

# JAVA.UTIL.COLLECTIONS.BINARYSEARCH METHOD

[http://www.tutorialspoint.com/java/util/collections\\_binarysearch\\_comparator.htm](http://www.tutorialspoint.com/java/util/collections_binarysearch_comparator.htm)

Copyright © tutorialspoint.com

## Description

The **binarySearch***List* < ?*extends* *T* > , *T*, *Comparator* < ?*super* *T* > method is used to search the specified list for the specified object using the binary search algorithm. The list must be sorted into ascending order according to the specified comparator.

## Declaration

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

```
public static <T> int binarySearch(List<? extends T> list, T key, Comparator<? super T> c)
```

## Parameters

- **list** -- This is the list to be searched.
- **key** -- This is the key to be searched for.
- **c** -- This is the comparator by which the list is ordered. A null value indicates that the elements' natural ordering should be used.

## Return Value

The method call returns the index of the search key, if it is contained in the list.

## Exception

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

## Example

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

```
package com.tutorialspoint;

import java.util.*;

public class CollectionsDemo {
    public static void main(String args[]) {
        // create arraylist
        ArrayList<String> arlst=new ArrayList<String>();

        // populate the list
        arlst.add("TP");
        arlst.add("PROVIDES");
        arlst.add("QUALITY");
        arlst.add("TUTORIALS");

        // search for key 'TUTORIALS' with natural ordering
        int index=Collections.binarySearch(arlst, "TUTORIALS", null);

        System.out.println("'TUTORIALS' is available at index: "+index);
    }
}
```

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

```
'TUTORIALS' is available at index: 3
```

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