More Inheritance Concepts 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: Create IPerson Interface

Right mouse-click on the OOPLab project and add a new interface named **IPerson**.

```
namespace OOPLab;

public interface IPerson {
  public string FirstName { get; set; }
  public string LastName { get; set; }

  public abstract string FullName();
}
```

Open the **Person.cs** file and inherit from this interface

```
public class Person : IPerson {
```

Try it Out

Run the application and ensure the output is still the same as in the last lab.

```
2 - John Smith
Smith, Inc.
John.Smith@smithinc.com
$75,000.00
Smith, Inc.: Smith, John
Customer ID: 2 - Smith, Inc.: Smith, John

Seashell, Sally
```

Lab 2: Abstract Class

Open the **IPerson.cs** file and add two new abstract methods

```
public abstract string GetInfo();
public abstract string ContactInfo();
```

Open the **Person.cs** file and add the abstract keyword

```
public abstract class Person : IPerson {
```

Add a new method

```
public abstract string ContactInfo();
```

Add a new GetInfo() method and have it return the last name, first name and the age.

```
public virtual string GetInfo() {
  return $"{LastName}, {FirstName} Age = {Age}";
}
```

Modify Customer Class

Open the **Customer.cs** file and implement the ContactInfo() and GetInfo() methods

```
public override string ContactInfo() {
  return GetInfo();
}

public override string GetInfo() {
  return $"Customer ID: {CustomerId}: {base.GetInfo()} -
Email Address = {EmailAddress}";
}
```

Try it Out

Open the **Program.cs** file and **try** to instantiate an instance of the **Person** class

```
using OOPLab;

Person entity = new();

Console.WriteLine(entity.GetInfo());
```

You should get a compiler error

Modify the code in the **Program.cs** file to look like the following:

```
using OOPLab;

Customer entity = new() {
  CustomerId = 1,
  FirstName = "John",
  LastName = "Smith",
  Age = 50,
  EmailAddress = "John@smithco.com"
};

Console.WriteLine(entity.GetInfo());
```

Run the application and view the output that should look like the following:

```
Customer ID: 1: Smith, John Age = 50 - Email Address =
John@smithco.com
```