

JAVA.MATH.BIGDECIMAL.BYTEVALUEEXACT METHOD

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

Copyright © tutorialspoint.com

Description

The **java.math.BigDecimal.byteValueExact** converts the BigDecimal to a byte, checking for lost information. If this BigDecimal has a nonzero fractional part or is out of the possible range for a byte result then an ArithmeticException is thrown.

Declaration

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

```
public byte byteValueExact()
```

Parameters

- NA

Return Value

This method returns the byte value of the BigDecimal Object.

Exception

- **ArithmeticException** - if BigDecimal object has a nonzero fractional part, or will not fit in a byte.

Example

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

```
package com.tutorialspoint;

import java.math.*;

public class BigDecimalDemo {

    public static void main(String[] args) {

        // create 2 BigDecimal objects
        BigDecimal bg1, bg2;

        // Create 2 byte objects
        byte i1, i2;

        // assign values to bg1 and bg2
        bg1 = new BigDecimal("-128");
        bg2 = new BigDecimal("48");

        // any BigDecimal value greater than 127 or less than -128,
        // will generate exception as it doesn't fit in byte range

        // this method also generates exception for decimal values like 12.10

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

        String str1 = "Exact byte value of " + bg1 + " is " + i1;
        String str2 = "Exact byte value of " + bg2 + " is " + i2;

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

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

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

```
Exact byte value of -128 is -128
```

```
Exact byte value of 48 is 48
```

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