**Air University Multan Campus**

****

**Name Zeeshan Altaf**

**Roll no 213185**

**Class BSCS-V-B**

**Level Undergraduate**

**Course Title Visual Programming**

**Course Code CS-382**

**Course Instructor Mam Aatka Ali**

**Assignment #03-**

**Codes:**

**Initial code:**

using Microsoft.EntityFrameworkCore.Migrations;

#nullable disable

namespace \_213185.Migrations

{

public partial class initial : Migration

{

protected override void Up(MigrationBuilder migrationBuilder)

{

migrationBuilder.CreateTable(

name: "User",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Name = table.Column<string>(type: "nvarchar(max)", nullable: false),

Email = table.Column<string>(type: "nvarchar(max)", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_User", x => x.Id);

});

}

protected override void Down(MigrationBuilder migrationBuilder)

{

migrationBuilder.DropTable(

name: "User");

}

}

}

**Index.cshtml.cs:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

using Microsoft.EntityFrameworkCore;

using \_213185.Data;

using \_213185.Models;

namespace \_213185.Pages

{

public class IndexModel : PageModel

{

private readonly \_213185.Data.\_213185Context \_context;

public IndexModel(\_213185.Data.\_213185Context context)

{

\_context = context;

}

public IList<User> User { get;set; } = default!;

public async Task OnGetAsync()

{

if (\_context.User != null)

{

User = await \_context.User.ToListAsync();

}

}

}

}

**Index.cshtml:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

using Microsoft.EntityFrameworkCore;

using \_213185.Data;

using \_213185.Models;

namespace \_213185.Pages

{

public class IndexModel : PageModel

{

private readonly \_213185.Data.\_213185Context \_context;

public IndexModel(\_213185.Data.\_213185Context context)

{

\_context = context;

}

public IList<User> User { get;set; } = default!;

public async Task OnGetAsync()

{

if (\_context.User != null)

{

User = await \_context.User.ToListAsync();

}

}

}

}

**Detail.cshtml.cs:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

using Microsoft.EntityFrameworkCore;

using \_213185.Data;

using \_213185.Models;

namespace \_213185.Pages

{

public class DeleteModel : PageModel

{

private readonly \_213185.Data.\_213185Context \_context;

public DeleteModel(\_213185.Data.\_213185Context context)

{

\_context = context;

}

[BindProperty]

public User User { get; set; } = default!;

public async Task<IActionResult> OnGetAsync(int? id)

{

if (id == null || \_context.User == null)

{

return NotFound();

}

var user = await \_context.User.FirstOrDefaultAsync(m => m.Id == id);

if (user == null)

{

return NotFound();

}

else

{

User = user;

}

return Page();

}

public async Task<IActionResult> OnPostAsync(int? id)

{

if (id == null || \_context.User == null)

{

return NotFound();

}

var user = await \_context.User.FindAsync(id);

if (user != null)

{

User = user;

\_context.User.Remove(User);

await \_context.SaveChangesAsync();

}

return RedirectToPage("./Index");

}

}

}

**Detail.cshtml:**

@page

@model \_213185.Pages.DetailsModel

@{

ViewData["Title"] = "Details";

}

<h1>Details</h1>

<div>

<h4>User</h4>

<hr />

<dl class="row">

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.User.Name)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.User.Name)

</dd>

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.User.Email)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.User.Email)

</dd>

</dl>

</div>

<div>

<**a** **asp-page**="./Edit" **asp-route-id**="@Model.User?.Id">Edit</**a**> |

<**a** **asp-page**="./Index">Back to List</**a**>

</div>

**Delete.cshtml:**

@page

@model \_213185.Pages.DeleteModel

@{

ViewData["Title"] = "Delete";

}

<h1>Delete</h1>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>User</h4>

<hr />

<dl class="row">

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.User.Name)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.User.Name)

</dd>

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.User.Email)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.User.Email)

</dd>

</dl>

<**form** method="post">

<**input** **type**="hidden" **asp-for**="User.Id" />

<input type="submit" value="Delete" class="btn btn-danger" /> |

<**a** **asp-page**="./Index">Back to List</**a**>

</**form**>

</div>

**Delete.cshtml.cs:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.RazorPages;

using Microsoft.EntityFrameworkCore;

using \_213185.Data;

using \_213185.Models;

namespace \_213185.Pages

{

public class DetailsModel : PageModel

{

private readonly \_213185.Data.\_213185Context \_context;

public DetailsModel(\_213185.Data.\_213185Context context)

{

\_context = context;

}

public User User { get; set; } = default!;

public async Task<IActionResult> OnGetAsync(int? id)

{

if (id == null || \_context.User == null)

{

return NotFound();

}

var user = await \_context.User.FirstOrDefaultAsync(m => m.Id == id);

if (user == null)

{

return NotFound();

}

else

{

User = user;

}

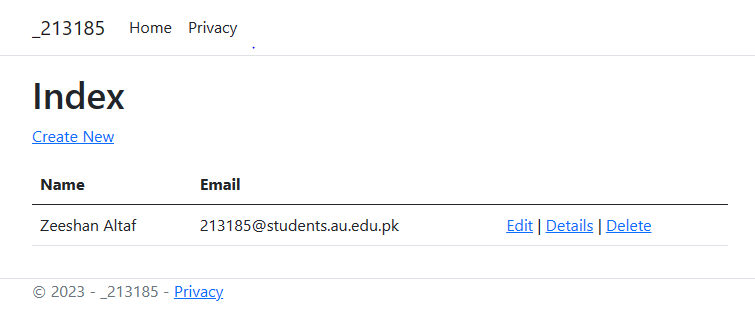
return Page();

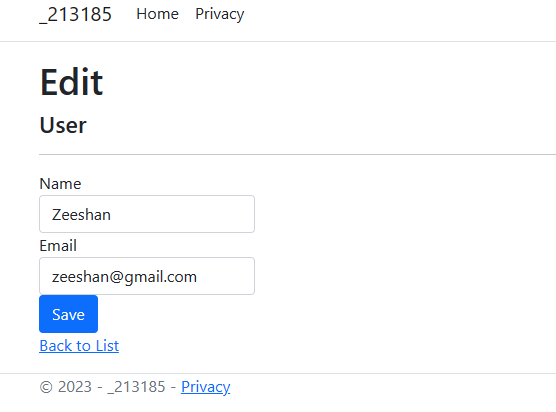
}

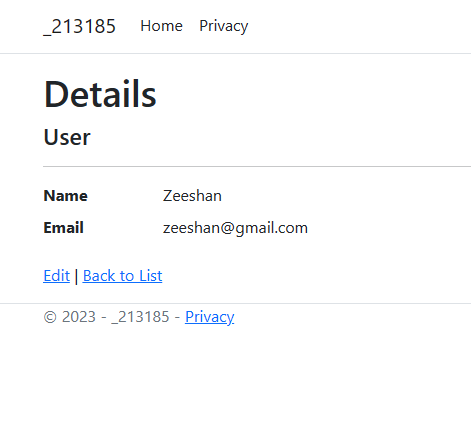
}

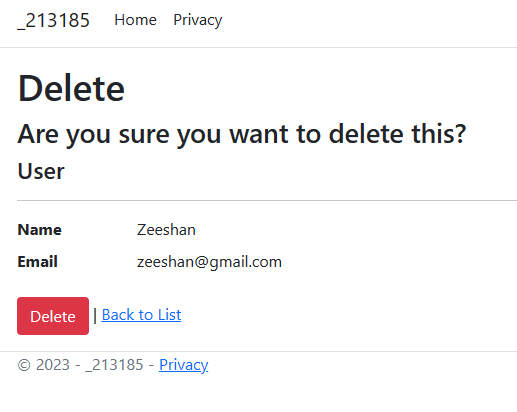
}

**Output:**

****

****

****

****