

JAVA.LANG.STRICTMATH.SCALB METHOD

http://www.tutorialspoint.com/java/lang/strictmath_scalb_double.htm

Copyright © tutorialspoint.com

Description

The **java.lang.StrictMath.scalb***doubled, intscaleFactor* method returns **$d \times 2^{\text{scaleFactor}}$** rounded as if performed by a single correctly rounded floating-point multiply to a member of the double value set. It includes these cases:

- If the first argument is NaN, NaN is returned.
- If the first argument is infinite, then an infinity of the same sign is returned.
- If the first argument is zero, then a zero of the same sign is returned.

Declaration

Following is the declaration for **java.lang.StrictMath.scalb** method

```
public static double scalb(double d, int scaleFactor)
```

Parameters

- **d** -- This is the number to be scaled by a power of two.
- **scaleFactor** -- This is the power of 2 used to scale d.

Return Value

This method returns the $d \times 2^{\text{scaleFactor}}$

Exception

- **NA**

Example

The following example shows the usage of java.lang.StrictMath.scalb method.

```
package com.tutorialspoint;

import java.lang.*;

public class StrictMathDemo {

    public static void main(String[] args) {

        double d1 = 3.8 , d2 = 4.9, d3 = 0.0;
        int power = 2;

        // returns (First argument*(pow(second argument,2) i.e  $d \times 2^{\text{scaleFactor}}$ )

        double scalbValue = StrictMath.scalb(d1, power);
        System.out.println("value = " + scalbValue);

        scalbValue = StrictMath.scalb(d2, power);
        System.out.println("value = " + scalbValue);

        scalbValue = StrictMath.scalb(d3, power);
        System.out.println("value = " + scalbValue);
    }
}
```

Let us compile and run the above program, this will produce the following result:

```
value = 15.2  
value = 19.6  
value = 0.0
```

Loading [Mathjax]/jax/output/HTML-CSS/jax.js