abstraction

encapsulation

polymorphism

cohesion

coupling

identity

Polymorphism - is the ability to have a single interface that has multiple implementations. It allows you to have to call methods that can call either the superclass or subclass methods.

Good Way to use would be when you have a superclass of student that has subclasses of undergrad and graduate students. If the student class had a method enroll, it would be good to be able to call enroll whether or not they are a grad student or undergraduate, but if they enroll in a 4000/5000 class the grad student could automatically choose the 5000 level version, where as the undergrad would choose the 4000 level class.

Bad way to use would be in a way that creates high coupling. If you need to make a change in the super class that affects the way that the subclasses work that would be bad. A little more concretely would be if your were taking an int ID and then decided to change it to a string to get more possible ID’s. This will cause you to have to search through all the subclasses and make sure there are no bugs with the code.