FPT Education

FPT POLYTECHNIC

LẬP TRÌNH C# 6

BÀI 5: WEB API UPLOAD/ DOWNLOAD FILES

P5.1

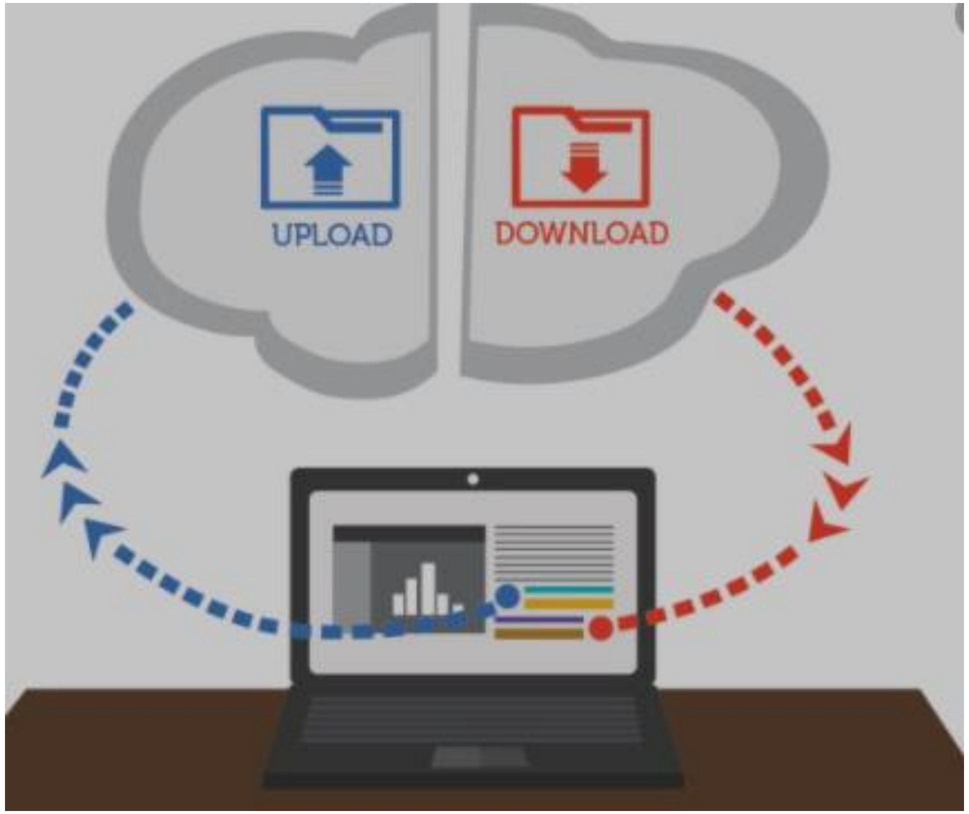
MỤC TIÊU

- Upload file
- Database và File System
- Upload File System-Database
- Download file

Upload file

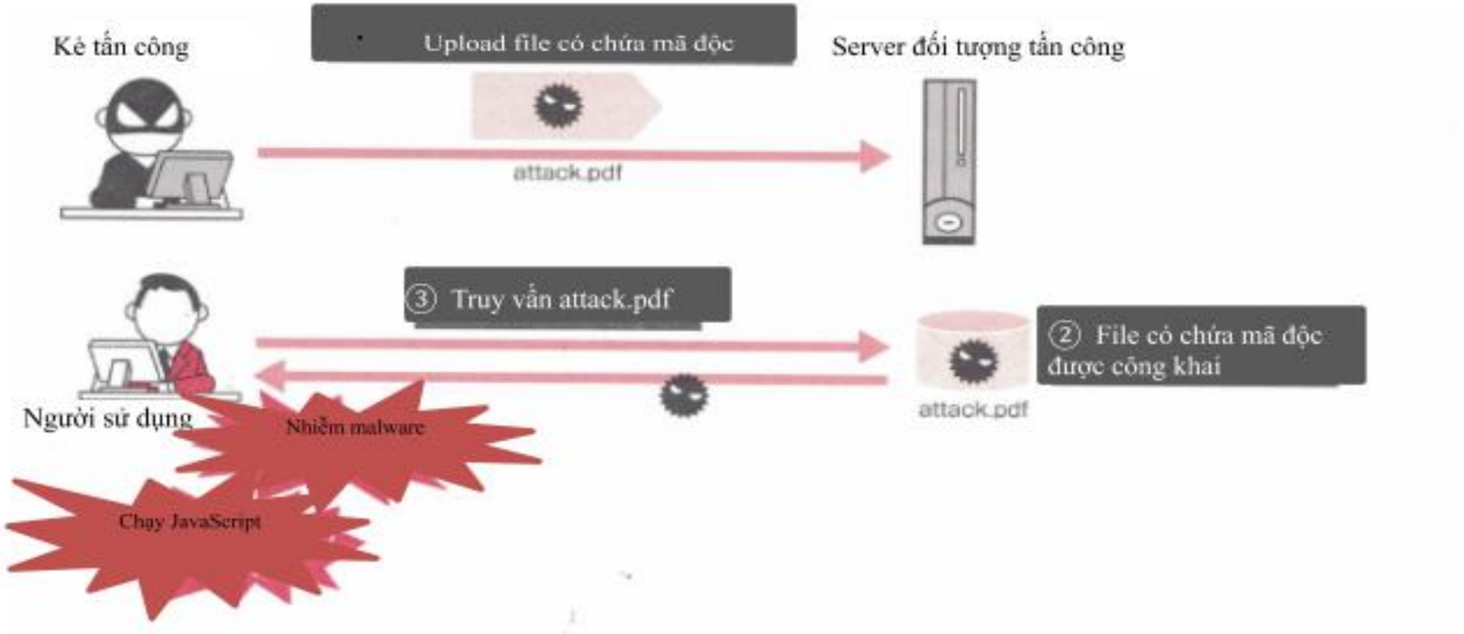
❑ upload files luôn là yêu cầu chính đối với nhiều ứng dụng web

❑ Lưu trữ dữ liệu upload trong system file - database

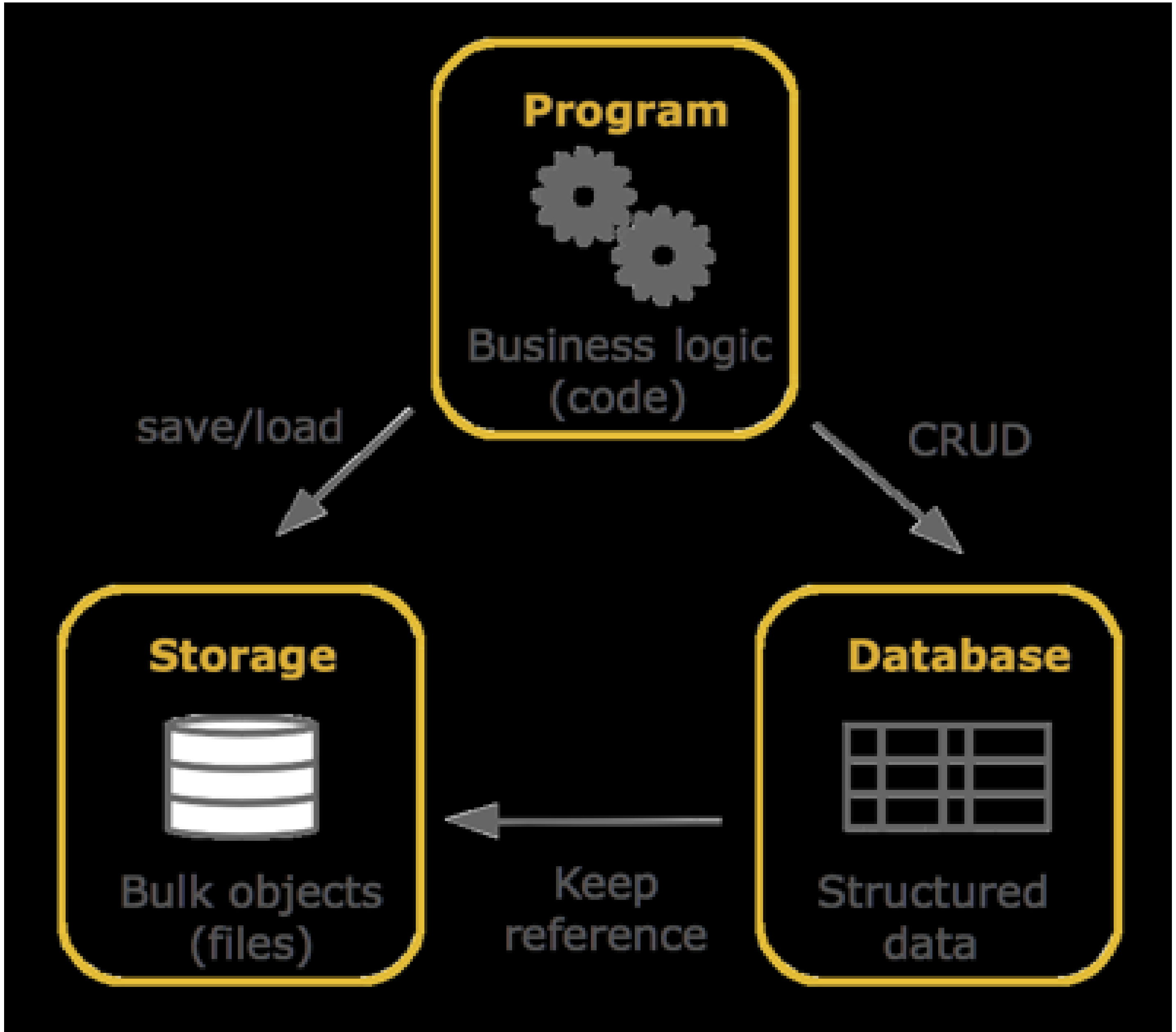


UPLOAD FILE

- ❑ DoS đối với chức năng upload
- ❑ Chạy file Script trên Server
- ❑ Người sử dụng download file có chứa mã độc
- ❑ Download vượt quá quyền hạn của file
- ❑ XSS bằng file hình ảnh



DATABASE VÀ FILE SYSTEM?



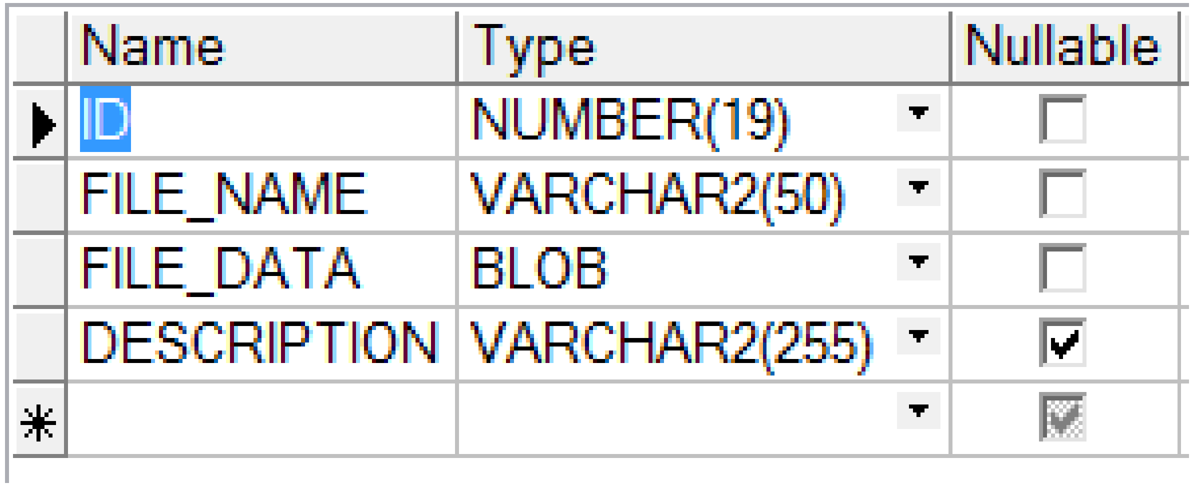
FILE SYSTEM?

- ❑ Performance
- ❑ Saving
- ❑ Migrating
- ❑ Cost
- ❑ Cloud storage
- ❑ Packed
- ❑ Security

DATABASE ?

- ❑ ACID
- ❑ Backups
- ❑ Secure
- ❑ Blob
- ❑ Edit files
- ❑ Memory
- ❑ Bandwidth db

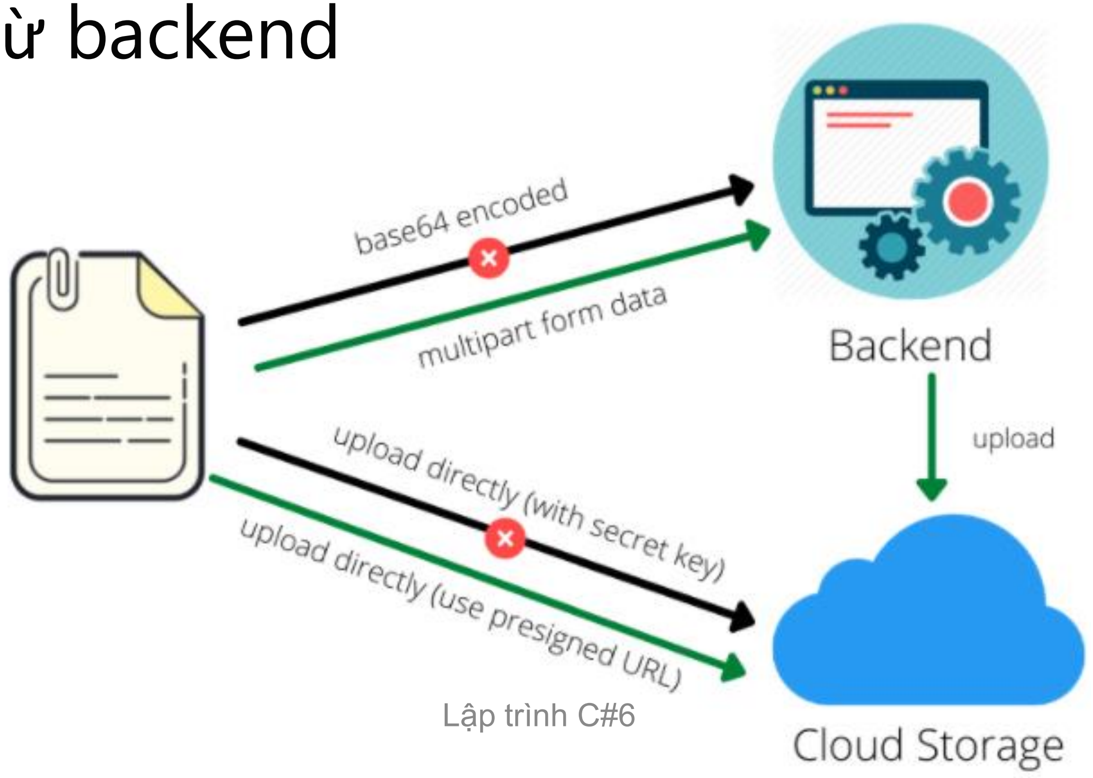
Attachment



	Name	Type	Nullable
▶	ID	NUMBER(19)	☐
	FILE_NAME	VARCHAR2(50)	☐
	FILE_DATA	BLOB	☐
	DESCRIPTION	VARCHAR2(255)	☑
*			☑

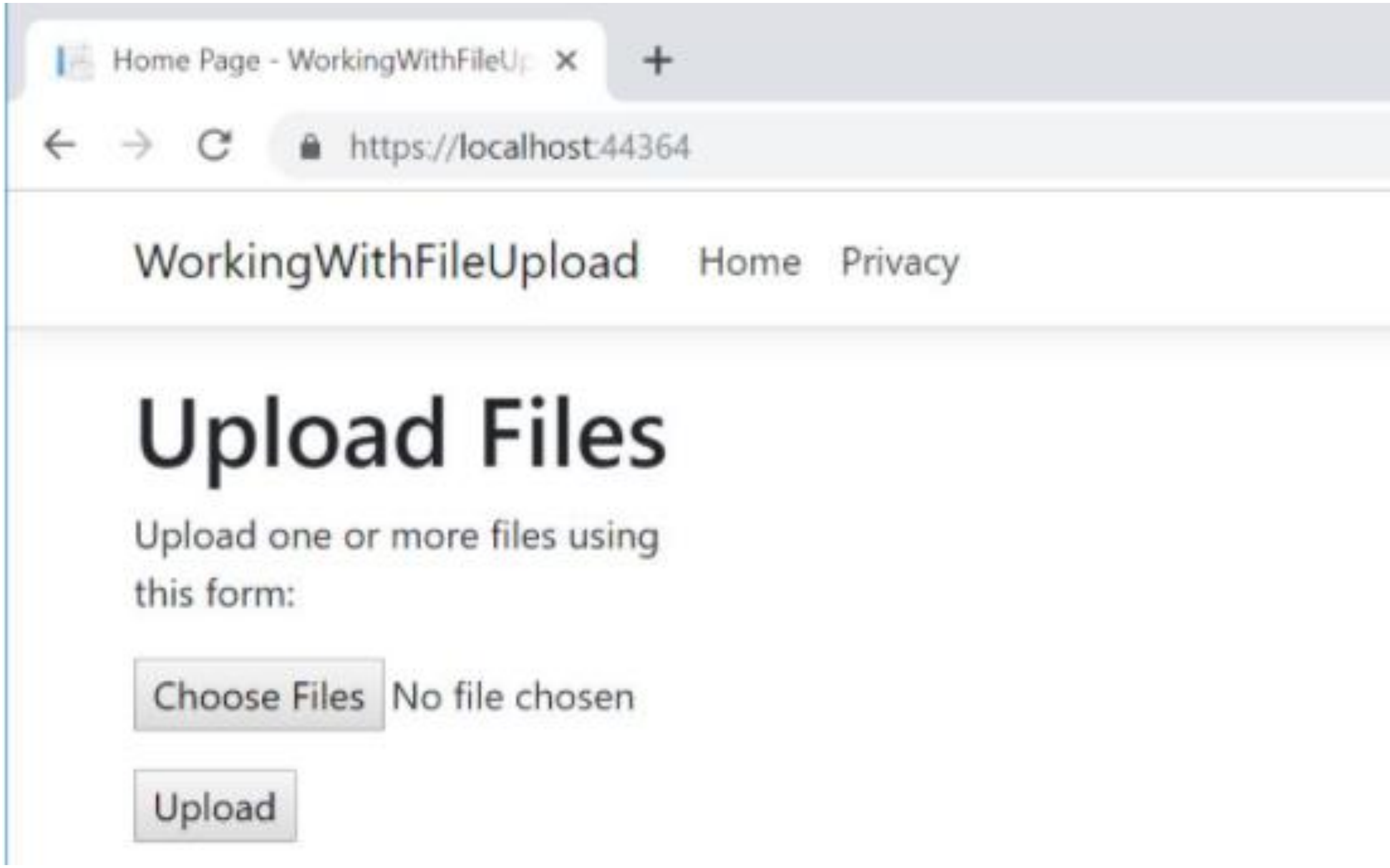
KỸ THUẬT UPLOAD FILE

- ❑ Client encode file (base64) rồi gởi về backend
- ❑ Client dùng API + Secret Key để thực hiện upload trực tiếp lên các Cloud Storage
- ❑ Sử dụng multipart form data
- ❑ Presigned URL từ backend


base64 encoded
multipart form data
Backend
upload
upload directly (with secret key)
upload directly (use presigned URL)
Cloud Storage

Upload file-IFormFile interface

- ❑ Creating a File Input Control
- ❑ The Role of Model Binding

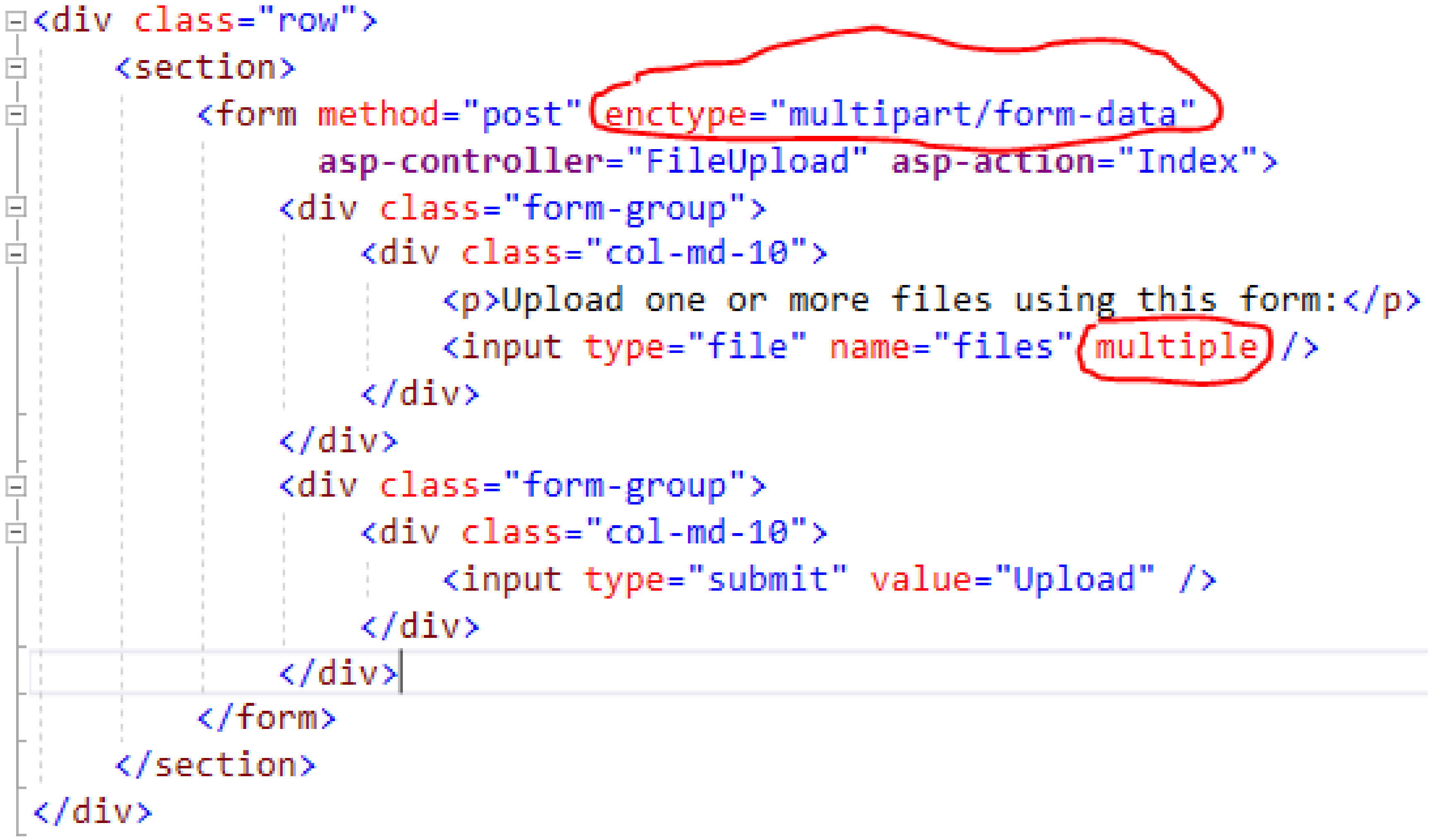


Upload Files

Upload one or more files using this form:

Upload file-IFormFile interface

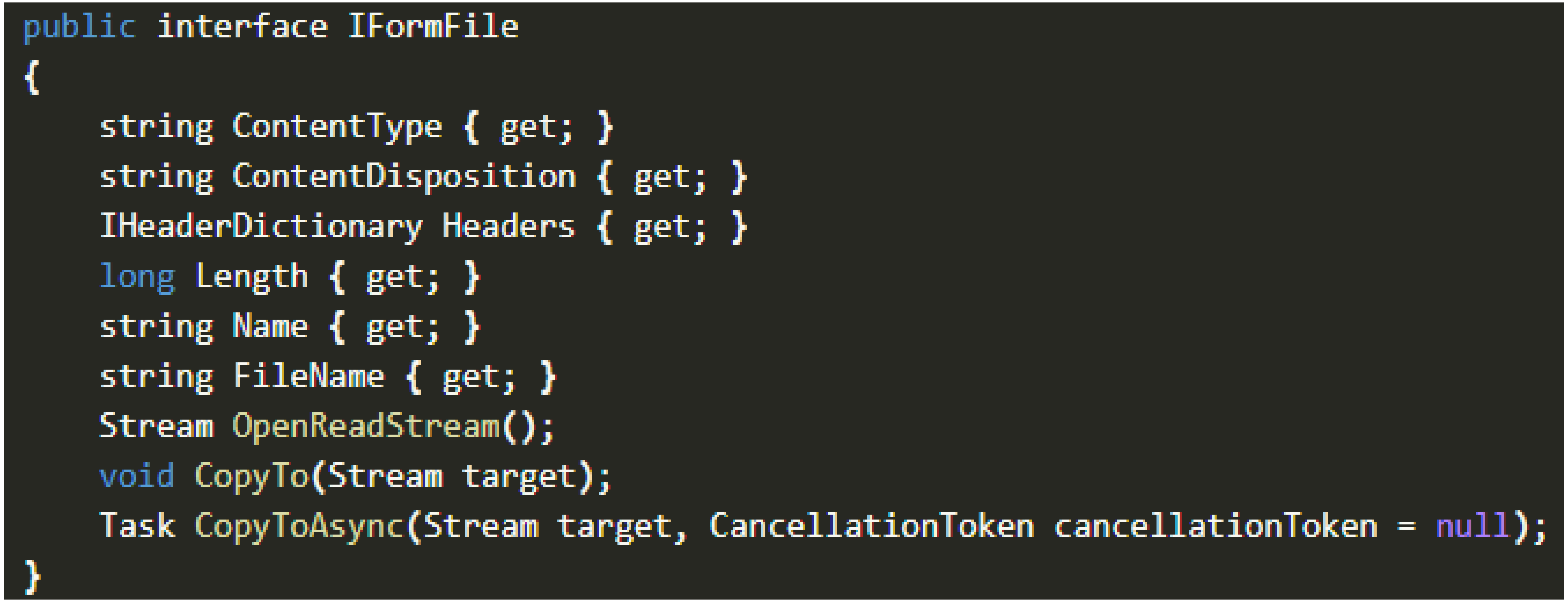
❑ Creating a File Input Control



```
<div class="row">
    <section>
        <form method="post" enctype="multipart/form-data"
              asp-controller="FileUpload" asp-action="Index">
            <div class="form-group">
                <div class="col-md-10">
                    <p>Upload one or more files using this form:</p>
                    <input type="file" name="files" multiple />
                </div>
            </div>
            <div class="form-group">
                <div class="col-md-10">
                    <input type="submit" value="Upload" />
                </div>
            </div>
        </form>
    </section>
</div>
```

UPLOAD FILE-IFORMFILE INTERFACE

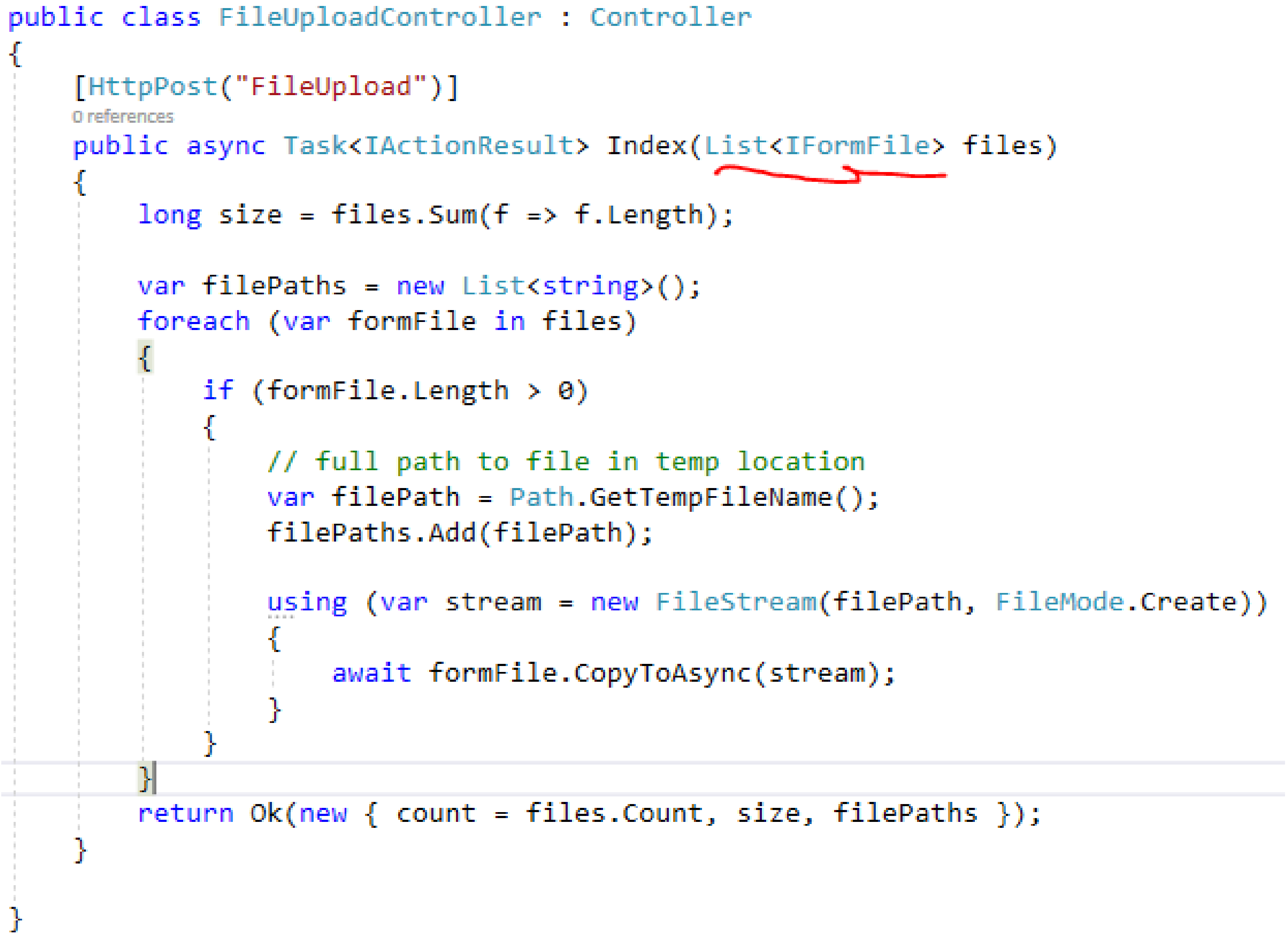
❑ The Role of Model Binding



```
public interface IFormFile
{
    string ContentType { get; }
    string ContentDisposition { get; }
    IHeaderDictionary Headers { get; }
    long Length { get; }
    string Name { get; }
    string FileName { get; }
    Stream OpenReadStream();
    void CopyTo(Stream target);
    Task CopyToAsync(Stream target, CancellationToken cancellationToken = null);
}
```

UPLOAD FILE-IFORMFILE INTERFACE

❑ The Role of Model Binding



```
public class FileUploadController : Controller
{
    [HttpPost("FileUpload")]
    0 references
    public async Task<IActionResult> Index(List<IFormFile> files)
    {
        long size = files.Sum(f => f.Length);

        var filePaths = new List<string>();
        foreach (var formFile in files)
        {
            if (formFile.Length > 0)
            {
                // full path to file in temp location
                var filePath = Path.GetTempFileName();
                filePaths.Add(filePath);

                using (var stream = new FileStream(filePath, FileMode.Create))
                {
                    await formFile.CopyToAsync(stream);
                }
            }
        }
        return Ok(new { count = files.Count, size, filePaths });
    }

}
```

DEMO 5.1

Upload file-IFormFile interface

FPT Education

FPT POLYTECHNIC

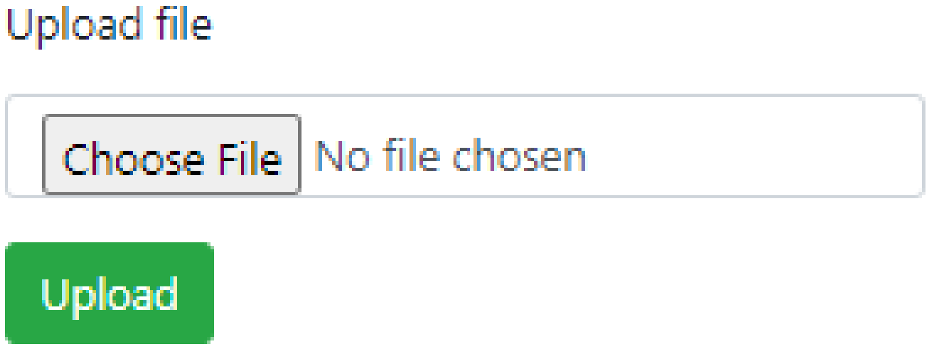
LẬP TRÌNH C# 6

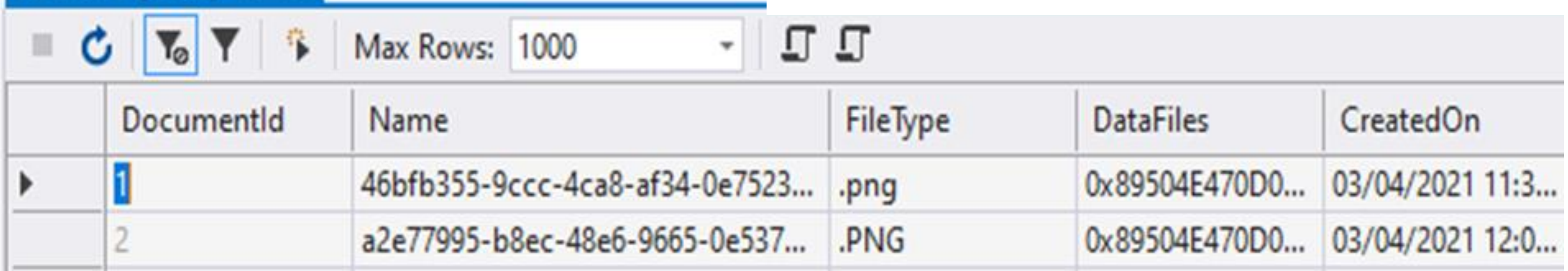
BÀI 5: WEB API UPLOAD/ DOWNLOAD FILES

P5.2

UPLOAD FILES - DATABASE

- ❑ Creating ASP.NET Core Application
- ❑ Database Design
- ❑ Adding Controller
- ❑ Adding View
- ❑ Adding Index Action Method to Handle POST Request
- ❑ Uploading Image
- ❑ Output


Upload file

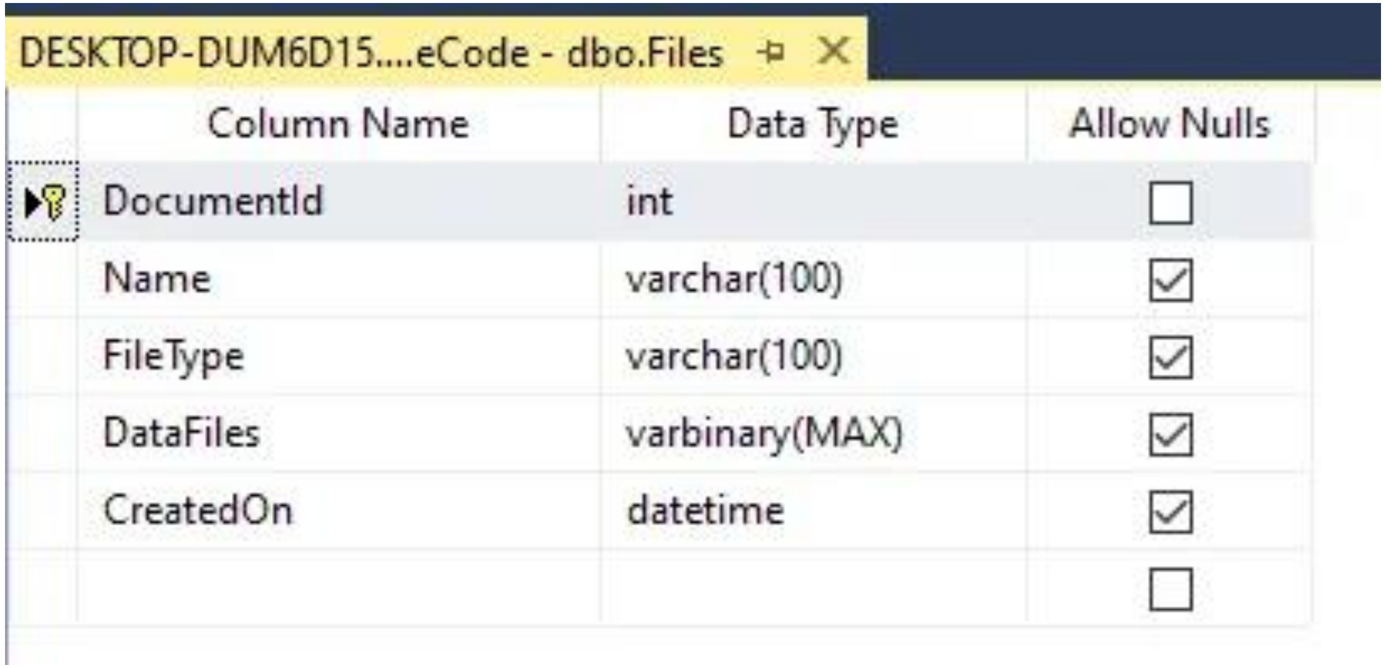

dbo.Files [Data]

Max Rows: 1000

	DocumentId	Name	FileType	DataFiles	CreatedOn
▶	1	46bfb355-9ccc-4ca8-af34-0e7523...	.png	0x89504E470D0...	03/04/2021 11:3...
	2	a2e77995-b8ec-48e6-9665-0e537...	.PNG	0x89504E470D0...	03/04/2021 12:0...

UPLOAD FILES - DATABASE

❑ Database Design

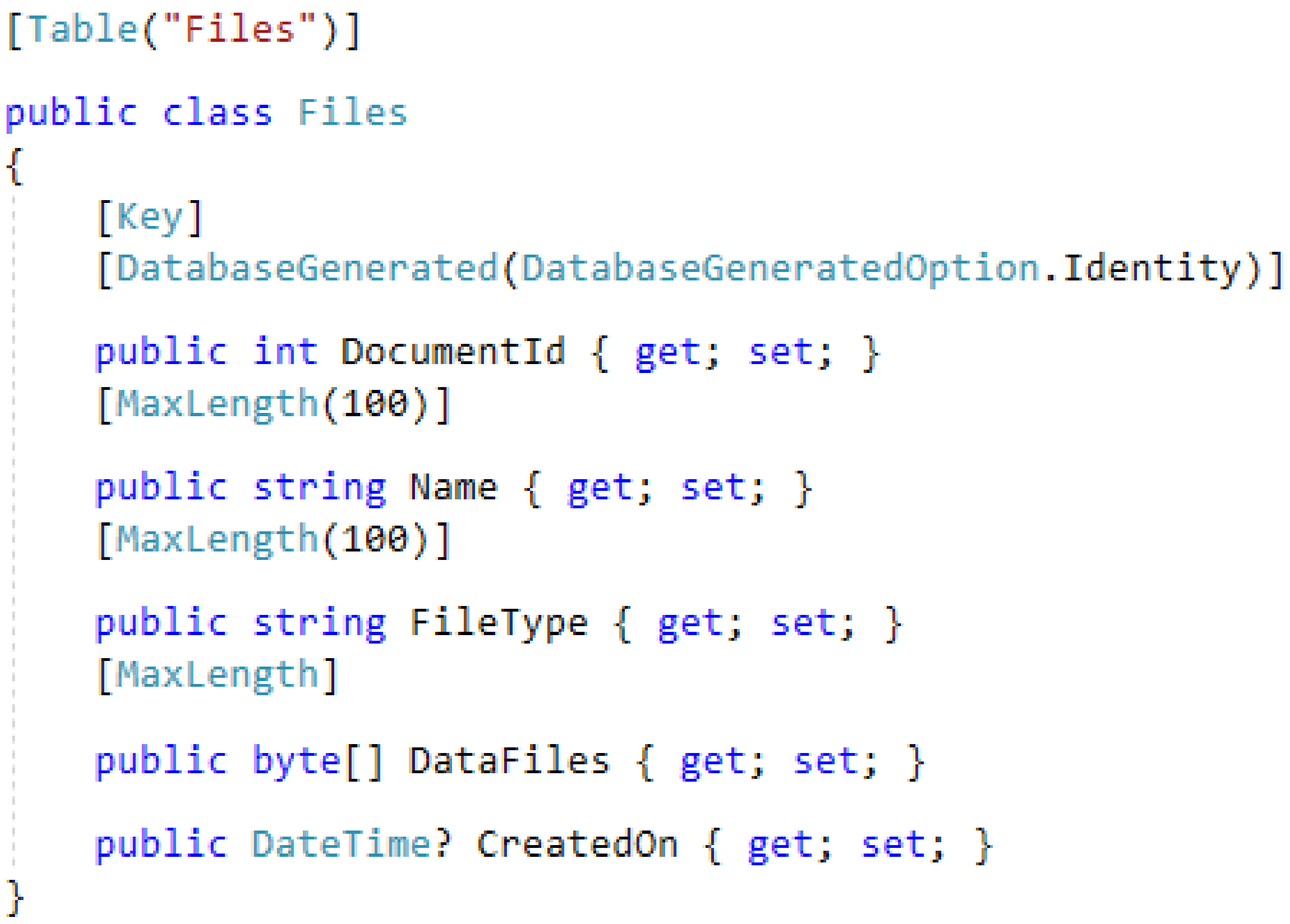


DESKTOP-DUM6D15....eCode - dbo.Files

	Column Name	Data Type	Allow Nulls
▶	DocumentId	int	□
	Name	varchar(100)	☑
	FileType	varchar(100)	☑
	DataFiles	varbinary(MAX)	☑
	CreatedOn	datetime	☑
			□

UPLOAD FILES - DATABASE

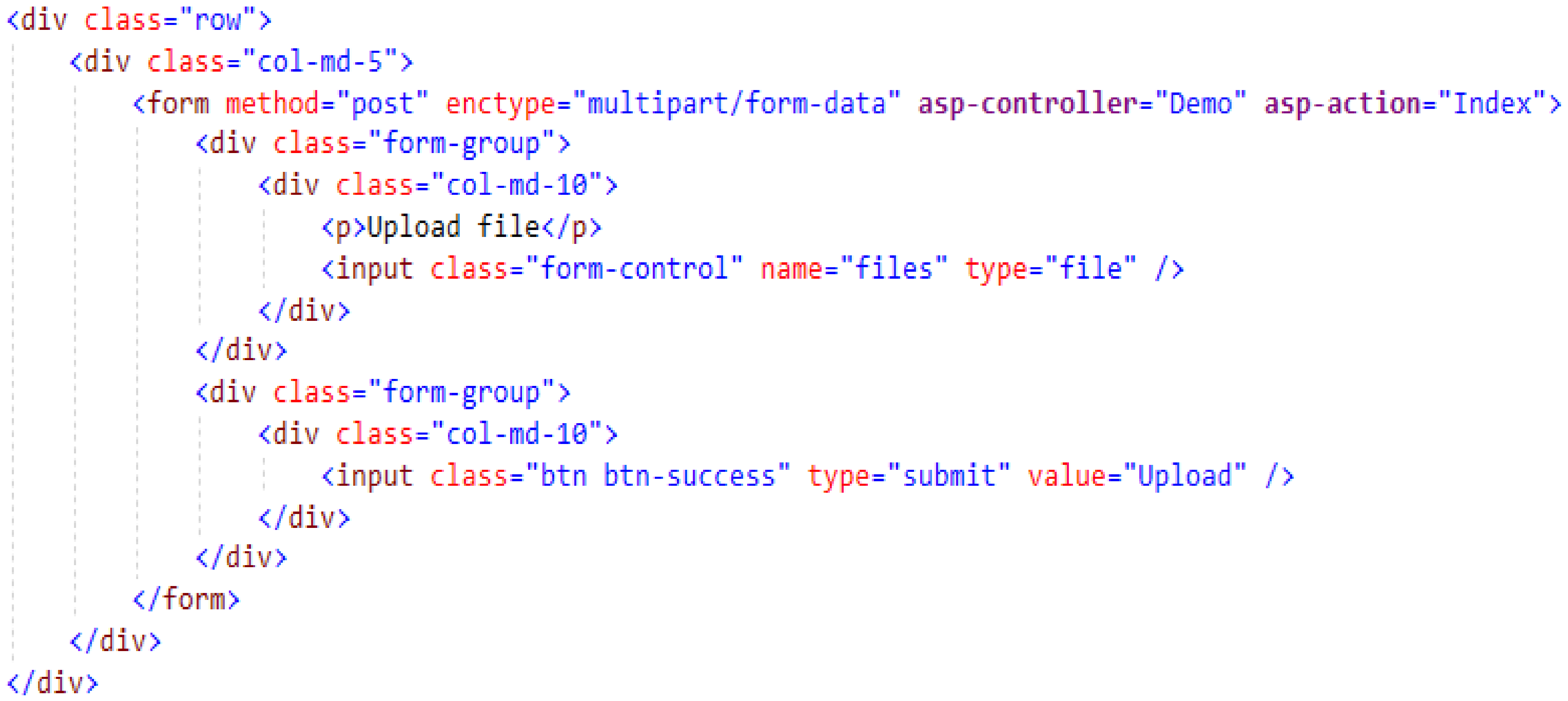
❑ Files Model



```
[Table("Files")]

public class Files
{
    [Key]
    [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

    public int DocumentId { get; set; }
    [MaxLength(100)]

    public string Name { get; set; }
    [MaxLength(100)]

    public string FileType { get; set; }
    [MaxLength]

    public byte[] DataFiles { get; set; }

    public DateTime? CreatedOn { get; set; }
}
```

UPLOAD FILES - DATABASE

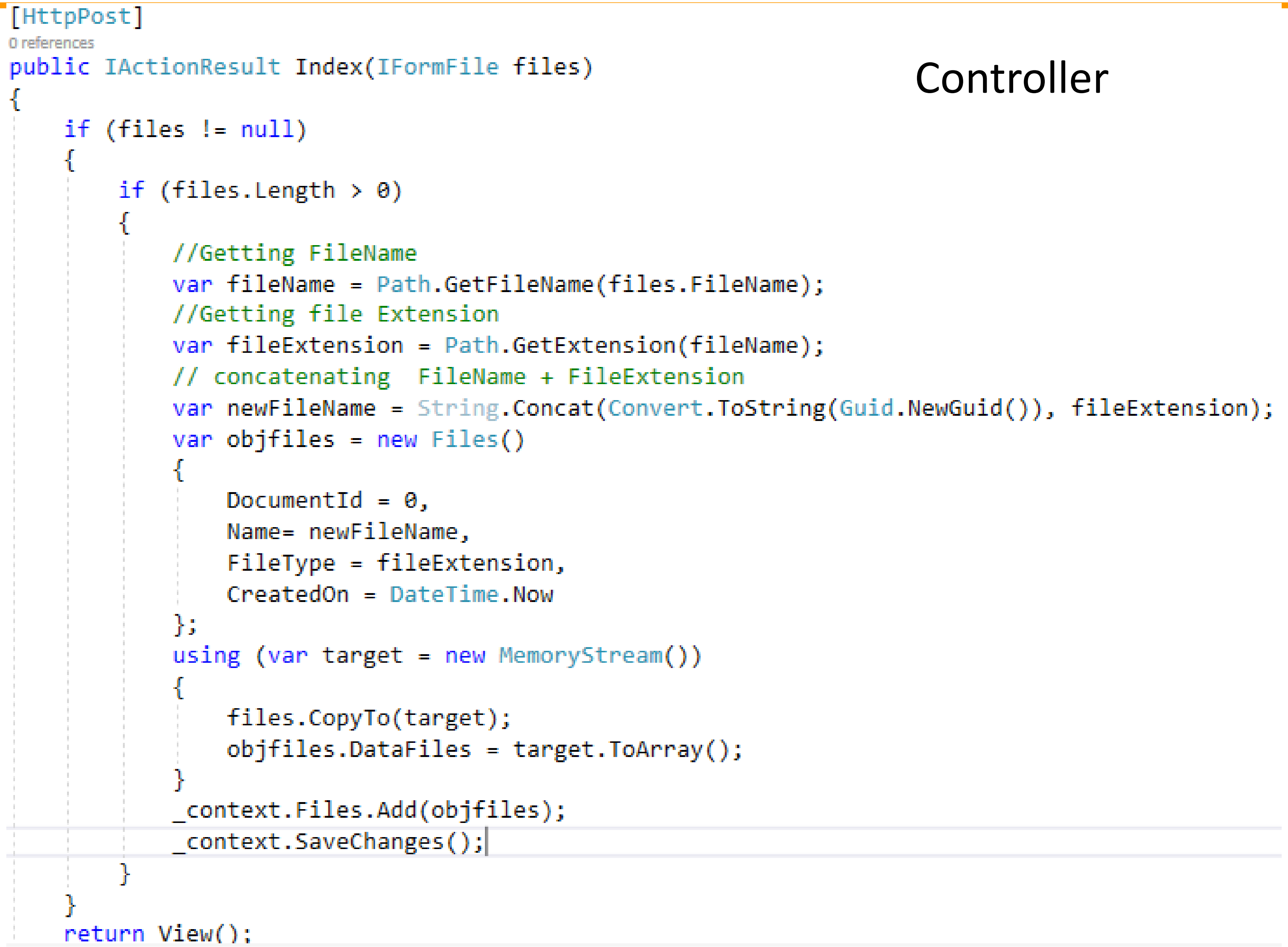
❑ View



```
<div class="row">
    <div class="col-md-5">
        <form method="post" enctype="multipart/form-data" asp-controller="Demo" asp-action="Index">
            <div class="form-group">
                <div class="col-md-10">
                    <p>Upload file</p>
                    <input class="form-control" name="files" type="file" />
                </div>
            </div>
            <div class="form-group">
                <div class="col-md-10">
                    <input class="btn btn-success" type="submit" value="Upload" />
                </div>
            </div>
        </form>
    </div>
</div>
```

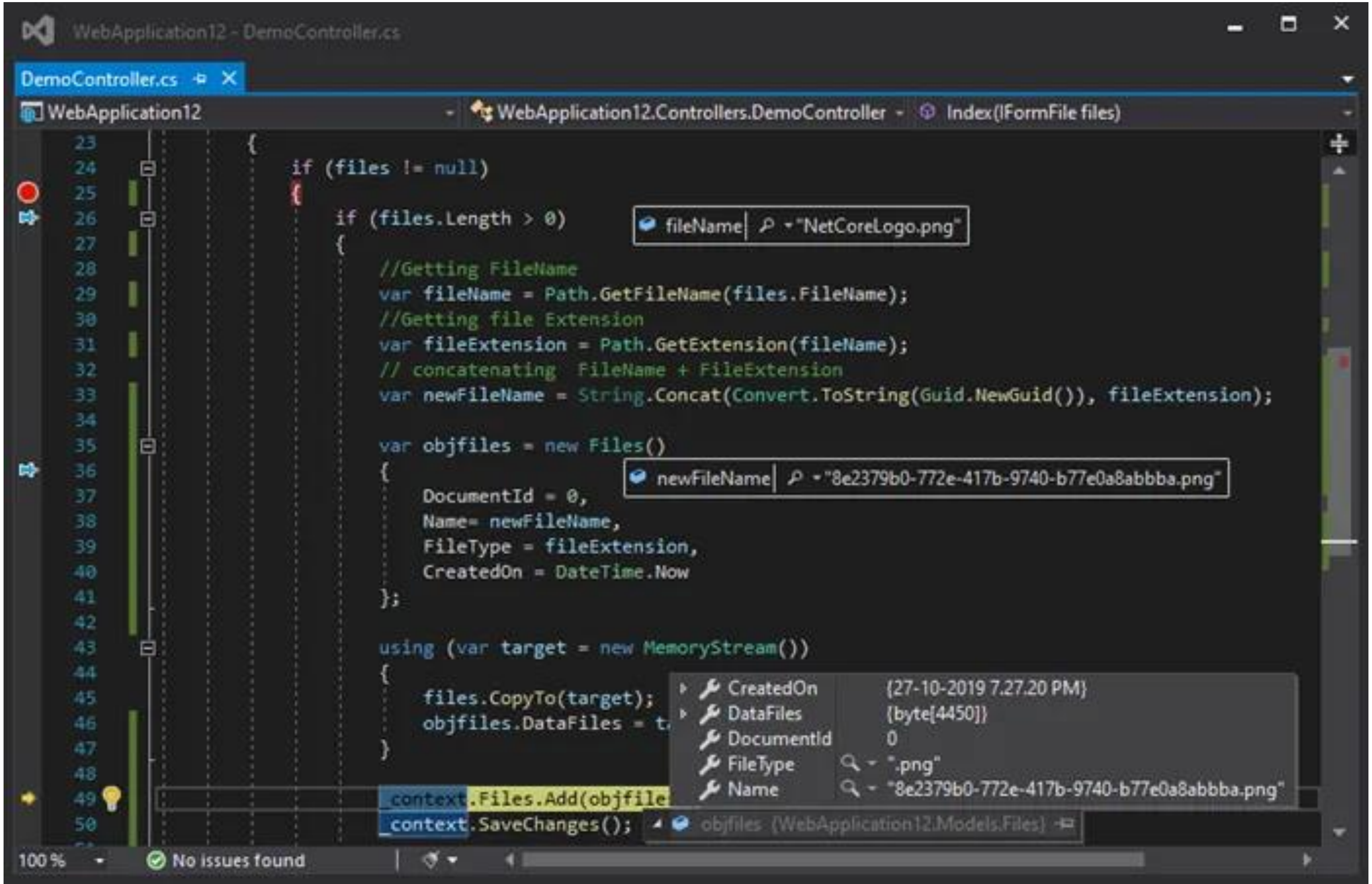
UPLOAD FILES - DATABASE

Controller



```
[HttpPost]
0 references
public IActionResult Index(IFormFile files)
{
    if (files != null)
    {
        if (files.Length > 0)
        {
            //Getting FileName
            var fileName = Path.GetFileName(files.FileName);
            //Getting file Extension
            var fileExtension = Path.GetExtension(fileName);
            // concatenating  FileName + FileExtension
            var newFileName = String.Concat(Convert.ToString(Guid.NewGuid()), fileExtension);
            var objfiles = new Files()
            {
                DocumentId = 0,
                Name= newFileName,
                FileType = fileExtension,
                CreatedOn = DateTime.Now
            };
            using (var target = new MemoryStream())
            {
                files.CopyTo(target);
                objfiles.DataFiles = target.ToArray();
            }
            _context.Files.Add(objfiles);
            _context.SaveChanges();
        }
    }
    return View();
```

UPLOAD FILES - DATABASE



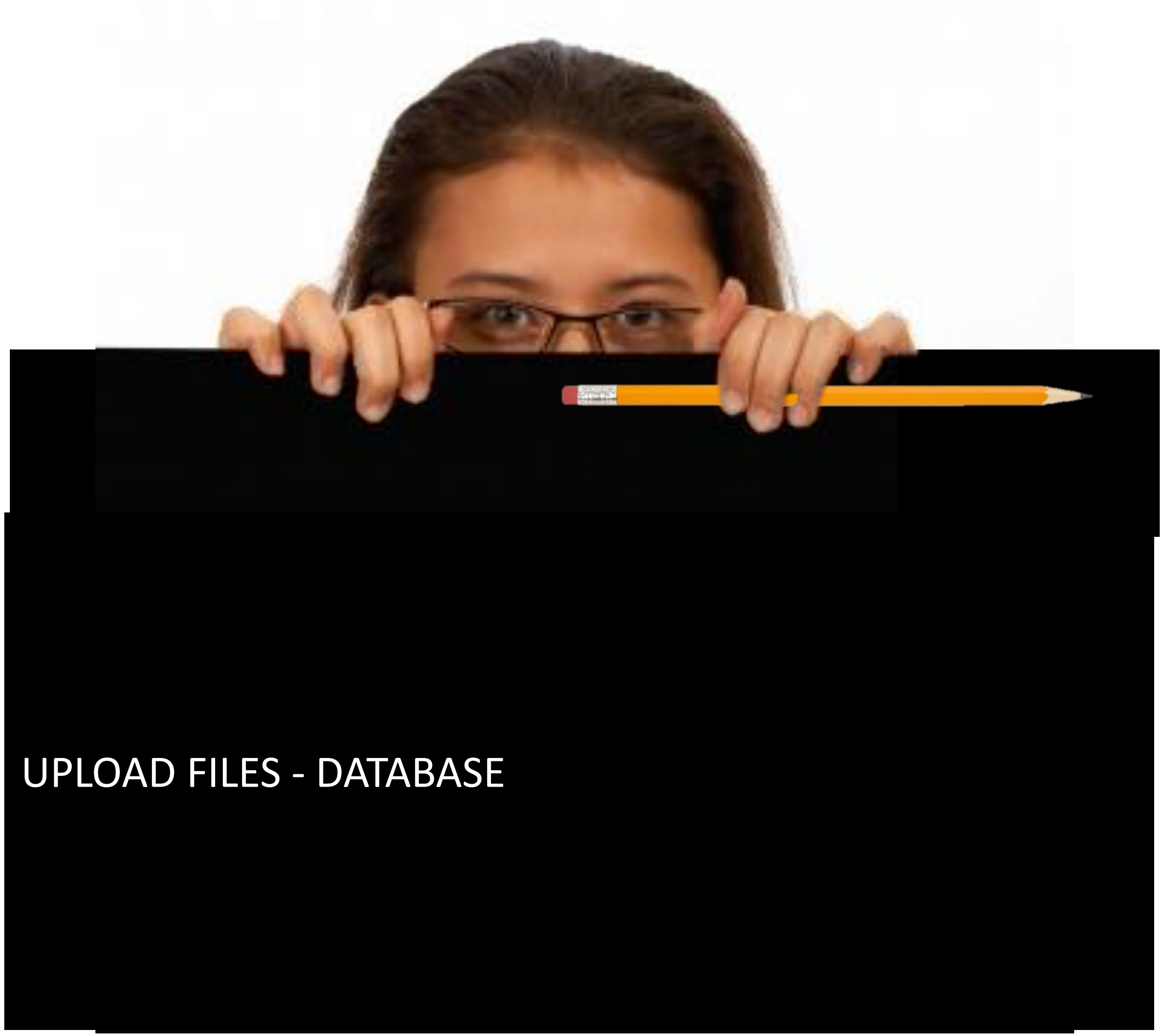
WebApplication12 - DemoController.cs
DemoController.cs
WebApplication12
WebApplication12.Controllers.DemoController
Index(IFormFile files)
100 %
No issues found

```
        {
            if (files != null)
            {
                if (files.Length > 0)
                {
                    //Getting FileName
                    var fileName = Path.GetFileName(files.FileName);
                    //Getting file Extension
                    var fileExtension = Path.GetExtension(fileName);
                    // concatenating  FileName + FileExtension
                    var newFileName = String.Concat(Convert.ToString(Guid.NewGuid()), fileExtension);

                    var objfiles = new Files()
                    {
                        DocumentId = 0,
                        Name= newFileName,
                        FileType = fileExtension,
                        CreatedOn = DateTime.Now
                    };

                    using (var target = new MemoryStream())
                    {
                        files.CopyTo(target);
                        objfiles.DataFiles = t
                    }

                    _context.Files.Add(objfile
                    _context.SaveChanges();
```

```
fileName | "NetCoreLogo.png"
newFileName | "8e2379b0-772e-417b-9740-b77e0a8abbba.png"
```

```
CreatedOn     {27-10-2019 7.27.20 PM}
DataFiles     {byte[4450]}
DocumentId    0
FileType      ".png"
Name          "8e2379b0-772e-417b-9740-b77e0a8abbba.png"
objfiles {WebApplication12.Models.Files}
```

DEMO 5.2



UPLOAD FILES - DATABASE

FPT Education

FPT POLYTECHNIC

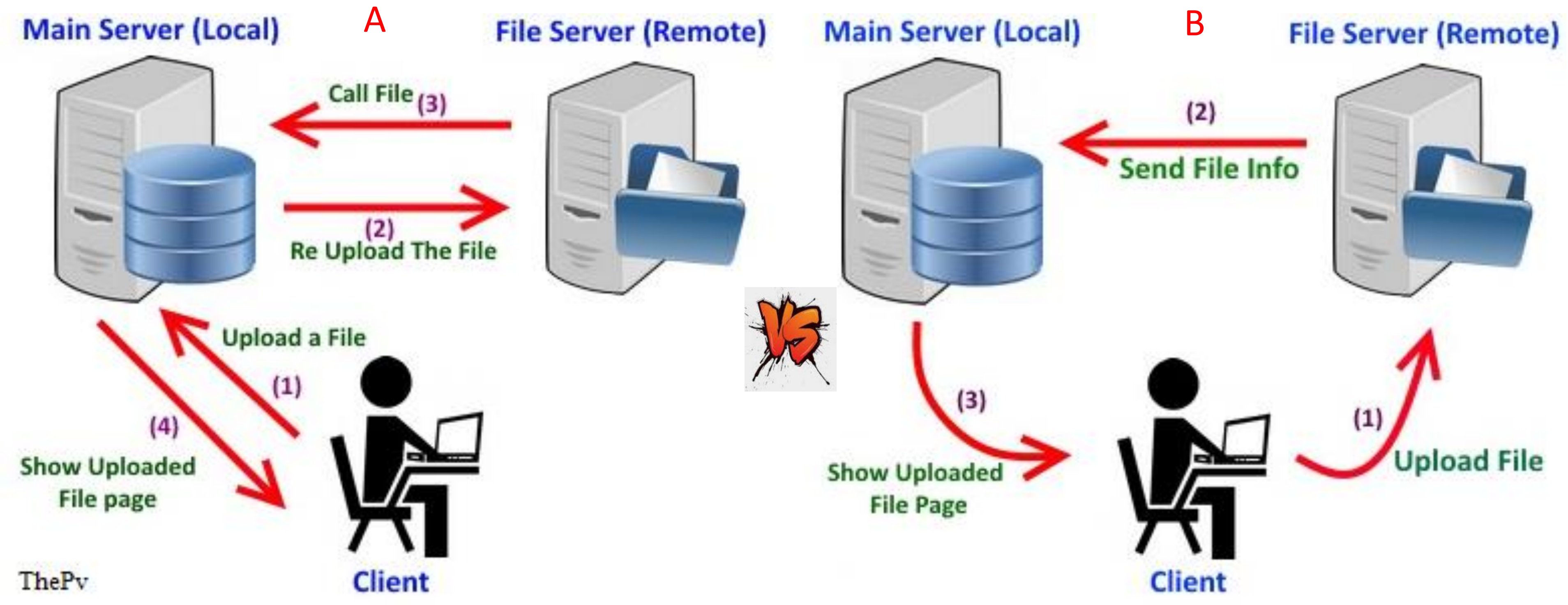
LẬP TRÌNH C# 6

BÀI 5: WEB API UPLOAD/ DOWNLOAD FILES

P5.3

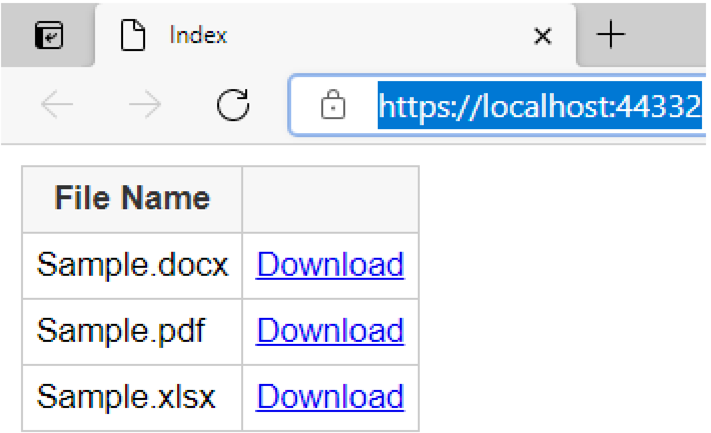
UPLOAD FILES

❑ Thảo luận chọn mô hình áp dụng

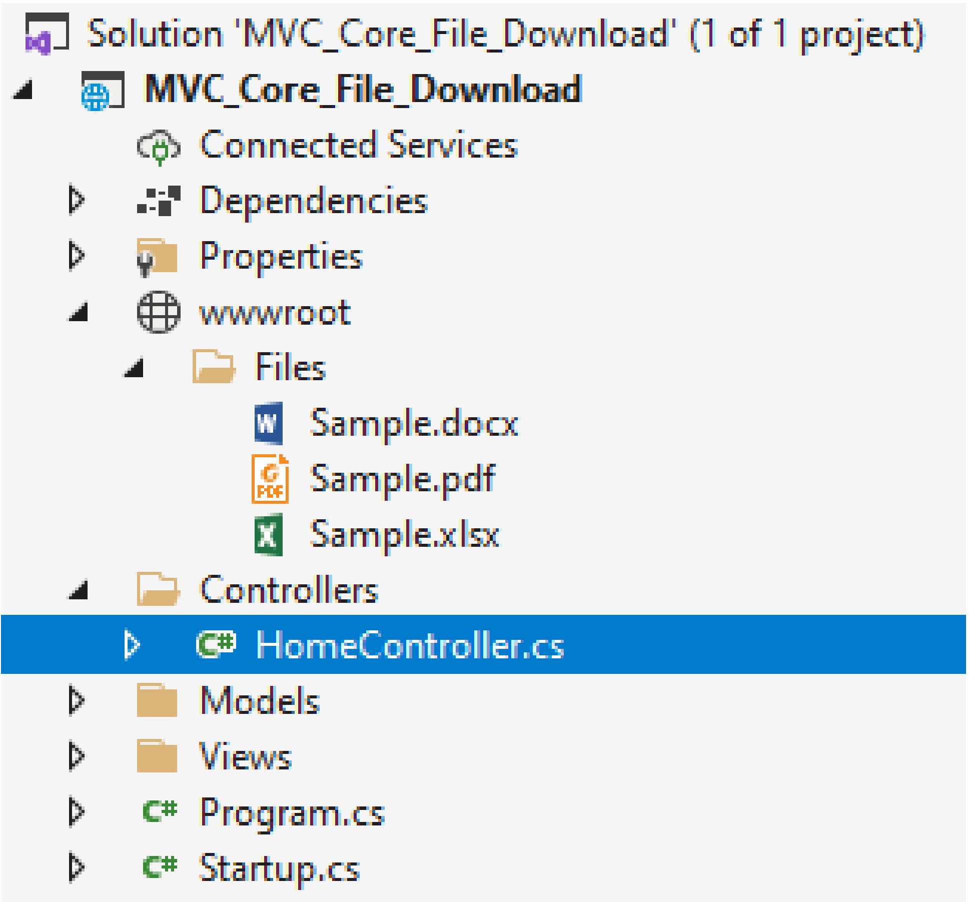


DOWNLOAD FILES

❑ FileResult-FileContentResult

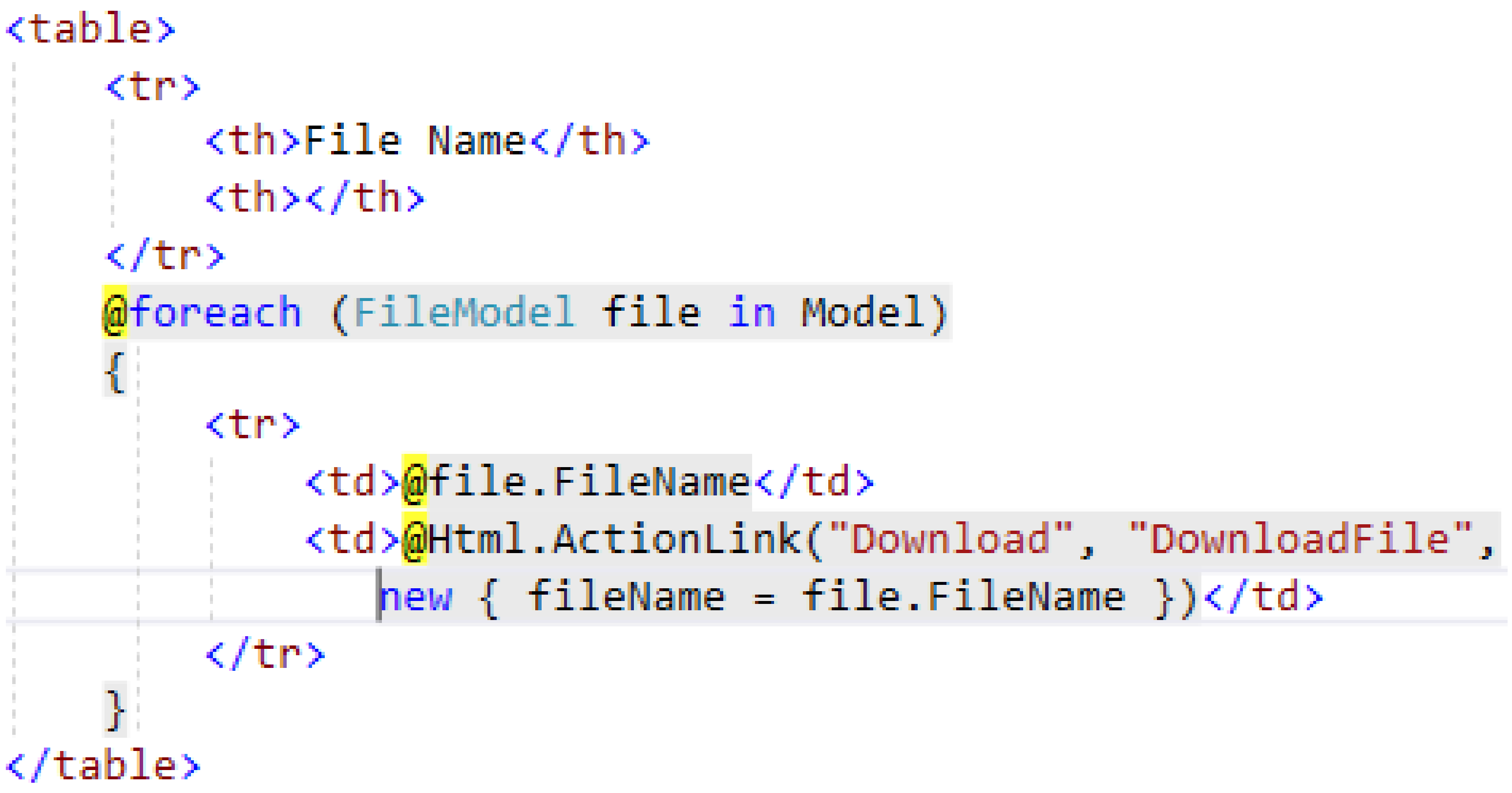


File Name	
Sample.docx	Download
Sample.pdf	Download
Sample.xlsx	Download

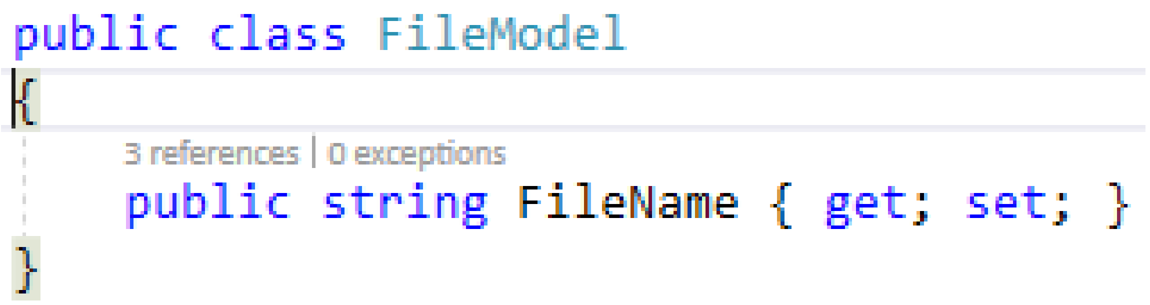


```
Solution 'MVC_Core_File_Download' (1 of 1 project)
◢ MVC_Core_File_Download
      Connected Services
   ▷  Dependencies
   ▷  Properties
   ◢  wwwroot
      ◢  Files
            Sample.docx
            Sample.pdf
            Sample.xlsx
   ◢  Controllers
      ▷  HomeController.cs
   ▷  Models
   ▷  Views
   ▷  Program.cs
   ▷  Startup.cs
```

DOWNLOAD FILES



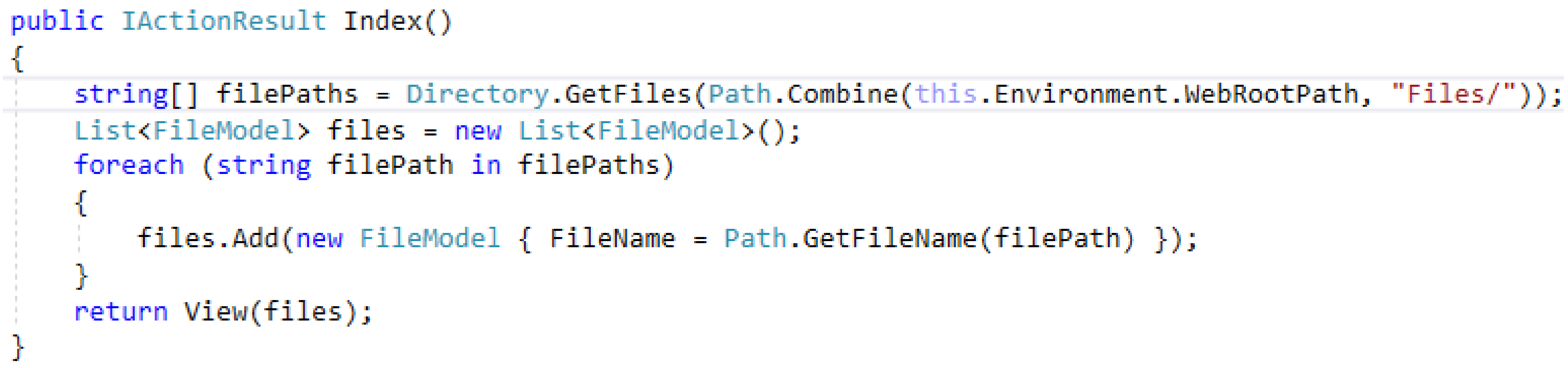
```
<table>
    <tr>
        <th>File Name</th>
        <th></th>
    </tr>
    @foreach (FileModel file in Model)
    {
        <tr>
            <td>@file.FileName</td>
            <td>@Html.ActionLink("Download", "DownloadFile",
               new { fileName = file.FileName })</td>
        </tr>
    }
</table>
```



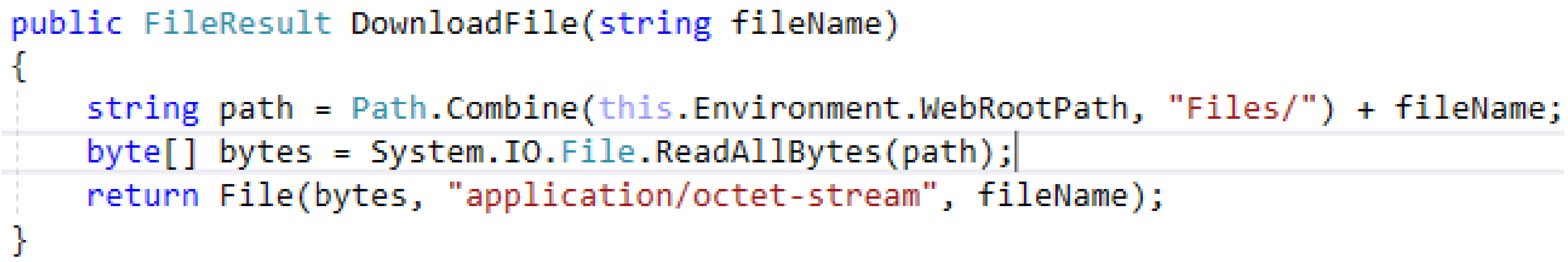
```
public class FileModel
{
    3 references | 0 exceptions
    public string FileName { get; set; }
}
```

DOWNLOAD FILES

❑ Controller

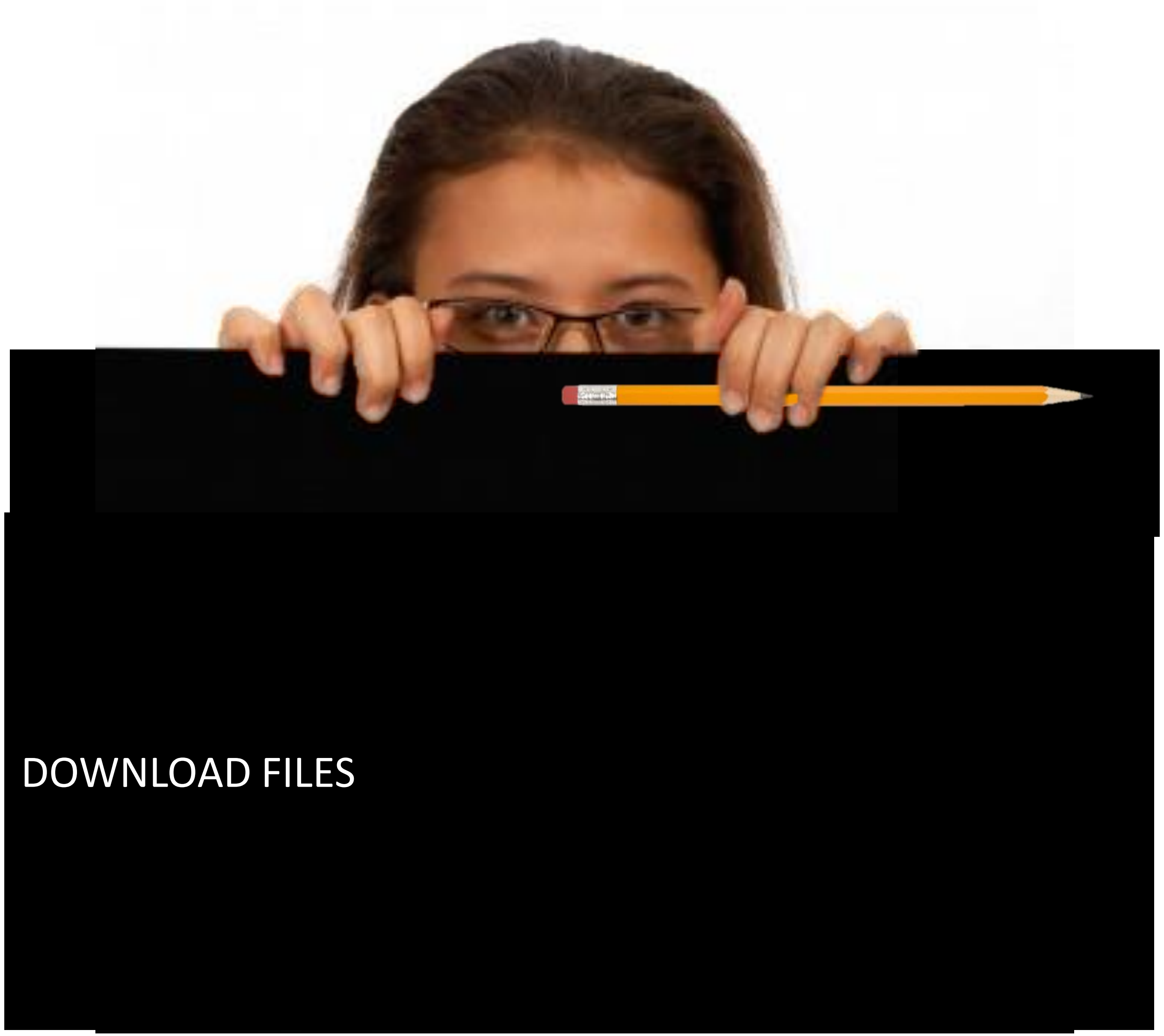


```
public IActionResult Index()
{
    string[] filePaths = Directory.GetFiles(Path.Combine(this.Environment.WebRootPath, "Files/"));
    List<FileModel> files = new List<FileModel>();
    foreach (string filePath in filePaths)
    {
        files.Add(new FileModel { FileName = Path.GetFileName(filePath) });
    }
    return View(files);
}
```



```
public FileResult DownloadFile(string fileName)
{
    string path = Path.Combine(this.Environment.WebRootPath, "Files/") + fileName;
    byte[] bytes = System.IO.File.ReadAllBytes(path);
    return File(bytes, "application/octet-stream", fileName);
}
```

DEMO 5.3



DOWNLOAD FILES

Tổng kết bài học

- Upload file
- Database và File System
- Upload File System-Database
- Download file

KẾT THÚC