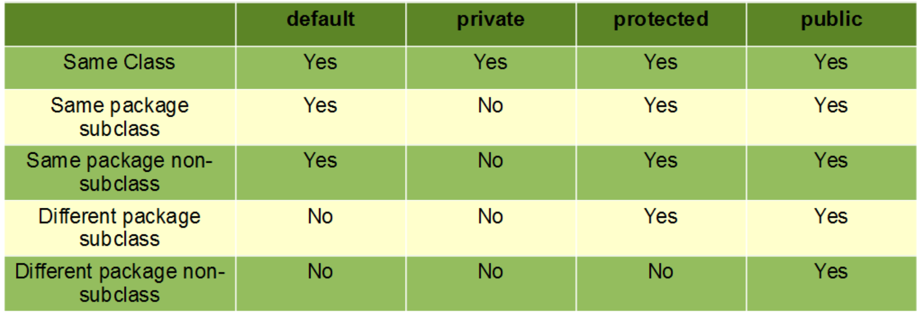
Access Modifires I

As the name suggests access modifiers in Java helps to restrict the scope of a class, constructor, variable, method, or data member.

✓ There are four types of access modifiers available in java:

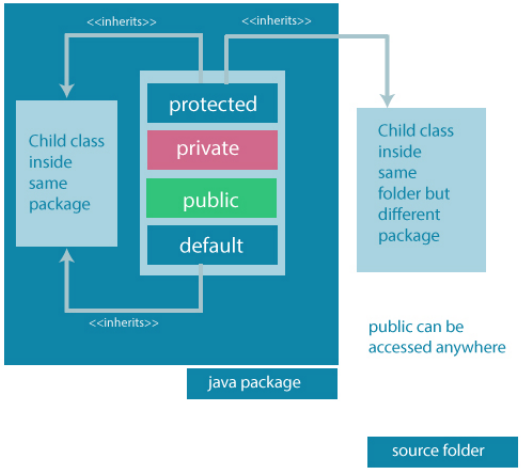
1. Default – No keyword required
2. Private
3. Protected
4. Public

Access Modifires II



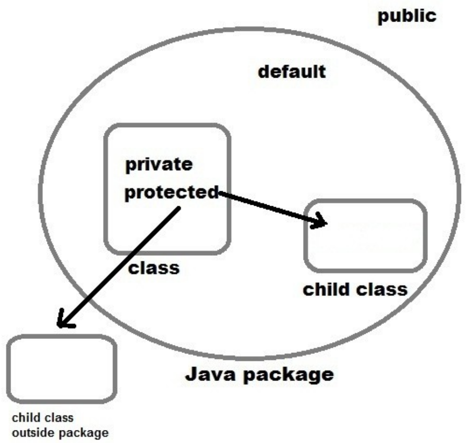
	default	private	protected	public
Same Class	Yes	Yes	Yes	Yes
Same package subclass	Yes	No	Yes	Yes
Same package non-subclass	Yes	No	Yes	Yes
Different package subclass	No	No	Yes	Yes
Different package non-subclass	No	No	No	Yes

Access Modifires III



<<inherits>>
<<inherits>>
protected
private
public
default
Child class inside same package
<<inherits>>
Child class inside same folder but different package
public can be accessed anywhere
java package
source folder

Access Modifires IV



Access Modifires V

✓ Default: When no access modifier is specified for a class, method, or data member – It is said to be having the default access modifier by default.

✓ The data members, class or methods which are not declared using any access modifiers i.e. having default access modifier are accessible only within the same package.

✓ In this example, we will create two packages and the classes in the packages will be having the default access modifiers and we will try to access a class from one package from a class of the second package.

Access Modifires VI

```java
// Java program to illustrate default modifier
package p1;
// Class McaMsc is having Default access modifier
class McaMsc
{
      void display()
      {
            System.out.println("Hello World!");
      }
}
```

```java
// Java program to illustrate error while using class from different package with default modifier
package p2;
import p1.*;
// This class is having default access modifier
class McaMscNew
{
      public static void main(String args[])
      {
            // Accessing class McaMsc from package p1
            McaMscs obj = new McaMsc();
            obj.display();
      }
```

Access Modifires VII

```
}
```

Private: The private access modifier is specified using the keyword private.

✓ The methods or data members declared as private are accessible only within the class in which they are declared.

✓ Any other class of the same package will not be able to access these members.

✓ Methods declared private are not inherited at all, so there is no rule for them.

✓ Top-level classes or interfaces can not be declared as private because private means “only visible within the enclosing class”. protected means “only visible within the enclosing class and any subclasses”.

Access Modifires VIII

✓ In this example, we will create two classes A and B within the same package p1. We will declare a method in class A as private and try to access this method from class B and see the result.

```java
// Java program to illustrate error while using class from different package with private modifier
package p1;
class A
{
        private void display()
        {
                System.out.println("McaMscsforMcaMscs");
        }
}
class B
{
        public static void main(String args[])
        {
                A obj = new A();
                // Trying to access private method
                // of another class
                obj.display();
        }
```

Access Modifires IX

```
}
Output:

error: display() has private access in A
obj.display();
```

Access Modifires X

protected: The protected access modifier is specified using the keyword protected.

✓ Variables, methods, and constructors, which are declared protected in a superclass can be accessed only by the subclasses in other package or any class within the package of the protected members' class.

✓ You can also say that the protected access modifier is similar to default access modifier with one exception that it has visibility in sub classes.

✓ The protected access modifier cannot be applied to class and interfaces. Methods, fields can be declared protected, however methods and fields in a interface cannot be declared protected.

✓ Protected access gives the subclass a chance to use the helper method or variable, while preventing a nonrelated class from trying to use it.

Access Modifires XI

✓ In this example, we will create two packages p1 and p2. Class A in p1 is made public, to access it in p2. The method display in class A is protected and class B is inherited from class A and this protected method is then accessed by creating an object of class B.

Access Modifires XII

```java
// Java program to illustrate protected modifier
package p1;
public class A
{
      protected void display()
      {
            System.out.println("Hello World");
      }
}
```

```java
// Java program to illustrate protected modifier
package p2;
import p1.*; // importing all classes in package p1
// Class B is subclass of A
class B extends A
{
      public static void main(String args[])
      {
            B obj = new B();
            obj.display();
      }

}
```

Access Modifires XIII

```
Output:
Hello World
```

Access Modifires XIV

```java
package package1;

public class Animal
{
    private String secret;  // Only accessible in Animal class
    protected String name;  // Accessible in package1 and subclasses

    public Animal(String name)
    {
        this.name = name;
        this.secret = "hidden";
    }

    private void privateMethod()
    {
        System.out.println("Private method");
    }

    protected void protectedMethod()
    {
        System.out.println("Protected method");
    }
}
```

Access Modifires XV

Access Modifires XVI

```java
package package2;

import package1.Animal;

public class Dog extends Animal
{
      public Dog(String name)
      {
            super(name);
      }

      public void demo()
      {
            // this.secret = "test";   // Compile error - private
            this.name = "Buddy";       // OK - protected
            // this.privateMethod();   // Compile error - private
            this.protectedMethod();    // OK - protected (subclass)
      }
}
```

Access Modifires XVII

public: The public access modifier is specified using the keyword public.

✓ The public access modifier has the widest scope among all other access modifiers.

✓ Classes, methods, or data members that are declared as public are accessible from everywhere in the program. There is no restriction on the scope of public data members.

```java
// Java program to illustrate
// public modifier
package p1;
public class A
{
      public void display()
      {
            System.out.println("McaMscsforMcaMscs");
      }
}
package p2;
import p1.*;
class B
{
      public static void main(String args[])
      {
```

Access Modifires XVIII

```
             A obj = new A;
             obj.display();
      }
}
```

✓ Access modifiers are mainly used for encapsulation. I can help us to control what part of a program can access the members of a class. So that misuse of data can be prevented.