Intro to Classes and Objects

January 19, 2004

Outline

- Classes
- Objects
- Methods
- Instance Variables
- Creating an Object
- Instance Variables and Methods in Objects
- Example Code

Classes

- What is a Class?
  - A template (blueprint) of something you are trying to represent
  - A definition of a general object
  - Declare using the *class* keyword
  - Syntax:

    public class <class name> {

      [inst. vars. and methods]

    }
  - Ex: Account, Car, Student, Book

Objects

- What is an object?
  - A single copy (or instance) of a class
  - The class is the object's data type
  - Object is created using the class template
  - A class must be defined before an object (instance) of the class is created.
  - Ex: An object of the Account class may be Alice or Bob (which identify bank accounts owned by Alice or Bob)

Methods

- Methods: define the **actions or behaviors** of objects – implements behavior
- Verbs of an object
- Syntax:

  public <return type> <method name>([params]) {

  [method content]

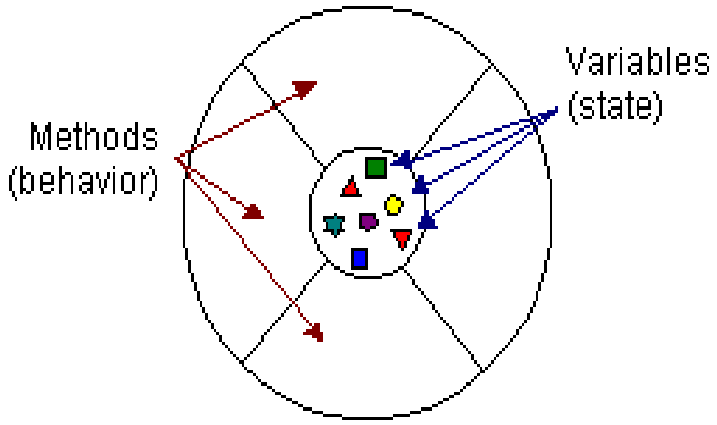
  }
- Use *void* return type for now
- Ex: An account will deposit, withdraw, and showBalance

Instance Variables

- Instance Variables: defines the data stored by a class - represents information that is common across objects, but may contain different values in each object
- Maintains state
- Nouns of an object
- Exist while an object exists
- Syntax:

  public <data type> <variable name> [= <initValue>];
- Ex: An account has a name and current balance

Graphical Model of a Class



From Learning the Java Language – OO Programming Concepts – address in References slide

Creating an Object

- Syntax:

  <object name> <variable name> = new <object name>();
- Ex: Account alice = new Account();

Instance Variables and Methods in Objects

- An object is an instance of a class
- Each object will have their own set of instance variables
- Each object of a class will use the same instance variable names, but the instance variables will have different values.
- Each object of a class will have the same methods, but the methods will act on the object it is called upon.

Example Code

- currentBalance – instance variable
- accountName – instance variable
- The main method creates two accounts. A change of instance variables in one account does not affect the instance variables in another account

Example Code (2)

- Running a method on one object does not affect other objects
- Each object has the same methods and data values available because they are all the same class
- Each can perform separate actions and store separate values because each is a separate object.

References

- Jason Schwarz's Lecture 3 slides: http://courses.ncsu.edu/csc116/
- Object-Oriented Articles: http://java.sun.com/docs/books/tutorial/java/concepts/