Java programming

Access modifiers in Java :-

There are two types of modifiers in Java

Access modifiers

Non- access modifiers

Access modifiers in Java specifies the accessibility of scope of a field, method, constructor or class.

We can change the access level of fields, constructors, methods & class by applying the access modifiers on it.

There are four types of access modifiers in Java they are:-

1. private
2. public
3. Default
4. protected.

private:-

The access level of a private modifier is only within the class. It cannot be accessed from outside the class.

Default :-

The access level of a default modifier is only within the package.

It cannot be accessed from outside the package. If you do not specify any access level, it will be the default.

protected :-

The access level of a protected modifier is within the package & outside the package through child class.

If you do not make the child class, it cannot be accessed from outside the package.

Public :-

The access level of a public modifier is everywhere. It can be accessed from within the class, outside the class, within the package & outside the package.

There are many non access modifiers, such as static, abstract, synchronized, native, volatile etc.

Access modifier	within class	within Package	Outside Package by subclass only	Outside Package
private	Y	N	N	N
Default	Y	Y	N	N
protected	Y	Y	Y	N
public	Y	Y	Y	Y

The private access modifier is accessible only within the class

example of private cloud

```
class A{
private int data = 40;
 private void msg()
 {System.out.println("Hello java");}
}
Public class simple {
public static void main(String [] args) {
   A obj = new A();
System.out.println(obj.data);
  obj.msg();
 }
}
```

example of Default

```
package my pack;
 import pack.*;
  class B{
 public static void
   main(String args[]
A obj = new A();
obj.msg();
 }
}
```

Example of protected

```
Package my pack;
import pack.*;
class B extends A {
public static void main
(string args [])
B obj = new B();
Obj. msg();
}
}
```

Example of public

```
package my pack;
import pack.*;
class B {
public static void
main (string args [])
A obj = new A();
obj. msg();
}
}
```

Java Access modifiers with method overriding

```
class A {
protected void msg()
{
system.out.println ("Hello java"); }
public class simple extends A {
void msg()
{ system. out. println ("Hello java"); }
public void main (String [] args) {
simple obj = new simple ();
obj. msg();
}
}
```