

# ASP.NET 平台多文件的上传和下载 \*

## Multiple files Uploading and Downloading in ASP.NET

商 杰 朱忠旭 (安徽工贸职业技术学院计算机系,安徽 淮南 232001)

### 摘 要

文件的上传和下载是网站最常用的功能。以项目实战方式利用动态 HTML 语言 (DHTML) 结合客户端 Javascript 脚本动态产生控件。并用 C# 语言编写后台代码真正实现 ASP.NET 多文件上传和下载,为 ASP.NET 平台中文件上传和下载的处理提供了一个编程框架。

**关键词:** ASP.NET, DHTML, C#, 上传, 下载

### Abstract

Multiple files uploading and downloading is the site most commonly used function. This paper generates the controls by using dynamic HTML language (DHTML) combined with JavaScript client script. This paper uses C# language to realize ASP.NET multiple file uploading and downloading. And it provides a framework for the processing of the file uploading and downloading in ASP.NET platform.

**Keywords:** ASP.NET, DHTML, C#, Upload, Download

上传是将文件按照一定的格式和规范,放到服务器指定的目录,同时为了保障网站服务器的安全,上传文件时可以限定上传文件的类型、大小。下载是将网站服务器上的共享文件保存到浏览者的计算机中。文件的上传和下载是网站最常用的功能,在很多网站中,都提供了多文件的上传和指定目录的多文件显示由用户选择下载。ASP.NET 提供了实现文件上传和下载的方法。

#### 1 多文件的上传

1) 在 .NET 平台上新建一个项目,在项目中添加一个 Web 页面“multifileupload.aspx”,设计一个多文件上传功能界面,如图 1 所示。

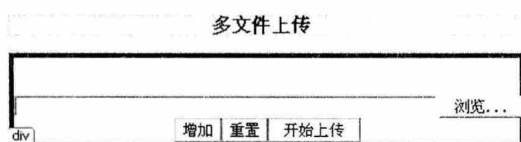


图 1 多文件上传功能界面

2) 添加增加按钮的客户端事件, <input type="button" value="增加" onclick="addFile()">, 动态添加文件上传标记, 如代码清单 1 所示。(注意 MyFile 是功能界面中的 div 的 id 值)

代码清单 1: “增加”按钮的客户端事件代码

```
var str = '<br/><br/><INPUT type="file" size="50" NAME="File">'  
MyFile.innerHTML=str+MyFile.innerHTML
```

3) 运行程序调度,单击页面中的“增加”按钮,添加文件上传控件,运行后的界面如图 2 所示。

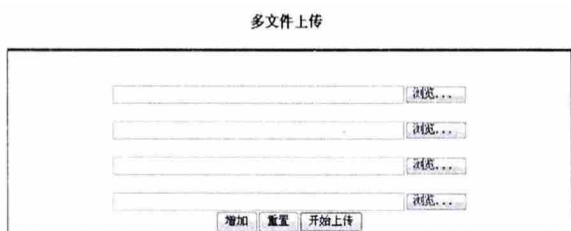


图 2 多文件上传的运行界面

4) 添加“开始上传”按钮事件 UploadButton\_Click, 实现多文件上传功能。如代码清单 2 所示。

代码清单 2: 多文件上传功能代码

```
//遍历 File 表单文件  
HttpFileCollection files = HttpContext.Current. Request.Files;  
//状态信息  
System.Text.StringBuilder strMsg = new System.Text.StringBuilder();  
strMsg.Append("<br><br><br>上传的文件分别是:<br color=red><br>");  
try  
{ for (int iFile = 0; iFile < files.Count; iFile++)  
    { //检查文件扩展名字  
      HttpPostedFile postedFile = files[iFile];  
      string fileName = Path.GetFileName (postedFile.FileName);  
      if (fileName != "")  
      { strMsg.Append(fileName + "<br>");  
        postedFile.SaveAs(Server.MapPath("~/uploadfile/") + fileName);  
        showResult.Text = strMsg.ToString();  
      }  
    }  
  catch (System.Exception Ex)  
  { showResult.Text = Ex.Message; }
```

相关知识: HttpContext 类封装了有关 HTTP 请求的所有 HTTP 特定的信息,也叫上下文信息。该类从客户端用户单击并产生了一个向服务器发送请求开始,到服务器处理完请求并生成返回到客户端为止,针对每个不同用户的请求,服务器都会创建一个新的 HttpContext 实例直到请求结束,服务器销毁这个实例。该类对 Request、Response、Server 等都进行了封装,并保证在整个请求周期内都可以随时随地地调用。

#### 2 文件的下载

文件上传后,浏览者能查看到上传的文件信息,并通过单击文件名进行文件下载。

1) 在项目中添加“downloadfiles.aspx”Web 页面,在页面

中添加一个显示文件列表的 Panel 控件,在控件上方添加提示信息“文件显示清单”,添加 Page\_Load 事件,如代码清单 3 所示。

代码清单 3:Page\_Load 事件代码

```
bool filetype = false;
string filepath = Server.MapPath("~/uploadfile/");
// 创建目录对象
DirectoryInfo Dirfile = new DirectoryInfo(filepath);
// 获取存放文件目录中所有文件的信息
FileInfo[] files = Dirfile.GetFiles();
for (int i = 0; i < files.Length; i++)
{
    // 过滤掉上传文件类型以外的文件
    string[] filetype = { ".doc", ".jpg", ".gif", ".xls" };
    string fileExt = Path.GetExtension(filepath + files[i].Name).ToLower();
    for (int j = 0; j < filetype.Length; j++)
    {
        if (fileExt == filetype[j]) { filetype = true; break; }
    }
    // 显示文件,并可以通过单击文件名转到下载页面进行下载
    if (filetype)
    {
        Panel1.Controls.Add(new LiteralControl("<a href= download.aspx?name="+files[i].Name+">" + files[i].Name + "</a>"));
        Panel1.Controls.Add(new LiteralControl("<br/>"));
    }
    filetype = false;
}
```

2) 在项目中添加下载文件的 Web 页面“download.aspx”,页面中不添加任何控件,在 Page\_Load 事件添加实现下载功能代码,如代码清单 4 所示。

代码清单 4:Page\_Load 事件代码

```
string filepath = Server.MapPath("~/uploadfile/");
// 获取文件清单页面传过来的文件名
string selectname = Request.QueryString["fname"];
// 调用下载文件的方法
savefile(filepath, selectname);
```

3) 添加 savefile 方法,如代码清单 5 所示。

代码清单 5:savefile 方法代码

```
void savefile(string filePath, string fileName) {
    string path = filePath + fileName;
    FileInfo fileInfo = new FileInfo(path);
    HttpContext.Current.Response.Clear();
    HttpContext.Current.Response.ClearContent();
    HttpContext.Current.Response.ClearHeaders();
    // 加上 UriEncode() 方法,防止文件下载时,文件名乱码
    HttpContext.Current.Response.AddHeader("Content-Disposition", "attachment; filename=" + HttpUtility.UrlEncode(fileName));
    HttpContext.Current.Response.AddHeader("Content-Length", fileInfo.Length.ToString());
    HttpContext.Current.Response.AddHeader("Content-Transfer-Encoding", "binary");
    HttpContext.Current.Response.ContentType = "application/octet-stream";
    HttpContext.Current.Response.ContentEncoding = System.Text.Encoding.GetEncoding("gb2312");
    HttpContext.Current.Response.WriteFile(fileInfo.FullName);
}
```

(上接第 41 页)

N 为所设倍率。如倍率 150% 对应转换表中的数据就应为:200-150=105。

对于上述计算公式的不同,如切削进给倍率 128% 是分界线,是因为一个字节用有符号数表示时数的范围为 -127 ~ +128,而在 FANUC 系统中功能指令数据表值的范围也是 -127 ~ +128,超过这个范围的数不能输入数控系统,所以在转换数据表中的数据根据数的范围不同,计算方法不同。

```
HttpContext.Current.Response.Flush();
HttpContext.Current.Response.End();
```

4) 运行“showfiles.aspx”Web 页面,进入如图 3 所示的文件列表显示页面。选择某个文件并单击,运行弹出如图 4 所示的“文件下载”对话框,在对话框中单击“保存”按钮,在弹出的“文件保存”对话框中保存文件到指定的目录。

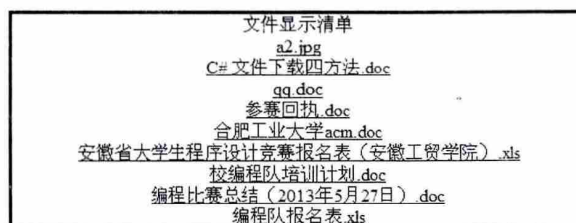


图 3 文件列表显示页面

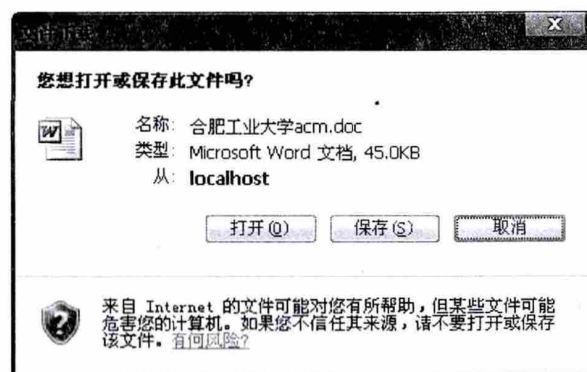


图 4 “文件下载”对话框

### 3 结束语

本文以项目实战方式利用动态 HTML 语言 (DHTML) 结合客户端 Javascript 脚本动态产生控件。并用 C# 语言编写后台代码真正实现 ASP.NET 多文件上传和下载,为 ASP.NET 平台中文件上传和下载的处理提供了一个编程框架。

### 参考文献

- [1] 李伟民,何伟,李平,等.基于 Web 的 SQL Server 数据库存取图片的 Delphi 实现[J].计算机工程与设计,2007(12):2944-2945
- [2] 王正,罗万明,阎保平.并行下载最优机制[J].软件学报,2009(8):2256-2257
- [3] 汤晓燕,王磊,等.ASP.NET 中基于流技术的二进制图片上传与显示[J].苏州市职业大学学报,2010(3):63-66
- [4] 龚赤兵.ASP.NET 2.0 网站开发案例教程[M].北京:水利水电出版社,2009
- [5] 杨兴鸿.基于 XML 的网络相册的设计与实现[D].北京:北京邮电大学,2010
- [6] 白金生.ASP.NET 下动态产生控件实现多文件随意上传[J].计算机系统应用,2007(10):120-123

[收稿日期:2013.7.5]

### 参考文献

- [1] 刘江,卢鹏程,许朝山.FANUC 数控系统 PMC 编程[M].北京:高等教育出版社,2011
- [2] 刘永久.数控机床故障诊断与维修技术(FANUC 系统)[M].北京:机械工业出版社,2012
- [3] 龚仲华.FANUC Oi 倍率信号的 PLC 控制[J].机床电器,2006(1)

[收稿日期:2013.9.9]