OOP]

l5c

l6

l7a

l7b

Class & Instances

In Java, a class is a definition of objects of the same kind. In other words, a class is a blueprint, template, or prototype that defines and describes the static attributes and dynamic behaviors common to all objects of the same kind.

An instance is a realization of a particular item of a class. In other words, an instance is an instantiation of a class. All the instances of a class have similar properties, as described in the class definition. For example, you can define a class called "Student" and create three instances of the class "Student" for "Peter", "Paul" and "Pauline".

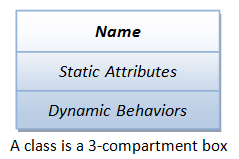
The term "object" usually refers to instance. But it is often used loosely, which may refer to a class or an instance.

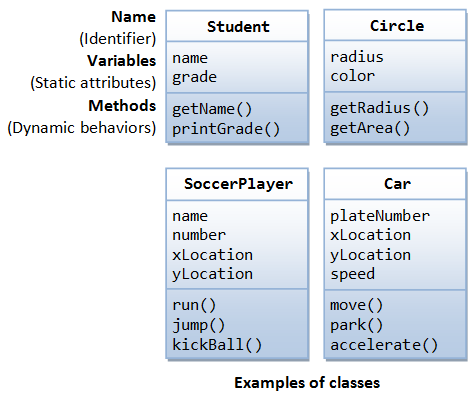
# A Class is a 3-Compartment Box encapsulating Data and Operations

A class can be visualized as a three-compartment box, as illustrated:

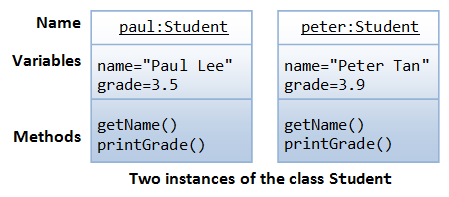
1. *Name* (or identity): identifies the class.
2. *Variables* (or attribute, state, field): contains the *static attributes* of the class.
3. *Methods* (or behaviors, function, operation): contains the *dynamic behaviors* of the class.

In other words, a class encapsulates the static attributes (data) and dynamic behaviors (operations that operate on the data) in a box.



The followings figure shows a few examples of classes:

The following figure shows two instances of the class Student, identified as "paul" and "peter".



**Unified Modeling Language (UML) Class and Instance Diagrams:** The above class diagrams are drawn according to the UML notations. A class is represented as a 3-compartment box, containing name, variables, and methods, respectively. Class name is shown in bold and centralized. An instance (object) is also represented as a 3-compartment box, with instance name shown asinstanceName:Classname and underlined.

##### Brief Summary

1. A class is a programmer-defined, abstract, self-contained, reusable software entity that mimics a real-world thing.
2. A class is a 3-compartment box containing the name, variables and the methods.
3. A class encapsulates the data structures (in variables) and algorithms (methods). The values of the variables constitute its state. The methods constitute its behaviors.
4. An instance is an instantiation (or realization) of a particular item of a class.