Page No.
Date

Name: Swara
Roll No: 9532210755
Class: CL

Lab Assignment 5B

Problem statement: Study of collection framework in Java

Objective: To study collections in Java

Theory:

ArrayList:

```
import java.util.ArrayList;

public class ArrayList {
    public static void main(String[] args) {
        ArrayList<String> fruits = new ArrayList<>();
            fruits.add("apple");
            fruits.add("banana");
            fruits.add("orange");
            fruits.add("berry");
        System.out.println("fruits: " + fruits);
        fruits.remove("banana");
    System.out.println("fruits after removing banana: " +
                                                fruits);
    System.out.println("size of fruits Arraylist: " + fruits.size
                                                ());
}
```

Teacher's Sign

2) Vector:

```
import java.util.Vector;
public class Vector2
    public class static void main (String [] args) {
        Vector <String> colors = new Vector <> ();
        colors.add ("red");
        colors.add ("green");
        colors.add ("blue");
        colors.add ("yellow");

    System.out.println ("Colors:  " + colors);
    colors.remove ("yellow");
    System.out.println (" colors after removing yellow: "+co
    System.out.println ("size of colors vector: "+ colors.size()
    }
}
```

3) Stack:

```
import java.util.Stack;
public class stack {
    public class static void main (String args []) {
        Stack <Integer> numbers = new Stack<>();
        numbers.push(1);
        numbers.push(2);
        numbers.push(3);
        numbers.push(4);
System.out.println ("Numbers  " + numbers);
int poppedNumber = numbers.pop();
System.out.println ("Popped No.:  " + poppedNumber);
System.out.println("size of Number stack:  "+ numbers.size());
}
}
```

Teacher's Sign.

Hashmap:

```
import
public
    public
    Hash
    stu
    St
    St
    St
System.ou
    int
System.out
System.out
    }
}
```

5) Linked List:

```
import j
public
    public
    Link
    co
System.
    ou
    Systen
System.ou
    }
}
```

```
main (String [] args) {
= new Vector <>();
);
);
");
");

" + colors );

fter removing yellow: "+c
s vector: " + colors.size()

ers );

" + poppedNumber);
er stack: " + numbers.size());
```

Teacher's Sign.:

Hashmap:

```
import java.util.HashMap;
public class HashMap {
    public static void main (String args[]) {
    Hashmap <String, Integer> studentgrades = new HashMap<>();

        studentgrades.put ("John", 90);
        Studentgrades.put ("Mary", 80);
        student grades.put ("Smith", 70);
        Studentgrades.put ("Patrick", 95);
System.out.println ("Student grades:  " + studentgrades);
    int  johngrade = studentgrade.get ("John");
System.out.println ("John's grade : " + johngrade);
System.out.println ("size of student Grades HashMap: " +
                                    studentgrade.size());

}
}
```

Linked List:

```
import java.util.LinkedList;
public class LinkedList {
    public static void main (String args[]) {
    LinkedList <String> countries = new LinkedList <>();
        countries.add ("India");
        countries.add ("USA");
        countries.add ("Russia");
System.out.println ("Countries:  " + countries);
    countries.remove ("USA");
System.out.println ("Countries after Removing USA:  "+
                                    countries);

System.out.println ("Size of Countries LinkedList:  " + countries.size
                                    ());

}
```

Teacher's Sign.:

Conclusion: Thus, we have successfully studied usage of collections in Java

FAQ's

Ans 1) The collection is a framework that provides an architecture to store, manipulate, and retrieve a group of objects. It includes interfaces, implementations, and algorithms.
Core Interfaces in collection framework are:
1) Collection
2) List
3) Set
4) Queue
5) Map

Ans 2) Array and collection are both used to store and manipulate groups of objects in Java, but they have some important differences:
1. Size: size of an array is fixed and cannot be changed once created, whereas size of collection can be changed dynamically.
2. Type: An array can store primitive type and objects of specific type, whereas collection can store objects of any
3. Initialization: Arrays must be initialized with a fixed size and default values for each element, whereas collection can be initialized with no size or with an initial element.
4. Accessing Elements: Arrays are accessed using index that corresponds to the position of an element in array, whereas collections provide iterators and methods to access elements.
5. Sorting: Arrays have built-in sorting mechanism, whereas collection do not.

Teacher's Sign.:

studied usage of

es an architecture to
f objects. It includes
s.
t are :

store and manipul
ve some important

cannot be changed
llection can be chan

and objects of
store objects of any
with a fixed size
whereas collection can
n initial element
d using index that
ent in array,
and methods to

g mechanism, whereas

Teacher's Sign:

Memory Management: Arrays consume memory for all elements, even if some elements are not used, whereas collection only consume memory for the elements in use.

Example:

```
import java.util.*;
public class cpllectionsDemo{
  public static void main(String[] args){
    Map m1 = new Hashmap();
      m1.put("Zara","8");
      m1.put("Mahnaz","31");
      m1.put("Ayan","12");
      m1.put("Daisy","14");
      System.out.println();
      System.out.println("Map Elements:");
      System.out.println("\t" + m1);}
    }
```

Output: Map Elements : {Daisy = 14, Ayan = 12, Zara = 8, Mahnaz = 31}

Linked List Example:

```
import java.util.*;
public class TestCollection7{
  public static void main(String args[])
  {
    LinkedList<String> al = new LinkedList<String>();
      al.add("Ravi");
      al.add("Vijay");
      al.add("Ravi");
      al.add("Ajay");
```

Iterator <String> itr = al.iterator();
while (itr.hasNext()){
System.out.println(itr.next());
3
3
3

Output:- Ravi Vijay Ravi Ajay.