

File Upload in ASP.NET

ASP.NET Web Forms
Telerik Software Academy
<http://academy.telerik.com>



Table of Contents

- ◆ File Upload Concepts
- ◆ Multipart HTTP Requests
- ◆ File Upload in ASP.NET Web Forms
 - ◆ To server file
 - ◆ To memory stream
- ◆ Using Kendo UI Upload
- ◆ Using Telerik ASP.NET AJAX



File Upload Concepts

How files are sent through the web



File Upload Concepts

- ◆ Done through forms
- ◆ Content type should be “multipart/form-data”
- ◆ Selected via input type “file” tag
- ◆ Data is sent as POST method through http
- ◆ Files are received and used at the server

```
<form id="FileUploadForm" runat="server">
    <asp:FileUpload ID="FileControl" runat="server" />
    <asp:Button runat="server" ID="UploadButton"
        Text="Upload" OnClick="Upload" />
</form>
```

Multipart HTTP Request

Different kind of POST method



Multipart HTTP Request

- ◆ Form is sent as multiple parts
- ◆ Enable multipart – type “multipart/form-data”
- ◆ Request is divided in parts

```
Content-Type: multipart/form-data; boundary=-----  
-----7d81b516112482
```

```
Content-Length: 324
```

```
-----7d81b516112482
```

```
Content-Disposition: form-data; name="file";  
filename="Exam solution.doc"
```

```
Content-Type: text/plain
```

```
SomeText
```

```
-----7d81b516112482
```

```
Content-Disposition: form-data; name="submit"
```

Multipart HTTP Request

Live Demo

ASP.NET Web Forms File Upload

Done within minutes

ASP.NET
MVC

ASP.NET
Web Forms

ASP.NET
Web Pages

ASP.NET Framework

.NET Framework

- ◆ **Request.Files**

- ◆ Gets the collection of sent files (dictionary)
 - ◆ Uses MIME format

- ◆ **HttpPostedFile**

- ◆ Provides access to individual files
 - ◆ **string FileName** property
 - ◆ **int ContentLength** in bytes
 - ◆ **InputStream** – stream of the file
 - ◆ **SaveAs(string fileName)** – saves the file

◆ FileUpload

- ◆ **bool HasFile** – checks if file is uploaded
- ◆ **byte[] FileBytes** – file content in binary
- ◆ **Stream FileContent** – file as stream
- ◆ **string FileName** – name of the file
- ◆ **HttpPostedFile PostedFile** – posted file
- ◆ **IList<HttpPostedFile> PostedFiles** – collection of all uploaded files

- ◆ Using <asp: FileUpload />

```
<asp:FileUpload ID="FileControl" runat="server" />
```

- ◆ In the code behind

```
if (FileControl.HasFile)
{
    string filename = Path.GetFileName
        (FileControl.FileName);

    FileControl.SaveAs(Server.MapPath
        ("~/Uploaded_Files/") + filename);
}
```

Simple File Upload

Live Demo

Restricted File Upload

- ◆ You can check for certain file types

```
if (FileControl.PostedFile.ContentType == "image/jpeg")
{
    // Use file
}
else
{
    // do not upload
}
```

- ◆ You can check for file size

```
if (FileControl.PostedFile.ContentLength < 102400)
{
    // Use file
}
...
...
```

Restricted File Upload

Live Demo

◆ Saving file to stream

```
byte[] fileData = null;
Stream fileStream = null;
int length = 0;

if (FileUploadControl.HasFile)
{
    length = FileControl.PostedFile.ContentLength;
    fileData = new byte[length + 1];
    fileStream = FileControl.PostedFile.InputStream;
    fileStream.Read(fileData, 0, length);
}

MemoryStream stream = new MemoryStream(fileData);
```

Uploading To Stream

Live Demo

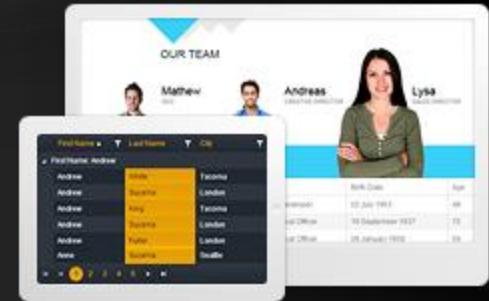
Kendo UI Upload Widget

Asynchronous and large files support



◆ Kendo Upload widget

- ◆ Easily done
- ◆ Have upload and delete
- ◆ Styled and ready to use
- ◆ Supports asynchronous file upload
- ◆ Supports multiple file upload at once
- ◆ Supports sending large files in parts



- ◆ Steps for asynchronous large multi-upload
 - ◆ Create landing page for the widget
 - ◆ Add input type “file” (no form required)
 - ◆ Add JavaScript for the widget
 - ◆ Add upload handler
 - ◆ Add remove handler (if you need it)
 - ◆ In web.config set request max length

```
<httpRuntime targetFramework="4.5"  
           executionTimeout="110" maxRequestLength="25000" />
```

◆ Landing page

```
<input name="uploaded" id="uploaded" type="file"
runat="server" />

<script>
    $(document).ready(function () {
        $("#uploaded").kendoUpload({
            async: {
                saveUrl: "Upload",
                removeUrl: "Remove",
                autoUpload: true,
            }
        });
    });
</script>
```

- ◆ Upload page
- ◆ Get name of the input from Request.Files
- ◆ Return empty response if successful

```
Response.Expires = -1;  
HttpPostedFile file = Request.Files["uploaded"];  
  
string savepath = Server.MapPath("~/Uploaded_Files/");  
string filename = file.FileName;  
file.SaveAs(savepath + filename);  
  
Response.ContentType = "application/json";  
Response.Write("{}");
```

- ◆ Remove page
- ◆ Get file from Request.Form["fileNames"]
- ◆ If file exists – delete it

```
var file = Request.Form["fileNames"];

if (file != null)
{
    var fileName = Path.GetFileName(file);
    var physicalPath = Path.Combine
        (Server.MapPath("~/Uploaded_Files"), fileName);

    if (File.Exists(physicalPath))
        File.Delete(physicalPath);
}
```

Kendo UI Upload Widget

Live Demo

Telerik ASP.NET AJAX

Designed for Web Forms



- ◆ Steps for asynchronous file upload
 - ◆ Install ASP.NET AJAX Controls
 - ◆ Add RadAsyncUpload control to a page
 - ◆ Add RadScriptManager (in web.config too)
 - ◆ Set MultipleFileSelection
 - ◆ Set TargetFolder
 - ◆ Add RadButton or normal ASP submit button

Telerik ASP.NET AJAX

Live Demo

Questions?

1. Create a Web Forms upload page using Kendo UI / ASP.NET AJAX Controls for archive with text files. The upload must work only with .zip file extension. You should not write the zip file into the server file system. Save the file to the memory and using external library, extract the text files and save their content into the database. The text files should not be saved to the file system either.

Free Trainings @ Telerik Academy

- ◆ C# Programming @ Telerik Academy

- ◆ csharpfundamentals.telerik.com



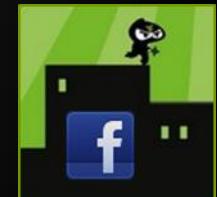
- ◆ Telerik Software Academy

- ◆ academy.telerik.com



- ◆ Telerik Academy @ Facebook

- ◆ facebook.com/TelerikAcademy



- ◆ Telerik Software Academy Forums

- ◆ forums.academy.telerik.com

