q-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.

```
package assignment4;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.ListIterator;
public class as4q1
{
    public static void main(String[] args) {
        float sum=0;
        ArrayList<Float> pb=new ArrayList<Float>();
        pb.add(12.12f);
        pb.add(4.5f);
        pb.add(6.4f);
        pb.add(8.7f);
        pb.add(3.1f);
        Iterator itr=pb.iterator();
        while(itr.hasNext())
        {
            float f=(float)itr.next();
            sum=sum+f;
        }
        System.out.println("Sum of values is: "+sum);
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=37219:/snap/int
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.as4q1
Sum of values is: 34.819996

Process finished with exit code 0
```

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

```
package assignment4;
import java.util.HashSet;
import java.util.Scanner;
class unique{
 public void cal()
    {
        System.out.println("Enter  a string");
        Scanner scan=new Scanner(System.in);
        String a=scan.nextLine();
        String c=a.toLowerCase();
        char [] b=c.toCharArray();
        HashSet <Character> h=new HashSet <Character>();
        for(int i=0;i<c.length();i++)
        {
            h.add(b[i]);
        }
        System.out.println("Unique characters are: "+h);
    }
}
public class as4q2 {
    public static void main(String[] args) {
       unique q=new unique();
       q.cal();
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=42213:/snap/intellij-idea-community/208
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.as4q2
Enter  a string
Siddharth
Unique characters are: [a, r, s, d, t, h, i]
```

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

```
package assignment4;
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;
class Ques3 {
    public void CountOccurrence(String userString) {
        char[] charArr = userString.toCharArray();
        Map<Character, Integer> countMap = new HashMap<Character, Integer>();
        for (Character c :
                charArr) {
            if (countMap.containsKey(c)) {
                countMap.put(c, countMap.get(c) + 1);
            } else
                countMap.put(c, 1);
        }
        for (Map.Entry entry :
                countMap.entrySet()) {
            System.out.println(entry.getKey() + " is Occurring :" + entry.getValue()+"
Times");
        }
    }
    public static void main(String[] args) {
        Ques3 object = new Ques3();
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string :");
        String userString = sc.nextLine();
        object.CountOccurrence(userString.toLowerCase());
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=41909:/snap/intellij-idea-community/208/bin -Dfile
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.Ques3
Enter the string :
Siddharth
a is Occurring :1 Times
r is Occurring :1 Times
s is Occurring :1 Times
d is Occurring :2 Times
t is Occurring :1 Times
h is Occurring :2 Times
i is Occurring :1 Times

Process finished with exit code 0
```

q-4 Write a program to sort HashMap by value.

```
package assignment4;
import java.util.*;
import java.lang.*;
class Ques4 {
    public static HashMap<String, Integer> sortByValue(HashMap<String, Integer> hm) {
        // Create a list from elements of HashMap
        List<Map.Entry<String, Integer>> list =
                new LinkedList<Map.Entry<String, Integer>>(hm.entrySet());
        // Sort the list
        Collections.sort(list, new Comparator<Map.Entry<String, Integer>>() {
            public int compare(Map.Entry<String, Integer> o1,
                               Map.Entry<String, Integer> o2) {
```

```
                return (o1.getValue()).compareTo(o2.getValue());
            }
        });
        // put data from sorted list to hashmap
        HashMap<String, Integer> temp = new LinkedHashMap<String, Integer>();
        for (Map.Entry<String, Integer> aa : list) {
            temp.put(aa.getKey(), aa.getValue());
        }
        return temp;
    }
    public static void main(String[] args) {
        HashMap<String, Integer> hm = new HashMap<String, Integer>();
        hm.put("Siddharth", 95);
        hm.put("Komal", 85);
        hm.put("Aayushi", 91);
        hm.put("Shivani", 95);
        hm.put("Aastha", 79);
        hm.put("Shobhit", 80);
        Map<String, Integer> hm1 = sortByValue(hm);
        // print the sorted hashmap
        for (Map.Entry<String, Integer> en : hm1.entrySet()) {
            System.out.println("Key = " + en.getKey() +
                    ", Value = " + en.getValue());
        }
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=41041:/snap/intellij-idea-community/208/bin -Dfile
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.Ques4
Key = Aastha, Value = 79
Key = Shobhit, Value = 80
Key = Komal, Value = 85
Key = Aayushi, Value = 91
Key = Siddharth, Value = 95
Key = Shivani, Value = 95

Process finished with exit code 0
```

q-5 Write a program to sort Employee objects based on highest salary using Comparator. Employee class{ Double Age; Double Salary; String Name

```
package assignment4;
import java.util.*;
class Employee
{
String name;
double age;
double salary;
Employee(String name,double age,double salary)
{
    this.name=name;
    this.age=age;
    this.salary=salary;
}
  public String toString()
    {
        return "Name "+name+" Age "+age+" Salary "+salary;
    }
}
class myComparator implements Comparator
{
    public int compare(Object o1,Object o2) {
        Employee e1=(Employee)o1;
        Employee e2=(Employee)o2;
        Double sal1= e1.salary;
        Double sal2=e2.salary;
        return sal2.compareTo(sal1);
```

```
    }
}
public class as4q5 {
    public static void main(String[] args) {
        Employee e1=new Employee("Siddharth",24,40000);
        Employee e2=new Employee("Shyam",28,35000);
        Employee e3=new Employee("Ram",32,50000);
        Employee e4=new Employee("Rahul",35,60000);
        TreeSet t=new TreeSet(new myComparator());
        t.add(e1);
        t.add(e2);
        t.add(e3);
        t.add(e4);
        System.out.println(t);
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=35941:/snap/intellij-idea-community/208/bin -Dfile
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.as4q5
[Name Rahul Age 35.0 Salary 60000.0, Name Ram Age 32.0 Salary 50000.0, Name Siddharth Age 24.0 Salary 40000.0, Name Shyam Age 28.0 Salary 35000.0]
```

q-6 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

```
package assignment4;
import java.util.*;
class Student
{
    String name;
    double score;
    double age;
    Student(String name,double score,double age)
    {
        this.name=name;
        this.score=score;
        this.age=age;
    }
    public String toString()
    {
        return "Name "+name+" Score "+score+" Age "+age;
    }
}
class myComparator2 implements Comparator
{
    public int compare(Object o1,Object o2) {
        Student e1=(Student)o1;
        Student e2=(Student)o2;
        Double sal1=e1.score;
        Double sal2=e2.score;
        int compareScore = sal1.compareTo(sal2);
            return sal2.compareTo(sal1);
    }
}
public class as4q6 {
    public static void main(String[] args) {
        Student e1=new Student("Siddharth",80,23);
        Student e2=new Student("Shyam",75,22);
        Student e3=new Student("Ram",82,24);
        Student e4=new Student("Rahul",78,25);
        TreeSet t=new TreeSet(new myComparator2());
        t.add(e1);
```

```
        t.add(e2);
        t.add(e3);
        t.add(e4);
        System.out.println(t);
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=37219:/snap/int
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.as4q1
Sum of values is: 34.819996

Process finished with exit code 0
```

q-7 Print the elements of an array in the decreasing frequency if 2 numbers have same frequency then print the one which came first.

```
package assignment4;
import java.util.*;
public class as2q7 {
    public static void main(String[] args) {
        int [] inputArray ={1,6,3,2,5,2,3,1,1,2,2,3,4,5,6,7,8};
        Map<Integer,Integer> linkedHashMap = new LinkedHashMap<>();
        int[] arr = inputArray;
        for(int i: arr){
            if(linkedHashMap.containsKey(i)){
                linkedHashMap.put(i,linkedHashMap.get(i)+1);
            }else
                linkedHashMap.put(i,1);
        }
        List<ValueFrequencyPair> list = new LinkedList<>();
        for(Map.Entry<Integer,Integer> e:linkedHashMap.entrySet()){
            list.add(new ValueFrequencyPair(e.getKey(),e.getValue()));
        }
        Collections.sort(list, new SortFrequency());
        for(ValueFrequencyPair v: list){
            System.out.println(v.i+"  "+v.value);
        }
    }
}
class ValueFrequencyPair{
    int i;
    Integer value;
    public ValueFrequencyPair(Integer i, Integer value) {
        this.i = i;
        this.value = value;
    }
}
class SortFrequency implements Comparator<ValueFrequencyPair>{
    @Override
    public int compare(ValueFrequencyPair v1, ValueFrequencyPair v2) {
        if(v1.value == v2.value){
            return 0;
        }else if(v1.value >v2.value)
            return -1;
        else
            return 1;
    }
}
```

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=36303:/snap/intellij-idea-commu
  .encoding=UTF-8 -classpath /home/siddharth/IdeaProjects/jvm_assignment/out/production/jvm_assignment assignment4.as2q7
2 4
1 3
3 3
6 2
5 2
4 1
7 1
8 1
```

q-8 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))

```
package assignment4;
import java.util.Stack;
class myStack {
    Stack<Integer> stack;
    Integer minElement;
    public myStack() {
        stack = new Stack<Integer>();
    }
    public void peek() {
        if (stack.isEmpty()) {
            System.out.println("Stack is Empty ....");
        }
        Integer t = stack.peek();
        if (t < minElement) {
            System.out.println("Top most element is :" + minElement);
        } else
            System.out.println("Top most element is :" + t);
    }
    public void push(Integer i) {
        if (stack.isEmpty()) {
            minElement = i;
            stack.push(i);
            System.out.println("Element inserted :" + i);
            return;
        }
        if (i < minElement) {
            stack.push(2 * i - minElement);
            minElement = i;
        } else
            stack.push(i);
        System.out.println("Element inserted :" + i);
    }
    public void pop() {
        if (stack.isEmpty()) {
            System.out.println("Stack is empty, cannot pop an element");
            return;
        }
        Integer t = stack.pop();
        if (t < minElement) {
            System.out.println("element removed :" + minElement);
            minElement = 2 * minElement - t;
        } else
            System.out.println("element removed :" + t);
    }
    public void getMinElement() {
        if (stack.isEmpty()) {
            System.out.println("stack is empty..");
        } else
            System.out.println("min element is :" + minElement);
    }
}
```

```java
public class as4q8 {
    public static void main(String[] args) {
        myStack stack = new myStack();
        stack.getMinElement();
        stack.pop();
        stack.push(3);
        stack.push(4);
        stack.push(2);
        stack.push(5);
        stack.push(1);
        stack.push(8);
        stack.getMinElement();
        stack.peek();
        stack.pop();
        stack.getMinElement();
        stack.pop();
        stack.getMinElement();
    }
}
```

```
stack is empty..
Stack is empty, cannot pop an element
Element inserted :3
Element inserted :4
Element inserted :2
Element inserted :5
Element inserted :1
Element inserted :8
min element is :1
Top most element is :8
element removed :8
min element is :1
element removed :1
min element is :2
```