JAVA.UTIL.ARRAYS.BINARYSEARCH METHOD

Description

The **java.util.Arrays.binarySearch***float*[]*a, intfromIndex, inttoIndex, floatkey* method searches a range of the specified array of floats for the specified value using the binary search algorithm. The range must be sorted before making this call.If it is not sorted, the results are undefined.

Declaration

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

```
public static int binarySearch(float[] a, int fromIndex, int toIndex, float key)
```

Parameters

- **a** -- This is the array to be searched.
- **fromIndex** -- This is the index of the first element *inclusive* to be searched.
- **toIndex** -- This is the index of the last element *exclusive* to be searched.
- **key** -- This is the value to be searched for.

Return Value

This method returns index of the search key, if it is contained in the array, else it returns −(*insertionpoint* - 1). The insertion point is the point at which the key would be inserted into the array; the index of the first element in the range greater than the key, or toIndex if all elements in the range are less than the specified key.

Exception

- **IllegalArgumentException** -- if fromIndex > toIndex
- **ArrayIndexOutOfBoundsException** -- if fromIndex < 0 or toIndex > a.length

Example

The following example shows the usage of java.util.Arrays.binarySearch method.

```
package com.tutorialspoint;

import java.util.Arrays;

public class ArrayDemo {

   public static void main(String[] args) {

   // initializing unsorted float array
   float floatArr[] = {5.2f,46.1f,42.9f,22.3f};

   // sorting array
   Arrays.sort(floatArr);

   // let us print all the elements available in list
   System.out.println("The sorted float array is:");
   for (float number : floatArr) {
   System.out.println("Number = " + number);
   }

   // entering the value to be searched
   float searchVal = 22.3f;
```

```
   // entering range of index
   int retVal = Arrays.binarySearch(floatArr,1,4,searchVal);

   System.out.println("The index of element 22.3 is : " + retVal);
   }
}
```

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

```
The sorted float array is:
Number = 5.2
Number = 22.3
Number = 42.9
Number = 46.1
The index of element 2.3 is : 1
```