# Subclasses and Polymorphism

Car and Motorcycle are subclasses of MotorVehicle. If you instantiate a Car or a Motorcycle with new, you can use that object anywhere you can use a MotorVehicle, because cars are motor vehicles. Similarly you can use a Motorcycle anywhere you can use a MotorVehicle. This use of a subclass object in place of a superclass object is the beginning of polymorphism. I'll say more about polymorphism later.

The converse is not true. Although all cars are motor vehicles, not all motor vehicles are cars. Some are motorcycles. Therefore if a method expects a Car object you shouldn't give it a MotorVehicle object instead.

Note that I said you shouldn't give a method that expects a Car a MotorVehicle. I didn't say you couldn't. Objects can be cast into their subclasses. This is useful when using data structures like Vectors that only handle generic objects. It's up to the programmer to keep track of what kind of object is stored in a Vector, and to use it accordingly.

The proper choice of classes and subclasses is a skill learned primarily through experience. There are often different ways to define classes.

toString() as example of polymorphism

[Previous](http://docs.google.com/14.html) | [Next](http://docs.google.com/16.html) | [Top](http://docs.google.com/index.html) | [Cafe au Lait](http://www.cafeaulait.org/)

Copyright 1997, 1998 Elliotte Rusty Harold

[elharo@metalab.unc.edu](mailto:elharo@metalab.unc.edu)

Last Modified October 20, 1998