6. Object Oriented Concepts and Techniques

Lesson 2:
Encapsulation – Part 1

6.7. Encapsulation

- Encapsulation enables you to hide, inside the object, both the data fields and the methods that act on that data

- In an Object encapsulated data members are global to objects methods local to the object

6.7. Encapsulation
6.7.1. Modifiers

- Encapsulation is achieved by using Modifiers
- Modifiers are Language Keywords that modify the definition of a Class, Method or a Variable
- Access to variables and methods in Java classes is accomplished through Access Modifiers
- Access modifiers define varying levels of access between class members and the outside world (other objects)

6.7. Encapsulation
6.7.2. Access modifiers

- Access modifiers are declared immediately before the type of a member variable or the return type of a method.

- There are four access modifiers :
  - Default
  - Public
  - Protected
  - Private

6.7. Encapsulation

6.7.2.1. Default access modifier

- The Default access modifier is applied by default in the absence of an access modifier.
- Specifies that only classes in the same package (groups of related classes and interfaces) can have access to class's variables and methods

```
long length;
void getlength()
{
    return length;
}
```

- Notice that neither the member variable nor the method, supply an access modifier.
- So they take on the default access modifier implicitly

6.7. Encapsulation

6.7.2.2. public access modifier

- Specifies that class variables and methods are accessible to anyone, both inside and outside the class

- This means that public class members have global visibility and can be accessed by any other objects.

```
public int count;
public boolean isActive;
```

6.7. Encapsulation

6.7.2.3. protected access modifier

- Specifies that class members are accessible only to
- methods in that class and subclasses of that class.

- This means that protected class members have visibility
- limited to subclasses.

```
protected char middleinitial;
protected char getMiddleInitial()
{
   return middleInitial;
}
```

6.7. Encapsulation

6.7.2.4. private access modifier

- Specifies that class members are only accessible by the class they are defined in.

- This means that no other class has access to private class members, even subclasses.