Formatters Lab

Perform these labs on your own computer using Visual Studio 2022 to ensure you understand the lessons presented in the corresponding videos and lectures.

Lab 1: Formatting Attributes

You can choose properties to ignore when serializing data

Open the **Customer.cs** file and add the following using statement.

```
using System.Text.Json.Serialization;
```

Add the following attribute above the **rowguid** property.

```
[JsonIgnore]
public string? MiddleName{ get; set; }
```

Add the [JsonPropertyName] attribute to the following properties.

```
[JsonPropertyName("Company")]
public string? CompanyName { get; set; }
[JsonPropertyName("Email")]
public string? EmailAddress { get; set; }
```

Add the [JsonPropertyOrder] attribute to the **ModifiedDate** property

```
[JsonPropertyOrder(-1)]
public DateTime ModifiedDate { get; set; }
```

Try it Out

Run the application and click on the **GET /api/Customer** method.

Notice the **ModifiedDate** property is listed first.

Notice the renaming of the two properties.

Notice the **MiddleName** property is no longer there.

Notice the **camelCasing** of the property names.

REMOVE/COMMENT the attributes you added in this demo.

Lab 2: Make Property Names Upper-Case

Open the **Program.cs** file and add a new using statement.

```
using Microsoft.AspNetCore.Http.Json;
```

Just before the code where you call **builder.Host.UseSerilog(…)**, add the following code.

```
// Configure JSON Options
builder.Services.Configure<JsonOptions>(options =>
{
  // Set property names to PascalCase
  options.SerializerOptions.PropertyNamingPolicy = null;
});
```

Try it Out

Run the application and click on the **GET /api/Customer** button.

Notice the property names in the result set are now in **PascalCase**.

Lab 3: Ignore ReadOnly Properties

Open the **Customer.cs** file and add a new property named ***IsActive.***

```
public bool IsActive { get; }
```

Try it Out

Run the application and see that the ***IsActive*** is added to the result set.

Set Other JSON Option

Open the **Program.cs** file and add another JSON option.

```
// Ignore "readonly" fields
options.SerializerOptions.IgnoreReadOnlyProperties =
true;
```

Try it Out

Run the application and see that ***IsActive*** no longer appears in the result.

Comment the line of code that nulls the PropertyNamingPolicy.

```
// Set property names to PascalCase
// options.SerializerOptions.PropertyNamingPolicy =
null;
```