Add a static **HttpClient** property to the Program class.

class Program

{

// New code:

static HttpClient client = new HttpClient();

}

using System;

using System.Net;

using System.Net.Http;

using System.Net.Http.Headers;

using System.Threading.Tasks;

namespace HttpClientSample

{

public class Product

{

public string Id { get; set; }

public string Name { get; set; }

public decimal Price { get; set; }

public string Category { get; set; }

}

class Program

{

static HttpClient client = new HttpClient();

static void ShowProduct(Product product)

{

Console.WriteLine($"Name: {product.Name}\tPrice: {product.Price}\tCategory: {product.Category}");

}

static async Task<Uri> CreateProductAsync(Product product)

{

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

response.EnsureSuccessStatusCode();

// return URI of the created resource.

return response.Headers.Location;

}

static async Task<Product> GetProductAsync(string path)

{

Product product = null;

HttpResponseMessage response = await client.GetAsync(path);

if (response.IsSuccessStatusCode)

{

product = await response.Content.ReadAsAsync<Product>();

}

return product;

}

static async Task<Product> UpdateProductAsync(Product product)

{

HttpResponseMessage response = await client.PutAsJsonAsync($"api/products/{product.Id}", product);

response.EnsureSuccessStatusCode();

// Deserialize the updated product from the response body.

product = await response.Content.ReadAsAsync<Product>();

return product;

}

static async Task<HttpStatusCode> DeleteProductAsync(string id)

{

HttpResponseMessage response = await client.DeleteAsync($"api/products/{id}");

return response.StatusCode;

}

static void Main()

{

RunAsync().Wait();

}

static async Task RunAsync()

{

client.BaseAddress = new Uri("http://localhost:55268/");

client.DefaultRequestHeaders.Accept.Clear();

client.DefaultRequestHeaders.Accept.Add(new

MediaTypeWithQualityHeaderValue("application/json"));

try

{

// Create a new product

Product product = new Product { Name = "Gizmo", Price = 100, Category = "Widgets" };

var url = await CreateProductAsync(product);

Console.WriteLine($"Created at {url}");

// Get the product

product = await GetProductAsync(url.PathAndQuery);

ShowProduct(product);

// Update the product

Console.WriteLine("Updating price...");

product.Price = 80;

await UpdateProductAsync(product);

// Get the updated product

product = await GetProductAsync(url.PathAndQuery);

ShowProduct(product);

// Delete the product

var statusCode = await DeleteProductAsync(product.Id);

Console.WriteLine($"Deleted (HTTP Status = {(int)statusCode})");

}

catch (Exception e)

{

Console.WriteLine(e.Message);

}

Console.ReadLine();

}

}

}