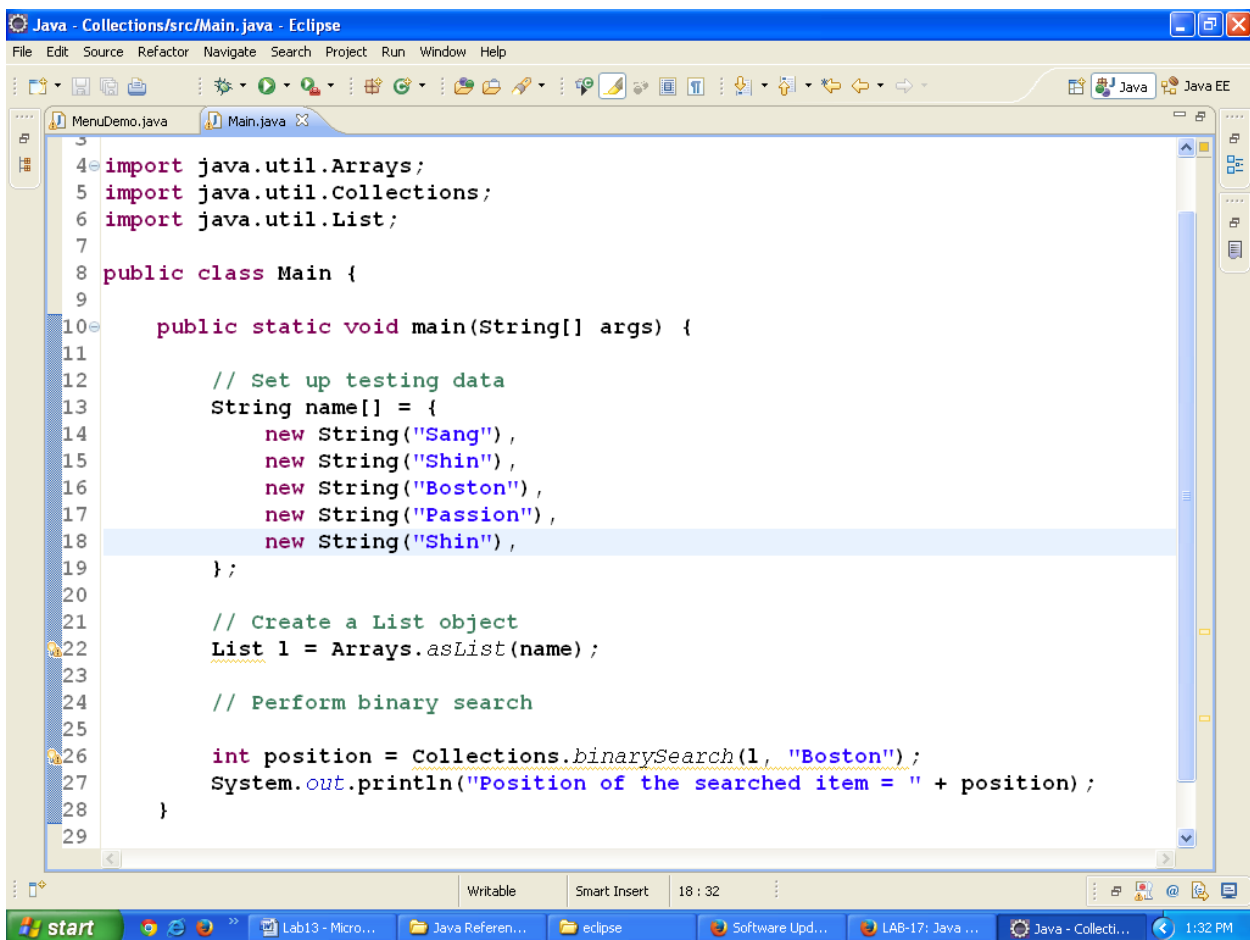


Lab-13 Assignments

Collections & Generics

Demo 1:List



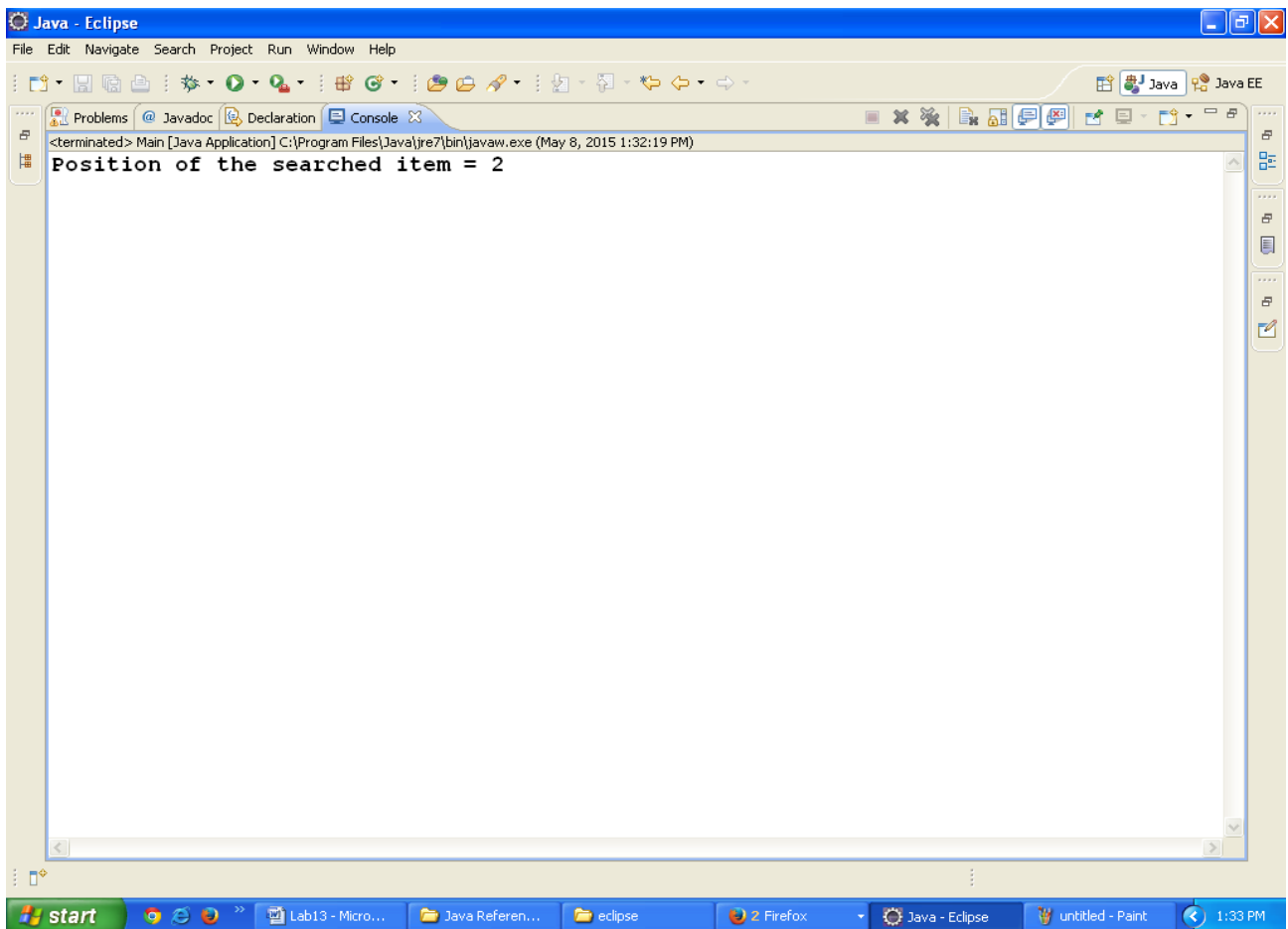
```
Java - Collections/src/Main.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

MenuDemo.java Main.java
3
4 import java.util.Arrays;
5 import java.util.Collections;
6 import java.util.List;
7
8 public class Main {
9
10     public static void main(String[] args) {
11
12         // Set up testing data
13         String name[] = {
14             new String("Sang"),
15             new String("Shin"),
16             new String("Boston"),
17             new String("Passion"),
18             new String("Shin"),
19         };
20
21         // Create a List object
22         List l = Arrays.asList(name);
23
24         // Perform binary search
25
26         int position = Collections.binarySearch(l, "Boston");
27         System.out.println("Position of the searched item = " + position);
28     }
29
```

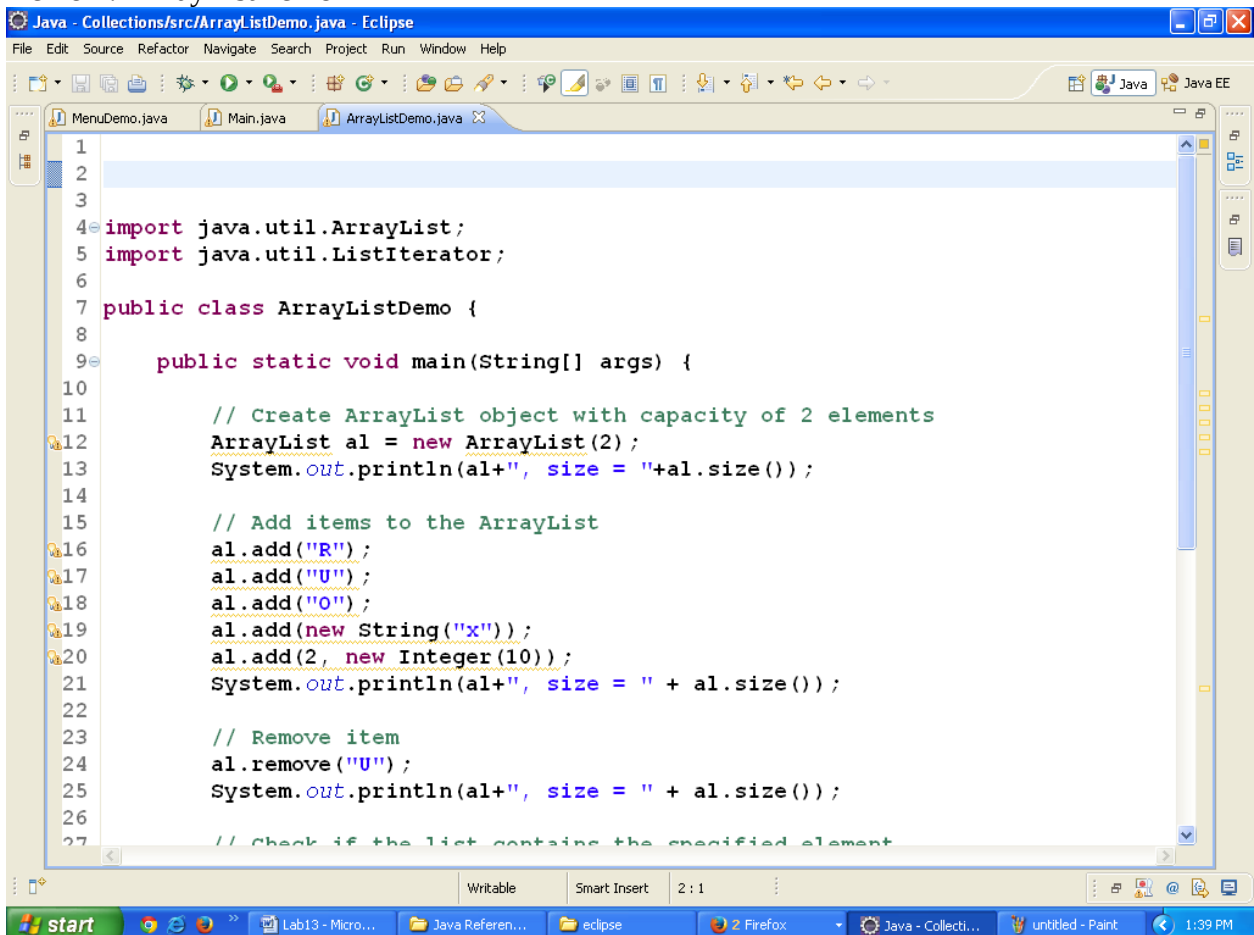
Writable Smart Insert 18 : 32

start Lab13 - Micro... Java Referen... eclipse Software Upd... LAB-17: Java ... Java - Collecti... 1:32 PM

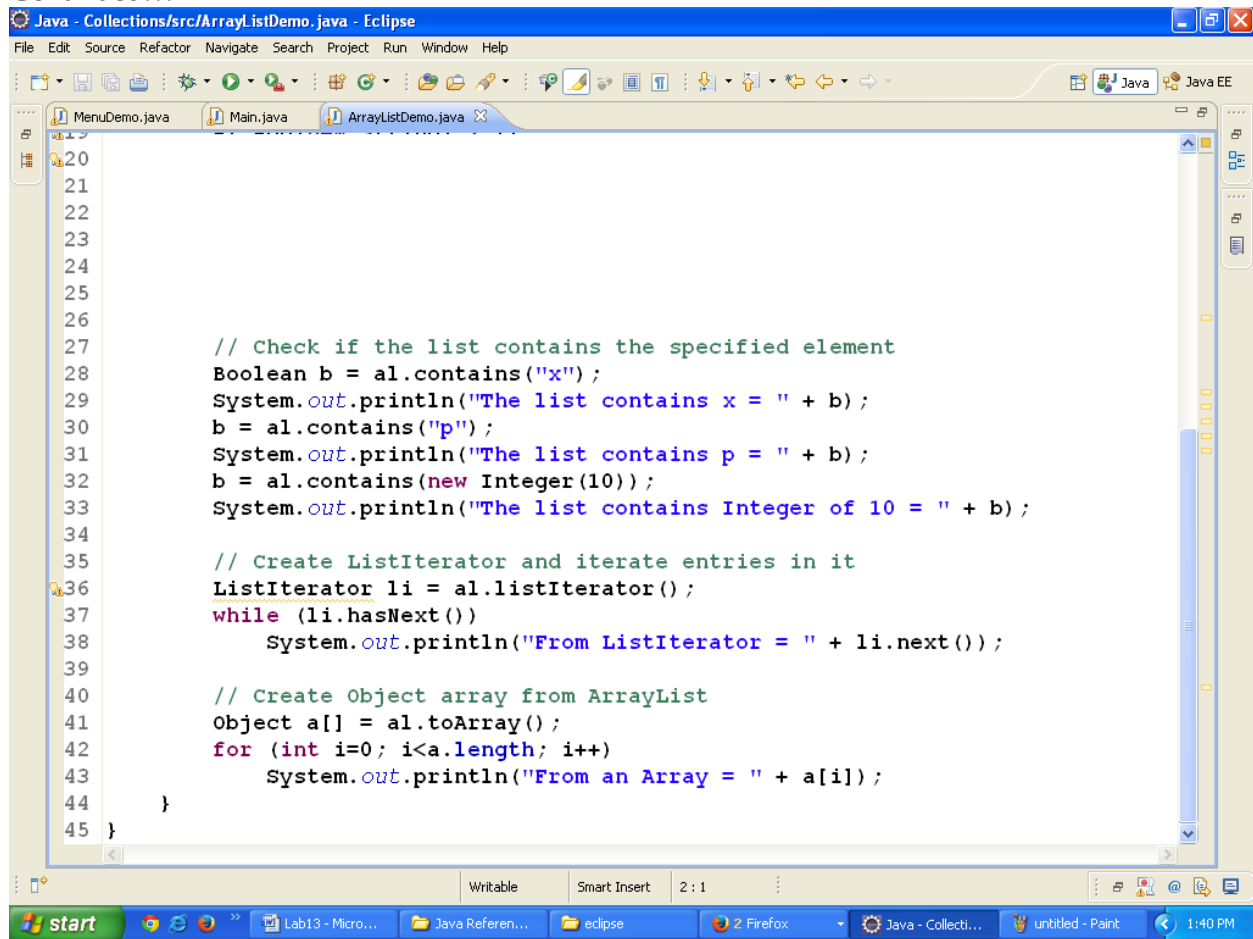
Run as -> Java Application



Demo 2: ArrayListDemo

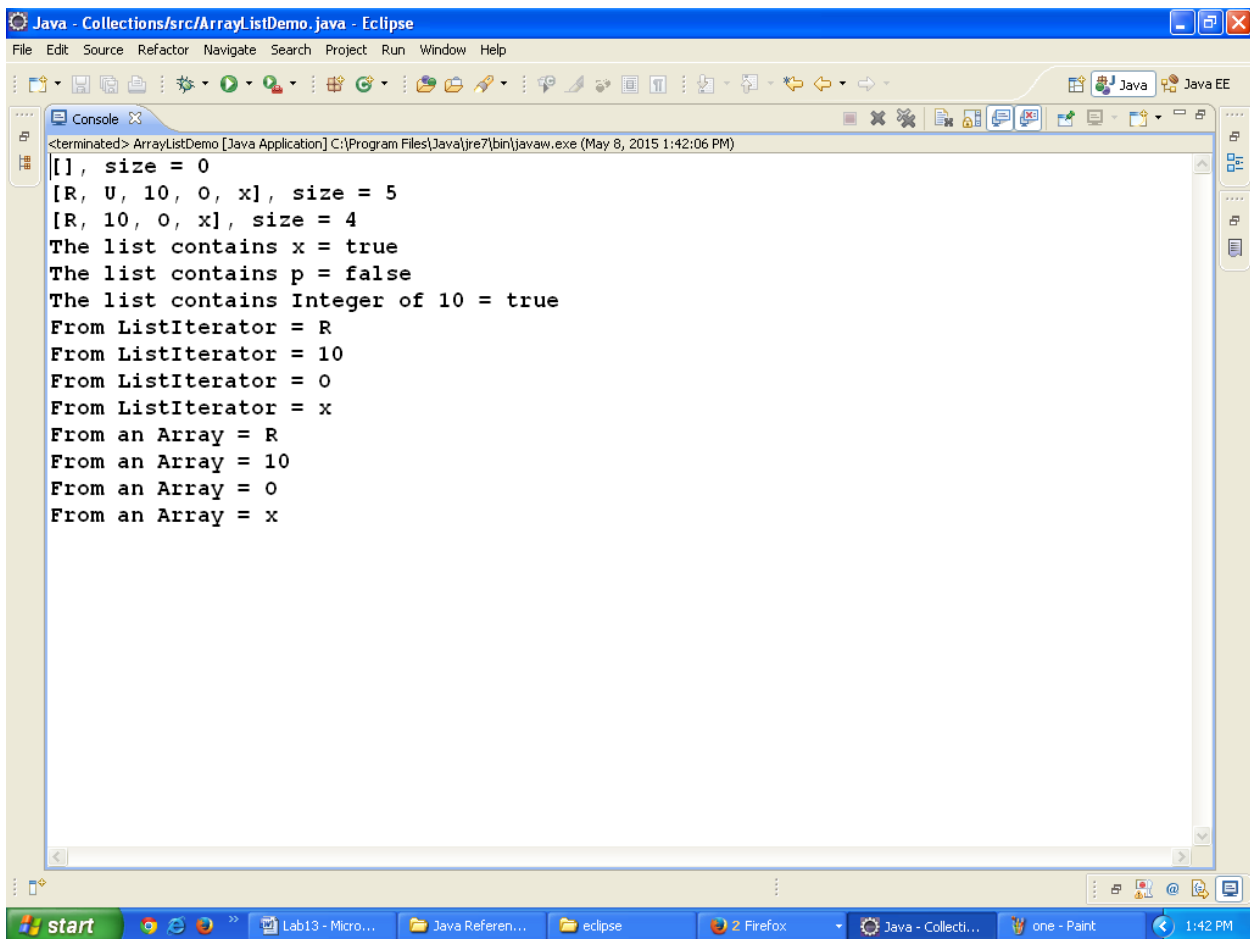


Continued...



```
20
21
22
23
24
25
26
27 // Check if the list contains the specified element
28 Boolean b = al.contains("x");
29 System.out.println("The list contains x = " + b);
30 b = al.contains("p");
31 System.out.println("The list contains p = " + b);
32 b = al.contains(new Integer(10));
33 System.out.println("The list contains Integer of 10 = " + b);
34
35 // Create ListIterator and iterate entries in it
36 ListIterator li = al.listIterator();
37 while (li.hasNext())
38     System.out.println("From ListIterator = " + li.next());
39
40 // Create Object array from ArrayList
41 Object a[] = al.toArray();
42 for (int i=0; i<a.length; i++)
43     System.out.println("From an Array = " + a[i]);
44 }
45 }
```

Run as -> Java Application



```
<terminated> ArrayListDemo [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (May 8, 2015 1:42:06 PM)
[], size = 0
[R, U, 10, 0, x], size = 5
[R, 10, 0, x], size = 4
The list contains x = true
The list contains p = false
The list contains Integer of 10 = true
From ListIterator = R
From ListIterator = 10
From ListIterator = 0
From ListIterator = x
From an Array = R
From an Array = 10
From an Array = 0
From an Array = x
```

Assignments To Solve :

1. Create a Product class with Product Id & Product Name. Write a program to accept information of 10 products and store that in Hash Table. Search a particular product in the Hash Table. Remove a particular product id and product name from the Hash Table.

The product list is as follows:

Product Id	Product Name
P001	Maruti800
P002	MarutiZen
P003	MarutiEsteem

- 2.WAP to generate the following GUI

Employee Management

ID

Name

Basic

Add **List All**

Search

Handle event for Add, List All and Search Buttons

Event for Add:

Retrieve the elements from the textfields, create an Object of Employee and add the Objects to Vector or ArrayList.

Event for List All

Display All Objects in TextArea

Event for Search

Iterate through the collection to find employee object for a particular id entered and display the details of that Object in TextArea..Or else print message Object not found for given id..

The screenshot shows a Java Swing window titled "PhoneBook" with a standard Windows-style title bar (minimize, maximize, close buttons). The window has a light gray background and contains the following elements:

- Form Fields:**
 - ID:** A text field containing the value "1002".
 - Name:** A text field containing the value "Sumit".
 - Basic:** A text field containing the value "85000".
- Buttons:**
 - Add:** A button located below the form fields.
 - List All:** A button located to the right of the "Add" button.
- Text Area:**
 - A multi-line text area with a scroll bar, containing the following text:

```
Collection :  
[ID =1001 Name=Harsh Basic=9000  
, ID =1002 Name=Sumit Basic=8500  
]
```
- Search Section:**
 - Search:** A button located at the bottom left.
 - Search Field:** A text field containing the value "1001" located to the right of the "Search" button.

3. Create Phone book having user interface like

1. Add new phone book entry
2. Search Phone Number
3. Quit.

Option 1 it allows add name and Phone no.

Option 2 it has to take name as input from the user based on that it should return phone

No

Option 3: will terminate the program

Use HashMap to store phone book entries.