

## Definition:

**Access modifiers/access specifiers:** are keywords in object-oriented languages that set the accessibility of classes, methods, and other members.

**Access modifiers** are a specific part of programming language syntax used to facilitate the encapsulation of components.

## In C++

There are only three access modifiers.

The three access modifiers are: **public, private and protected.** (see in the table below)

**C#** extends the number of them to six. These are: **private, private protected, protected internal, protected, internal and public.**

## Java

Has four access modifiers, but three keywords for this purpose.

These includes: **public, private, package and protected.** (see in the table below)

When a class is declared as:

**Public:-** it is accessible to other classes defined in the same package as well as those defined in other packages. This is the most commonly used specifier for classes.

A class cannot be declared as **private**. If no access specifier is stated, the default access restrictions will be applied. The class will be accessible to other classes in the same package but will be inaccessible to classes outside the package.

When we say that a class is **inaccessible**, it simply means that we cannot create an object of that class or declare a variable of that class type. The protected access specifier too cannot be applied to a class.

The access modifier **package** is the default, and is used if any other access modifier keyword is missing.

The meaning of these modifiers may differ from one language to another.

A comparison of the keywords, ordered from the most restrictive to the most open, and their meaning in these three languages follows. Their visibility ranges from the same class to the package where the class is defined to a general access permission.

Below, the maximal access is written into the table.

Keyword	C#	C++	Java
<b>private</b>	class	class	class
<b>private protected</b>	derived classes in the same assembly	-	-
<b>protected internal</b>	same assembly <i>and/or</i> derived classes	-	-
<b>protected</b>	derived classes	derived classes	derived classes <i>and/or</i> within same package
<b>package</b>	-	-	within its package
<b>internal</b>	same assembly	-	-
<b>public</b>	everybody	everybody	everybody