

JAVA.UTIL.COLLECTIONS.MIN METHOD

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

Copyright © tutorialspoint.com

Description

The **minCollection** *< ?extendsT > , Comparator < ?superT >* method is used to return the minimum element of the given collection, according to the order induced by the specified comparator.

Declaration

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

```
public static <T> T min(Collection<? extends T> coll, Comparator<? super T> comp)
```

Parameters

- **coll** -- The collection whose minimum element is to be determined.
- **comp** -- The comparator with which to determine the minimum element.

Return Value

The method call returns the minimum element of the given collection, according to the specified comparator.

Exception

- **ClassCastException** -- This is thrown if the collection contains elements that are not mutually comparable using the specified comparator.
- **NoSuchElementException** -- This is thrown if the collection is empty.

Example

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

```
package com.tutorialspoint;

import java.util.*;

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

        // populate the list
        list.add(-18);
        list.add(40);
        list.add(-45);
        list.add(12);

        // comparing using natural ordering
        System.out.println("Min val: " + Collections.min(list, null));
    }
}
```

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

```
Min val: -45
```

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