

## Access Modifiers

In object-oriented programming, an access modifier is a keyword that determines the visibility and accessibility of class members (such as variables, methods, and constructors) from other parts of a program. The access modifier specifies the level of access or restriction placed on the member.

In most programming languages, there are typically three main access modifiers:

**Public:** When a member is declared as public, it can be accessed from any part of the program, including outside the class or even in other classes. Public members are not subject to any access restrictions.

**Private:** Private members are accessible only within the class in which they are defined. They cannot be accessed from outside the class, including other classes. Private members are often used to encapsulate implementation details and provide data hiding.

**Protected:** Protected members are accessible within the package. They are not accessible outside the class hierarchy. Protected members allow subclasses to inherit and use the member, but they are hidden from other classes.

**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.

## Understanding Java Access Modifiers

Let's understand the access modifiers in Java by a simple table.

<b>Access Modifier</b>	<b>within class</b>	<b>within package</b>	<b>outside package by subclass only</b>	<b>outside package</b>
<b>Private</b>	Y	N	N	N
<b>Default</b>	Y	Y	N	N
<b>Protected</b>	Y	Y	Y	N
<b>Public</b>	Y	Y	Y	Y