show : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("用户：" + Request.Params["username"]+"<br>");  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("密码：" + Request.Params["password"]+"<br>");  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("用户：" + Request.QueryString["username"]+"<br>");//get方法时用Request.QueryString  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("密码：" + Request.QueryString["password"]+"<br>");  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("用户：" + Request["username"]+"<br>");  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("密码：" + Request["password"]+"<br>");  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

再看看post方法

直接看例子：(Post.aspx和post\_acc.aspx)  
该例子和上面例子差不多，是说在Post.aspx页面上注册用户名和密码，然后跳转到另一个页面上，在另一个页面上取其用户名和密码。

http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Post.aspx.cs" Inherits="ASPNETState.Post" %>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif<html xmlns="http://www.w3.org/1999/xhtml" >  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif<head runat="server">  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    <title>无标题页</title>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif</head>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif<body>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif     <!--注意action="post\_acc.aspx" method="post"-->  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    <form action="post\_acc.aspx" method="post">  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    <div>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        用户：<input id="txtUser" type="text" name="username" /><br />  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        <br />  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        密码：<input id="txtPwd" type="text" name="pwd" /><br />  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        <br />  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        <input id="Button1" type="submit" value="提交"  /></div>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        <!--注意按钮的type为submit-->  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    </form>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif</body>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif</html>

在另一个页面上取用户名和密码:

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class post\_acc : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            if (!IsPostBack)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Response.Write(Server.MapPath("post.aspx"));  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Response.Write("用户："　+　Request.Form["username"]);//post方法时用Request.Form  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Response.Write("密码：" 　+ Request.Params["pwd"]);  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**第三部分 基于服务器的状态管理**  
信息存储在服务器上，尽管其安全型较高，但会占用较多的web服务器资源。服务器端通常用以下方式实现状态管理：  
**一、Application对象  
二、Session对象**  
**一、Application状态**  
1、应用程序级别的状态存取（就是说服务器上的应用程序，各个客户端都可以访问它）  
2、变量状态的存储和提取  
   存储： Application["username"] = "xieex";  
   提取： string strUserName = Application["username"];  
3、同时访问要加锁,防止并发冲突  
   Application.Lock();  
   Application.Unlock();  
Application是保存在服务器内存中的。  
参见实例（ApplicationState.aspx）  
该实例用于记录访问该页面的访问者个数，用Application存储变量，这样不会因为一次会话结束而把访问记录清零。

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class ApplicationState : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application.Lock();  //加锁，防止并发，保证同一时间只有一个用户对其访问  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            if (Application["count"] != null)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Application["count"] = (Int32)Application["count"] + 1;  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            else  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif            {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif                Application["count"] = 1;  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application.UnLock();  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Write("您是第"+Application["count"]+"位访问者！");    
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif           //每访问一次都加1，只有当ＩＩＳ服务重启时才会清零  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

另一个例子（模拟网站的当前用户人数和访问总人数）  
Global.asax

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.SessionState;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.IO;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public class Global : System.Web.HttpApplication  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        /// <summary>  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        /// 必需的设计器变量。  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        /// </summary>  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        private System.ComponentModel.IContainer components = null;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        private FileStream fileStream;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        private StreamReader reader;//读字符流  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        private StreamWriter writer;//写字符流  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        public Global()  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            InitializeComponent();  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }      
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Application\_Start(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application["CurrentGuests"] = 0;//初始花为０；  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            fileStream = File.Open(Server.MapPath("counts.text"), FileMode.OpenOrCreate);//文件不存在，创建文件  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            reader = new StreamReader(fileStream);//要读取的完整路径  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application["AllGuests"] = Convert.ToInt32(reader.ReadLine()); //从当前流中读取一行字符并将数据作为字符串返回  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            reader.Close();//关闭流  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Session\_Start(Object sender, EventArgs e)//当用户访问网站时，在线用户+１，总访问数+１  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application.Lock();//同步,避免同时写入  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application["CurrentGuests"] = (int)Application["CurrentGuests"] + 1;//总在线用户数  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application["AllGuests"] = (int)Application["AllGuests"] + 1;//访问网站的总用户数  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            fileStream = new FileStream(Server.MapPath("counts.text"), FileMode.OpenOrCreate, FileAccess.ReadWrite);//  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            writer = new StreamWriter(fileStream);//实现一个写入流，使其以一种特定的编码向流中写入字符  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            writer.WriteLine(Application["AllGuests"].ToString());//把访问网站的总用户数再次写入到文件  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            writer.Close();//关闭写入流  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application.UnLock();//同步结束  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Application\_BeginRequest(Object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Application\_EndRequest(Object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Application\_AuthenticateRequest(Object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Application\_Error(Object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Session\_End(Object sender, EventArgs e)//当前用户退出网站时,在线用户数量-1,  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application.Lock();  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application["CurrentGuests"] = (int)Application["CurrentGuests"] - 1;//总在线用户数量-1  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Application.UnLock();  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Application\_End(Object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif              
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        private void InitializeComponent()  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            this.components = new System.ComponentModel.Container();  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

在页面上显示，需写代码：

http://www.cnblogs.com/Images/OutliningIndicators/None.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            this.Label1.Text = "正在访问站点的用户数：" + Application["CurrentGuests"].ToString();  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            this.Label2.Text = "访问过站点的总用户数：" + Application["AllGuests"].ToString();   
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif        }

Application对象的使用建议：  
1、对于频繁使用（很多用户都要使用的）的数据使用该对象  
2、不要把太多的信息放在该对象中  
3、如果站点有很大的通信量，建议使用Web.Config

**二、Session状态**  
对网站的一次访问叫做会话（Session），超时后，自动结束会话（一般是20分钟），Session也是保存在服务器内存中的。  
使用Session时的情况，如：  
1、购物车：网络用户决定购买的商品列表  
2、用户信息：访问者的姓名  
3、用户设置：个性化界面  
等等

ASP.NET会话状态模块在Web.config文件中像这样配置（不进行额外设置，以下是默认设置）的：  
<sessionState mode="InProc" cookieless="false" timeout="20" />  
mode属性设为InProc(默认值)，表明会话状态要由ASP.NET存储在内存中  
cookieless属性设为false,表明不用Cookie来传递会话ID,这就避免了用户禁用了Cookie，Session对象无数据可用。  
timeout属性设为20,表示登录网站后，如果20分钟不对其进行操作，则该会话结束，需要重新登录。

Session属性和方法：  
1、TimeOut属性：  
获取和设置会话结束之前的时间段，以分钟为单位，默认为20分钟  
2、Abandon()：  
结束当前会话，会话中的所有信息都被清空  
3、Clear()：  
删除当前会话中的所有信息，但不结束会话  
4、IsNewSession:  
如果会话是在用户访问页面时创建的，则这个属性返回true,当会话需要对某些数据进行初始化后才能使用时，就可以使用这个属性

参见实例（SessionState.aspx和SessionState\_Redirect.aspx）  
该例子是说在SessionState.aspx页面上注册用户名和密码，然后跳转到另一个页面上，在另一个页面上取其用户名和密码。  
在该页面上注册用户名和密码

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class SessionState : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Button1\_Click(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Session["user"] = txtUser.Text;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Session["pwd"] = txtPwd.Text;  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Response.Redirect("SessionState\_Redirect.aspx");  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

在另一个页面上取用户名和密码:

http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Data;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Configuration;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Collections;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.Security;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.WebControls.WebParts;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing System.Web.UI.HtmlControls;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifnamespace ASPNETState  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif{  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    public partial class SessionState\_Redirect : System.Web.UI.Page  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif    {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif        protected void Page\_Load(object sender, EventArgs e)  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif        {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Label1.Text = Session["user"].ToString();  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif            Label2.Text = Session["pwd"].ToString();  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

Application和Session状态的区别：  
简单的说，Application是应用程序级别的状态存储，Session是会话级别的状态存储。  
另外作用域不同，  
Application对象针对所有用户都生效  
Session对象则相反，每个用户都有自己的Session对象，它的生命周期起始于服务器产生对用户请求页面的响应，终止于用户断开与服务器的连接。  
举例来说，当手机开机时，相当于一个Applicaion开始，然后当一个朋友打电话过来，此时一个Session开始，挂电话时相当于这个Session结束，然后又有一个朋友打电话过来，此时另一个Session开始。手机未关机说明Application还未结束

由于Application和Session状态都存储在内存中，但是当服务器重新启动时，保留的状态就会消失了，为了保留其状态，就必须将状态保存到数据库。如网站计数器

**最后对各个状态进行总结：  
为了更清楚的了解,我们总结出每一种对象应用的具体环境,如下表所示:** 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 方法 | 信息量大小 | 保存时间 | 应用范围 | 保存位置 |
| Application | 任意大小 | 整个应用程序的生命期 | 所有用户 | 服务器端 |
| Session | 小量,简单的数据 | 用户活动时间+一段延迟时间(一般 为20分钟) | 单个用户 | 服务器端 |
| Cookie | 小量,简单的数据 | 可以根据需要设定 | 单个用户 | 客户端 |
| Viewstate | 小量,简单的数据 | 一个Web页面的生命期 | 单个用户 | 客户端 |
| Cache | 任意大小 | 可以根据需要设定 | 所有用户 | 服务器端 |
| 隐藏域 | 小量,简单的数据 | 一个Web页面的生命期 | 单个用户 | 客户端 |
| 查询字符串 | 小量,简单的数据 | 直到下次页面跳转请求 | 单个用户 | 客户端 |
| Web.Config文件 | 不变或极少改变的小量数据 | 直到配置文件被更新 | 单个用户 | 服务器端 |

1、ViewState对象  
    ViewState 常用于保存单个用户的状态信息，有效期等于页面的生存期。ViewState容器可以保持大量的数据，但是必须谨慎使用，因为过多使用会影响应用程序的性能。所有Web服务器控件都使用ViewState在页面回发期音保存自己的状态信息。如果某个控件不需要在回发期间保存状态信息,最好关闭该对象的ViewState，避免不必要的资源浪费。通过给@Page指令添加“EnableViewState=false”属性可以禁止整个页面的ViewState。

2、隐藏域  
  Hidden控件是属于HTML类型的服务器控件，使用此控件可以实现隐藏域的功能。其实此控件和其它服务器控件的使用没有太大区别，只是它不会在用户端的浏览器中显示，始终处于隐藏状态。但是每次页面提交的时候，此控件和其它服务器控件一同提交到服务器端，因此在服务器端可以使用Value属性获取或保存一些数据信息。

3、Cookie对象  
    Cookie用于保存客户浏览器请求服务器页面的请求信息,程序员也可以用它存放非敏感性的用户信息，信息保存的时间可以根据需要设置.如果没有设置Cookie失效日期,它们仅保存到关闭浏览器程序为止.如果将Cookie对象的Expires属性设置为Minvalue,则表示Cookie永远不会过期.Cookie存储的数据量很受限制,大多数浏览器支持最大容量为4096,因此不要用来保存数据集及其他大量数据.由于并非所有的浏览器都支持Cookie,并且数据信息是以明文文本的形式保存在客户端的计算机中,因此最好不要保存敏感的,未加密的数据,否则会影响网站的安全性。

4、查询字符串  
  查询字符串的方式是将要传递的值连接在URL后面，然后通过Response.Redirect方法实现客户端的重定向。这种方式可以实现在两个页面之间传递信息。由于URL的长度有一定的限制，因此不能传递太大的信息，加外安全性也不是很好。

5、Application对象   
    Application用于保存所有用户的公共的数据信息,如果使用Application对象,一个需要考虑的问题是任何写操作都要在Application\_OnStart事件(global.asax)中完成.尽管使用Application.Lock和Applicaiton.Unlock方法来避免写操作的同步,但是它串行化了对Application对象的请求,当网站访问量大的时候会产生严重的性能瓶颈.因此最好不要用此对象保存大的数据集合。

6、Session对象  
    Session用于保存每个用户的专用信息.她的生存期是用户持续请求时间再加上一段时间(一般是20分钟左右).Session中的信息保存在Web服务器内容中,保存的数据量可大可小.当Session超时或被关闭时将自动释放保存的数据信息.由于用户停止使用应用程序后它仍然在内存中保持一段时间,因此使用Session对象使保存用户数据的方法效率很低.对于小量的数据,使用Session对象保存还是一个不错的选择。

7、Cache对象  
   Cache对象用于在HTTP请求间保存页面或数据。该对象的使用可以极大地提高整个应用程序的效率。它允许将频繁访问的大量服务器资源存储在内存中，当用户发出相同的请求后服务器不再次处理而是将Cache中保存的信息返回给用户，节省了服务器处理请求的时间。此对象的实例是每个应用程序专用的，其生存期依赖于该应用程序的生存期。当重新启动应用程序时，将重新创建其Cache对象的实例。使用Cache对象保存信息的代码如下。   //存放信息  
  Cache["nameID"]="0001";  
  //存放信息  
  Cache.Insert("nameID","0001"1);  
  //读取信息  
  string NameID=Cache["nameID"].ToString();  
  
具体例子代码见附件：  
[ASPNETState.rar](http://files.cnblogs.com/xieex/ASPNETState.rar)