Return Statement in Java

What is a return statement in Java?

In Java programming, the return statement is used for returning a value when the execution of the block is completed. The return statement inside a loop will cause the loop to break and further statements will be ignored by the compiler.

Returning a Value from a Method

1. In Java, every method is declared with a return type such as int, float, double, string, etc.
2. These return types required a return statement at the end of the method. A return keyword is used for returning the resulted value.

1. The void return type doesn't require any return statement. If we try to return a value from a void method, the compiler shows an error.

Following are the important points must remember while returning a value:

* The return type of the method and type of data returned at the end of the method should be of the same type. For example, if a method is declared with the float return type, the value returned should be of float type only.
* The variable that stores the returned value after the method is called should be a similar data type otherwise, the data might get lost.
* If a method is declared with parameters, the sequence of the parameter must be the same while declaration and method call.

Syntax:

The syntax of a return statement is the return keyword is followed by the value to be returned.

**return** returnvalue;

The following Java programs demonstrate the use of return statements.

**SampleReturn1.java**

**public** **class** SampleReturn1

{

    /\* Method with an integer return type and no arguments \*/

**public** **int** CompareNum()

    {

**int** x = 3;

**int** y = 8;

        System.out.println("x = " + x + "\ny = " + y);

**if**(x>y)

**return** x;

**else**

**return** y;

    }

    /\* Driver Code \*/

**public** **static** **void** main(String ar[])

    {

        SampleReturn1 obj = **new** SampleReturn1();

**int** result = obj.CompareNum();

        System.out.println("The greater number among x and y is: " + result);

    }

}

**Output:**

x = 3

y = 8

The greater number among x and y is: 8