

## Access modifiers or access specifiers

=====

- It is used to define or declare the scope of object, variable, method and class.
  - There are 4 types of access modifiers in JAVA
    1. Private
    2. Protected
    3. Default
    4. Public
1. **Private** - The method or data member (variable) declared as private are accessible only within the class in which they are declared.

E.x

```
package testJava;
public class aclass
{
    private int i =10;           // variable with private access modifier
    private void test()         // method with private access modifier
    {
        System.out.println("test method");
    }

    public static void main(String[] args)
    {
        aclass ac = new aclass();
        ac.test();
        System.out.println(ac.i);
    }
}
```

Output >> test method  
10

-----

2. **Protected** - The method or data member (variable) declared as protected are accessible within the same package or in sub classes in different packages by using inheritance.

E.x Super class >>

```
package testJava;           // same package
```

```

public class aclass
{
    protected int i = 10;           // variable with protected access modifier
    protected void test()          // method with protected access modifier
    {
        System.out.println("test method");
    }

    public static void main(String[] args)
    {
        aclass ac = new aclass();
        ac.test();
        System.out.println(ac.i);
    }
}

```

---

Sub class >>

```

package testJava;                  // same package
public class sclass extends aclass
{
    public static void main(String[] args)
    {
        sclass sc = new sclass();
        sc.test();                  // Calling method having protected access
        System.out.println(sc.i);   // Calling variable having protected access
    }
}

```

---

Sub class >>

```

package testingClass;              // different package

import testJava.aclass;            // import class from another package

public class Variables extends aclass
{
    public static void main(String[] args)
    {
        Variables ver = new Variables();
        ver.test();
        System.out.println(ver.i);
    }
}

```

Output >>     test method  
                 10

---

3. **Public** - Classes, methods or data member (variable) declared as public are accessible from everywhere in the project.
- 

4. **Default** - The data member(variable), class or method which are not declared using any access modifiers i.e, having default access modifiers, which are accessible only within the same package.

E.x Super class >>

```
package testJava;           // same package
class aclass                 // class with default access modifier
{
    int i =10;               // variable with default access modifier
    void test()              // method with default access modifier
    {
        System.out.println("test method");
    }
}
```

Sub class >>

```
package testJava;           // same package
public class sclass extends aclass
{
    public static void main(String[] args)
    {
        sclass sc = new sclass();
        sc.test();           // Calling method having default access modifier
        System.out.println(sc.i); // Calling variable having default access modifier
    }
}
```

Output >>     test method  
                 10

---

Final Keyword

=====

Final is a keyword which is used to apply the restrictions on class, method and variable.

1. If the programmer declared the class with the final keyword then that class can not be inherited.
2. If the programmer declared the method with the final keyword then that method can not be override.
3. If the programmer declared the variable with the final keyword then that variable can not be changed or reinitiate.

E.x

```
package testJava;
```

```
final class aclass           // class declared with final keyword
{
    final int i= 10;         // variable declared with final keyword

    final void test()        // method declared with final keyword
    {
        System.out.println("test method");
    }
}
```