Unit 9
- Why Use Classes
- Anatomy of a Class

Encapsulation

- A class (in OOP) encapsulates (groups together) data and actions relating to an object.

- Encapsulation hides the implementation details of the class from the users (clients) of the class.

Encapsulation (cont'd)

```
public class MyClass
{
    // Private fields:
    private  <sometype>  myField;
    ...

    // Constructors:
    public MyClass (...)   { ... }
    ...
    // Public methods:
    public  <sometype>  myMethod (...)   { ... }
    ...
    // Private methods:
    private  <sometype>  myMethod (...)  { ... }
    ...
}
```

Public interface: public constructors and methods

1. Fields (or instance variables)

- Fields describe the data of an object (*think nouns!!*).

- Fields are generally labeled private and therefore hidden from the client (only accessible from **within** the class).

2. Constructors

- Constructors describe ways to create an object of a class and initialize the fields.
- Constructors are always labeled public and have the same name as the class.
- There may be more than one constructor for a class. If more than one, each must have different arguments (or parameters).

3. Public Methods

- Methods describe the actions or questions that can be asked about an object (*think verbs!!*).

- Methods are generally labeled public. private (helper) methods can only be used within the class.

3. Public Methods (cont.)

- In general, 3 classifications of methods:
  1. Accessor – "gets" information about the object.
  2. Modifier – "sets" information about the object (or changes the state of the object).
  3. boolean – asks a true or false question about the object.

3. Public Methods (cont.)

```
public [or private]  returnType
                     methodName (type1 name1, ..., typeN nameN)
{
    ...
}
```

Body

Header

- To define a method:
  - decide between public and private (usually public)
  - give it a name
  - specify the types of parameters and give them names
  - specify the method's return type or choose void
  - write the method's code