

ASP.NET动态网页设计

执教：朱亚林



南通师范高等专科学校
Nantong Normal College



ASP.NET内置对象

ASP.NET 内置对象

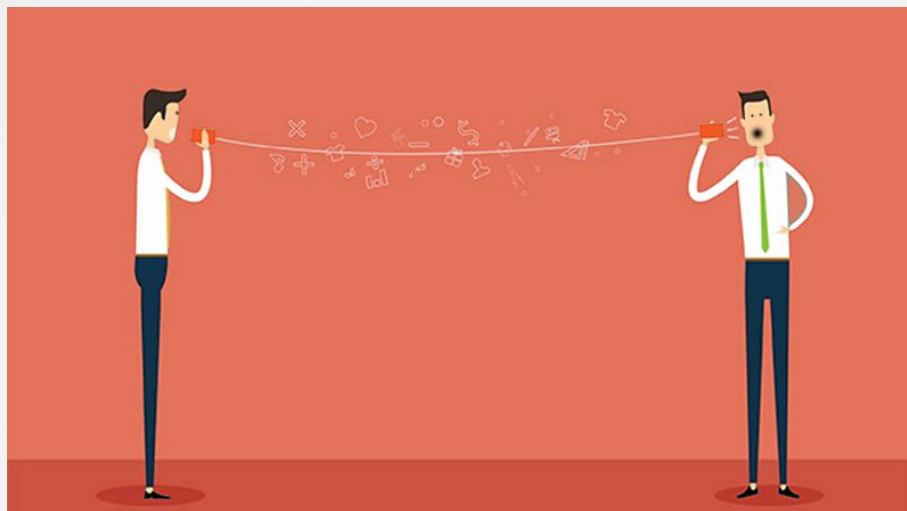
- Response对象
- Request对象
- Application对象
- Session对象
- Cookie对象
- Server对象

这些对象使用户更容易收集通过浏览器请求发送的信息、响应浏览器以及存储用户信息，以实现其他特定的状态管理和页面信息的传递。

Response对象

Response对象

- Response对象用于将数据从服务器发送回浏览器。它允许将数据作为请求的结果发送到浏览器中，并提供有关响应的信息。还可以用来在页面中输入数据、在页面中跳转，并传递各个页面之间的参数。
- 如果把用户向服务器请求信息的过程比喻成客户到柜台买商品的过程，那么在客户对商品进行了描述（发送参数）之后，售货员就会把商品摆在客户面前，而这个根据要求摆出商品的过程，就是Response对象将数据从服务器发送回浏览器的过程。



返回数据
页面跳转
传递参数

在页面中输出数据

- Response对象通过Write或者WriteFile方法将数据输出到用户页面上。这些数据可以包括页面能够呈现的所有信息。

```
char c = 'a';  
Response.Write("输出的字符是: "+c+"<br>");  
string s = "Hello,World";  
Response.Write("输出的字符串是"+s+"<br>");  
char[] cArray = {'H','e','l','l','o'};  
Response.Write("以下输出的是字符数组: ");  
Response.Write(cArray,0,cArray.Length);  
Response.Write("以下输出的是一个文件");  
Response.WriteFile(@"e:\123.txt"); // 注: 此处的输出对象必须是已经存在的
```

Response 页面跳转

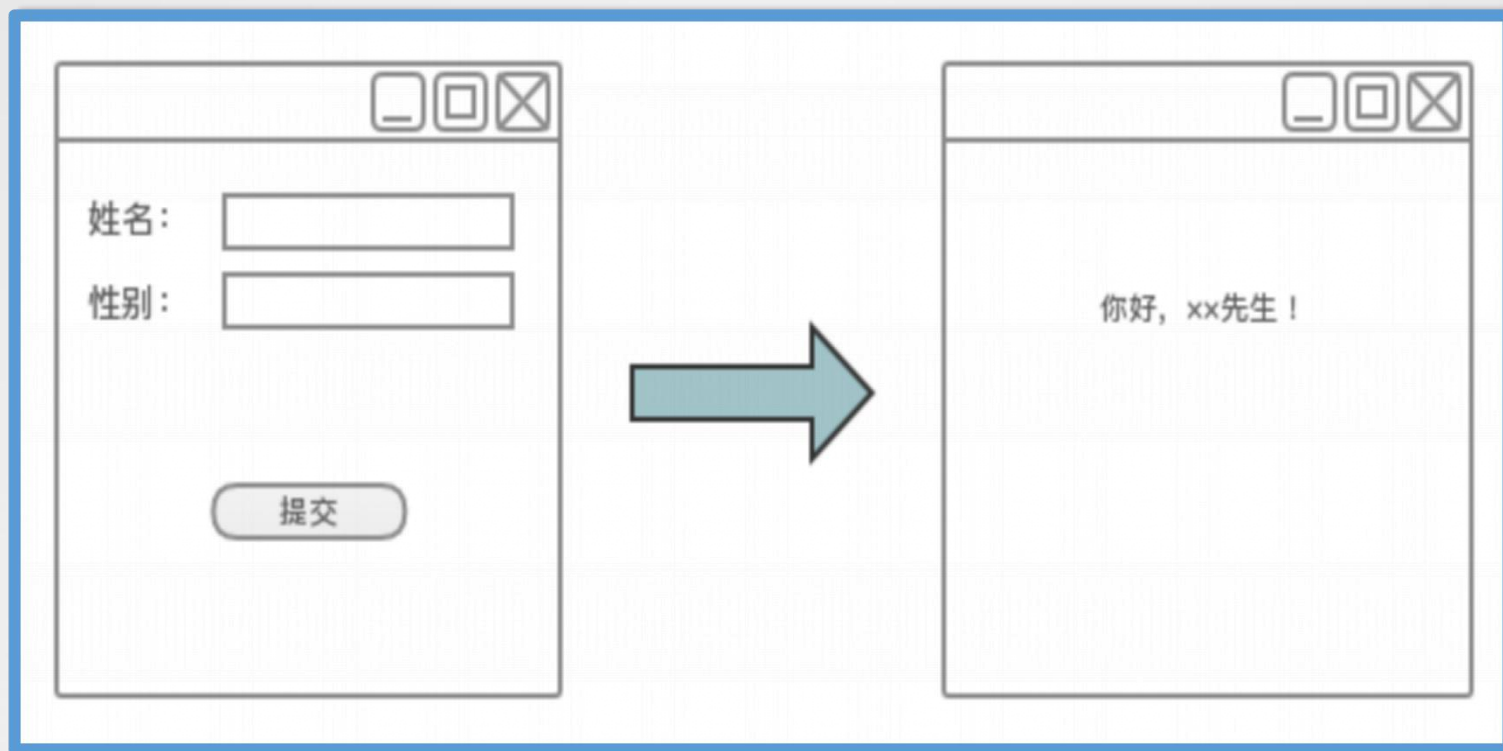
- Response对象通过Redirect方法可以实现页面重定向的功能，即在当前网址跳转到另一个网址上。同时在跳转时可以传递参数。

```
Response.Redirect("~/welcome.aspx");
```

```
Response.Redirect("~/welcome.aspx?parameter1=one&parameter2=two");
```

Response 页面跳转

- 案例：页面跳转并传递参数。



Response输出图像

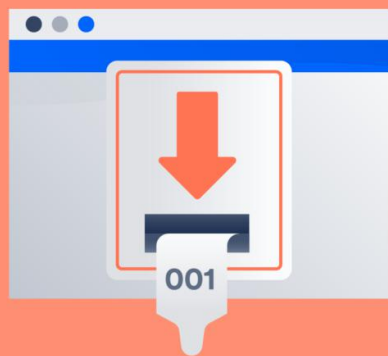
- 案例：使用BinaryWrite方法向网页输出二进制数据，如图像、图片等。

```
FileStream stream = new FileStream(Server.MapPath("pic.gif"), FileMode.Open);  
long FileSize = stream.Length;  
byte[] Buffer = new byte[(int)FileSize];  
stream.Read(Buffer, 0, (int)FileSize);  
stream.Close();  
Response.BinaryWrite(Buffer);
```

Request对象

Request对象

- Request对象用于检索从浏览器向服务器发送的请求中的信息。它提供对当前页请求的访问，包括标题、Cookie、客户端证书、查询字符串等，与HTTP协议的请求消息相对应。
- 以商场购物为例，可以描述为客户向销售员描述要购买的商品的同时，销售员也在记录客户端的描述。



Request对象

获取浏览器信息

Request.Browser.Beta

Request.Browser.Version

Request.Browser.Platform

Request.Browser.Cookies

Request.Browser.Type

检测浏览器是否Beta版本

反馈浏览器的版本号

反馈浏览器运行的平台

检测浏览器是否支持Cookies

反馈浏览器的类型

Request对象

获取客户端请求的URL中的信息

Request.UrlReferrer.Port

Request.UrlReferrer.Authority

Request.UrlReferrer.AbsolutePath

Request.UrlReferrer.Host

Request.UrlReferrer.HostNameType

获取该URL的端口号

获取主机名或IP地址和端口号

获取URI的绝对路径

获取此实例的主机部分

获取URL中指定的主机名的类型

Request对象

■ 案例：获取页面间传送的值

```
Response.Write(“使用Request[string key]方法” +Request[“value” ]+” <br/>” );
```

```
Response.Write(“使用Request.Params[string key]方法” +Request.Params[“value” ]+” <br/>” );
```

```
Response.Write(“使用Request.QueryString[string key]方法” +Request.QueryString[“value” ]+” <br/>” );
```

Request对象

■ 案例：获取客户端浏览器的信息

```
HttpBrowserCapabilities b = Request.Browser;  
Response.Write("客户端浏览器信息: ");  
Response.Write("<hr />");  
Response.Write("类型: " + b.Type + "<br>");  
Response.Write("名称: " + b.Browser + "<br>");  
Response.Write("版本: " + b.Version + "<br>");  
Response.Write("操作平台: " + b.Platform + "<br>");  
Response.Write("是否支持框架: " + b.Frames + "<br>");  
Response.Write("是否支持表格: " + b.Tables + "<br>");  
Response.Write("是否支持Cookies: " + b.Cookies + "<br>");
```

Application对象

Application对象

- Application对象用于共享应用程序级信息，即可以让多个用户共享服务器上创建的一个Application对象。
- 其工作原理是，当一个用户请求ASP.NET文件时，将启动应用程序并创建一个Application对象。一旦Application对象被创建，就可以共享和管理整个应用程序的信息。在应用程序关闭前，Application对象将一直存在。

Application对象常用方法

- 丨 Add, 新增一个Application对象变量
- 丨 Clear, 清除全部Application对象变量
- 丨 Lock, 锁定全部Application对象变量
- 丨 Remove, 使用变量名称移除一个Application对象变量
- 丨 RemoveAll, 移除全部Application对象变量
- 丨 Set, 使用变量名称更新一个Application变量的内容
- 丨 UnLock, 解除锁定的Application对象变量

Application对象变量的创建

```
Application.Lock(); //设置之前需要先锁定
```

```
Application[ "Name" ]= " Jack" ;//定义一个名为Name的Application变量
```

```
Application.Unlock(); //用完解锁
```

```
Response.Write(Application[ "Name" ].ToString()); //输出其中内容
```

制作一个在线人数计数器

- 制作思路：配置一个全局应用程序类，当新的用户访问站点时，建立一个Session对象，并触发Application对象，为其计数，对在线人数加1。当用户退出时，再次触发Application对象，并对在线人数减1。

```
void Application_Start(object sender,EventArgs e){
    Application["count"]=0;}
void Session_Start(object sender,EventArgs e){
    Application.Lock();
    Application["count"]=(int)Application["count"]+1;
    Application.Unlock();}
void Session_End(object sender,EventArgs e){
    Application.Lock();
    Application["count"]=(int)Application["count"]-1;
    Application.Unlock();}
```

```
void Page_Load(object sender,EventArgs e)
{
    Label1.Text=Application["count"].ToString()+"访
问者";
}
```