

缩略图

使用画图工具制作缩略图

```
public void ProcessRequest(HttpContext context)
{
    context.Response.ContentType = "text/plain";
    string FilePath = context.Request.MapPath("/Image/img.jpg");

    #region
    using (Bitmap map = new Bitmap(140, 140))
    {
        using (Graphics g = Graphics.FromImage(map))
        {
            using (Image img = Image.FromFile(FilePath))
            {
                g.DrawImage(img, new Rectangle(0, 0, map.Width,
map.Height));
                string newFileName = Guid.NewGuid().ToString();
                map.Save(context.Request.MapPath("/Image/" + newFileName +
".jpg"), System.Drawing.Imaging.ImageFormat.Jpeg);
            }
        }
    }
    #endregion
}
```

使用工具类完成缩略图

```
string newFileName=Guid.NewGuid().ToString();
ImageClass.MakeThumbnail(FilePath, context.Request.MapPath("/Image/"
+ newFileName + ".jpg"), 140, 140, "Cut");
context.Response.Write("111");
```

验证码

制作登录表单

```
<form method="post" action="Login.ashx">  
    用户名: <input type="text" name="txtName" /><br />  
    密码:<input type="password" name="txtPwd" /><br />  
    验证码:<input type="text" name="txtCode" /> <a href="javascript:void(0)"  
id="url"></a>  
</form>
```

使用一般处理程序生成验证码

```
public void ProcessRequest(HttpContext context)  
{  
    context.Response.ContentType = "text/plain";  
    Common.ValidateCode validateCode = new Common.ValidateCode();  
    string code=validateCode.CreateValidateCode(4);  
    validateCode.CreateValidateGraphic(code, context);  
}
```

实现点击加载验证码

```
<script type="text/javascript">  
    window.onload = function () {  
        var url = document.getElementById("url");  
        url.onclick = function () {  
            var src = document.getElementById("imgCode").src;  
            document.getElementById("imgCode").src = src+"?d="+new  
Date().getMilliseconds();  
        };  
    }  
</script>
```

下载

创建下载页面

```
<a href="aaa.rar">下载</a>  
<a href="DownLoad.ashx">下载 2</a>
```

如果 `HttpHandler` 输出的是 `html`、`txt`、`jpeg` 等类型的信息，那么浏览器会直接显示，如果希望弹出保存对话框，则需要添加 Header：`string encodeFileName = HttpUtility.UrlEncode(“过滤词.txt”); Response.AddHeader(“Content-Disposition”, string.Format(“attachment;filename=“{0}\ “”, encodeFileName));`其中 `filename` 后为编码后的文件名。`filename` 段为建议的保存文件名。http 报文中普通方法看不到的那些东西就是“报文头 (Header)”

```
context.Response.ContentType = "text/plain";  
string encodeFileName = "aaa.txt";  
context.Response.AddHeader("Content-Disposition",  
string.Format("attachment;filename=\"{0}\"", encodeFileName));//在响应头中加上  
Content-Disposition, attachment 表示以附件形式下载。  
context.Response.WriteFile("aaa.txt");
```