

Test Watermark

حضرت خواجہ سیدنا معین الدین حسن چشتی سنجاری اجمیری رحمۃ اللہ علیہ



Driving mobility
Mitsubishi Electric

HOME

BLOGS

FORUMS

VIDEOS

SEARCH

INTERVIEW

COMMUNITY

CONTACT

1



Ex Send Email ASP.Net

ASP.Net Core Razor Pages: Display PDF files from Database in View

Posted: 01 Jul 2022 Author: [Mudassar Khan](#) Views: 11500

ASP.Net

JavaScript

SQL Server

HTML5

PDF

Core

Razor Pages

[Download Code](#)

Free Spire.Office for .NET Community Edition

Free Spire.Office for .NET is a Community Edition of the
Spire.Office for .NET API

Disclaimer

Free Edition

This site makes use of Cookies. Please refer [Privacy Policy](#) for more details.

Allow

Reject

In this article I will explain with an example how to upload and display PDF files from Database

PDF.js plugin

The PDF file will be displayed (rendered) in Browser using [PDF.js](#) JavaScript plugin.


The following files of [PDF.js](#) JavaScript plugin will be used.

1. pdf_viewer.min.css
2. pdf.min.js
3. pdf.worker.min.js

2

Database

This article makes use of a table named tblFiles whose schema is defined as follows.

MUDASSAR-PC\SQL20...les - dbo.tblFiles			
	Column Name	Data Type	Allow Nulls
	id	int	<input type="checkbox"/>
	Name	varchar(50)	<input type="checkbox"/>
	ContentType	nvarchar(200)	<input type="checkbox"/>
	Data	varbinary(MAX)	<input type="checkbox"/>

Note: You can download the database table SQL by clicking the download link below.

[Download SQL file](#)

Namespaces

You will need to import the following namespaces:

```
using System.IO;  
using System.Data.SqlClient;  
using System.Collections.Generic;  
using Microsoft.AspNetCore.Mvc;  
using Microsoft.Extensions.Logging;
```

Disclaimer

This site makes use of Cookies. Please refer [Privacy Policy](#) for more details.

Allow

Reject

the records of the Files from database.

```
public class FileModel
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string ContentType { get; set; }
    public byte[] Data { get; set; }
}
```

3

Razor PageModel (Code-Behind)

The PageModel consists of three Handler methods.

Handler method for handling GET operation

Inside this Handler method, the GetFiles method is called and the public property Files is set.

Inside the GetFiles method, the records from the tblFiles table are fetched using DataReader and generic list of FileModel class objects is populated.

Note: For details about reading Connection String from AppSettings.json, please refer my article, Net Core: Read Connection String from AppSettings.json file.

Finally, the generic list of FileModel public property is returned to the Razor Page.

Handler method for handling POST operation for uploading Files

This Handler method gets called when a File is selected and the Upload Button is clicked, and it gets the uploaded file in the IFormFile

Note: In case the IFormFile parameter is null, IFormFile always returns null.

The uploaded file is converted to a byte array and inserted into the database table.

Disclaimer

This site makes use of Cookies. Please refer [Privacy Policy](#) for more details.

[Allow](#)[Reject](#)

After successful insert of the File, RedirectToPage is called which redirects to the IndexModel Get

When the View Link inside the HTML Table (Grid) is clicked, then the ID of the particular PDF file is sent to this Handler method and using the File ID, the File's binary data is fetched from the database.

Note: The following Handler method performs File display and hence the return type is set to Json Result. For more details, please refer my article Using jQuery AJAX in ASP.Net Core Razor Pages.

Finally, JSON is returned back to the jQuery AJAX function.

4

```
public class IndexModel : PageModel
{
    public List<FileModel> Files { get; set; }
    private IConfiguration Configuration;

    public IndexModel(IConfiguration configuration)
    {
        Configuration = _configuration;
    }

    public void OnGet()
    {
        this.Files = this.GetFiles();
    }

    public IActionResult OnPostUploadFile(IFormFile postedFile)
    {
        string fileName = Path.GetFileName(postedFile.FileName);
        string contentType = postedFile.ContentType;
        using (MemoryStream ms = new MemoryStream())
        {
            postedFile.CopyTo(ms);
            string content = ms.ToArray();

            using (SqlConnection conn = new SqlConnection(
                "Data Source=.;Initial Catalog=PDFFiles;Integrated Security=True;"))
            {
                conn.Open();
                string sql = "INSERT INTO PDFFiles (FileName, Content) VALUES (@FileName, @Content)";
                SqlCommand cmd = new SqlCommand(sql, conn);
                cmd.Parameters.AddWithValue("@FileName", fileName);
                cmd.Parameters.AddWithValue("@Content", content);
                cmd.ExecuteNonQuery();
            }
        }
    }
}
```

Disclaimer

This site makes use of Cookies. Please refer [Privacy Policy](#) for more details.

Allow

Reject

```
string query = "INSERT INTO tblFiles VALUES (@Name, @Con
```

```

        cmd.Connection = con;
        cmd.Parameters.AddWithValue("@Name", fileName);
        cmd.Parameters.AddWithValue("@ContentType", contentType);

        cmd.Parameters.AddWithValue("@Data", ms.ToArray());
        con.Open();
        cmd.ExecuteNonQuery();
        con.Close();
    }
}
}

```

```
return RedirectToPage("Index");
```

```
}
```

```
public JsonResult OnPostGetPDF(int fileId)
```

```
{
```

```
    byte[] bytes;
```

```
    string fileName, contentType;
```

```
    string constr = this.Configuration.GetConnectionString("MyCon
```

```
n");
```

```
    using (SqlConnection con = new SqlConnection(constr))
```

```
    {
```

```
        using (SqlCommand cmd = new SqlCommand())
```

```
        {
```

```
            cmd.CommandText = "SELECT Name, Data, ContentType FROM t
```

```
tblFiles WHERE Id=@Id";
```

```
            cmd.Parameters.AddWithValue("@Id", fileId);
```

```
            cmd.Connection = con;
```

```
            con.Open();
```

```
            using
```

```
            {
```

```
                sd
```

```
                by
```

```
                co
```

```
                fi
```

Disclaimer

This site makes use of Cookies. Please refer [Privacy Policy](#) for more details.

Allow

Reject