

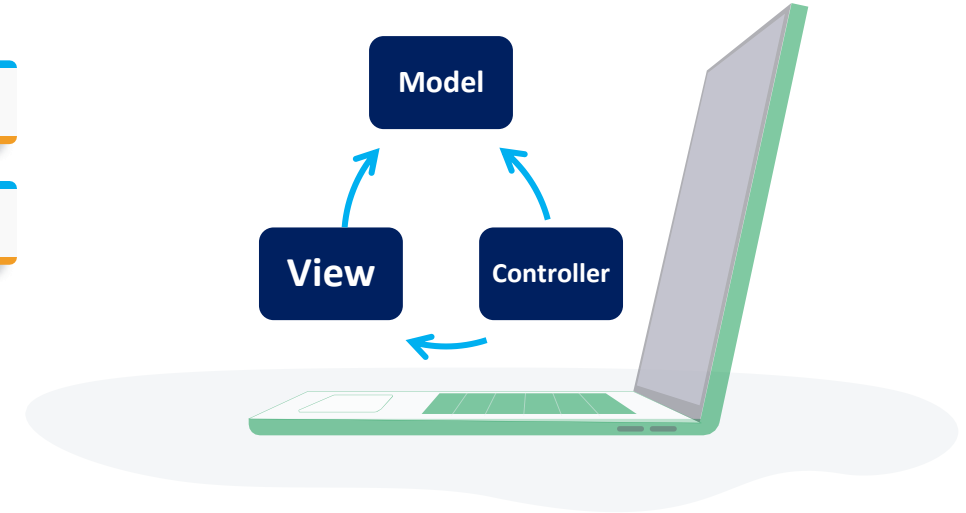
# ASP.NET Core MVC 6.0

Education and Training Solutions 2022



1 Display Images

2 Upload Images

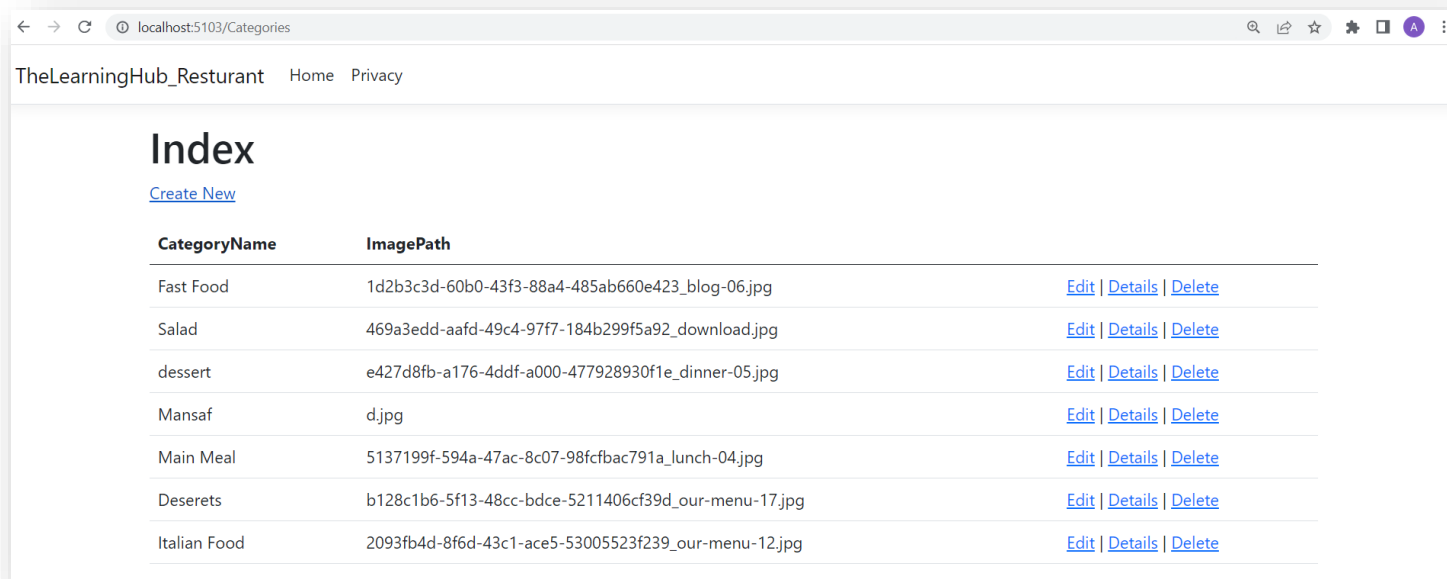


# Display Images

➤ Go To View => Index => Open with browser

The table content Category name and Image path.

**Note:** The image path **only** is saved in databases.



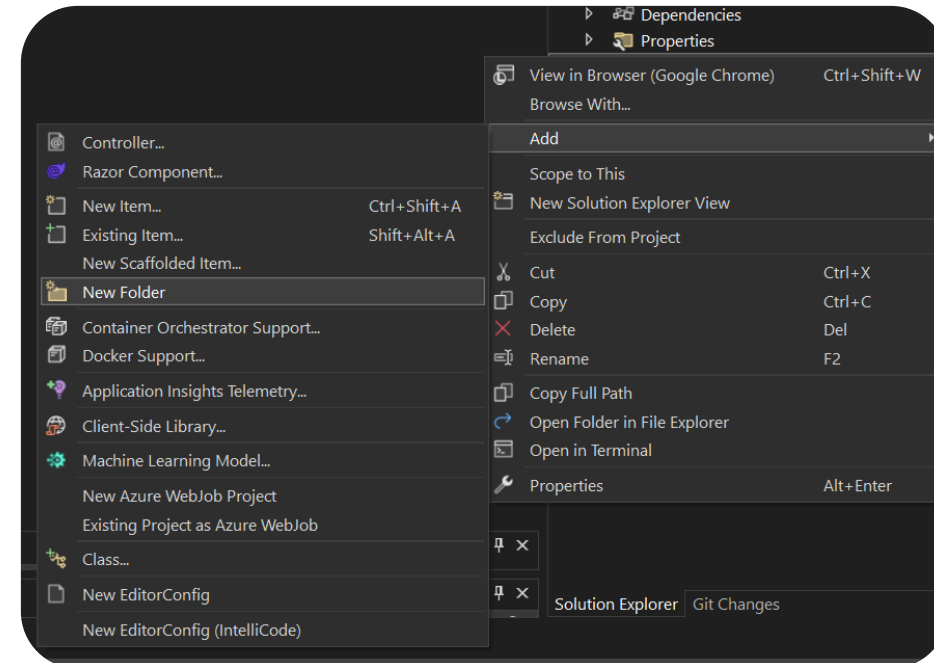
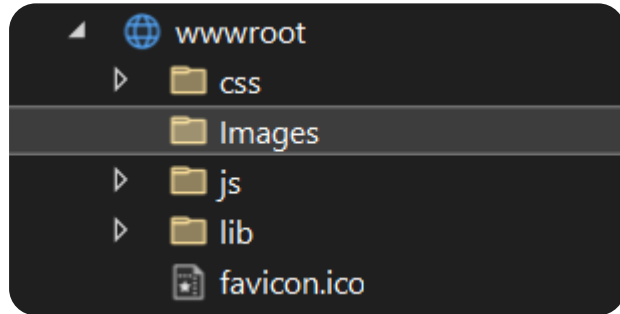
The screenshot shows a web browser window with the address bar displaying 'localhost:5103/Categories'. The page title is 'TheLearningHub\_Resturant' with navigation links for 'Home' and 'Privacy'. The main content area is titled 'Index' and includes a link for 'Create New'. Below this is a table with the following data:

CategoryName	ImagePath	
Fast Food	1d2b3c3d-60b0-43f3-88a4-485ab660e423_blog-06.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Salad	469a3edd-aafd-49c4-97f7-184b299f5a92_download.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
dessert	e427d8fb-a176-4ddf-a000-477928930f1e_dinner-05.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Mansaf	d.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Main Meal	5137199f-594a-47ac-8c07-98fcfbac791a_lunch-04.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Deserets	b128c1b6-5f13-48cc-bdce-5211406cf39d_our-menu-17.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Italian Food	2093fb4d-8f6d-43c1-ace5-53005523f239_our-menu-12.jpg	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>

## To display the image instead of the path of the image

1. We need to upload the image to the server.
  - Create an Image folder in the www root folder to save the uploaded image in it.

**Folder Name: Images**



2. Edit The Index.cshtml View Page under categories view folder.

Diagram illustrating the structure of the `<img>` tag in the Index.cshtml view page:

```
<img src=@Url.Content("~/Images/" + item.ImagePath)
asp-append-version="true" style="border-radius:20%;
width="100" height="100" />
```

Annotations:

- Image tag** points to the `<img>` tag.
- Folder name** points to the `Images` folder in the path.
- Image path** points to the `item.ImagePath` property.
- Style** points to the `style="border-radius:20%; width="100" height="100"` attributes.





```
@foreach (var item in Model) {  
    <tr>  
        <td>  
            @Html.DisplayFor(modelItem => item.CategoryName)  
        </td>  
        <td>  
            <img src=@Url.Content("~/Images/" + item.ImagePath)  
                asp-append-version="true" style="border-radius:20%;"  
                width="100px" height="100px" />  
        </td>  
        <td>  
            <a asp-action="Edit" asp-route-id="@item.Id">Edit</a> |  
            <a asp-action="Details" asp-route-id="@item.Id">Details</a> |  
            <a asp-action="Delete" asp-route-id="@item.Id">Delete</a>  
        </td>  
    </tr>  
}  
</tbody>  
</table>
```

```
</f9pfe>  
</fpoql>  
}  
</f>
```

Now we see the image icon instead of the image path.

## Index

[Create New](#)

CategoryName	ImagePath	
Fast Food		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Salad		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Main Meal		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Dessert		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>





# Upload Images

Display Category Create page.

# Create Category

---

CategoryName

ImagePath

Create

[Back to List](#)

## To Upload the image instead the path of the image

1. Edit Category Create View Page.

**Add** enctype="multipart/form-data" in the form tag

```
<form asp-action="Create" enctype="multipart/form-data">
  <div asp-validation-summary="ModelOnly" class="text-danger"></div>
  <div class="form-group">
    <label asp-for="CategoryName" class="control-label"></label>
    <input asp-for="CategoryName" class="form-control" />
    <span asp-validation-for="CategoryName" class="text-danger"></span>
  </div>
  <div class="form-group">
    <label asp-for="ImageFile" class="control-label"></label>
    <input asp-for="ImageFile" class="form-control" />
    <span asp-validation-for="ImageFile" class="text-danger"></span>
  </div>
  <div class="form-group">
    <input type="submit" value="Create" class="btn btn-primary" />
  </div>
</form>
```

2. Add an Image file Property in the category class to add a new category.

```
[NotMapped]  
public virtual IFormFile ImageFile { get; set; }
```

```
public partial class Category  
{  
    17 references  
    public decimal Id { get; set; }  
    16 references  
    public string? CategoryName { get; set; }  
    13 references  
    public string? ImagePath { get; set; }  
    [NotMapped]  
    6 references  
    public virtual IFormFile ImageFile { get; set; }  
    1 reference  
    public virtual ICollection<Product> Products { get; set; } = new List<Product>();  
}
```

### [NotMapped]

Use to specify Entity or property is not mapped to table in a database.

### IFormFile

An interface is used to upload the image.

2. Replace ImagePath with ImageFile Property that added to category class :

```
<form asp-action="Create" enctype="multipart/form-data">
  <div asp-validation-summary="ModelOnly" class="text-danger"></div>
  <div class="form-group">
    <label asp-for="CategoryName" class="control-label"></label>
    <input asp-for="CategoryName" class="form-control" />
    <span asp-validation-for="CategoryName" class="text-danger"></span>
  </div>
  <div class="form-group">
    <label asp-for="ImageFile" class="control-label"></label>
    <input asp-for="ImageFile" class="form-control" />
    <span asp-validation-for="ImageFile" class="text-danger"></span>
  </div>
  <div class="form-group">
    <input type="submit" value="Create" class="btn btn-primary" />
  </div>
</form>
```

The Category Create page will be like:

## Create Category

---

CategoryName

ImageFile

Choose File

No file chosen

Create

[Back to List](#)

### 3. Update Create a function (CategoriesController).

Replace imagepath with **ImageFile** Property

Add the  
code  
here.

```
[HttpPost]
[ValidateAntiForgeryToken]
0 references
public async Task<IActionResult> Create([Bind("Id,CategoryName,ImageFile")] Category cate
{
    if (ModelState.IsValid)
    {
        _context.Add(category);
        await _context.SaveChangesAsync();
        return RedirectToAction(nameof(Index));
    }
    return View(category);
}
```



## Code:

**Add this variable to the category controller**

```
private readonly IWebHostEnvironment _webHostEnviroment;
```

**Update the constructer**

```
public CategoriesController(ModelContext context , IWebHostEnvironment webHostEnviroment)
{
    _context = context;
    _webHostEnviroment = webHostEnviroment;
}
```

```
if (category.ImageFile != null)
{
    string wwwRootPath = _webHostEnvironment.WebRootPath;
    string fileName = Guid.NewGuid().ToString() + "_" +
        category.ImageFile.FileName;
    string path = Path.Combine(wwwRootPath + "/Images/",
        fileName);
    using (var fileStream = new FileStream(path,
        FileMode.Create))
    {
        await category.ImageFile.CopyToAsync(fileStream);
    }
    category.ImagePath = fileName;
}
```



#### 4. Create new category and Upload new Image:

## Create Category

CategoryName

ImageFile

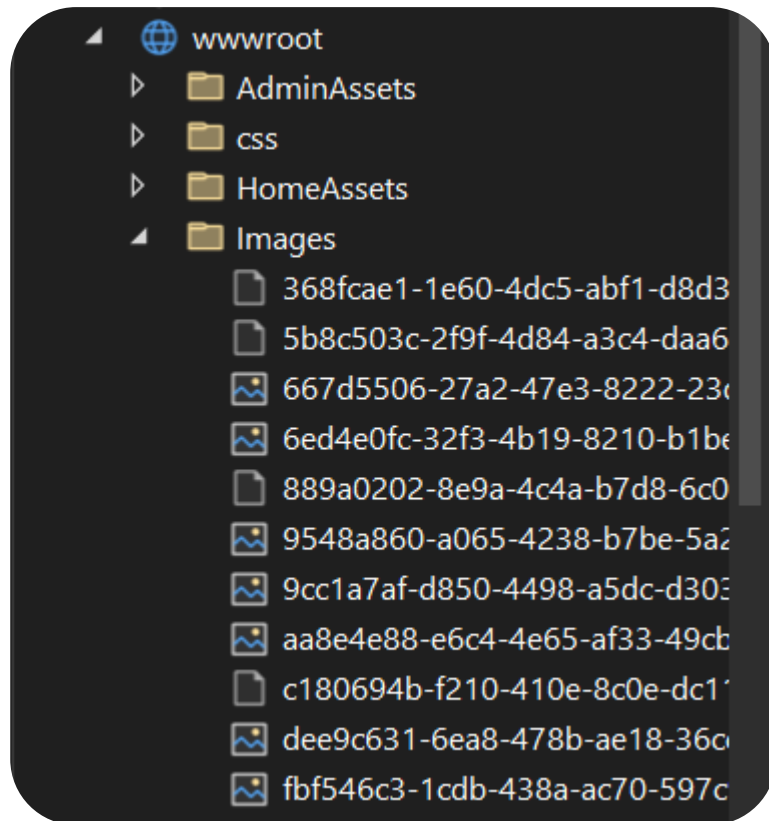
Choose File

blog-08.jpg

Create





[Back to List](#)

The Image is saved in the Images folder.



## Index

[Create New](#)

CategoryName	ImagePath	
Fast Food		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Salad		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Main Meal		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Dessert		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>

- Now build the code for the Edit function for uploading the image, not the Image path.
- For the Delete and Details view page make sure that the Image displayed not the Image Path.



Break





Thank You

