

JAVA.UTIL.TREESET.HIGHER METHOD

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

Copyright © tutorialspoint.com

Description

The **higher** method is used to return the least element in this set strictly greater than the given element, or null if there is no such element.

Declaration

Following is the declaration for **java.util.TreeSet.higher** method.

```
public E higher(E e)
```

Parameters

- **e** -- This is the value to be matched.

Return Value

The method call returns the least element greater than e, or null if there is no such element.

Exception

- **ClassCastException** -- This exception is thrown if the specified element cannot be compared with the elements currently in the set.
- **NullPointerException** -- if the specified element is null and this set uses natural ordering, or its comparator does not permit null elements.

Example

The following example shows the usage of java.util.TreeSet.higher method.

```
package com.tutorialspoint;

import java.util.TreeSet;

public class TreeSetDemo {
    public static void main(String[] args) {
        // creating a TreeSet
        TreeSet<Integer>treeadd = new TreeSet<Integer>();

        // adding in the tree set
        treeadd.add(12);
        treeadd.add(11);
        treeadd.add(16);
        treeadd.add(15);

        // getting the higher value for 13
        System.out.println("Higher value of 13: "+treeadd.higher(13));
    }
}
```

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

```
Higher value of 13: 15
```

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