457. Classical Binary Search

* [Description](http://lintcode.com/en/problem/classical-binary-search/#description)
* [Notes](http://lintcode.com/en/problem/classical-binary-search/#note)
* [Testcase](http://lintcode.com/en/problem/classical-binary-search/#testcase)
* [Judge](http://lintcode.com/en/problem/classical-binary-search/#judge)

Find any position of a target number in a sorted array. Return -1 if target does not exist.

Have you met this question in a real interview?

Yes

**Example**

Given [1, 2, 2, 4, 5, 5].

For target = 2, return 1 or 2.

For target = 5, return 4 or 5.

For target = 6, return -1.

<http://lintcode.com/en/problem/classical-binary-search/#>

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication5;

import java.util.\*;

/\*\*

\*

\* @author Usuario

\*/

public class JavaApplication5 {

public static int findPosition(int[] nums, int target) {

// write your code here

int ind = Arrays.binarySearch(nums, target);

return ind < -1 ? -1 : ind ;

}

public static void main(String[] args) {

// TODO code application logic here

int[] a = {1, 2, 2, 4, 5, 5};

System.out.println( findPosition(a, 5));

}

}