using BookServiceUsingRepo.Models;

using Microsoft.AspNetCore.Mvc;

using Newtonsoft.Json;

using System.Diagnostics;

using System.Net.Http.Headers;

using WebApplication1.Models;

namespace WebApplication1.Controllers

{

public class HomeController : Controller

{

private readonly ILogger<HomeController> \_logger;

public HomeController(ILogger<HomeController> logger)

{

\_logger = logger;

}

//static async Task GetStudentByID()

//{

// using (var client = new HttpClient())

// {

// //Send HTTP requests from here.

// client.BaseAddress = new Uri("https://localhost:44322/");

// client.DefaultRequestHeaders.Accept.Clear();

// client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

// //GET Method

// HttpResponseMessage response = await client.GetAsync("api/Students/7");

// if (response.IsSuccessStatusCode)

// {

// Student student = await response.Content.ReadAsAsync<Student>();

// Console.WriteLine("Id:{0}\tName:{1}", student.StudentId, student.Name);

// // Console.WriteLine("No of Employee in Department: {0}", department.Employees.Count);

// }

// else

// {

// Console.WriteLine(response.ReasonPhrase);

// Console.WriteLine("Internal server Error");

// }

// }

//}

//static async Task GetStudents()

//{

// using (var client = new HttpClient())

// {

// //Send HTTP requests from here.

// client.BaseAddress = new Uri("https://localhost:44322/");

// client.DefaultRequestHeaders.Accept.Clear();

// client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

// //GET Method

// HttpResponseMessage response = await client.GetAsync("api/Students");

// if (response.IsSuccessStatusCode)

// {

// var jsonString = response.Content.ReadAsStringAsync();

// jsonString.Wait();

// var student = JsonConvert.DeserializeObject<List<Student>>(jsonString.Result);

// foreach (var temp in student)

// {

// Console.WriteLine("Id:{0}\tName:{1}", temp.StudentId, temp.Name);

// // Console.WriteLine("No of Employee in Department: {0}", department.Employees.Count);

// }

// }

// else

// {

// Console.WriteLine(response.ReasonPhrase);

// Console.WriteLine("Internal server Error");

// }

// }

//}

public async Task<ActionResult> Index()

{

using (var client = new HttpClient())

{

//Send HTTP requests from here.

client.BaseAddress = new Uri("http://localhost:5041/");

client.DefaultRequestHeaders.Accept.Clear();

client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

//GET Method

List<Book> books = new List<Book>();

HttpResponseMessage response = await client.GetAsync("api/book");

if (response.IsSuccessStatusCode)

{

var jsonString = response.Content.ReadAsStringAsync();

jsonString.Wait();

var student = JsonConvert.DeserializeObject<List<Book>>(jsonString.Result);

books = student.ToList();

//foreach(Book book in student)

//{ Book book = new Book()

//{

//}

}

else

{

ViewBag.msg = "sss";

}

return View(books);

}

}

public async Task<ActionResult> Details(int id)

{

using (var client = new HttpClient())

{

//Send HTTP requests from here.

client.BaseAddress = new Uri("http://localhost:5041/");

client.DefaultRequestHeaders.Accept.Clear();

client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

//GET Method

Book book = new Book();

HttpResponseMessage response = await client.GetAsync("api/book/" + id);

if (response.IsSuccessStatusCode)

{

var jsonString = response.Content.ReadAsStringAsync();

jsonString.Wait();

book = JsonConvert.DeserializeObject<Book>(jsonString.Result);

}

else

{

ViewBag.msg = "sss";

}

return View(book);

}

}

public async Task<ActionResult> Create()

{

return View();

}

[HttpPost]

public async Task<ActionResult> Create(Book book)

{

using (var client = new HttpClient())

{

//Send HTTP requests from here.

client.BaseAddress = new Uri("http://localhost:5041/");

client.DefaultRequestHeaders.Accept.Clear();

client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

//GET Method

HttpResponseMessage response = await client.PostAsJsonAsync("api/book", book);

if (response.IsSuccessStatusCode)

{

var jsonString = response.Content.ReadAsStringAsync();

jsonString.Wait();

book = JsonConvert.DeserializeObject<Book>(jsonString.Result);

}

}

return RedirectToAction("Index");

}