**Chapter 8 Terms**

Class - A software blueprint for implementing objects of a given type.

Object - Something designed and create by code. We use classes in programming to create the desired object.

State - Characteristics of an object; those things that define it.

Object reference (or instance) - A variable that represents an object. Example: Circle c = new Circle(); c is Circle instance

Data fields (or instance variables) - Current state of a given object is maintained by these. These are the variables that we declare inside our class blueprint.

Methods - The behaviors exhibited by the object and the operations that control the object.

Encapsulation - Combining an object's data and methods together into a single unit.

Information hiding - Restricting access to parts of a class through the use of the *private* keyword.

Constructor - Creates an object of a class. The constructor will also be the same name as the class, but may have several different parameters lists.

Default constructor (or no-arg constructor) - A constructor with no arguments or empty parameter list.

Driver - A class with a main method that is used to test another class by instantiating and using the object.

Scope - The scope of a variable or method is the locations in which that variable or method can be accessed. For example, the scope of a private variable is its own class.

Null reference (or null pointer) - An object that has not been initialized is considered null and any attempt to access or modify the object will result in a compile-time error. When designing classes, however, the failure to initialize a instance variable will result in a run-time error. (Program will compile, but will crash when that null variable is referenced)

Public Class- Implies this class is usable by other classes. If a class is not declared public, it is only usable by the classes inside its own package.

Public method - Accessible to all client programs.

Private method - Can only be accessed by that class, not from the outside.

Private variable - Can only be directly accessed by that class, not from the outside.

Static variable (or class variable) - Contains a value that is shared by all instances of the class. "Static" means that memory allocation happens once.

Constant (or static final variables) - These are assigned a value at declaration and cannot be changed.

Accessor (or getter or get method) - A method that accesses a class object without changing anything.

Mutator (or setter or set method) - A method that changes the state of an object by modifying at least one of its instance variables.

Static method - There is no reference to a specific instance in a static method. It cannot reference any instances of the class, nor can it reference any instance variables. It is typically used to work with variables or constants which were declared as static.

Instance method - A method that focuses on one instance of class and doesn't affect any other instance of that class.