

JAVA.MATH.BIGDECIMAL.INTVALUE METHOD

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

Copyright © tutorialspoint.com

Description

The **java.math.BigDecimal.intValue** converts this BigDecimal to an int. This conversion is analogous to the narrowing primitive conversion from double to short. Any fractional part of this BigDecimal will be discarded, and if the resulting "BigInteger" is too big to fit in an int, only the low-order 32 bits are returned.

This conversion can lose information about the overall magnitude and precision of this BigDecimal value as well as return a result with the opposite sign.

Declaration

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

```
public int intValue()
```

Specified by

- intValue in class **Number**

Parameters

- NA

Return Value

This method returns the int value of the BigDecimal Object

Exception

- NA

Example

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

```
package com.tutorialspoint;

import java.math.*;

public class BigDecimalDemo {

    public static void main(String[] args) {

        // create 3 BigDecimal objects
        BigDecimal bg1, bg2;

        //Create 2 int Object
        int i1, i2;

        bg1 = new BigDecimal("1234");

        //assign a larger value to bg2
        bg2 = new BigDecimal("3383878445");

        // assign the int value of bg1 and bg2 to i1,i2 respectively
        i1 = bg1.intValue();
        i2 = bg2.intValue();

        String str1 = "int value of " + bg1 + " is " + i1;
```

```
String str2 = "int value of " + bg2 + " is " + i2;

    // print i1,i2
    System.out.println( str1 );
    System.out.println( str2 );
}
}
```

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

```
int value of 1234 is 1234
int value of 3383878445 is -911088851
```

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