Difference between private, protected, public and package modifier or keyword in Java

private vs public vs protected vs package in Java

Java has four access modifiers namely private, protected, and public. package level access is the **default access** level provided by Java if no access modifier is specified. These access modifiers are used to restrict the accessibility of a class, method, or variable on which it applies. We will start from the private access modifier which is the most restrictive access modifier and then go towards the public which is the least restrictive access modifier, along the way we will see some best practices while using access modifier in Java and some examples of using private and protected keywords.

private keyword in Java

private keyword or modifier in java can be applied to member field, method, or nested class in Java. you can not use the private modifier on top-level classes. private variables, methods, and classes are only accessible on the class on which they are declared.

private is the highest form of Encapsulation Java API provides and should be used as much as possible. It's best coding practice in Java to declare variables private by default. a private method can only be called from the class where it has been declared.

As per Rules of method overriding in Java, a private method can not be overridden as well. the private keyword can also be applied to the constructor and if you make constructor private you prevent it from being sub-classed.

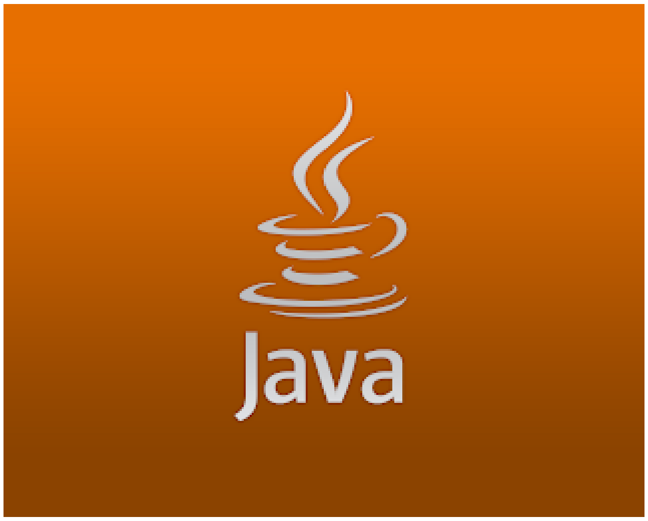
A popular example of making the constructor private is Singleton class in Java which provides getInstance() method to get object instead of creating a new object using the constructor in Java. here are some differences between private and protected, public and package level access

package or default access level in Java

there is no access modifier called package instead `package` is a keyword which is used to declare a package in Java, a package is a directory on which a class in Java belongs. Package or default access level is second highest restrictive access modifier after `private` and any variable, method or class declared as `package-private` is only accessible on the package it belongs. the good thing about default modifier is that top level class can also be package-private if there is no class level access modifier.

protected keyword in Java

The difference between `private` and `protected` keyword is that protected method, variable or nested class not only accessible inside a class, inside the package but also outside of package on a subclass. if you declare a variable `protected` means anyone can use it if they extend your class. the top level class can not be make `protected` as well.



public keyword in Java

`public` is the least restrictive access modifier in Java programming language and its bad practice to declare field, method or class by default public because once you make it public it's very difficult to make any change on the internal structure of class as it affects all clients using it.

Making class or instance variable public also violated the principle of Encapsulation which is not good at all and affects maintenance badly. instead of making variable `public` you should make it `private` and provided public getter and setter. `the public` modifier can also be applied to a top-level class. In Java name of the file must be the same as the public class declared in the file.

That's all difference between `private`, `protected`, `package,` and `public` access modifiers. As you have seen the difference between private and public lies in how accessible a particular field, method, or class would have. public means you can access it anywhere while private means you can only access it inside its own clas