

## \* Set A

- 1] Accept 'n' integers from the user and store them in a collection. Display them in the sorted order. The collection should not accept duplicate elements. (Use a suitable collection). Search for a particular element using predefined search method in the collection framework.

→ Ass1SetA1.java.

```
import java.util.*;  
import java.io.*;
```

```
class Ass1SetA1  
{
```

```
    public static void main (String args[])  
        throws Exception  
    {
```

```
        int n, i;
```

```
        String srch;
```

```
        BufferedReader br = new BufferedReader (new  
            InputStreamReader (System.in));
```

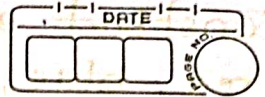
```
        TreeSet ts = new TreeSet ();
```

```
        System.out.println ("Enter how many Elements  
        you want? = ");
```

```
        n = Integer.parseInt (br.readLine ());
```

EXPERIMENT :

No.



```
for (i=0; i<n; i++)
```

```
ts.add (br.readLine ());
```

```
System.out.println (ts);
```

```
System.out.println ("Enter Elements to  
Search = ");
```

```
srch = br.readLine ();
```

```
if (ts.contains (srch))
```

```
System.out.println ("In It" + srch + " Found");
```

```
else
```

```
System.out.println ("In It" + srch + " NOT Found");
```

```
}
```

```
}
```



2) Construct a linked list containing names of colors: red, blue, yellow and orange. Then extend your program to do the following:

- i) Display the contents of the list using an Iterator.
- ii) Display the contents of the list in reverse order using a ListIterator.
- iii) Create another list containing pink and green. Insert the elements of this list between blue and yellow.

→ import java.util.\*;

```
class Ass1SetA2
```

```
{
```

```
    public static void main (String args[])
```

```
    {
```

```
        LinkedList colors = new LinkedList();
```

```
        colors.add ("red");
```

```
        colors.add ("blue");
```

```
        colors.add ("yellow");
```

```
        colors.add ("orange");
```

```
        System.out.println (colors);
```

```
        System.out.println ("\\n\\n Contents of List  
using an Iterator :");
```

```
        Iterator i = colors.iterator ();
```

```
        while (i.hasNext ())
```

```
            System.out.println (i.next());
```

```
        System.out.println ("\\n\\n Reverse Contents  
of list using ListIterator :");
```

```
        ListIterator li = colors.listIterator (colors.size());  
        // Iterate in reverse.
```

EXPERIMENT :

No.

DATE

```
while (li.hasPrevious())  
    System.out.println(li.previous());
```

```
LinkedList colors2 = new LinkedList();  
colors2.add("pink");  
colors2.add("green");
```

```
colors.addAll(C2, colors2);
```

```
System.out.println("In\nAfter adding  
another List:");
```

```
System.out.println(colors);
```



3) Create a Hash table containing student name and percentage. Display the details of the hash table. Also search for a specific student and display percentage of that student.

→ import java.io.\*;  
import java.util.\*;

class AsstSetA3

{

public static void main (String arg[])  
throws IOException

{

HashTable HT= new HashTable ();

String str;

BufferedReader br= new BufferedReader  
(new InputStreamReader (System.in));

HT.put ("Nagesh", new Double (77.88));

HT.put ("Amrut", new Double (57.87));

HT.put ("Pallavi", new Double (88.67));

HT.put ("Ramesh", new Double (80.58));

System.out.println (HT+"\n\n");

System.out.println ("Enter student name  
to search :");

str = br.readLine ();

if (HT.containsKey (str))

System.out.println ("Name : "  
+str+ "Parc : "+HT.get (str));

EXPERIMENT :

No.

DATE

--	--	--	--

else

system.out.println("Not student information  
Not found");

}

}

'c'

Gautam  
08/10/2024

Teacher's Sign.: \_\_\_\_\_