

JAVA.UTIL.COLLECTIONS.SORT METHOD

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

Copyright © tutorialspoint.com

Description

The **sort***List* *< T >* , *Comparator* *< ?superT >* method is used to sort the specified list according to the order induced by the specified comparator.

Declaration

Following is the declaration for **java.util.Collections.sort** method.

```
public static <T> void sort(List<T> list, Comparator<? super T> c)
```

Parameters

- **list** -- This is the list to be sorted.
- **c** -- This is the comparator to determine the order of the list.

Return Value

- NA

Exception

- **ClassCastException** -- Throws if the list contains elements that are not mutually comparable using the specified comparator.
- **UnsupportedOperationException** -- Throws if the specified list's list-iterator does not support the set operation.

Example

The following example shows the usage of **java.util.Collections.sort**

```
package com.tutorialspoint;

import java.util.*;

public class CollectionsDemo {
    public static void main(String args[]) {
        // create linked list object
        LinkedList<Integer> list = new LinkedList<Integer>();

        // populate the list
        list.add(-28);
        list.add(20);
        list.add(-12);
        list.add(8);

        // sort the list
        Collections.sort(list, null);

        System.out.println("List sorted in natural order: ");
        for(int i : list){
            System.out.println(i+ " ");
        }
    }
}
```

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

List sorted in natural order:

-28

-12

8

20

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