

JAVA.UTIL.BITSET CLASS

http://www.tutorialspoint.com/java/util/java_util_bitset.htm

Copyright © tutorialspoint.com

Introduction

The **java.util.BitSet** class implements a vector of bits that grows as needed. Following are the important points about BitSet:

- A BitSet is not safe for multithreaded use without external synchronization.
- All bits in the set initially have the value false.
- Passing a null parameter to any of the methods in a BitSet will result in a NullPointerException.

Class declaration

Following is the declaration for **java.util.BitSet** class:

```
public class BitSet
    extends Object
    implements Cloneable, Serializable
```

Class constructors

S.N.	Constructor & Description
1	BitSet This constructor creates a new bit set.
2	BitSet(int nbits) This constructor creates a bit set whose initial size is large enough to explicitly represent bits with indices in the range 0 through nbits-1.

Class methods

S.N.	Method & Description
1	<u>void and(BitSet set)</u> This method performs a logical AND of this target bit set with the argument bit set.
2	<u>void andNot(BitSet set)</u> This method clears all of the bits in this BitSet whose corresponding bit is set in the specified BitSet.
3	<u>int cardinality</u> This method returns the number of bits set to true in this BitSet.

4

[void clear](#)

This method sets all of the bits in this BitSet to false.

5

[void clearintbitIndex](#)

This method sets the bit specified by the index to false.

6

[void clearintfromIndex, intoIndex](#)

This method sets the bits from the specified fromIndex *inclusive* to the specified toIndex *exclusive* to false.

7

[Object clone](#)

This method clones this BitSet and produces a new BitSet that is equal to it.

8

[boolean equalsObjectobj](#)

This method compares this object against the specified object.

9

[void flipintbitIndex](#)

This method sets the bit at the specified index to the complement of its current value.

10

[void flipintfromIndex, intoIndex](#)

This method sets each bit from the specified fromIndex *inclusive* to the specified toIndex *exclusive* to the complement of its current value.

11

[boolean getintbitIndex](#)

This method returns the value of the bit with the specified index.

12

[BitSet getintfromIndex, intoIndex](#)

This method returns a new BitSet composed of bits from this BitSet from fromIndex *inclusive* to toIndex *exclusive*.

13

[int hashCode](#)

This method returns the value of the bit with the specified index.

14

[boolean intersectsBitSetset](#)

This method returns true if the specified BitSet has any bits set to true that are also set to true in this BitSet.

15	<u>boolean isEmpty</u> This method returns true if this BitSet contains no bits that are set to true.
16	<u>int length</u> This method returns the "logical size" of this BitSet: the index of the highest set bit in the BitSet plus one.
17	<u>int nextClearBit(int fromIndex)</u> This method returns the index of the first bit that is set to false that occurs on or after the specified starting index.
18	<u>int nextSetBit(int fromIndex)</u> This method returns the index of the first bit that is set to true that occurs on or after the specified starting index.
19	<u>void or(BitSet set)</u> This method performs a logical OR of this bit set with the bit set argument.
20	<u>void set(int bitIndex)</u> This method sets the bit at the specified index to true.
21	<u>void set(int bitIndex, boolean value)</u> This method sets the bit at the specified index to the specified value.
22	<u>void set(int fromIndex, int toIndex)</u> This method sets the bits from the specified fromIndex <i>inclusive</i> to the specified toIndex <i>exclusive</i> to true.
23	<u>void set(int fromIndex, int toIndex, boolean value)</u> This method sets the bits from the specified fromIndex <i>inclusive</i> to the specified toIndex <i>exclusive</i> to the specified value.
24	<u>int size</u> This method returns the number of bits of space actually in use by this BitSet to represent bit values.
25	<u>String toString</u> This method returns a string representation of this bit set.

[void xorBitSetset](#)

This method performs a logical XOR of this bit set with the bit set argument.

Methods inherited

This class inherits methods from the following classes:

• [java.util.Object](#)

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