

# JAVA.MATH.BIGDECIMAL.REMAINDER METHOD

[http://www.tutorialspoint.com/java/math/bigdecimal\\_remainder\\_mc.htm](http://www.tutorialspoint.com/java/math/bigdecimal_remainder_mc.htm)

Copyright © tutorialspoint.com

## Description

The **java.math.BigDecimal.remainder***BigDecimal divisor, MathContext mc* returns a BigDecimal whose value is *this*, with rounding according to the context settings.

The MathContext settings affect the implicit divide used to compute the remainder. The remainder computation itself is by definition exact. Therefore, the remainder may contain more than *mc.getPrecision* digits.

The remainder is given by *this.subtract(this.divideToIntegralValue(divisor, mc.multiply(divisor))*. This is not the modulo operation i.e the result can be negative.

## Declaration

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

```
public BigDecimal remainder(BigDecimal divisor, MathContext mc)
```

## Parameters

- **divisor** - value by which this BigDecimal is to be divided
- **mc** - the context to use

## Return Value

This method returns the remainder when BigDecimal object is divided by the divisor i.e, this % divisor, rounded as necessary

## Exception

- **ArithmeticException** - if divisor == 0
- **ArithmeticException** - if the result is inexact but the rounding mode is UNNECESSARY, or *mc.precision > 0* and the result of *this.divideToIntegralValue(divisor)* would require a precision of more than *mc.precision* digits.

## Example

The following example shows the usage of `math.BigDecimal.remainder` method

```
package com.tutorialspoint;

import java.math.*;

public class BigDecimalDemo {

    public static void main(String[] args) {

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

        MathContext mc = new MathContext(2); // 2 precision

        bg1 = new BigDecimal("-144.144");
        bg2 = new BigDecimal("16.12");

        // bg2 divided by bg1 using mc gives bg3 as remainder
        bg3 = bg2.remainder(bg1, mc);

        String str = "The remainder is " + bg3;
```

```
// print the value of bg3
    System.out.println( str );
}
}
```

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

```
The remainder is -15.184
Loading [Mathjax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js
```