# ‘ Final ’ keyword

**In Java, the final keyword is used to declare constants, define immutable variables, and indicate that a method or class cannot be overridden. Here are the common places where the final keyword can be used in Java:**

1. **Variables : You can use the final keyword to declare a variable as a constant,**

**which means its value cannot be changed once assigned.**

**For example :-**

JAVA CODE

final int MAX\_VALUE = 100;

final String MESSAGE = "Hello”;

1. **Methods : When you declare a method with the final keyword, it means the method**

**cannot be overridden by subclasses. This is useful when you want to**

**prevent any modifications to the method implementation.**

**For example :-**

JAVA CODE

public final void printMessage() {

System.out.println("Message");

}

1. **Classes : You can use the final keyword to declare a class that cannot be extended**

**by other classes. This is useful when you want to ensure that a class is**

**not subclassed or overridden.**

**For example :-**

JAVA CODE

Public final class FinalClass {

// Class implementation

}

1. **Method parameters: By using the final keyword on a method parameter, you can**

**indicate that the value of that parameter cannot be modified**

**within the method. This is useful for documenting the**

**intention of the code and preventing accidental**

**modifications.**

**For example :-**

JAVA CODE

public void process(final int value) {

// value cannot be modified within this method

}

1. **Local variables: Similar to method parameters, you can use the final keyword on**

**local variables to indicate that their values should not be**

**modified once assigned. This is useful for ensuring the**

**immutability of variables within a method or block.**

**For example :-**

JAVA CODE

Public void doSomething( ) {

final int count = 10;

// count cannot be modified beyond this point

}

**Note that :- The usage of the final keyword depends on the specific requirements of your program. It is a good practice to use it judiciously, focusing on cases where immutability or prevention of overriding is necessary.**