

JAVA.MATH.BIGINTEGER.SHIFTLEFT METHOD

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

Copyright © tutorialspoint.com

Description

The **java.math.BigInteger.shiftLeft***int**n* returns a BigInteger whose value is *this* << *n*. The shift distance, *n*, may be negative, in which case this method performs a right shift. It computes floor(*this* * 2^{*n*}).

Declaration

Following is the declaration for **java.math.BigInteger.shiftLeft** method

```
public BigInteger shiftLeft(int n)
```

Parameters

- **n** - shift distance, in bits

Return Value

This method returns a BigInteger object whose value is *this* << *n* .

Exception

- **ArithmeticException** - if the shift distance is Integer.MIN_VALUE

Example

The following example shows the usage of math.BigInteger.shiftLeft method

```
package com.tutorialspoint;

import java.math.*;

public class BigIntegerDemo {

    public static void main(String[] args) {

        // create 3 BigInteger objects
        BigInteger bi1, bi2, bi3;

        bi1 = new BigInteger("10");

        // perform leftshift operation on bi1 using 2 and -2
        bi2 = bi1.shiftLeft(2);
        bi3 = bi1.shiftLeft(-2);

        String str1 = "Leftshift on " + bi1 + ", 2 times gives " +bi2;
        String str2 = "Leftshift on " + bi1 + ", -2 times gives " +bi3;

        // print bi2, bi3 values
        System.out.println( str1 );
        System.out.println( str2 );
    }
}
```

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

```
Leftshift on 10, 2 times gives 40
Leftshift on 10, -2 times gives 2
Loading [MathJax]/jax/output/HTML-CSS/jax.js
```