JAVA.UTIL.TREESET.COMPARATOR METHOD

Description

The **comparator** method returns the comparator used to order the elements in this set, or null if this set uses the natural ordering of its elements.

Declaration

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

```
public Comparator<? super E> comparator()
```

Parameters

- **NA**

Return Value

The method call returns the comparator used to order the elements in this set, or null if this set uses the natural ordering of its elements.

Exception

- **NA**

Example

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

```
package com.tutorialspoint;

import java.util.Iterator;
import java.util.TreeSet;

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

   // adding in the tree
   tree.add(12);
   tree.add(13);
   tree.add(14);
   tree.add(15);
   tree.add(16);
   tree.add(17);

   // using comparator
   treecomp = (TreeSet)tree.comparator();

   if(treecomp!=null){
   for (Integer element : treecomp)
   System.out.println(element + " ");
   }
   else{
   System.out.println("treecomp value: "+treecomp);
   System.out.println("So it is using natural ordering");
   }
   }
}
```

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

```
treecomp value: null
So it is using natural ordering
```