NAME: VENNELA G

REG NO: 20BDS0146

SUBJECT: JAVA PROGRAMMING

LAB SLOT: L13+14

ASSESSMENT NO: 3

1. Write a program to demonstrate the knowledge of students in working with Java collection framework. Eg., Assume only a maximum of 3 courses can be registered by a student for week end semester classes. Create a hashmap ‘h1’ with ‘n’ key-value pairs where keys are the names of students and values are the courses registered by them. Create another hashmap ‘h2’ with ‘m’key-value pairs where keys are the names of courses offered for B.Tech-IT and values are the names of faculty handling the courses. Write appropriate code to - Add or remove a student from h1 - Iterate over the maps and display the key-value pairs stored in them - Given a student name, fetch the names of all those who teach him/her.

   Eg:, if the elements of h1 are

Stud name	Courses registered
A	Python, maths, c
B	c, c++
C	C++, physics,chemistry

   And if the elements of h2 are

Course name	Faculty
Python	111
Maths	222
C	333
C++	444
Physics	555
Chemistry	666
Digital electronics	777

For the student “B”, faculty should be displayed as 333 and 444.

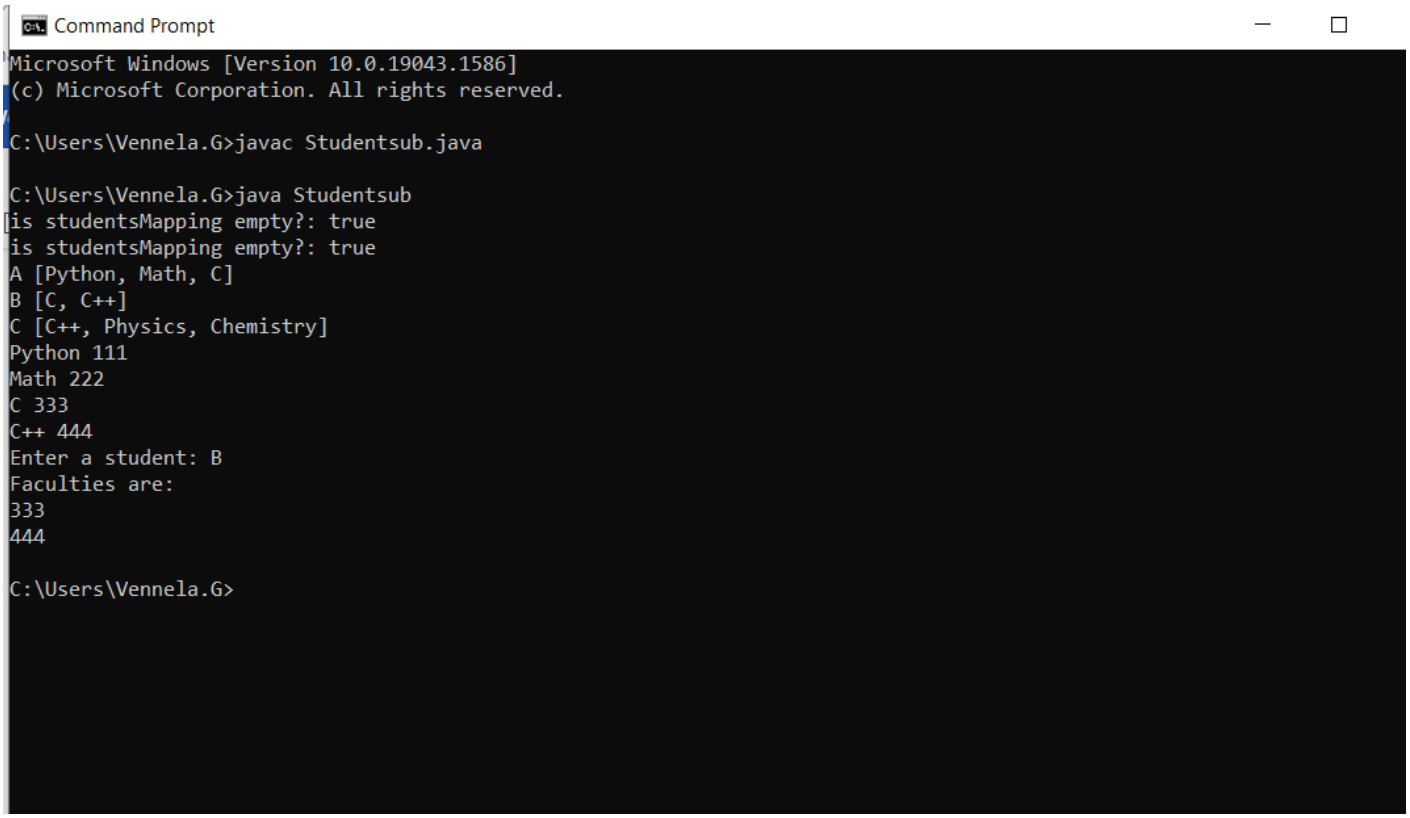
CODE:

```
import java.util.*;


public class Studentsub {
    public static void main(String[] args) {
        HashMap<String, List<String>> h1 = new LinkedHashMap<>();
        HashMap<String, String> h2 = new LinkedHashMap<>();

        System.out.println("is studentsMapping empty?: " + h1.isEmpty());
        System.out.println("is studentsMapping empty?: " + h2.isEmpty());

       List<String> subjects = Arrays.asList("Python", "Math", "C");
```

```
h1.put("A",subjects);
subjects = Arrays.asList("C","C++");
h1.put("B",subjects);
subjects = Arrays.asList("C++","Physics","Chemistry");
h1.put("C",subjects);


h2.put("Python","111");
h2.put("Math","222");
h2.put("C","333");
h2.put("C++","444");


for(Map.Entry m:h1.entrySet()){
   System.out.println(m.getKey()+" "+m.getValue());
}

for(Map.Entry m:h2.entrySet()){
   System.out.println(m.getKey()+" "+m.getValue());
}

Scanner sc = new Scanner(System.in);
```

```
        System.out.print("Enter a student: ");
        String s = sc.nextLine();

        System.out.println("Faculties are: ");
        h1.forEach((k, v) -> {
            if(k.equals(s))
            v.forEach(w -> {for(Map.Entry m:h2.entrySet()){
                if(m.getKey().equals(w))
                    System.out.println(m.getValue());
            }});
        });
    }
}
```

OUTPUT:



2. Create a class tourist which has the data members name, state, famous_spot . Create a list of all the states in the south. Add the tourist places to the list. Display the list in sorted order based on state. Search a tourist spot from the list and display the details. Raise an exception if the details are not present.

CODE:

```
import java.util.*;

public class tourist

{

public static void main (String[]args)

{

TouristClass t = new TouristClass ("Vennela", "Karnataka");
```

```
t.printDetails ();
}
}
class TouristClass
{
private final Map < String, List < String >> myMaps =new HashMap < String, List < String >> ();
public String name;
public String state;
public static List < String > famousSpots = new ArrayList < String > ();
TouristClass (String n, String s)
{
List < String > states = new ArrayList < String > ();
states.add ("Kerala");
states.add ("Karnataka");
states.add ("Tamil-Nadu");
states.add ("Andhra-Pradesh");
states.add ("Telangana");
List < List < String >> ls2d = new ArrayList < List < String >> ();
List < String > spot_for_kerala = new ArrayList < String > ();
spot_for_kerala.add ("Alleppey");
spot_for_kerala.add ("Cochin");
```

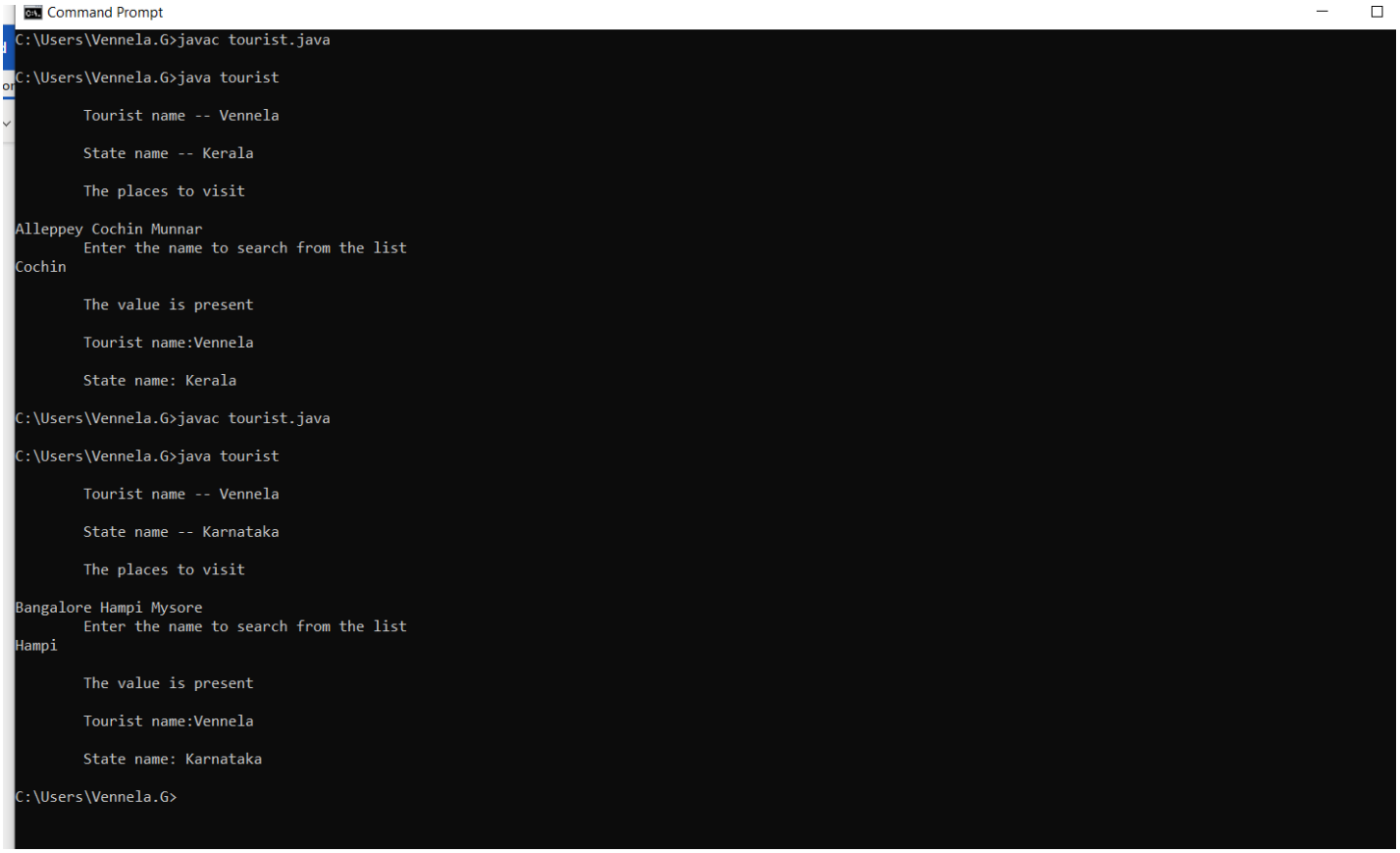
```
spot_for_kerala.add ("Munnar");

List < String > spot_for_karnataka = new ArrayList < String > ();

spot_for_karnataka.add ("Hampi");
spot_for_karnataka.add ("Mysore");
spot_for_karnataka.add ("Bangalore");

List < String > spot_for_tamil_nadu = new ArrayList < String > ();
spot_for_tamil_nadu.add ("Chennai");
spot_for_tamil_nadu.add ("Kanyakumari");
spot_for_tamil_nadu.add ("Trichy");
List < String > spot_for_andhrapradesh = new ArrayList < String > ();
spot_for_andhrapradesh.add ("Tirupati");
spot_for_andhrapradesh.add ("Vizag");
spot_for_andhrapradesh.add ("Amaravati");
List < String > spot_for_telangana = new ArrayList < String > ();
spot_for_telangana.add ("Hyderabad");
spot_for_telangana.add ("Warangal");
spot_for_telangana.add ("Shamshabad");

ls2d.add (spot_for_kerala);
```

```
ls2d.add (spot_for_karnataka);
ls2d.add (spot_for_tamil_nadu);
ls2d.add (spot_for_andhrapradesh);
ls2d.add (spot_for_telangana);
for (int i = 0; i <3; i++)
{
myMaps.put (states.get (i), ls2d.get (i));
}
name = n;
state = s;
switch (state)
{
case "Kerala":
famousSpots = myMaps.get ("Kerala");
break;
case "Karnataka":
famousSpots = myMaps.get ("Karnataka");
break;
case "Tamil-Nadu":
famousSpots = myMaps.get ("Tamil-Nadu");
break;
case "Andhra-Pradesh":
```

```
famousSpots = myMaps.get ("Andhra-Pradesh");
break;
case "Telangana":
famousSpots = myMaps.get ("Telangana");
break;
}
}
void printDetails ()
{
System.out.println ("\n\tTourist name -- " + name);
System.out.println ("\n\tState name -- " + state);
System.out.println ("\n\tThe places to visit\n\t");
Collections.sort (famousSpots);
for (int i = 0; i < famousSpots.size (); i++)
{
System.out.print (famousSpots.get (i) + " ");
}
System.out.println ("\n\tEnter the name to search from the list");
Scanner sc = new Scanner (System.in);
String val = sc.next ();
if (famousSpots.contains (val) == true)
{
```

```
System.out.println ("\n\tThe value is present");
System.out.println ("\n\tTourist name:" + name);
System.out.println ("\n\tState name: " + state);
}
else
{
System.out.println ("\n\tNo value found");
}
}
}
```

OUTPUT:



3. The following list gives the amount of rainfall (in cms) recorded at a particular place for 12 months.

10.2, 11.9, 8.0, 11.2, 10.8, 6.9, 8.2, 11.5, 10.4, 8.7, 7.8, 7.5.

Store these values in an queue. Find the average rainfall and display the count of the number of months in which the rainfall is more than the average.

CODE:

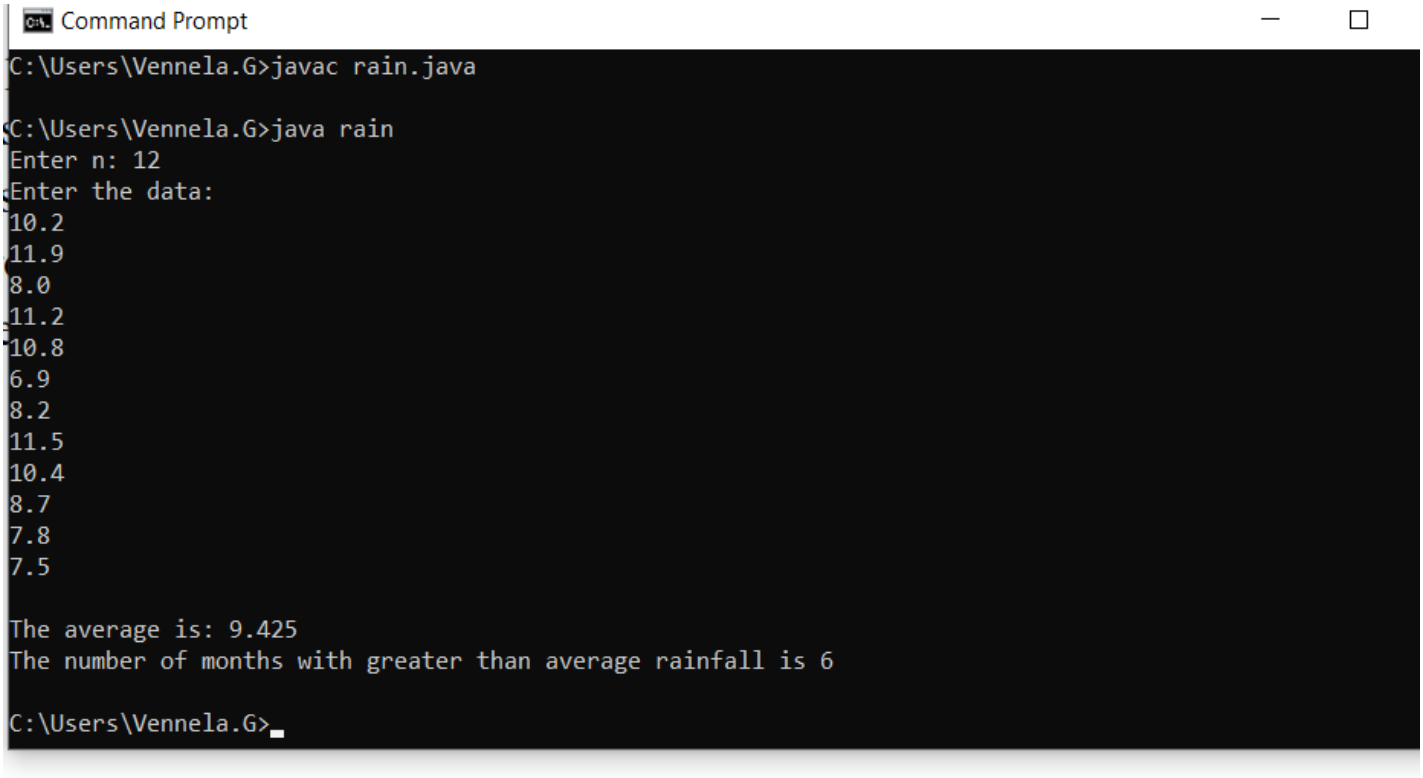
```
import java.util.*;

public class rain {

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

```
Scanner sc = new Scanner(System.in);
Queue<Float> q = new LinkedList<>();
System.out.print("Enter n: ");
int n = sc.nextInt();
float sum = 0f;
System.out.println("Enter the data: ");
for(int i =0;i<n;i++){
float val = sc.nextFloat();
q.add(val);
sum+= val;
}
float avg = (sum/n);
System.out.println("\nThe average is: "+avg);
int count=0;
for(int i =0;i<n;i++){
float ele = q.remove();
if(ele > avg){
count++;
}
}
System.out.println("The number of months with greater than average rainfall is "+count);
```

}

}

OUTPUT:



4. The librarian would like to maintain a list which has the information about the book name, author, price, no_of_copies in the library. Max 5 books can be placed in a rack. Create a hashmap of the book object with the rack no. Write a method search to read a book name and return its rack no. Write a method sort to display the book name in a particular rack.

CODE:

Rack.java

import java.lang.*;

import java.util.*;

```
public class Rack {
public static Scanner sc = new Scanner(System.in);
static HashMap<Integer,ArrayList<Book>> map=new HashMap<Integer,
ArrayList<Book>>();
public static void main(String[] args){
int choice = 0;
while(choice != 4){
System.out.print("1.Add\n2.Search\n3.Sort\n4.Exit\n\nEnter Your Choice: ");
choice = sc.nextInt();
switch(choice){
case 1: add();break;
case 2: System.out.print("Enter Book name to be searched: ");
String name = sc.next();
int rack = search(name);
if(rack == -1){
System.out.println("Book not found in any rack! ");
}else {
System.out.println("Book found in "+rack+" rack! ");
}break;
case 3: sort();break;
}
}
}
private static void sort() {
ArrayList<Integer> sortedKeys = new ArrayList<Integer>(map.keySet());
```

```
Collections.sort(sortedKeys);
for (Integer x : sortedKeys){
System.out.println("Rack = "+x);
ArrayList<Book> arr = map.get(x);
for(Book book: arr){
System.out.println("Name = "+book.name);
System.out.println("Author = "+book.author);
System.out.println("Price = "+book.price);
System.out.println("Number of copies = "+book.no_of_copies+"\n");
}
}
}
private static int search(String name) {
for (Map.Entry<Integer, ArrayList<Book>> e : map.entrySet()){
ArrayList<Book> arr = e.getValue();
for (Book book : arr){
if(book.name.equals(name)){
return e.getKey();
}
}
}
return -1;
}
private static void add() {
```

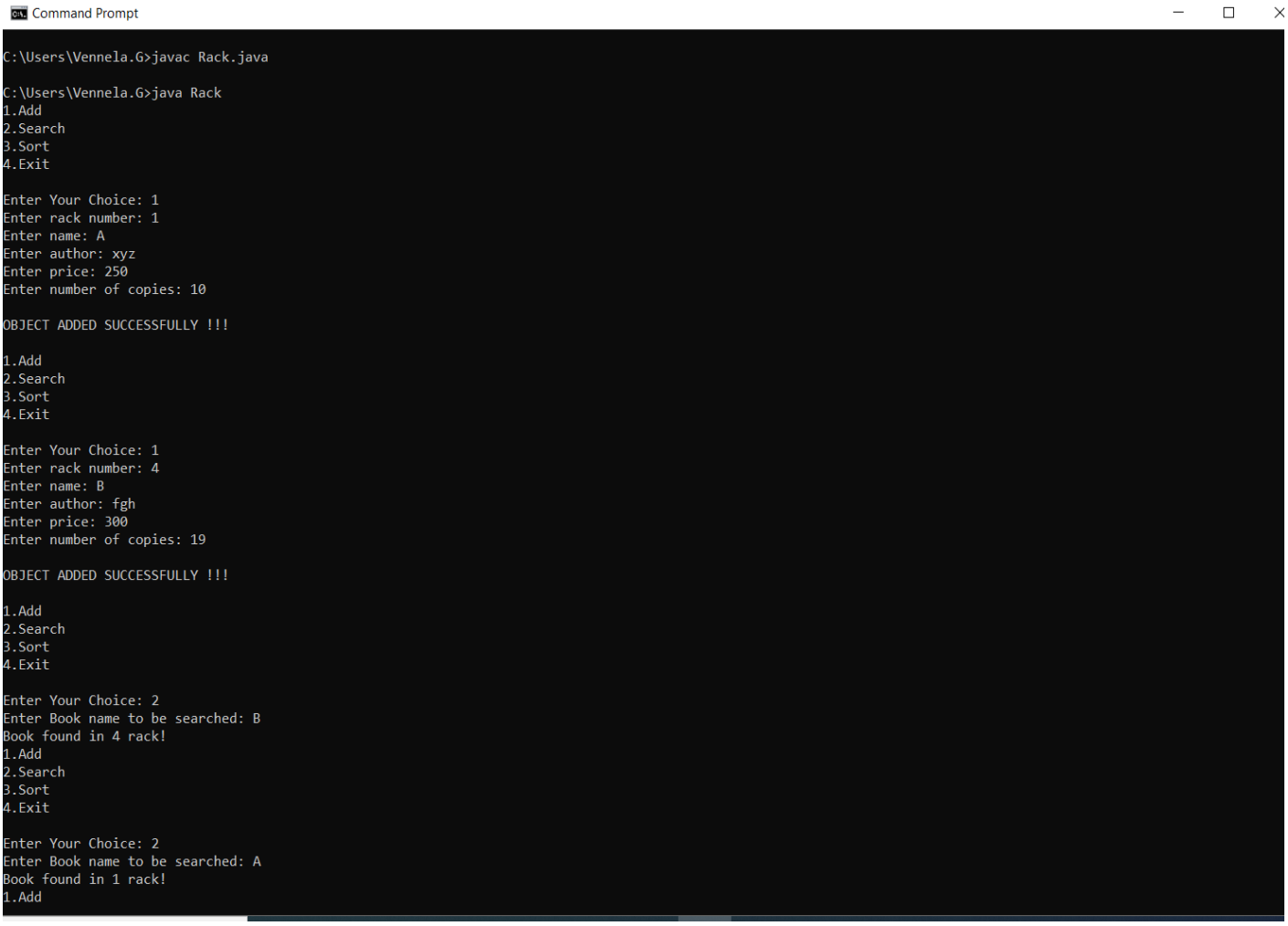
```
System.out.print("Enter rack number: ");
int rack = sc.nextInt();
System.out.print("Enter name: ");
String name = sc.next();
System.out.print("Enter author: ");
String author = sc.next();
System.out.print("Enter price: ");
float price = sc.nextFloat();
System.out.print("Enter number of copies: ");
int no_of_copies = sc.nextInt();
Book book = new Book(name,author,price,no_of_copies);
boolean isKeyPresent = map.containsKey(rack);
if(isKeyPresent){
ArrayList<Book> arr = map.get(rack);
arr.add(book);
map.put(rack,arr);
}
if(!isKeyPresent){
ArrayList<Book> arr = new ArrayList<Book>(5);
arr.add(book);
map.put(rack,arr);
}
System.out.println("\nOBJECT ADDED SUCCESSFULLY !!!\n");
}
```

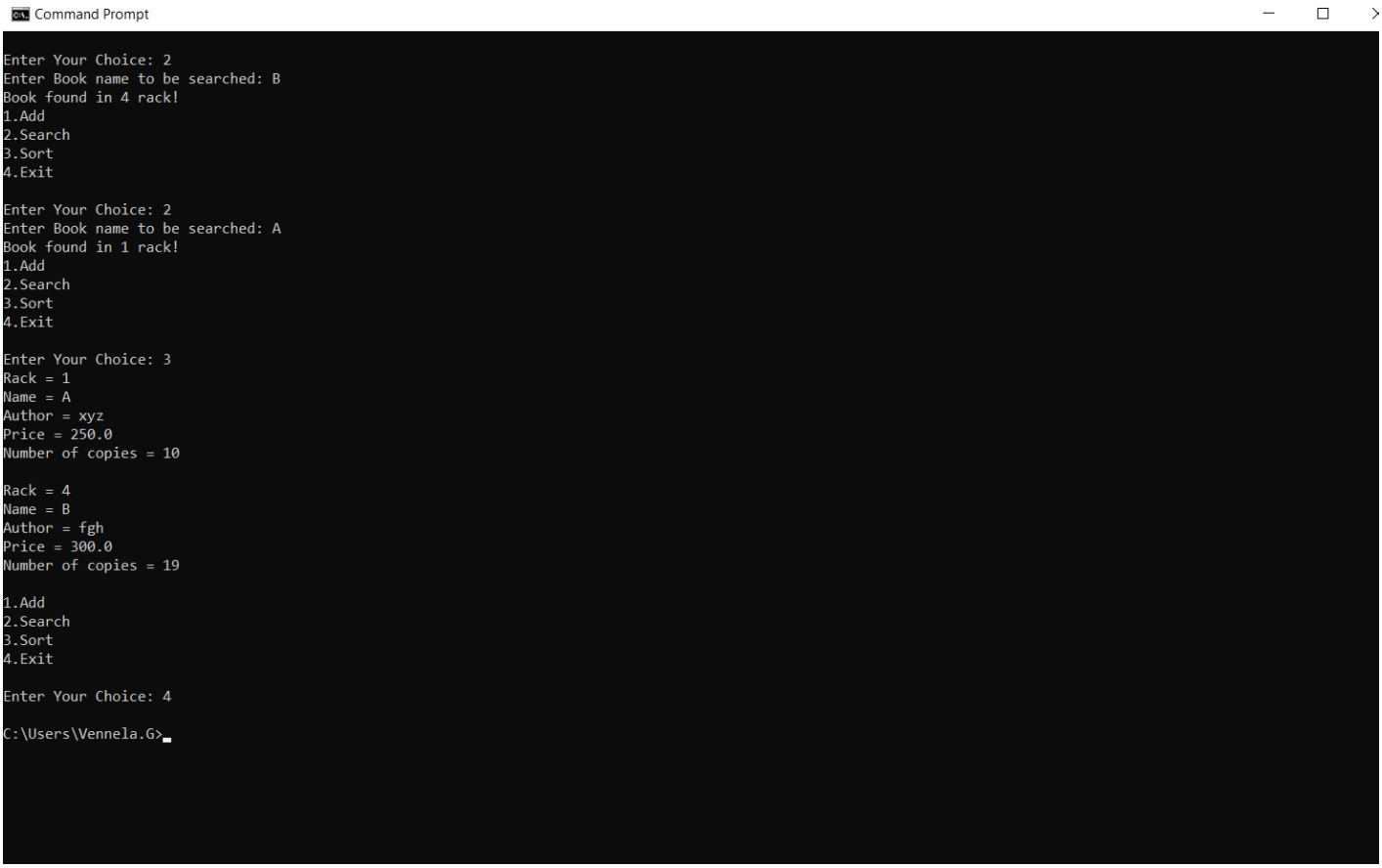
```
}
```

Book.java

```
public class Book {
String name;
String author;
float price;
int no_of_copies;
public Book(){
}
public Book(String name, String author, float price, int no_of_copies) {
this.name = name;
this.author = author;
this.price = price;
this.no_of_copies = no_of_copies;
} }
```

OUTPUT:


Command Prompt
C:\Users\Vennela.G>javac Rack.java
C:\Users\Vennela.G>java Rack
1.Add
2.Search
3.Sort
4.Exit
Enter Your Choice: 1
Enter rack number: 1
Enter name: A
Enter author: xyz
Enter price: 250
Enter number of copies: 10
OBJECT ADDED SUCCESSFULLY !!!
1.Add
2.Search
3.Sort
4.Exit
Enter Your Choice: 1
Enter rack number: 4
Enter name: B
Enter author: fgh
Enter price: 300
Enter number of copies: 19
OBJECT ADDED SUCCESSFULLY !!!
1.Add
2.Search
3.Sort
4.Exit
Enter Your Choice: 2
Enter Book name to be searched: B
Book found in 4 rack!
1.Add
2.Search
3.Sort
4.Exit
Enter Your Choice: 2
Enter Book name to be searched: A
Book found in 1 rack!
1.Add



5. An Industry collects the product sample measurements (product id, diameter, length, weight) for quality test and sends it to the quality assurance (QA) department in a serialized manner. The QA departments deserialize the samples and checks if the length=10cm, diameter=3cm, weight=100gms. The product id of defective samples are stored in a set. The product id of correct samples are stored in another set. Sort the correct samples in set.

CODE:

```
import java.io.*;

import java.util.*;
```

```
class Main implements Serializable
{
public String pid;
public int diameter;
public int length;
public int weight;

public Main(String pid, int diameter, int length, int weight)
{
this.pid = pid;
this.diameter = diameter;
this.length = length;
this.weight = weight;
}
}
public class Product{
public static void main(String[] args)
{
Scanner scanner = new Scanner(System.in);
String pid;
int diameter;
int length;
int weight;
ArrayList<String> list = new ArrayList<>();
```

```
int i = 0;
int n;
System.out.println("Enter the number of product samples: ");
n = scanner.nextInt();
for(int l=0; l<n; l++)
{
System.out.println("Enter the details of product "+(l+1));
pid = scanner.next();
length = scanner.nextInt();
diameter = scanner.nextInt();
weight = scanner.nextInt();
Sample object = new Sample(pid, diameter, length, weight);
String filename = "file.txt";
try
{
FileOutputStream file = new FileOutputStream(filename);
ObjectOutputStream out = new ObjectOutputStream(file);
out.writeObject(object);
out.close();
file.close();
}
catch (IOException ex)
{
System.out.println("IOException is caught");
```

```
}
Sample object2 = null;
try {
FileInputStream file = new FileInputStream(filename);
ObjectInputStream in = new ObjectInputStream(file);
object2 = (Sample) in.readObject(); in.close();
file.close();
if (object2.length != 10 || object2.diameter != 3 || object2.weight != 100)
{
list.add((object2.pid).toString());
}
}
catch (IOException ex)
{
System.out.println("IOException is caught");
}
catch (ClassNotFoundException ex)
{
System.out.println("ClassNotFoundException is caught");
}
}
System.out.println("The product id of defective sample is:");
for(String j:list)
{
```

```
System.out.println(j);
}
}
}
```

OUTPUT: