# **Access Modifiers in Java**

In Java, Access modifiers help to restrict the scope of a class, constructor, variable, method, or data member. It provides security, accessibility, etc to the user depending upon the access modifier used with the element.

**Types of Access Modifiers in Java**

There are four types of access modifiers available in Java:

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

### **1. Default Access Modifier**

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, classes, or methods that are not declared using any access modifiers i.e. having default access modifiers are accessible **only within the same package**.

### **2. Private Access Modifier**

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.

### **3. Protected Access Modifier**

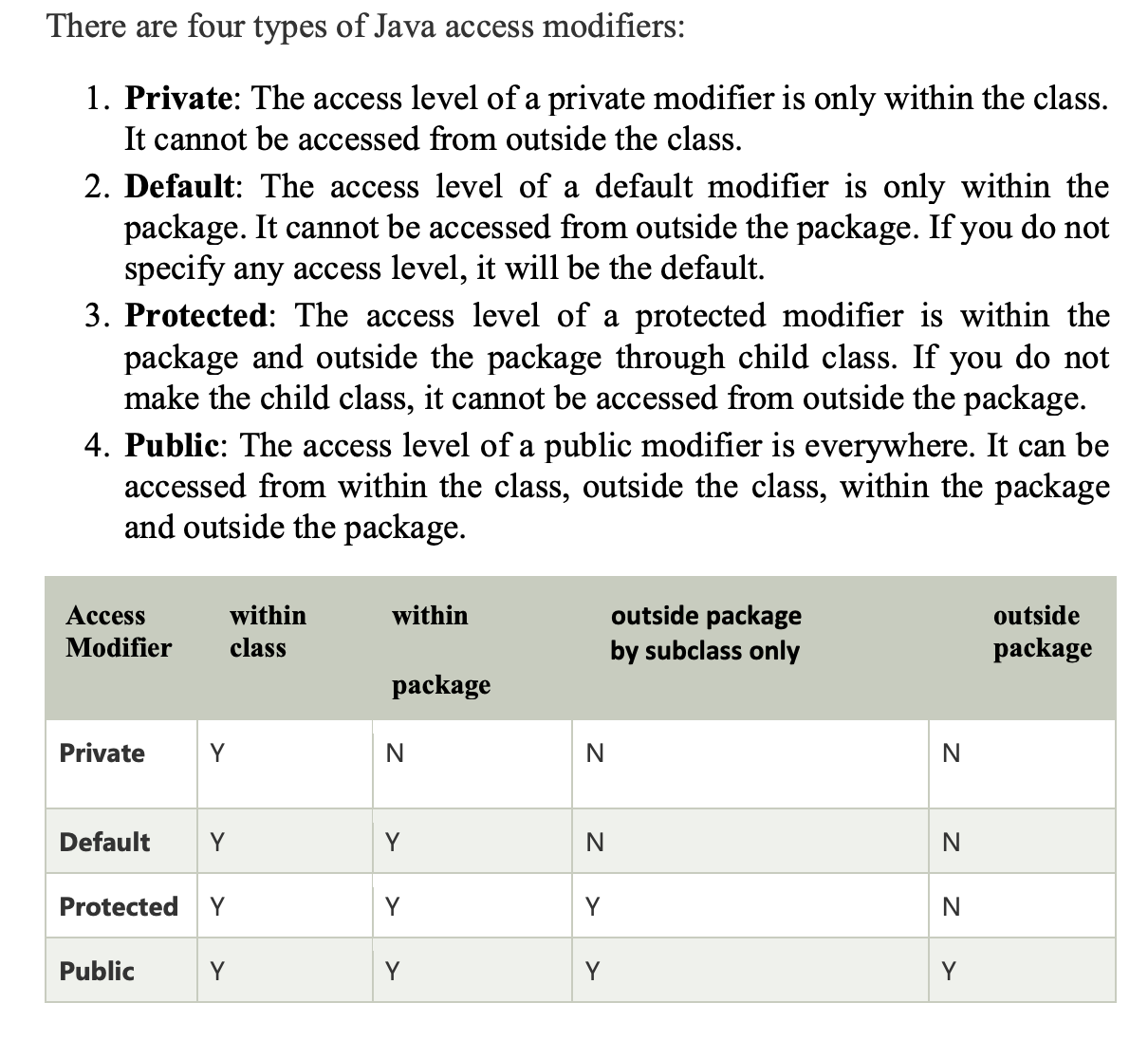
The protected access modifier is specified using the keyword **protected**.

The methods or data members declared as protected are **accessible within the same package or subclasses in different packages.**

### **4. Public Access Modifier**

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.



### **1. What are access modifiers in Java?**

*Access modifiers in Java are the keywords that are used for controlling the use of the methods, constructors, fields, and methods in a class.*

### **2. What is void in Java?**

*Void in Java is used to specify no return value with the method.*

### **3. What are the 12 modifiers in Java?**

*12 Modifiers in Java are public, private, protected, default, final, synchronized, abstract, native, strictfp, transient, and volatile.*