

JAVA.MATH.BIGDECIMAL.HASHCODE METHOD

http://www.tutorialspoint.com/java/math/bigdecimal_hashcode.htm

Copyright © tutorialspoint.com

Description

The **java.math.BigDecimal.hashCode** returns the hash code for this BigDecimal. Two BigDecimal objects that are numerically equal but differ in scale *like 2.0 and 2.00* will generally not have the same hash code.

Declaration

Following is the declaration for **java.math.BigDecimal.hashCode** method

```
public int hashCode()
```

Overrides

- hashCode in class **Object**

Parameters

- NA

Return Value

This method returns the hashCode Value of the BigDecimal Object

Exception

- NA

Example

The following example shows the usage of math.BigDecimal.hashCode method

```
package com.tutorialspoint;

import java.math.*;

public class BigDecimalDemo {

    public static void main(String[] args) {

        // create 3 BigDecimal objects
        BigDecimal bg1, bg2, bg3;

        // create 3 int objects
        int i1, i2, i3;

        bg1 = new BigDecimal("125");
        bg2 = new BigDecimal("125.50");
        bg3 = new BigDecimal("125.80");

        // assign the HashCode value of bg1, bg2, bg3 to i1, i2, i3
        // respectively
        i1 = bg1.hashCode();
        i2 = bg2.hashCode();
        i3 = bg3.hashCode();

        String str1 = "HashCode of " + bg1 + " is " + i1;
        String str2 = "HashCode of " + bg2 + " is " + i2;
        String str3 = "HashCode of " + bg3 + " is " + i3;

        // print i1, i2, i3 values
```

```
        System.out.println( str1 );  
        System.out.println( str2 );  
        System.out.println( str3 );  
    }  
}
```

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

```
HashCode of 125 is 3875  
HashCode of 125.50 is 389052  
HashCode of 125.80 is 389982
```

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