

JAVA.LANG.INTEGER.HIGHESTONEBIT METHOD

http://www.tutorialspoint.com/java/lang/integer_highestonebit.htm

Copyright © tutorialspoint.com

Description

The **java.lang.Integer.highestOneBit** method returns an int value with at most a single one-bit, in the position of the highest-order " *leftmost* " one-bit in the specified int value. It returns zero if the specified value has no one-bits in its two's complement binary representation, that is, if it is equal to zero.

Declaration

Following is the declaration for **java.lang.Integer.highestOneBit** method

```
public static int highestOneBit(int i)
```

Parameters

- **i** -- This is the int value.

Return Value

This method returns an int value with a single one-bit, in the position of the highest-order one-bit in the specified value, or zero if the specified value is itself equal to zero.

Exception

- **NA**

Example

The following example shows the usage of java.lang.Integer.highestOneBit method.

```
package com.tutorialspoint;

import java.lang.*;

public class IntegerDemo {

    public static void main(String[] args) {

        int i = 170;
        System.out.println("Number = " + i);

        /* returns the string representation of the unsigned integer value
        represented by the argument in binary (base 2) */
        System.out.println("Binary = " + Integer.toBinaryString(i));

        // returns the number of one-bits
        System.out.println("Number of one bits = " + Integer.bitCount(i));

        /* returns an int value with at most a single one-bit, in the position
        of the highest-order ("leftmost") one-bit in the specified int value */
        System.out.println("Highest one bit = " + Integer.highestOneBit(i));
    }
}
```

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

```
Number = 170
Binary = 10101010
Number of one bits = 4
Highest one bit = 128
```

