

JAVA.MATH.BIGDECIMAL.DIVIDEANDREMAINDER METHOD

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

Copyright © tutorialspoint.com

Description

The **java.math.BigDecimal.divideAndRemainder** *BigDecimal* divisor returns a two-element BigDecimal array containing the result of divideToIntegralValue followed by the result of remainder on the two operands calculated with rounding according to the context settings.

If both the integer quotient and remainder are needed, this method is faster than using the divideToIntegralValue and remainder methods separately because the division need only be carried out once.

Declaration

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

```
public BigDecimal[] divideAndRemainder(BigDecimal divisor)
```

Parameters

- **divisor** - value by which this BigDecimal is to be divided, and the remainder computed

Return Value

This method returns a two element BigDecimal array: the quotient *theresultofdivideToIntegralValue* is the initial element and the remainder is the final element.

Exception

- **ArithmeticException** - if divisor == 0

Example

The following example shows the usage of math.BigDecimal.divideAndRemainder method,

```
package com.tutorialspoint;

import java.math.*;

public class BigDecimalDemo {

    public static void main(String[] args) {

        // create 2 BigDecimal objects
        BigDecimal bg1, bg2;

        bg1 = new BigDecimal("143.145");
        bg2 = new BigDecimal("10.01");

        // BigDecimal array bg stores result of bg1/bg2
        BigDecimal bg[] = bg1.divideAndRemainder(bg2);

        // print quotient and remainder
        System.out.println("Division result");
        System.out.println("Quotient is " + bg[0] );
        System.out.println("Remainder is " + bg[1] );
    }
}
```

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

```
Division result
```

Quotient is 14.0
Remainder is 3.005

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