**Major Themes of this course:**

* Abstraction
* Encapsulation
* Inheritance
* Polymorphism
* Delegation
* Design Patterns
* Asynchronous Programming

**Types:**

* **Value Types** hold a value:
  + Integers
  + Real Numbers
  + Boolean
  + Characters
* **Reference Types** can hold a reference to something else:
  + String
  + Array
  + Objects typed by their class
  + Classes themselves

**Method Signature:**

1. Access Modifier:
   1. Public, Private, Protected
   2. if not specified, *Package* access
2. Method Type
   1. *Static* or default
   2. The keyword *static* indicates a “class method”
   3. default is an “instance method”
3. Return Type
   1. type of value that the method returns
   2. if no return value, return is *void*
4. Method Name
   1. must start with a letter, $, or \_\_\_
   2. can contain letters, numbers, $, or \_\_\_
5. Parameter List
   1. List of named values that the method expects

**Principle of Encapsulation**

* Do not expose internal state of an object directly
* Separates exposed behavior from internal behavior
  + exposed behavior:
    - procedures / functions to be interacted with
  + internal behavior:
    - Procedures / functions not to be interacted with

**JavaBeans Conventions**

JavaBeans: Software engineering framework that relies on certain convention

Getters and Setters (Setters are void)

**Access Modifiers:**

* **Public:** Everyone has access
* **Private:** only class has access
* **Protected**: Class and subclasses have access

**Model View / Model View Controller**:

\*\*copy image from PPT