COLLECTIONS AND DATES

-YUKTI
SHARMA

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.**

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

public class Ques1 {
    static  float sum=0.0F;
    public static void main(String[] args) {

        List<Float> list= new ArrayList<Float>();
        list.add(2.4F);
        list.add(1.34F);
        list.add(93.31F);
        list.add(33.22F);
        list.add(1.2003F);

        Iterator<Float> listitr= list.iterator();

        System.out.println("elements of list are-:\n");
        while (listitr.hasNext())
        {
            System.out.println(listitr.next());
        }

        Iterator<Float> list1= list.iterator();
        while (list1.hasNext())
        {
            float n= list1.next();
            sum += n;
        }
        System.out.println("========\n"+sum+ "  <== sum of list");
    }
}
```

```
Ques1
/home/yukti/.sdkman/candidates/java/8.0.202-amzn/
elements of list are-:

2.4
1.34
93.31
33.22
1.2003
========
131.47029  <== sum of list

Process finished with exit code 0
```

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

```java
import java.util.HashSet;
import java.util.Iterator;
import java.util.Scanner;

public class Ques2 {

    public static void main(String[] args) {
        System.out.println("enter a string");
        Scanner sc= new Scanner(System.in);
        String input= sc.next();
           int size= countUniqueChars(input);
         System.out.println("\n\nNumber of unique characters in "+input+" are ==> "+size);
    }

    public static int countUniqueChars(String input) {
        HashSet<Character> hash = new HashSet<>();
        input = input.toUpperCase();
        for (int i = 0; i < input.length(); i++)
            hash.add(input.charAt(i));

        System.out.println("unique characters are- \n");

        Iterator iterator = hash.iterator();
        while (iterator.hasNext())
            System.out.print(iterator.next()+" \t");

              return hash.size();
    }


}
```

```
Ques2
/home/yukti/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
enter a string
hellohowareyouherethere
unique characters are-

A   R   T   E   U   W   H   Y   L   O

Number of unique characters in hellohowareyouherethere are ==> 10

Process finished with exit code 0
```

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

```java
import java.util.HashMap;
import java.util.Scanner;

public class Ques3 {
    static void CharacterCount(String input){
        HashMap<Character,Integer> charCount= new HashMap<>();
        char[] array= input.toCharArray();
        for (char i: array){
            if(charCount.containsKey(i))
            {
                charCount.put(i,charCount.get(i)+1);
            }
            else
            {
                charCount.put(i,1);
            }
        }
        System.out.println(charCount);
    }

    public static void main(String[] args)
    {
        Scanner sc= new Scanner(System.in);
        System.out.println("enter string");
        String entered= sc.next();
        CharacterCount(entered);
    }
}
```

```
/home/yukti/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
enter string
yuktisharmayuksyukti
{a=2, r=1, s=2, t=2, u=3, h=1, y=3, i=2, k=3, m=1}

Process finished with exit code 0
```

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

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

public class Ques4 {
    public static void main(String[] args) {
        Employee e1 = new Employee( age: 23,  salary: 99999,  name: "yukti");
        Employee e2 = new Employee( age: 21,  salary: 13399,  name: "priya");
        Employee e3 = new Employee( age: 20,  salary: 92299,  name: "seema");
        Employee e4 = new Employee( age: 30,  salary: 102299,  name: "Reema");
        List<Employee> EmployeeList = new ArrayList<Employee>();
        EmployeeList.add(e1);
        EmployeeList.add(e2);
        EmployeeList.add(e3);
        EmployeeList.add(e4);

        for (Employee emp:EmployeeList)
            System.out.println("Name is- "+emp.name+"\tAge is- "+ emp.age+"\tSalary is- "+emp.salary);


        Collections.sort(EmployeeList,new Sorting());
        System.out.println("\nSorted Employees on basis of highest salary are-\n");

        for (Employee emp:EmployeeList)
            System.out.println("Name is- "+emp.name+"\tAge is- "+ emp.age+"\tSalary is- "+emp.salary);
    }
}
```

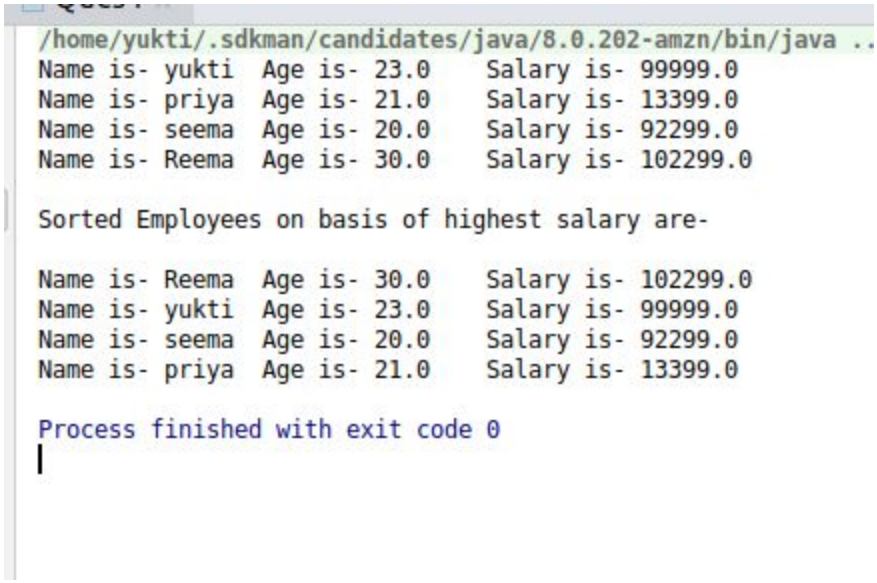
```java
class Employee{
   double age;
   double salary;
    String name;

    public Employee(double age, double salary, String name) {
        this.age = age;
        this.salary = salary;
        this.name = name;
    }

    @Override
    public String toString() {
        return this.name+" "+this.age+" "+this.salary;
    }
}

class Sorting implements Comparator<Employee>{

    @Override
    public int compare(Employee o1, Employee o2) {
        return (int)(o2.salary-o1.salary);
    }
}
```



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**

```java
class Student{
    String Name;
    Double Score;
    Double Age;


    public Student(String Name, Double score, Double age) {
        this.Name = Name;
        Score = score;
        Age = age;
    }

    @Override
    public String toString() {
        return this.Name+" "+this.Age+" "+this.Score;
    }
}

class SortingMarks implements Comparator<Student>{

    @Override
    public int compare(Student o1, Student o2) {
        if(o1.Score.equals(o2.Score))
            return o1.Name.compareTo(o2.Name);
        return (int)(o2.Score-o1.Score);
    }
}
```

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

public class Ques5 {
    public static void main(String[] args) {
        Student e1 = new Student( Name: "yukti",  score: 60.0,  age: 20.0);
        Student e2 = new Student( Name: "Shivam",  score: 100.0,  age: 21.0);
        Student e3 = new Student( Name: "Rahul",  score: 70.0,  age: 19.0);
        Student e4 = new Student( Name: "Rishabh",  score: 78.5,  age: 18.0);
        Student e5 = new Student( Name: "Anky",  score: 78.5,  age: 21.0);

        List<Student> StudentList = new ArrayList<Student>();
        StudentList.add(e1);
        StudentList.add(e2);
        StudentList.add(e3);
        StudentList.add(e4);
        StudentList.add(e5);


        for (Student stu:StudentList)
            System.out.println("Name is- "+stu.Name+"\tAge is- "+ stu.Age+"\tMarks are- "+stu.Score);


        Collections.sort(StudentList,new SortingMarks());
        System.out.println("\nSorted Students on basis of marks are-\n");

        for (Student stu:StudentList)
            System.out.println("Name is- "+stu.Name+"\tAge is- "+ stu.Age+"\tMarks are- "+stu.Score);
    }
}
```

```
/home/yukti/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
Name is- yukti  Age is- 20.0    Marks are- 60.0
Name is- Shivam Age is- 21.0    Marks are- 100.0
Name is- Rahul  Age is- 19.0    Marks are- 70.0
Name is- Rishabh    Age is- 18.0    Marks are- 78.5
Name is- Anky   Age is- 21.0    Marks are- 78.5

Sorted Students on basis of marks are-

Name is- Shivam Age is- 21.0    Marks are- 100.0
Name is- Anky   Age is- 21.0    Marks are- 78.5
Name is- Rishabh    Age is- 18.0    Marks are- 78.5
Name is- Rahul  Age is- 19.0    Marks are- 70.0
Name is- yukti  Age is- 20.0    Marks are- 60.0

Process finished with exit code 0
```

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

```
    public static void main(String[] args)
    {
        Scanner sc= new Scanner(System.in);
        System.out.println("enter size of array");
        int size= sc.nextInt();
        String[] entered= new String[size];
        System.out.println("enter the array");
        for(int i=0;i<size;i++) {
            entered[i] = sc.next();
        }
        CharacterCount(entered);
    }
}
```

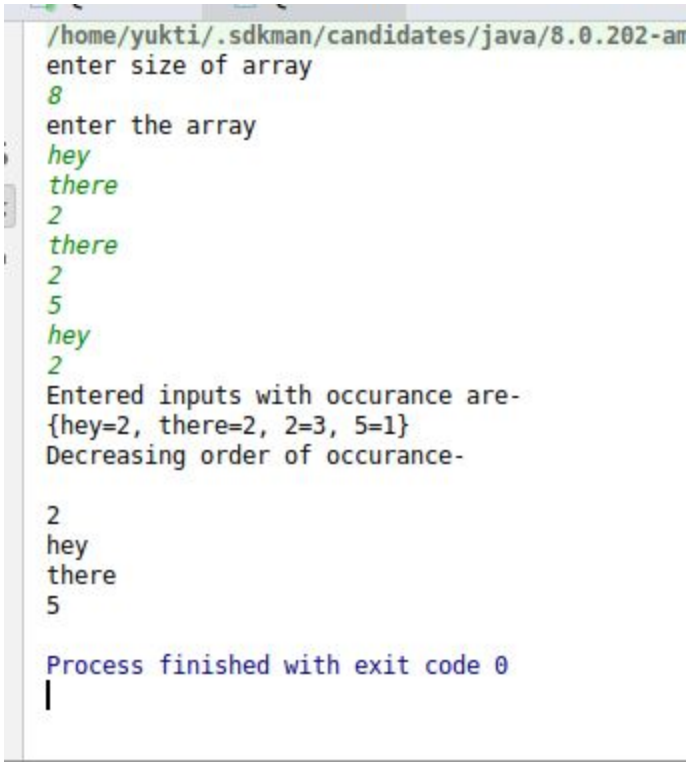
```
import java.util.*;

public class Ques6 {
    static void CharacterCount(String[] input){
        LinkedHashMap<String,Integer> charCount= new LinkedHashMap<>();

        for (String i: input){
            if(charCount.containsKey(i))
            {
                charCount.put(i,charCount.get(i)+1);
            }
            else
            {
                charCount.put(i,1);
            }
        }
        System.out.println("Entered inputs with occurance are- \n"+charCount);
        System.out.println("Decreasing order of occurance-");

        List<Map.Entry<String,Integer>> list = new ArrayList<>(charCount.entrySet());
        Collections.sort(list, new Comparator<Map.Entry<String, Integer>>() {
            @Override
            public int compare(Map.Entry<String, Integer> o1, Map.Entry<String, Integer> o2) {
                return o2.getValue()-o1.getValue();
            }
        });
        for(Map.Entry<String,Integer> e : list){

            System.out.println(e.getKey());
        }
    }
```



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))**

```java
import java.util.Stack;

public class Ques7  extends Stack<Integer> {
    private Stack<Integer>minStack=new Stack<Integer>();
    public void push(int element)
    {
        if(isEmpty())
        {
            super.push(element);
            minStack.push(element);
            System.out.println("pushed- "+element);
        }
        else
        {   super.push(element);
            int minEle=minStack.peek();
            if(element<minEle)
                minStack.push(element);
            else
                minStack.push(minEle);
            System.out.println("pushed- "+element);
        }
    }
    public Integer pop()
    {
        Integer poppedElement=null;
        if(!isEmpty())
            poppedElement=super.pop();
        if(minStack.isEmpty()==false)
            minStack.pop();
        return poppedElement;
    }
    public Integer getMin()
```

```
    public Integer getMin()
    {
        if(isEmpty())
            return null;
        return minStack.peek();
    }

    public static void main(String[] args) {
        Ques7 o= new Ques7();
        o.push( element: 4);
        o.push( element: 3);
        o.push( element: 8);
        o.push( element: 1);
        o.push( element: 6);
        System.out.println("minimum is "+ o.getMin());
        System.out.println("Popped "+ o.pop());
        System.out.println("Popped "+ o.pop());
        System.out.println("minimum is "+ o.getMin());
        System.out.println("Popped "+ o.pop());
        System.out.println("Popped "+ o.pop());
        System.out.println("minimum is "+ o.getMin());
        System.out.println("Popped "+ o.pop());
        System.out.println("Popped "+ o.pop());
    }
}
```

```
Ques6 × Ques7 ×
/home/yukti/.sdkman/candidates/java/8.0.202-amzn/bin
pushed- 4
pushed- 3
pushed- 8
pushed- 1
pushed- 6
minimum is 1
Popped 6
Popped 1
minimum is 3
Popped 8
Popped 3
minimum is 4
Popped 4
Popped null

Process finished with exit code 0
```

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

```java
import java.text.SimpleDateFormat;
import java.util.Calendar;

public class Ques8 {
    public static void main(String[] args) {
        Calendar date= Calendar.getInstance();
        //date.set(2016,2,21);
        date.set( year: 2016,Calendar.MARCH, date: 21);
        SimpleDateFormat date2 = new SimpleDateFormat( pattern: "dd-MMMM-yyyy");
        System.out.println(date2.format(date.getTime()));


    }
}
```

```
/home/yukti/.sdkman/candidates/java/8.0.202-
21-March-2016

Process finished with exit code 0
```

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

```java
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( pattern: "dd-MMMM-yyyy 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("UK"));
        String UK = df.format(today);
        System.out.println("Date in UK Timezone : " +UK);


    }

}
```

```
/home/yukti/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
Date in Indian Timezone (IST) : 25-February-2019 00:11:313 IST
Date in PST Timezone : 24-February-2019 10:41:313 PST
Date in UK Timezone : 24-February-2019 18:41:313 GMT

Process finished with exit code 0
```