

Create a Superclass

A superclass is defined just like any other class:

```
class Vehicle
{
}
```

And a subclass inherits, or “extends”, a superclass using colon syntax (:):

```
class Sedan : Vehicle
{
}
```

A class can extend a superclass and implement an interface with the same syntax. Separate them with commas and make sure the superclass comes before any interfaces:

```
class Sedan : Vehicle, IAutomobile
{
}
```

The above code means that `Sedan` will inherit all the functionality of the `Vehicle` class, and it “promises” to implement all the functionality in the `IAutomobile` interface.

☒ Instructions

1.

In **Vehicle.cs**, build an empty `Vehicle` class.

Hint



Declare a class using the `class` keyword. For example:

```
class Faker
{
}
```

2.

In **Vehicle.cs**, define:

`string LicensePlate` property (getter only)

`double Speed` property (getter and private setter)

`int Wheels` property (getter only)

`void Honk()` method

`SpeedUp()` method

`SlowDown()` method

Hint



The format of an automatic property with `get` and `private set` is:

```
public bool IsFake
{ get; protected set; }
```

`SpeedUp()` and `SlowDown()` should add and remove 5 from `Speed`, respectively.

`Honk()` should write to the console.

3.

In **Sedan.cs**, use colon syntax to announce that `Sedan` inherits the `Vehicle` class.

Hint



Make sure that `Vehicle` is listed before `IAutomobile`. The inherited class comes before any interfaces.

4.

`Sedan` now inherits the members you defined in `Vehicle`. Remove them from **`Sedan.cs`**. You may still see errors and that's okay! We'll fix those in the next exercise.

Hint



The constructor should be the only member remaining in `Sedan`.