Access Modifiers in java

There are two types of modifiers in [java](http://www.javatpoint.com/access-modifiers): [**access modifiers**](http://www.javatpoint.com/access-modifiers) and **non-**[**access modifiers**](http://www.javatpoint.com/access-modifiers).

The [access modifiers in java](http://www.javatpoint.com/access-modifiers) specifies accessibility (scope) of a [data member](http://www.javatpoint.com/access-modifiers), [method](http://www.javatpoint.com/access-modifiers), [constructor](http://www.javatpoint.com/access-modifiers) or [class](http://www.javatpoint.com/access-modifiers).

There are 4 types of [java](http://www.javatpoint.com/access-modifiers) [access modifiers](http://www.javatpoint.com/access-modifiers):

1. [private](http://www.javatpoint.com/access-modifiers)
2. [default](http://www.javatpoint.com/access-modifiers)
3. [protected](http://www.javatpoint.com/access-modifiers)
4. [public](http://www.javatpoint.com/access-modifiers)

There are many non-[access modifiers](http://www.javatpoint.com/access-modifiers) such as static, abstract, synchronized, native, volatile, transient etc. Here, we will learn [access modifiers](http://www.javatpoint.com/access-modifiers).

**1)** [**private**](http://www.javatpoint.com/access-modifiers) **access modifier**

|  |
| --- |
| The [private](http://www.javatpoint.com/access-modifiers) access modifier is accessible only within [class](http://www.javatpoint.com/access-modifiers). |

**Simple example of** [**private**](http://www.javatpoint.com/access-modifiers) **access modifier**

|  |
| --- |
| In this example, we have created two classes A and Simple. A [class](http://www.javatpoint.com/access-modifiers) contains [private](http://www.javatpoint.com/access-modifiers) [data member](http://www.javatpoint.com/access-modifiers) and [private](http://www.javatpoint.com/access-modifiers) [method](http://www.javatpoint.com/access-modifiers). We are accessing these [private](http://www.javatpoint.com/access-modifiers) members from outside the [class](http://www.javatpoint.com/access-modifiers), so there is compile time error. |

1. [class](http://www.javatpoint.com/access-modifiers) A{
2. [private](http://www.javatpoint.com/access-modifiers) int data=40;
3. [private](http://www.javatpoint.com/access-modifiers) void msg(){System.out.println("Hello [java](http://www.javatpoint.com/access-modifiers)");}
4. }
6. [public](http://www.javatpoint.com/access-modifiers) [class](http://www.javatpoint.com/access-modifiers) Simple{
7. [public](http://www.javatpoint.com/access-modifiers) static void main(String args[]){
8. A obj=new A();
9. System.out.println(obj.data);//Compile Time Error
10. obj.msg();//Compile Time Error
11. }
12. }

**Role of** [**Private**](http://www.javatpoint.com/access-modifiers)[**Constructor**](http://www.javatpoint.com/access-modifiers)

|  |
| --- |
| If you make any [class](http://www.javatpoint.com/access-modifiers) [constructor](http://www.javatpoint.com/access-modifiers) [private](http://www.javatpoint.com/access-modifiers), you cannot create the instance of that [class](http://www.javatpoint.com/access-modifiers) from outside the [class](http://www.javatpoint.com/access-modifiers). For example: |

1. [class](http://www.javatpoint.com/access-modifiers) A{
2. [private](http://www.javatpoint.com/access-modifiers) A(){}//[private](http://www.javatpoint.com/access-modifiers) [constructor](http://www.javatpoint.com/access-modifiers)
3. void msg(){System.out.println("Hello [java](http://www.javatpoint.com/access-modifiers)");}
4. }
5. [public](http://www.javatpoint.com/access-modifiers) [class](http://www.javatpoint.com/access-modifiers) Simple{
6. [public](http://www.javatpoint.com/access-modifiers) static void main(String args[]){
7. A obj=new A();//Compile Time Error
8. }
9. }

**Note: A** [**class**](http://www.javatpoint.com/access-modifiers) **cannot be** [**private**](http://www.javatpoint.com/access-modifiers) **or** [**protected**](http://www.javatpoint.com/access-modifiers) **except nested** [**class**](http://www.javatpoint.com/access-modifiers)**.**

**2)** [**default**](http://www.javatpoint.com/access-modifiers) **access modifier**

|  |
| --- |
| If you don't use any modifier, it is treated as [**default**](http://www.javatpoint.com/access-modifiers) bydefault. The [default](http://www.javatpoint.com/access-modifiers) modifier is accessible only within package. |

**Example of** [**default**](http://www.javatpoint.com/access-modifiers) **access modifier**

|  |
| --- |
| In this example, we have created two packages pack and mypack. We are accessing the A [class](http://www.javatpoint.com/access-modifiers) from outside its package, since A [class](http://www.javatpoint.com/access-modifiers) is not [public](http://www.javatpoint.com/access-modifiers), so it cannot be accessed from outside the package. |

1. //save by A.[java](http://www.javatpoint.com/access-modifiers)
2. package pack;
3. [class](http://www.javatpoint.com/access-modifiers) A{
4. void msg(){System.out.println("Hello");}
5. }
6. //save by B.[java](http://www.javatpoint.com/access-modifiers)
7. package mypack;
8. import pack.\*;
9. [class](http://www.javatpoint.com/access-modifiers) B{
10. [public](http://www.javatpoint.com/access-modifiers) static void main(String args[]){
11. A obj = new A();//Compile Time Error
12. obj.msg();//Compile Time Error
13. }
14. }

In the above example, the scope of [class](http://www.javatpoint.com/access-modifiers) A and its [method](http://www.javatpoint.com/access-modifiers) msg() is [default](http://www.javatpoint.com/access-modifiers) so it cannot be accessed from outside the package.

**3)** [**protected**](http://www.javatpoint.com/access-modifiers) **access modifier**

The [**protected**](http://www.javatpoint.com/access-modifiers) **access modifier** is accessible within package and outside the package but through inheritance only.

The [protected](http://www.javatpoint.com/access-modifiers) access modifier can be applied on the [data member](http://www.javatpoint.com/access-modifiers), [method](http://www.javatpoint.com/access-modifiers) and [constructor](http://www.javatpoint.com/access-modifiers). It can't be applied on the [class](http://www.javatpoint.com/access-modifiers).

**Example of** [**protected**](http://www.javatpoint.com/access-modifiers) **access modifier**

In this example, we have created the two packages pack and mypack. The A [class](http://www.javatpoint.com/access-modifiers) of pack package is [public](http://www.javatpoint.com/access-modifiers), so can be accessed from outside the package. But msg [method](http://www.javatpoint.com/access-modifiers) of this package is declared as [protected](http://www.javatpoint.com/access-modifiers), so it can be accessed from outside the [class](http://www.javatpoint.com/access-modifiers) only through inheritance.

1. //save by A.[java](http://www.javatpoint.com/access-modifiers)
2. package pack;
3. [public](http://www.javatpoint.com/access-modifiers) [class](http://www.javatpoint.com/access-modifiers) A{
4. [protected](http://www.javatpoint.com/access-modifiers) void msg(){System.out.println("Hello");}
5. }
6. //save by B.[java](http://www.javatpoint.com/access-modifiers)
7. package mypack;
8. import pack.\*;
10. [class](http://www.javatpoint.com/access-modifiers) B extends A{
11. [public](http://www.javatpoint.com/access-modifiers) static void main(String args[]){
12. B obj = new B();
13. obj.msg();
14. }
15. }

Output:Hello

**4)** [**public**](http://www.javatpoint.com/access-modifiers) **access modifier**

|  |
| --- |
| The [**public**](http://www.javatpoint.com/access-modifiers) **access modifier** is accessible everywhere. It has the widest scope among all other modifiers. |

**Example of** [**public**](http://www.javatpoint.com/access-modifiers) **access modifier**

1. //save by A.[java](http://www.javatpoint.com/access-modifiers)
3. package pack;
4. [public](http://www.javatpoint.com/access-modifiers) [class](http://www.javatpoint.com/access-modifiers) A{
5. [public](http://www.javatpoint.com/access-modifiers) void msg(){System.out.println("Hello");}
6. }
7. //save by B.[java](http://www.javatpoint.com/access-modifiers)
9. package mypack;
10. import pack.\*;
12. [class](http://www.javatpoint.com/access-modifiers) B{
13. [public](http://www.javatpoint.com/access-modifiers) static void main(String args[]){
14. A obj = new A();
15. obj.msg();
16. }
17. }

Output:Hello

**Understanding all** [**java**](http://www.javatpoint.com/access-modifiers)[**access modifiers**](http://www.javatpoint.com/access-modifiers)

Let's understand the [access modifiers](http://www.javatpoint.com/access-modifiers) by a simple table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Access Modifier** | **within** [**class**](http://www.javatpoint.com/access-modifiers) | **within package** | **outside package by subclass only** | **outside package** |
| [**Private**](http://www.javatpoint.com/access-modifiers) | Y | N | N | N |
| [**Default**](http://www.javatpoint.com/access-modifiers) | Y | Y | N | N |
| [**Protected**](http://www.javatpoint.com/access-modifiers) | Y | Y | Y | N |
| [**Public**](http://www.javatpoint.com/access-modifiers) | Y | Y | Y | Y |

[**Java**](http://www.javatpoint.com/access-modifiers)[**access modifiers**](http://www.javatpoint.com/access-modifiers) **with** [**method**](http://www.javatpoint.com/access-modifiers) **overriding**

If you are overriding any [method](http://www.javatpoint.com/access-modifiers), overridden [method](http://www.javatpoint.com/access-modifiers) (i.e. declared in subclass) must not be more restrictive.

1. [class](http://www.javatpoint.com/access-modifiers) A{
2. [protected](http://www.javatpoint.com/access-modifiers) void msg(){System.out.println("Hello [java](http://www.javatpoint.com/access-modifiers)");}
3. }
5. [public](http://www.javatpoint.com/access-modifiers) [class](http://www.javatpoint.com/access-modifiers) Simple extends A{
6. void msg(){System.out.println("Hello [java](http://www.javatpoint.com/access-modifiers)");}//C.T.Error
7. [public](http://www.javatpoint.com/access-modifiers) static void main(String args[]){
8. Simple obj=new Simple();
9. obj.msg();
10. }
11. }

|  |
| --- |
| The [default](http://www.javatpoint.com/access-modifiers) modifier is more restrictive than [protected](http://www.javatpoint.com/access-modifiers). That is why there is compile time error. |