



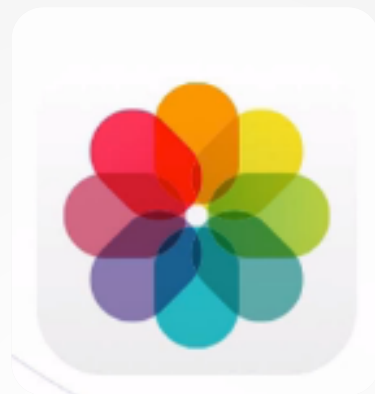
动态网站设计

执教：朱亚林



南通师范高等专科学校
Nantong Normal College





内 置 对 象

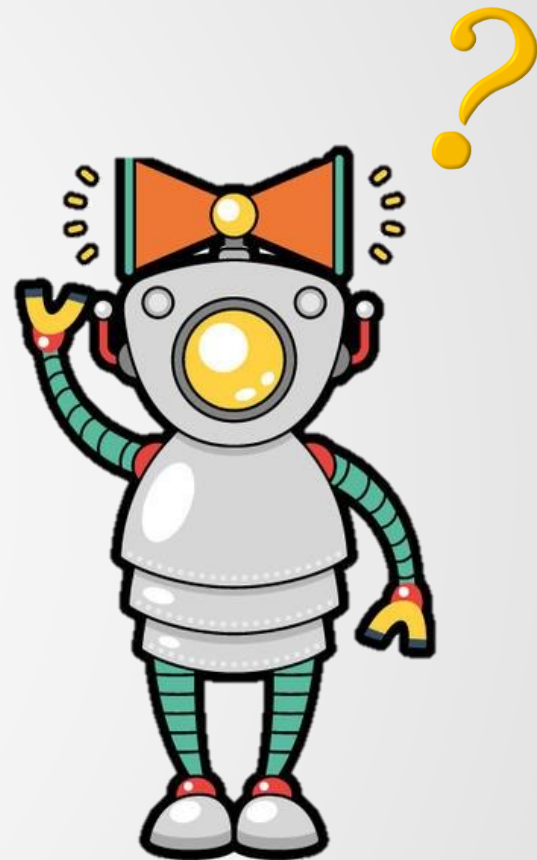


Cookie对象



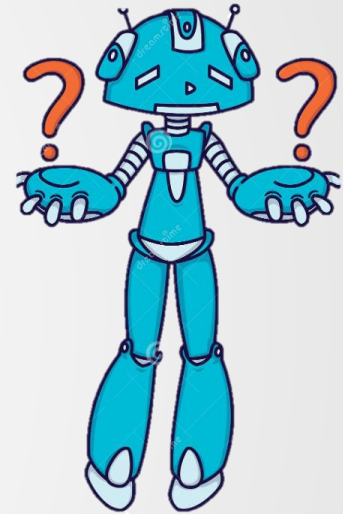
Cookie对象是用来做什么的？

- 🌐 Cookie 为 Web 应用程序保存用户相关信息提供的一种方法。使用Cookie将信息保存在客户端。
- 🌐 例如，当用户访问您的站点时，您可以利用Cookie 保存用户首选项或其他信息，这样，当用户下次再访问您的站点时，应用程序就可以检索以前保存的信息。



什么是Cookie?

- 🌐 Cookie 是一小段文本信息，伴随着用户请求和页面在 Web 服务器和浏览器之间传递。用户每次访问站点时，Web 应用程序都可以读取 Cookie 包含的信息。

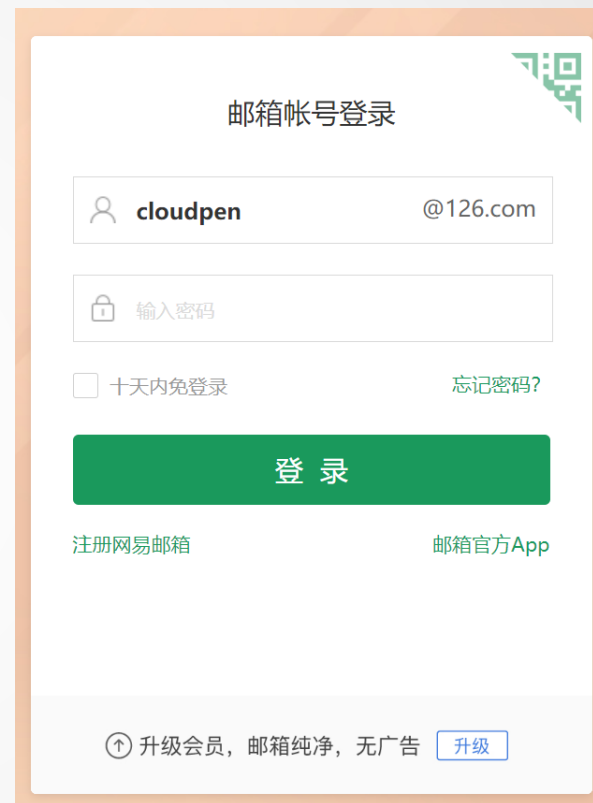


A screenshot of a Google search result for "edge for linux". The search bar shows "edge for linux" and the search button. Below the search bar, there are navigation links for "全部", "新闻", "图书", "购物", "图片", and "更多". The search results show "找到约 203,000,000 条结果 (用时 0.58 秒)". The first result is from "www.microsoftedgeinsider.com" with the title "Microsoft Edge Insider". The snippet below the title reads: "Microsoft Edge on Linux is coming soon. Want to be one the first to preview what's new? We will continue to update our Insider channels with the latest features, ...". Below the snippet, there are links for "Download", "Straight from the cutting edge", "Community", and "What's Next". A red box highlights the text "您访问过该网页很多次。上次访问日期: 20年10月8日".



Cookie应用场景

- 🌐 使用Cookie进行免登录设置
指定时间内不需要输入密码
- 🌐 使用Cookie进行投票限制
每日一票



邮箱帐号登录

cloudpen @126.com

输入密码

十天内免登录 [忘记密码?](#)

登录

[注册网易邮箱](#) [邮箱官方App](#)

⬆️ 升级会员，邮箱纯净，无广告 [升级](#)



Cookie常用属性和方法

名称	类型	说明
Expires	属性	设定Cookie变量的有效时间，默认为1000分钟。若设为0，则可以实时删除Cookie变量
Name	属性	取得Cookie变量的名称
Value	属性	获取或设置Cookie变量的内容值
Path	属性	获取或设置Cookie适用的URL
Equals	方法	确定指定Cookie是否等于当前的Cookie
ToString	方法	返回此Cookie对象的一个字符串表示形式



如何创建Cookie对象？

🌐 `Response.Cookies[varName].Value=值;`

如何调用Cookie对象？

🌐 `a=Request.Cookies[varName].Value;`

如何设置Cookie对象期限？

🌐 `Response.Cookies["cookieName"].Expires=日期;`

🌐 例如：

`Response.Cookies["cookieName"].Expires=DateTime.Now.AddMonths(1);`

`Response.Cookies["cookieName"].Expires=DateTime.Parse("10/26/2020");`



Cookie对象举例

案例：用户免密码登录

思路：用户登录成功后，再次打开页面时会自动读取已保存的Cookie信息并进行登录验证。



Cookie对象举例

```
if (TextBox1.Text == "admin" && TextBox2.Text == "admin")
{
    Session["user"] = TextBox1.Text;

    Response.Cookies["user"].Value=Session["user"].ToString();
    Response.Cookies["passwd"].Value = TextBox2.Text;
    Response.Cookies["user"].Expires = DateTime.MaxValue;
    Response.Cookies["passwd"].Expires = DateTime.MaxValue;
    Response.Cookies["passwd"].Value = TextBox2.Text;
    Response.Redirect("~/welcome.aspx");
}
```

设置Cookie

```
protected void Button2_Click(object sender, EventArgs e)
{
    for (int i = 0; i < Request.Cookies.Count; i++)
    {
        Response.Cookies[Request.Cookies[i].Name].Expires = DateTime.Now.AddDays(-1);
    }
}
```

清除Cookie

```
protected void Page_Load(object sender, EventArgs e)
{
    if (Request.Cookies["user"] != null && Request.Cookies["passwd"] != null)
    {
        if (Request.Cookies["user"].Value == "admin" && Request.Cookies["passwd"].Value == "admin")
        {
            Session["user"] = Request.Cookies["user"].Value;
            Response.Redirect("~/welcome.aspx");
        }
    }
}
```

加载Cookie

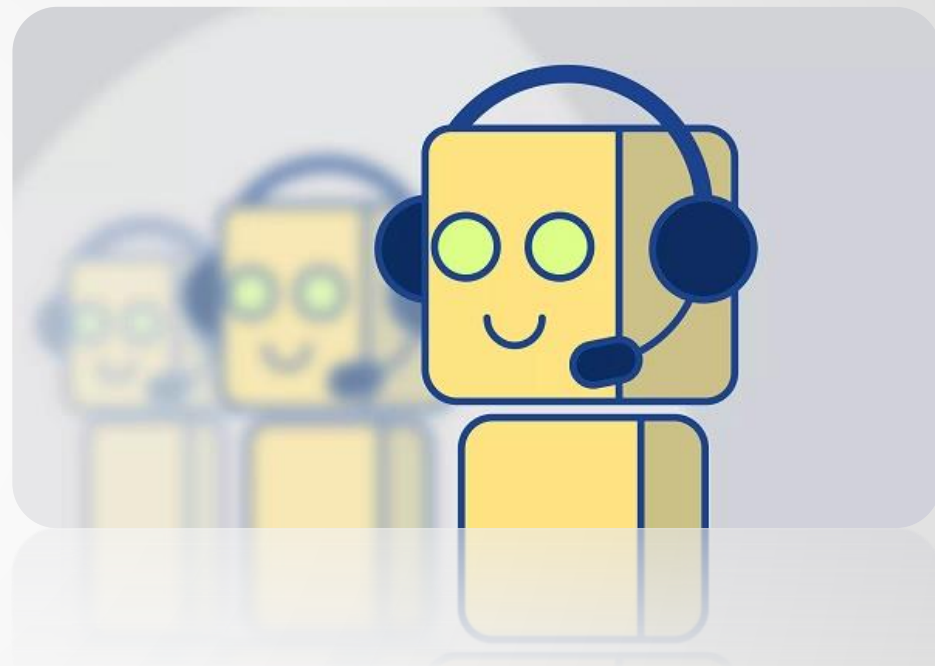


Server对象



Server对象的用途

- Server对象提供对服务器上的方法和属性的访问，获取有关服务器的信息。



Server之HtmlEncode与HtmlDecode

- ④ 如果要在网页上显示HTML标记时，若在网页中直接输出则会被浏览器识别为HTML的内容，应使用Server.HtmlEncode方法编码后输出；若要将编码后的内容恢复成原来的内容，应使用Server.HtmlDecode方法进行反编码操作



Server之MapPath

- Server对象的MapPath()方法可以将指定的相对路径映射为Web服务器端的实际物理路径。

```
string strFilePath = Server.MapPath("default.aspx");  
Response.Write(strFilePath);
```



一种跳转Execute

- ④ Execute，在当前请求的上下文中执行指定资源的处理程序，然后将控制返回给该处理程序。（执行完毕还回到原页面）



```
7 <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
8 <title></title>
9 </head>
0 <body>
1
2 <div id="content">
3
4 <%
5     Response.Write("我是页面1");
6     Server.Execute("WebForm2.aspx");
7     %>
8
9 </div>
0
1
2 <div id="foot">
3     <% Response.Write("看到我就证明页面1，执行见底了"); %>
4 </div>
5
6 </body>
7 </html>
8
```



我是页面1
我是页面2
看到我就证明页面1，执行见底了



另一种跳转Transfer

- ④ Transfer，终止当前页面的执行，并为当前请求开始执行新页。（执行到哪里，就到哪里）



```
WebForm1.aspx  + ×
http-equiv="Content-Type" content="text/html; charset=utf-8" />
</title>
</head>
<body id="content">
<%
Response.Write("我是页面1");
Server.Transfer("WebForm2.aspx");
%>
```

WebForm1.aspx执行到
Server.Transfer()方法

服务器

执行WebForm2.aspx
页面并输出

其实执行到Server.Transfer方法就终止了当前页面的执行，转向去执行一个新的页面，新页面通过URL传参指定

组成一个新页面

WebForm1里从
Server.Transfer()方法以
下的内容

WebForm2内容

上部分

下部分

