**Java Access Specifiers**

Java Access Specifiers regulate access to classes, fields and methods in Java.

It is also known as Visibility Specifiers .

These Specifiers determine whether a field or method in a class, can be used or invoked by another method in another class or sub-class.

Access Specifiers can be used to restrict access.

**Access Specifiers In Java**

**Types Of Access Specifiers :**  
  
In java we have four Access Specifiers and they are listed below.  
 **1. Public**

Public Specifiers achieves the highest level of accessibility. Classes, methods, and fields declared as public can be accessed from any class in the Java program, whether these classes are in the same package or in another package.  
  
**Example :**

public class Demo { // public class

public x, y, size; // public instance variables

}

**2. Private**

Private Specifiers achieves the lowest level of accessibility.private methods and fields can only be accessed within the same class to which the methods and fields belong. private methods and fields are not visible within subclasses and are not inherited by subclasses.

**Example :**

public class Demo { // public class

private double x, y; // private (encapsulated) instance variables

public set(int x, int y) { // setting values of private fields

this.x = x;

this.y = y;

}

public get() { // setting values of private fields

return Point(x, y);

}

}

**3. Protected**

Methods and fields declared as protected can only be accessed by the subclasses in other package or any class within the package of the protected members' class. The protected access specifier cannot be applied to class and interfaces.  
**4. default(no specifier)**

When you don't set access specifier for the element, it will follow the default accessibility level.

