

JAVA.LANG.STRICTMATH.CBRT METHOD

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

Copyright © tutorialspoint.com

Description

The **java.lang.StrictMath.cbrt** method returns the cube root of a double value. For positive finite x , **cbrt**- $x == -\text{cbrt}x$; that is, the cube root of a negative value is the negative of the cube root of that value's magnitude. This includes these cases:

- If the argument is NaN, then the result is NaN.
- If the argument is infinite, then the result is an infinity with the same sign as the argument.
- If the argument is zero, then the result is a zero with the same sign as the argument.

Declaration

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

```
public static double cbrt(double a)
```

Parameters

- **a** -- This is the double value

Return Value

This method returns the cube root of a .

Exception

- **NA**

Example

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

```
package com.tutorialspoint;

import java.lang.*;

public class StrictMathDemo {

    public static void main(String[] args) {

        double d1 = 8.05 , d2 = 729, d3 = 0;

        // returns the cube root of a double value
        double cbrtValue = StrictMath.cbrt(d1);
        System.out.println("Cube root of " + d1 + " = " + cbrtValue);

        cbrtValue = StrictMath.cbrt(d2);
        System.out.println("Cube root of " + d2 + " = " + cbrtValue);

        cbrtValue = StrictMath.cbrt(d3);
        System.out.println("Cube root of " + d3 + " = " + cbrtValue);
    }
}
```

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

```
Cube root of 8.05 = 2.0041580161269152
Cube root of 729 = 9.0
Cube root of 0 = 0.0
```

