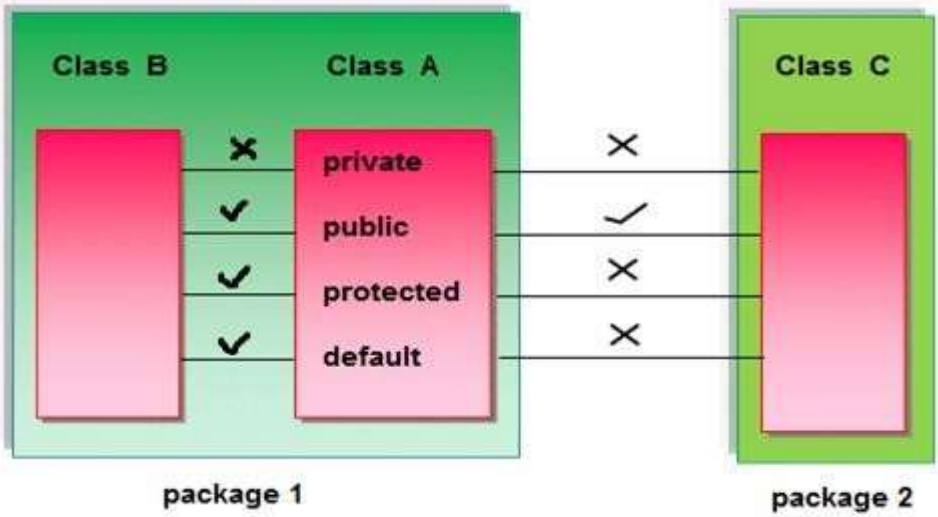
Access Modifiers in Java

Access modifiers are those which are applied before data members or methods of a class. These are used to suggest "where to access and where not to access the data members or methods".
In java programming these are classified into four types:

- Private
- Default (not a keyword)
- Protected
- Public

Note: Default is not a keyword (where as public, private, protected are keywords)

If we are not using private , protected and public keywords then JVM is by default takesdefault access mode.
Access modifiers are always used for how to reuse the features within the package and access the package between class to class, interface to interface and interface to class. Access modifiers provide features accessing and controlling mechanism among the classes and interfaces.



Note: A protected member of class is accessible within the same class and other class of same package and also accessible in inherited class of other package.

Rules for access modifiers:

The following diagram gives rules for Access

Modifiers	Within Same Class	Within other class of Same package	Within derived class of other package	Within external Class of other package
Private (Class level A.S)	Yes	No	No	No
Default (Package level A.S)	Yes	Yes	No	No
Protected (Derived level A.S)	Yes	Yes	Yes	No
Public (Universal A.S)	Yes	Yes	Yes	Yes

A.S --> Access Specifier

modifiers.

Private: private members of a class are only accessible within the class, but not accessible anywhere in the program. Private is also called class level access modifier.

Example

```
class
Hello{
private int a=20;
private void
show(){
System.out.println("Hello java");
    }
}
public class Demo{
 public static void main(String args[]){
  Hello obj=new Hello();
  System.out.println(obj.a); //Compile Time Error, you can't access private data
  obj.show();  //Compile Time Error, you can't access private methods
    }
}
```

public: public members of any class are accessible anywhere in the program: inside the same class and outside of class, within the same package and outside of the package. public is also called as universal access modifier.

Example

```
class
Hello{
public int a=20;
public void
show(){
System.out.println("Hello java");
    }
}
public class Demo{
 public static void main(String args[]){
  Hello obj=new Hello();
  System.out.println(obj.a);
  obj.show();
     }
}
```

Output

```
20
Hello Java
```

Protected: protected members of a class are accessible within the same class and other class of the same package and also accessible in inherited class of the other package. Protected is also called derived level access modifier.

In below example we have created two packages pack1 and pack2. In pack1, class A is public so we can access this class outside of pack1 but method show is declared as a protected so it is only access outside of package pack1 only through inheritance.

Example

```
// save A.java
package pack1;
public class A
```