.NET 9 App Dev Hands-On Lab

EF Lab 9 (Optional) – MXL Serialization Support

This lab involves updating the entities to support XML serialization. This is an optional lab unless you do the optional content negotiation lab in the RESTful services module. Before starting this lab, you must have completed EF Lab 6. This entire lab works on the `AutoLot.Models` project.

Copilot Agent Mode

This entire lab can be completed with Copilot Agent Mode.

```
Setup Prompt: Always use file scoped namespaces. Always combine attributes on a single line when
possible. The project does not use nullable reference types. There is a GlobalUsings.cs file that
includes common usings, don't include using statements in new files if they are already in the
globalusings.cs file. I prefer expression bodied members when possible. Single line if statements
should still use braces. Use ternary operators when appropriate. Use internal over private. All
classes and methods are public unless told otherwise.

Prompt: All work is to be done in the AutoLot.Models project.

Prompt:Add the following global usings to the GlobalUsings.cs file if they do not already exist:
global using System.Xml.Serialization;

Prompt:Add the Serializable attribute to the Car, CarDriver, Driver, make, and Radio classes.
For each navigation property in those classes, add the XmlIgnore attribute.
```

- After running the following prompts, verify the files were updated correctly, and you have completed this Lab.

Manual

Follow the steps below to complete this lab.

Part 1: Update the Global Usings

- Add the following line to the `GlobalUsings.cs` file:

```
global using System.Xml.Serialization;
```

Part 2: Update the Entities

The section updates the entities to support XML serialization. The classes must be marked as `[Serializable]`, and the navigation properties must be marked with `[XmlIgnore]`.

Step 1: Update the Car Entity

- Update the `Car.cs` class in the `Entities` folder by adding the following **bold** changes (extra lines omitted for brevity):

```
namespace AutoLot.Models.Entities;
[Serializable]
//omitted for brevity
public class Car : BaseEntity
{
  //omitted for brevity

  [ForeignKey(nameof(MakeId))]
  [InverseProperty(nameof(Make.Cars))]
  [XmlIgnore]
  public Make MakeNavigation { get; set; }
  [InverseProperty(nameof(Radio.CarNavigation))]
  [XmlIgnore]
  public Radio RadioNavigation { get; set; }
  [InverseProperty(nameof(Driver.Cars))]
  [XmlIgnore]
  public IEnumerable<Driver> Drivers { get; set; } = new List<Driver>();
  [InverseProperty(nameof(CarDriver.CarNavigation))]
  [XmlIgnore]
  public IEnumerable<CarDriver> CarDrivers { get; set; } = new List<CarDriver>();

  //omitted for brevity
}
```

Step 2: Update the CarDriver Entity

- Update the `CarDriver.cs` class in the `Entities` folder by adding the following **bold** changes (extra lines omitted for brevity):

```
namespace AutoLot.Models.Entities;
[Serializable]
//omitted for brevity
public class CarDriver : BaseEntity
{
  //omitted for brevity

  [ForeignKey(nameof(DriverId))]
  [XmlIgnore]
  public Driver DriverNavigation { get; set; }
  [ForeignKey(nameof(CarId))]
  [XmlIgnore]
  public Car CarNavigation { get; set; }

  //omitted for brevity
}
```

Step 3: Update the Driver Entity

- Update the `Driver.cs` class in the `Entities` folder by adding the following **bold** changes (extra lines omitted for brevity):

```
namespace AutoLot.Models.Entities;
[Serializable]
//omitted for brevity
public class Driver : BaseEntity
{
  //omitted for brevity
  [InverseProperty(nameof(Car.Drivers))]
  [XmlIgnore]
  public IEnumerable<Car> Cars { get; set; } = new List<Car>();
  [InverseProperty(nameof(CarDriver.DriverNavigation))]
  [XmlIgnore]
  public IEnumerable<CarDriver> CarDrivers { get; set; } = new List<CarDriver>();
}
```

Step 4: Update the Make Entity

- Update the `Make.cs` class in the `Entities` folder by adding the following **bold** changes (extra lines omitted for brevity):

```
namespace AutoLot.Models.Entities;
[Serializable]
//omitted for brevity
public class Make : BaseEntity
{
  //omitted for brevity
    [InverseProperty(nameof(Car.MakeNavigation))]
    [XmlIgnore]
    public IEnumerable<Car> Cars { get; set; } = new List<Car>();
}
```

Step 5: Update the Radio Entity

- Update the `Radio.cs` class in the `Entities` folder by adding the following **bold** changes (extra lines omitted for brevity):

```
namespace AutoLot.Models.Entities;
[Serializable]
//omitted for brevity
public class Radio : BaseEntity
{
  //omitted for brevity
    [ForeignKey(nameof(CarId))]
    [XmlIgnore]
    public Car CarNavigation { get; set; }
}
```

Summary

In this lab, you updated the entities to support XML serialization.