Introduction to Inheritance

Duplicated code leads to errors. Say you have two classes `Sedan` and `Truck`. They're different types, but they share a few properties and methods, like `SpeedUp()` and `SlowDown()`. If one of those members (say it's `SpeedUp()`) has to change, then we would have to change the code in every location where `SpeedUp()` is defined.

In this case it's two classes, but in other programs it may be many more! There are two reasons we don't want to make the same change on code across multiple files:

> It's a waste of time
>
> More importantly, it is a big risk for making mistakes

In this lesson you'll learn about a solution to this problem: *inheritance*. With inheritance, you can define one superclass that contains the shared members (like `SpeedUp()` and `SlowDown()`). All classes that need those members can *inherit* them from the superclass.

☑Instructions

Check out **Sedan.cs** and **Truck.cs**. What code is duplicated across these types?