Collections Exercise

1. Write Java code to define List . Insert 5 floating point numbers in List, and using an iterator, find the sum of the numbers in List.

Sol – Program File Folder name – program1

```
import java.util.*;

class Ques1
{
	public static void main(String args[])
	{
		List<Float> al = new ArrayList<Float>();
		al.add(10.4f);
		al.add(20.5f);
		al.add(30.1f);
		al.add(40.7f);
		al.add(50.4f);
		al.add(60.3f);

		System.out.println("Floating values in the list:");
		float sum = 0f;
		Iterator i = al.iterator();
		while(i.hasNext())
		{
			float x = (float)i.next();
			System.out.println(x);
			sum = sum + x;
		}

		System.out.println("Total Sum: "+ sum);
	}
```

```
ttn@ttn:program1 $ javac Ques1.java
ttn@ttn:program1 $ java Ques1
Floating values in the list:
10.4
20.5
30.1
40.7
50.4
60.3
Total Sum: 212.40001
```

2.Write a method that takes a string and returns the number of unique characters in the string.

Sol – Program File Folder name – program2

```
import java.util.*;


public class Ques2 {
	public static int uniqueChar(String s1){
		StringBuffer sb = new StringBuffer(s1);
		for (int i = 0; i < sb.length(); i++) {
			int count = 0;
			for (int j = i + 1; j < sb.length(); j++) {
				if (sb.charAt(i) == sb.charAt(j)) {
					sb.deleteCharAt(j);
					j--;
					count++;
				}
			}
			if (count >= 1) {
				sb.deleteCharAt(i);
				i--;
			}
		}
		int val = sb.length();
	return val;
	}

	public static void main(String[] args){
		Scanner sc = new Scanner(System.in);
```

```
        System.out.println("Enter String:");
        String str = sc.nextLine();
        int v = uniqueChar(str);
        System.out.println("Number of unique characters in the string: "+ v);
    }
}
```

```
ttn@ttn:program2 $ vim Ques2.java
ttn@ttn:program2 $ javac Ques2.java
ttn@ttn:program2 $ java Ques2
Enter String:
naveen
Number of unique characters in the string: 2
```

3.Write a method that takes a string and print the number of occurrence of each character characters in the string.

Sol – Program File Folder name – program3

```
import java.util.*;
class Ques3 {
    static final int MAX_CHAR = 256;

    static void occuringChar(String str)
    {
        int count[] = new int[MAX_CHAR];
        int len = str.length();
        for (int i = 0; i < len; i++)
            count[str.charAt(i)]++;

        char ch[] = new char[str.length()];
        for (int i = 0; i < len; i++) {
            ch[i] = str.charAt(i);
            int find = 0;
            for (int j = 0; j <= i; j++) {
                if (str.charAt(i) == ch[j])
```

```
                    find++;
                }

                if (find == 1)
                    System.out.println("Number of Occurrence of "+
str.charAt(i)+ " is:" + count[str.charAt(i)]);
            }
        }

        public static void main(String[] args)
        {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter String:");
            String str = sc.nextLine();
            occuringChar(str);
        }
}
```

```
ttn@ttn:collections $ javac Ques3.java
ttn@ttn:collections $ java Ques3
Enter String:
helloworld
Number of Occurrence of h is:1
Number of Occurrence of e is:1
Number of Occurrence of l is:3
Number of Occurrence of o is:2
Number of Occurrence of w is:1
Number of Occurrence of r is:1
Number of Occurrence of d is:1
```

4. Write a program to sort Employee objects based on highest salary using Comparator. Employee class{ Double Age; Double Salary; String Name

Sol – Program File Folder name – program4

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
```

```
import java.util.Comparator;

class Employee {
    public int empAge;
    public String empName;
    public double empSalary;

    public Employee(int empAge, String empName, double empSalary) {
        this.empAge = empAge;
        this.empName = empName;
        this.empSalary = empSalary;
    }

}

class EmployeeSortBySalary implements Comparator < Employee > {
    public int compare(Employee emp1, Employee emp2) {
        int value = 0;
        if (emp1.empSalary > emp2.empSalary)
            value = 1;
        else if (emp1.empSalary < emp2.empSalary)
            value = -1;
        else if (emp1.empSalary == emp2.empSalary)
            value = 0;

        return value;
    }
}

public class ques4 {
```

```
    public static void main(String[] args) {
        List <Employee> employees = new ArrayList <Employee> ();
        employees.add(new Employee(22, "Dinesh", 18000));
        employees.add(new Employee(24, "Pankaj", 19000));
        employees.add(new Employee(28, "Mayur", 14000));
        employees.add(new Employee(32, "Pravin", 22000));

        System.out.println("Sort By Employee Salary");
        Collections.sort(employees, new EmployeeSortBySalary());

        printEmployees(employees);
    }

    public static void printEmployees(List <Employee> employees) {
        for (Employee e: employees) {
            System.out.println("Age:" + e.empAge + "  Name:" + e.empName + "
Salary:" + e.empSalary);
        }
    }

}
```

```
ttn@ttn:collections $ javac ques4.java
ttn@ttn:collections $ java ques4
Sort By Employee Salary
Age:28  Name:Mayur  Salary:14000.0
Age:22  Name:Dinesh  Salary:18000.0
Age:24  Name:Pankaj  Salary:19000.0
Age:32  Name:Pravin  Salary:22000.0
ttn@ttn:collections $
```

5. Write a program to sort the Student objects based on Score , if the score are same then sort on First Name . Class Student{ String Name; Double Score; Double Age

Sol – Program File Folder name – program5

```
import java.util.*;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import java.util.Comparator;

class Student{
   public double age;
   public String fname;
   public double score;
   public Student(double age, String fname, double score) {
      this.age = age;
      this.fname = fname;
      this.score = score;
   }
}

class StudentSortByScore implements Comparator < Student > {
    public int compare(Student s1, Student s2) {
        if(s2.score>s1.score){
                    return 1;
                }else if(s2.score<s1.score){
                    return -1;
                }
                return s1.fname.compareTo(s2.fname);
    }
}

public class Ques5
{
        public static void main(String[] args){
```

```
        List<Student> studentList = new ArrayList<Student>();


        studentList.add(new Student(22, "Naveen", 190.5));
    studentList.add(new Student(22, "Dinesh", 180.4));
    studentList.add(new Student(24, "Pankaj", 190.5));
    studentList.add(new Student(28, "Rahul", 140.0));
    studentList.add(new Student(32, "Ankit", 175.3));
        System.out.println("Sort the Student objects based on Score , if the score are same then sort on First Name");


        Collections.sort(studentList, new StudentSortByScore());


        printEmployees(studentList);
    }


    public static void printEmployees(List <Student> studentList) {
    for (Student s: studentList) {
                System.out.println("Age:" + s.age + "  Name:" + s.fname + " Salary:" + s.score);
    }


    }
}
```

```
ttn@ttn:program5 $ javac Ques5.java
ttn@ttn:program5 $ java Ques5
Sort the Student objects based on Score , if the score are same then sort on First Name
Age:22.0  Name:Naveen  Salary:190.5
Age:24.0  Name:Pankaj  Salary:190.5
Age:22.0  Name:Dinesh  Salary:180.4
Age:32.0  Name:Ankit  Salary:175.3
Age:28.0  Name:Rahul  Salary:140.0
ttn@ttn:program5 $
```

6.Print the elements of an array in the decreasing frequency if 2 numbers have same frequency then print the one which came first.

Sol – Program File Folder name – program6

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

```
public class Ques6 {
    public static void main(String[] args)
    {
        int[] array = { 4, 4, 2, 2, 2, 2, 3, 3, 1, 1, 7, 7, 5,10 };

        Map<Integer, Integer> map = new HashMap<>();
        List<Integer> outputArray = new ArrayList<>();

        for (int current : array) {
            int count = map.getOrDefault(current, 0);
            map.put(current, count + 1);
            outputArray.add(current);
        }

        SortComparator comp = new SortComparator(map);
        Collections.sort(outputArray, comp);
        for (Integer i : outputArray) {
            System.out.print(i + " ");
        }
         System.out.print("\n");
    }
}


class SortComparator implements Comparator<Integer> {
    private final Map<Integer, Integer> freqMap;

    SortComparator(Map<Integer, Integer> tFreqMap)
    {
        this.freqMap = tFreqMap;
```

```
    }

    public int compare(Integer k1, Integer k2)
    {
        int freqCompare = freqMap.get(k2).compareTo(freqMap.get(k1));
        int valueCompare = k1.compareTo(k2);

         // If frequency is equal, then just compare by value, otherwise compare
by the frequency
        if (freqCompare == 0)
            return valueCompare;
        else
            return freqCompare;
    }
}
```

```
ttn@ttn:collections $ vim Ques6.java
ttn@ttn:collections $ javac Ques6.java
ttn@ttn:collections $ java Ques6
2 2 2 2 1 1 3 3 4 4 7 7 5 10
ttn@ttn:collections $
```

7.Design a Data Structure SpecialStack that supports all the stack operations like push(), pop(), isEmpty(), isFull() and an additional operation getMin() which should return minimum element from the SpecialStack. (Expected complexity O(1))

Sol – Program File Folder name – program7

```
import java.util.*;

class SpecialStack
{
    Stack<Integer> s;
    Integer minEle;

    SpecialStack() {
```

```
        s = new Stack<Integer>();
}

void getMin()
{
    if (s.isEmpty())
        System.out.println("Stack is empty");

    else
        System.out.println("Minimum Element in the " +
                           " stack is: " + minEle);
}

void peek()
{
    if (s.isEmpty())
    {
        System.out.println("Stack is empty ");
        return;
    }

    Integer t = s.peek();
    System.out.print("Top Most Element is: ");

    if (t < minEle)
        System.out.println(minEle);
    else
        System.out.println(t);
}

// Removes the top element
```

```
void pop()
{
    if (s.isEmpty())
    {
        System.out.println("Stack is empty");
        return;
    }

    System.out.print("Top Most Element Removed: ");
    Integer t = s.pop();

    if (t < minEle)
    {
        System.out.println(minEle);
        minEle = 2*minEle - t;
    }

    else
        System.out.println(t);
}

void push(Integer x)
{
    if (s.isEmpty())
    {
        minEle = x;
        s.push(x);
        System.out.println("Number Inserted: " + x);
        return;
    }
```

```
        if (x < minEle)
        {
            s.push(2*x - minEle);
            minEle = x;
        }

        else
            s.push(x);

        System.out.println("Number Inserted: " + x);
    }
};

public class Ques7
{
    public static void main(String[] args)
    {
        SpecialStack s = new SpecialStack();
        s.push(21);
        s.push(50);
         s.push(15);
         s.push(34);
        s.getMin();
        s.push(20);
        s.push(4);
        s.getMin();
        s.pop();
        s.getMin();
        s.pop();
        s.peek();
    }
```

}

```
ttn@ttn:program7 $ java Ques7
Number Inserted: 21
Number Inserted: 50
Number Inserted: 15
Number Inserted: 34
Minimum Element in the  stack is: 15
Number Inserted: 20
Number Inserted: 4
Minimum Element in the  stack is: 4
Top Most Element Removed: 4
Minimum Element in the  stack is: 15
Top Most Element Removed: 20
Top Most Element is: 34
ttn@ttn:program7 $
```

8. Write a program to format date as example "21-March-2016"

Sol – Program File Folder name – program8

```
import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.Locale;


public class Ques8 {

	public static void main(String[] args) {

		Date date = new Date();

		SimpleDateFormat formatter = new
SimpleDateFormat("MM/dd/yyyy");

		String strDate = formatter.format(date);

		System.out.println("Date Format with MM/dd/yyyy : "+strDate);


		formatter = new SimpleDateFormat("dd-MMMM-yyyy");

	strDate = formatter.format(date);

	System.out.println("Date Format with dd-MMMM-yyyy : "+strDate);


		formatter = new SimpleDateFormat("dd-M-yyyy hh:mm:ss");

		strDate = formatter.format(date);
```

```
            System.out.println("Date Format with dd-M-yyyy hh:mm:ss :
"+strDate);


            formatter = new SimpleDateFormat("dd MMMM yyyy");
            strDate = formatter.format(date);
            System.out.println("Date Format with dd MMMM yyyy : "+strDate);


            formatter = new SimpleDateFormat("dd MMMM yyyy zzzz");
            strDate = formatter.format(date);
            System.out.println("Date Format with dd MMMM yyyy zzzz :
"+strDate);


            formatter = new SimpleDateFormat("E, dd MMM yyyy HH:mm:ss
z");
            strDate = formatter.format(date);
            System.out.println("Date Format with E, dd MMM yyyy HH:mm:ss z :
"+strDate);
      }
}
```

```
ttn@ttn:collections $ vim Ques8.java
ttn@ttn:collections $ javac Ques8.java
ttn@ttn:collections $ java Ques8
Date Format with MM/dd/yyyy : 02/17/2021
Date Format with dd-MMMM-yyyy : 17-February-2021
Date Format with dd-M-yyyy hh:mm:ss : 17-2-2021 05:18:05
Date Format with dd MMMM yyyy : 17 February 2021
Date Format with dd MMMM yyyy zzzz : 17 February 2021 India Standard Time
Date Format with E, dd MMM yyyy HH:mm:ss z : Wed, 17 Feb 2021 17:18:05 IST
```

9.Write a program to display times in different country format.

Sol – Program File Folder name – program9

```
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;

public class Ques9{

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

```
        Date today = new Date();
        DateFormat df = new SimpleDateFormat("dd-MM-yy HH:mm:SS z");

         df.setTimeZone(TimeZone.getTimeZone("Asia/Kolkata"));
        String IST = df.format(today);
        System.out.println("Date in Indian Timezone (IST) : " + IST);

        df.setTimeZone(TimeZone.getTimeZone("America/Los_Angeles"));
        String PST = df.format(today);
        System.out.println("Date in PST Timezone : " + PST);

         df.setTimeZone(TimeZone.getTimeZone("Australia/Sydney"));
        String AET = df.format(today);
        System.out.println("Date in AET Timezone : " + AET);

         df.setTimeZone(TimeZone.getTimeZone("Africa/Cairo"));
        String ART = df.format(today);
        System.out.println("Date in ART Timezone : " + ART);

         df.setTimeZone(TimeZone.getTimeZone("Europe/Paris"));
        String ECT = df.format(today);
        System.out.println("Date in ECT Timezone : " + ECT);

         df.setTimeZone(TimeZone.getTimeZone("Asia/Tokyo"));
        String JST = df.format(today);
        System.out.println("Date in JST Timezone : " + JST);


    }

}
```

```
ttn@ttn:program9 $ java Ques9
Date in Indian Timezone (IST) : 17-02-21 17:36:728 IST
Date in PST Timezone : 17-02-21 04:06:728 GMT-08:00
Date in AET Timezone : 17-02-21 23:06:728 AEDT
Date in ART Timezone : 17-02-21 14:06:728 EET
Date in ECT Timezone : 17-02-21 13:06:728 CET
Date in JST Timezone : 17-02-21 21:06:728 JST
ttn@ttn:program9 $
```