Collections binarySearch(..) method

binarySearch is a static method in Collections class. This method is used to search an element in List.

It leverages the random access nature of list and searches for element in O(log n) time, provided that the list is sorted.

If the specified list in parameter does not support random access then it takes O(n) time for search.

**public** **static** List<String> populateLanguages() {

List<String> languages = **new** ArrayList<String>();

languages.add("Java");

languages.add("JavaScript");

languages.add("C#");

languages.add("Python");

**return** languages;

}

**public** **static** **void** binarySearchDemo() {

List<String> languages = *populateLanguages*();

System.***out***.println("List is " + languages);

Collections.*sort*(languages);

System.***out***.println("Sorted List is " + languages);

String key = "Python";

**int** index = Collections.*binarySearch*(languages, key);

System.***out***.println(key + " is at index " + index);

key = "C";

index = Collections.*binarySearch*(languages, key);

System.***out***.println(key + " is at index " + index);

}

Output is

List is [Java, JavaScript, C#, Python]

Sorted List is [C#, Java, JavaScript, Python]

Python is at index 3

C is at index -1

If the implementation used is ArrayList<E> then the random access is utilized and hence binary search algorithm works in O(log n) time.

Things change if the implementation is LinkedList<E>. LinkedList<E> does not support random access hence the time required for searching is O(n).