VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Name :- Atishay Jain
Registration number :- 19BDS0033
Java CSE 1007 slot (C1 + TC1)
Theory Digital Assignment

Arrays :-

Q1) Let there be a class of 40 students. Let the faculty divide the class into 5 groups. Writa a java progarmme such that it would return the summation of the students with the highest roll numbers in their respective groups.

Ans)

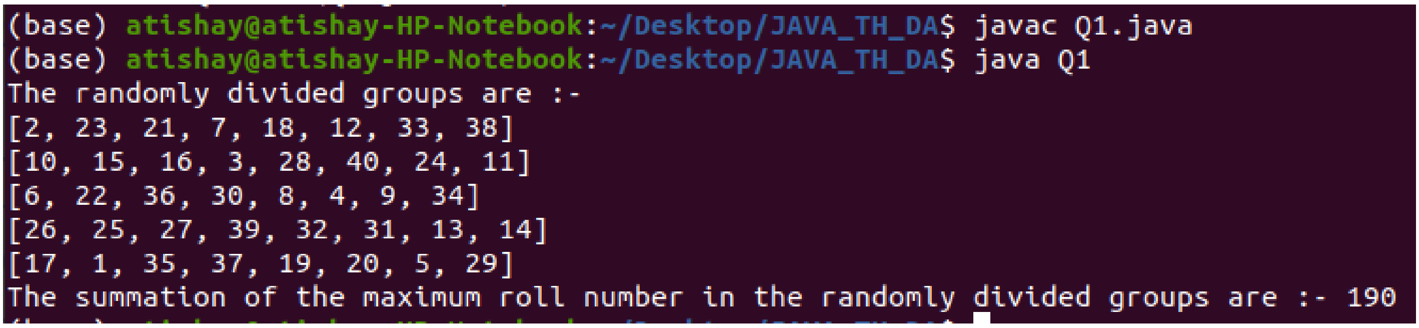
Code :-

```
import java.util.*;

public class Q1{
    public static void main(String[] args){
    int[] array = new int[40];
    for(int i = 1;i <= 40;i++){
        array[i-1] = i;
    }

    Random rand = new Random();

    for (int i = 0; i < array.length; i++) {
        int randomIndexToSwap = rand.nextInt(array.length);
        int temp = array[randomIndexToSwap];
        array[randomIndexToSwap] = array[i];
        array[i] = temp;
    }

    int[] groupA = new int[8];
    int[] groupB = new int[8];
    int[] groupC = new int[8];
    int[] groupD = new int[8];
```

```
int[] groupE = new int[8];

for(int j = 0;j < 40;j++){
    if(j<8){
        groupA[j] = array[j];
    }
    else if(j>=8 && j<16){
        groupB[j-8] = array[j];
    }
    else if(j>=16 && j<24){
        groupC[j-16] = array[j];
    }
    else if(j>=24 && j<32){
        groupD[j-24] = array[j];
    }
    else if(j>=32 && j<40){
        groupE[j-32] = array[j];
    }
}

System.out.println("The randomly divided groups are :- ");
System.out.println(Arrays.toString(groupA));
System.out.println(Arrays.toString(groupB));
System.out.println(Arrays.toString(groupC));
System.out.println(Arrays.toString(groupD));
System.out.println(Arrays.toString(groupE));

int[] maxArray = new int[5];

int maxA = groupA[0];
for (int i = 1; i < groupA.length; i++)
    if (groupA[i] > maxA)
        maxA = groupA[i];
maxArray[0] = maxA;

int maxB = groupB[0];
for (int i = 1; i < groupB.length; i++)
    if (groupB[i] > maxB)
        maxB = groupB[i];
maxArray[1] = maxB;

int maxC = groupC[0];
for (int i = 1; i < groupC.length; i++)
    if (groupC[i] > maxC)
        maxC = groupC[i];
maxArray[2] = maxC;

int maxD = groupD[0];
for (int i = 1; i < groupD.length; i++)
    if (groupD[i] > maxD)
        maxD = groupD[i];
maxArray[3] = maxD;
```

```
    int maxE = groupE[0];
    for (int i = 1; i < groupE.length; i++)
        if (groupE[i] > maxE)
            maxE = groupE[i];
    maxArray[4] = maxE;

    int sum_ = 0;
    for(int i =0;i < 5;i++){
        sum_ += maxArray[i];
    }

    System.out.println("The summation of the maximum roll number in the randomly divided groups
are :- " + sum_);
}}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q1.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q1
The randomly divided groups are :-
[2, 23, 21, 7, 18, 12, 33, 38]
[10, 15, 16, 3, 28, 40, 24, 11]
[6, 22, 36, 30, 8, 4, 9, 34]
[26, 25, 27, 39, 32, 31, 13, 14]
[17, 1, 35, 37, 19, 20, 5, 29]
The summation of the maximum roll number in the randomly divided groups are :- 190
```

String :-

Q2) Let the user input 2 words. Assume that each letter has a value attached to it. This value is based on alphabetical order(a=1, b=2, c=3......). Calculate the word weight by adding the numeric value corresponding to each alphabet. Write a java programme to tell how similiar two words are based on this word weight.

Ans)

Code :-

Output :-

Inheritance :-

Q3) Create a class Medicine containing attributes like manufacture date, expiry date, etc which are common to all medicines. Create 2 subclasses of medicine called “HighDoseMedicine” and “MildMedicine” which will inherit from the main class. Give these subclasses attributes that are suitable to their type. Like for MildMedicine you could give an attribute telling weather it’s allopathic or ayurvedic, etc.

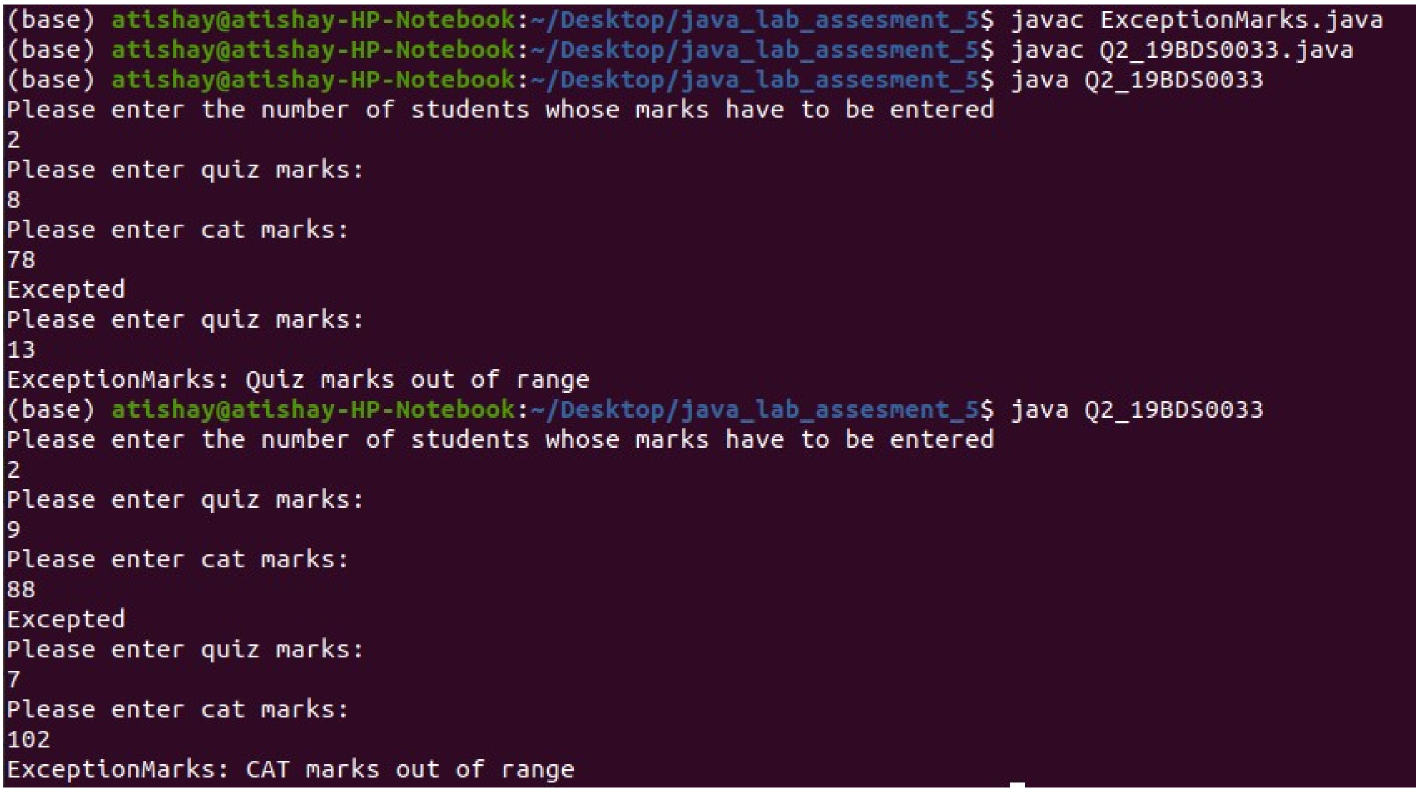
Ans)

Code :-

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

```
class Medicine {

    Date manufactureDate;
    int shelfMonthDuration;

}

class MildMedicine extends Medicine{

    String typeMedicine;

    public void setDate(Date manufactureDate){
        this.manufactureDate = manufactureDate;
    }
    public void setDuration(int shelfMonthDuration){
        this.shelfMonthDuration = shelfMonthDuration;
    }
    public void setType(String typeMedicine){
        this.typeMedicine = typeMedicine;
    }
    public void show(){
        System.out.println("manufacture date :- "+manufactureDate);
        System.out.println("Shelf life in months :- "+shelfMonthDuration);
        System.out.println("Type :- "+typeMedicine);
    }

}

class HighDoseMedicine extends Medicine{

    String emergencyLevel; // critical,severe,high

    public void setDate(Date manufactureDate){
        this.manufactureDate = manufactureDate;
    }
    public void setDuration(int shelfMonthDuration){
        this.shelfMonthDuration = shelfMonthDuration;
    }
    public void setEmergeny(String emergencyLevel){
        this.emergencyLevel = emergencyLevel;
    }

    public void show(){
        System.out.println("manufacture date :- "+manufactureDate);
        System.out.println("Shelf life in months :- "+shelfMonthDuration);
        System.out.println("Severity :- "+emergencyLevel);
    }

}

public class Q2{
    public static void main(String args[]){

        MildMedicine medicine1 = new MildMedicine();
        HighDoseMedicine medicine2 = new HighDoseMedicine();

        Calendar calendar1 = Calendar.getInstance();
        calendar1.set(2020, 06, 29, 59, 59, 59);
        Date date1 = calendar1.getTime();

        medicine1.setDate(date1);
        medicine1.setDuration(15);
        medicine1.setType("Homeopathy");

        Calendar calendar2 = Calendar.getInstance();
        calendar2.set(2020, 05, 21, 59, 59, 59);
        Date date2 = calendar2.getTime();

        medicine2.setDate(date2);
        medicine2.setDuration(12);
        medicine2.setEmergeny("critical");

        System.out.println("For modicine 1 :- ");
        medicine1.show();
        System.out.println("");
        System.out.println("For modicine 2 :- ");
        medicine2.show();
    }
}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/java_lab_assesment_5$ javac ExceptionMarks.java
(base) atishay@atishay-HP-Notebook:~/Desktop/java_lab_assesment_5$ javac Q2_19BDS0033.java
(base) atishay@atishay-HP-Notebook:~/Desktop/java_lab_assesment_5$ java Q2_19BDS0033
Please enter the number of students whose marks have to be entered
2
Please enter quiz marks:
8
Please enter cat marks:
78
Excepted
Please enter quiz marks:
13
ExceptionMarks: Quiz marks out of range
(base) atishay@atishay-HP-Notebook:~/Desktop/java_lab_assesment_5$ java Q2_19BDS0033
Please enter the number of students whose marks have to be entered
2
Please enter quiz marks:
9
Please enter cat marks:
88
Excepted
Please enter quiz marks:
7
Please enter cat marks:
102
ExceptionMarks: CAT marks out of range
```

Interfaces :-

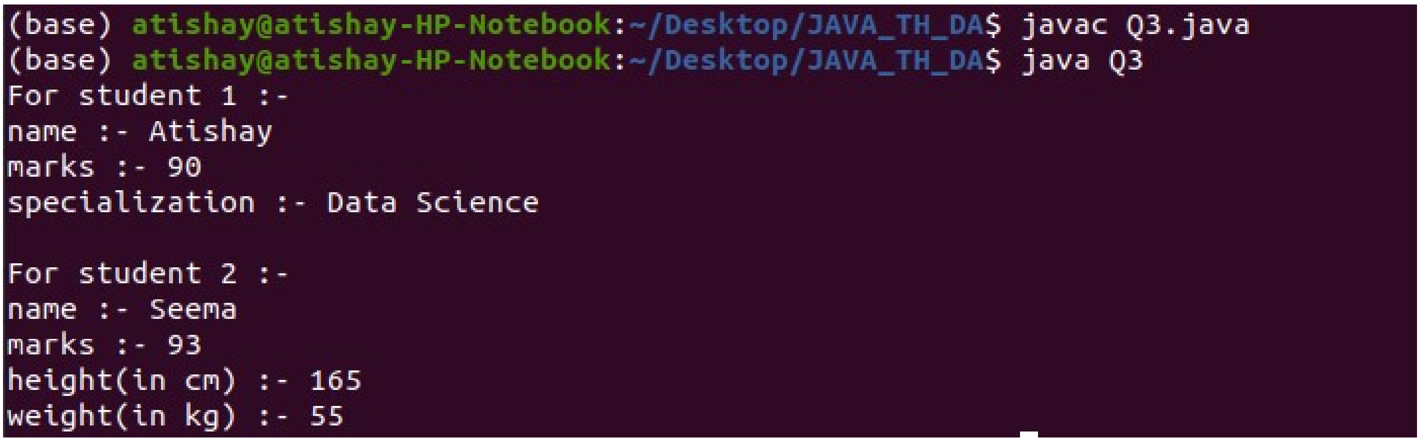
Q4) Create an interface Student. Then create a concrete class “ScienceStudent” and an abstarct class “SportsStudent” that implement Student. Give (name, marks, specialization) as the attributes for ScienceStudent and (height and weight) as attributes for Sports student. Create object of these classes and print their values.

Ans)

Code :-

```
import java.util.*;

interface Student {

   public void setName(String name);
   public void setMarks(int marks);

}

class ScienceStudent implements Student{

   String name;
   int marks;
   String specialization;

   public void setName(String name){
      this.name = name;
   }
   public void setMarks(int marks){
      this.marks = marks;
   }
   public void setSpecialization(String specialization){
      this.specialization = specialization;
   }
   public void show(){
      System.out.println("name :- "+name);
      System.out.println("marks :- "+marks);
      System.out.println("specialization :- "+specialization);
   }

}

class SportsStudent implements Student{

   String name;
   int marks;
   int weight;
   int height;

   public void setName(String name){
        this.name = name;
    }
    public void setMarks(int marks){
        this.marks = marks;
    }
    public void setWeight(int weight){
        this.weight = weight;
    }
    public void setHeight(int height){
        this.height = height;
    }
    public void show(){
        System.out.println("name :- "+name);
        System.out.println("marks :- "+marks);
        System.out.println("height(in cm) :- "+height);
        System.out.println("weight(in kg) :- "+weight);
    }


}

public class Q3{

    public static void main(String args[]){

        ScienceStudent student1 = new ScienceStudent();
        SportsStudent student2 = new SportsStudent();

        student1.setName("Atishay");
        student1.setMarks(90);
        student1.setSpecialization("Data Science");

        student2.setName("Seema");
        student2.setMarks(93);
        student2.setWeight(55);
        student2.setHeight(165);

        System.out.println("For student 1 :- ");
        student1.show();
        System.out.println("");
        System.out.println("For student 2 :- ");
        student2.show();
    }
}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q3.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q3
For student 1 :-
name :- Atishay
marks :- 90
specialization :- Data Science

For student 2 :-
name :- Seema
marks :- 93
height(in cm) :- 165
weight(in kg) :- 55
```

Packages :-

Q5) Create a java package called "MyOffice" which will have different files containing classes for different designations such that all classes inherit from one base class which will have basic properties shared for all employees irrespective of their designation.

Ans)

Code :-

Output :-

User defined Exception :-

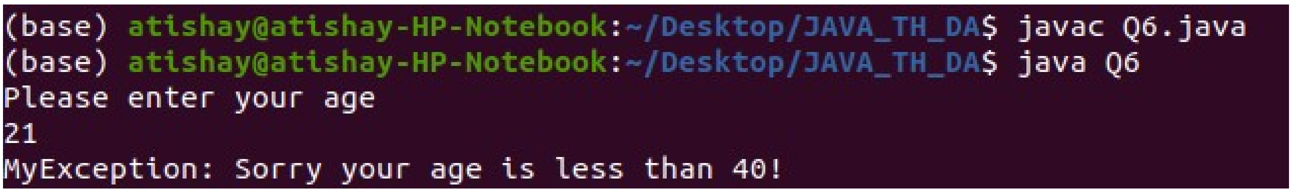
Q6) Create a java programme to verify weather a person if fit to take covid vaccine shot. The criterea are as follows :- (Age > 40) (No underlying health condition) (Should be a frontline worker) (Should work for more than 40 hours/week) If the user does not meet the above conditions then programme should throw an exception with the reason in the above mentioned order.

Ans)

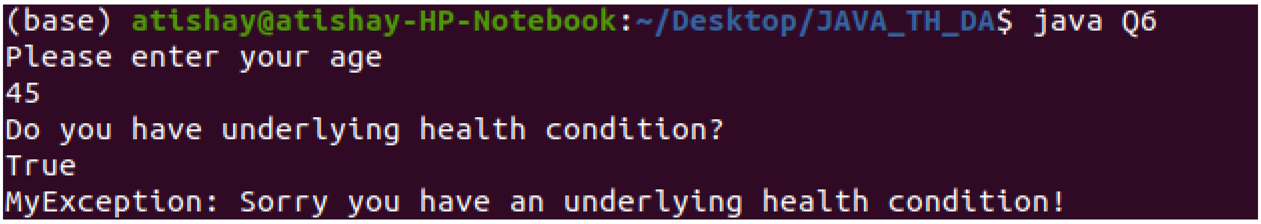
Code :-

```
import java.util.Scanner;

public class Q6{
    public static void main(String args[]){

        int age;
        boolean healthCondition;
        boolean frontlineWorker;
        int hoursWork;

        Scanner sc = new Scanner(System.in);

        System.out.println("Please enter your age");
        age = sc.nextInt();
        try{
            if(age<40){
                throw new MyException("Sorry your age is less than 40!");
            }}catch(MyException e){
                System.out.println(e);
            }


        System.out.println("Do you have underlying health condition?");
        healthCondition = sc.nextBoolean();
        try{
            if(healthCondition==true){
                throw new MyException("Sorry you have an underlying health condition!");
            }}catch(MyException e){
                System.out.println(e);
            }


        System.out.println("Are you a frontline worker?");
        frontlineWorker = sc.nextBoolean();
        try{
            if(frontlineWorker==false){
                throw new MyException("Sorry you are not a frontline worker");
            }}catch(MyException e){
                System.out.println(e);
            }


        System.out.println("How many hours a week do you work?");
        hoursWork = sc.nextInt();
        try{
            if(hoursWork<40){
                throw new MyException("Sorry you work less than 40 hours a week");
            }}catch(MyException e){
                System.out.println(e);
            }

            System.out.println("You satisfy thecriteria to get vaccinated!");

    }
}

class MyException extends Exception{
    public MyException(String msg){
        super(msg);
    }
}
```

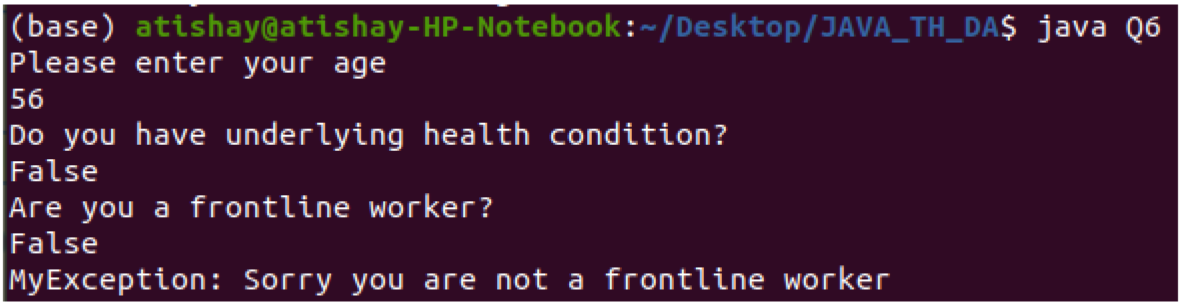
Output :-



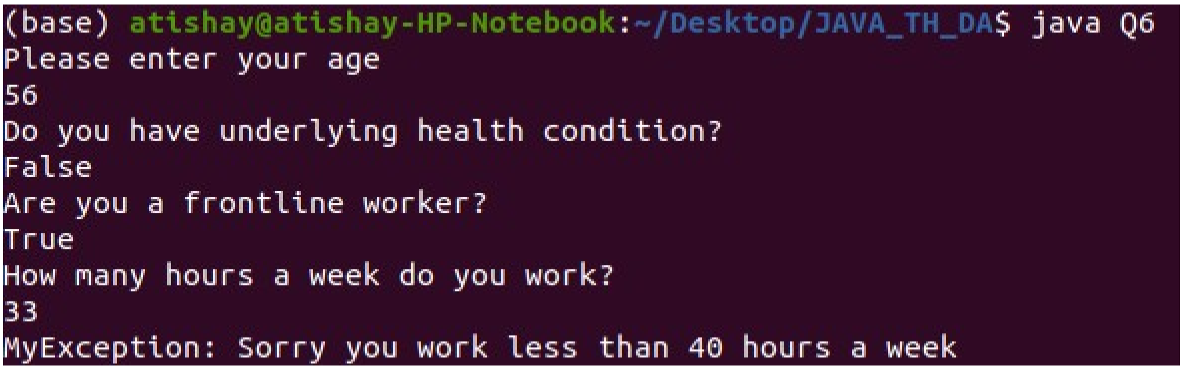
```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q6.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q6
Please enter your age
21
MyException: Sorry your age is less than 40!
```



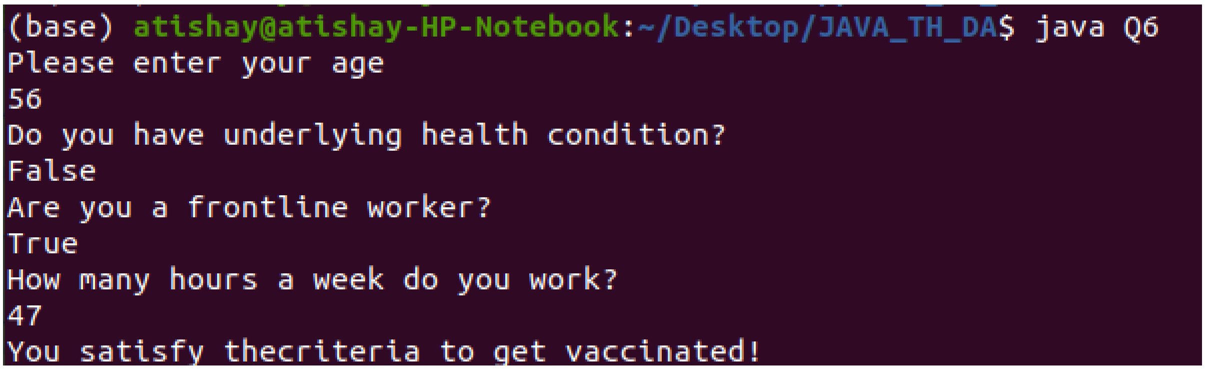
```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q6
Please enter your age
45
Do you have underlying health condition?
True
MyException: Sorry you have an underlying health condition!
```



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q6
Please enter your age
56
Do you have underlying health condition?
False
Are you a frontline worker?
False
MyException: Sorry you are not a frontline worker
```



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q6
Please enter your age
56
Do you have underlying health condition?
False
Are you a frontline worker?
True
How many hours a week do you work?
33
MyException: Sorry you work less than 40 hours a week
```



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q6
Please enter your age
56
Do you have underlying health condition?
False
Are you a frontline worker?
True
How many hours a week do you work?
47
You satisfy thecriteria to get vaccinated!
```

Multithreading (synchronization – interthread communication):-

Q7) Writa a java programme that created two threads to print consecultive numbers till 12. After 12 numbers have been printed then create a new thred that prints numbers from 13 to 20. Print the numbers in such a way that it is visible that which thread is printing those numbers.

Ans)

Code :-

Output :-

Files :-

Q8) Let’s say that you have an machine learning algorithm that detects people in a video and assigns id to them. This algorithm gives output as a txt file with data in a csv format with each row corresponding to a unique track in a given frame. Data format :- <frame_id, track_id, bbox_tlx, bbox_tly, bbox_width, bbox_height>. Here bbox_tlx, bbox_tly mean the x and y coordinates of the top left corner of the bounding box respectively. Assuming that each frame has equal number of tracks, write a java programme to output the bounding box coordinates of a given track in a given frame.

Ans)
Input file :-

GroungTruth.txt

```
1,1,1703,385,157,339,1,-1,-1,-1
1,2,1293,455,83,213,1,-1,-1,-1
1,3,259,449,101,261,1,-1,-1,-1
1,4,1253,529,55,127,1,-1,-1,-1
2,1,1699,383,159,341,1,-1,-1,-1
2,2,1293,455,83,213,1,-1,-1,-1
2,3,261,447,101,263,1,-1,-1,-1
2,4,1253,529,55,127,1,-1,-1,-1
3,1,1697,383,159,343,1,-1,-1,-1
3,2,1293,455,83,213,1,-1,-1,-1
3,3,263,447,101,263,1,-1,-1,-1
3,4,1255,529,55,127,1,-1,-1,-1
4,1,1695,383,159,343,1,-1,-1,-1
4,2,1293,455,83,213,1,-1,-1,-1
4,3,265,447,101,263,1,-1,-1,-1
4,4,1257,529,55,127,1,-1,-1,-1
5,1,1693,381,159,347,1,-1,-1,-1
5,2,1295,455,83,213,1,-1,-1,-1
5,3,267,447,101,263,1,-1,-1,-1
5,4,1259,529,55,129,1,-1,-1,-1
6,1,1689,381,161,349,1,-1,-1,-1
6,2,1297,455,83,213,1,-1,-1,-1
6,3,269,447,103,263,1,-1,-1,-1
6,4,1261,529,55,129,1,-1,-1,-1
7,1,1687,381,161,349,1,-1,-1,-1
7,2,1299,455,83,213,1,-1,-1,-1
7,3,271,447,103,263,1,-1,-1,-1
7,4,1263,529,55,129,1,-1,-1,-1
8,1,1685,379,161,353,1,-1,-1,-1
8,2,1301,455,83,213,1,-1,-1,-1
8,3,273,447,103,263,1,-1,-1,-1
8,4,1265,529,55,131,1,-1,-1,-1
9,1,1683,379,161,353,1,-1,-1,-1
9,2,1303,455,83,213,1,-1,-1,-1
9,3,275,447,103,263,1,-1,-1,-1
9,4,1267,529,53,131,1,-1,-1,-1
10,1,1679,379,163,355,1,-1,-1,-1
10,2,1305,455,83,213,1,-1,-1,-1
10,3,277,445,105,265,1,-1,-1,-1
```

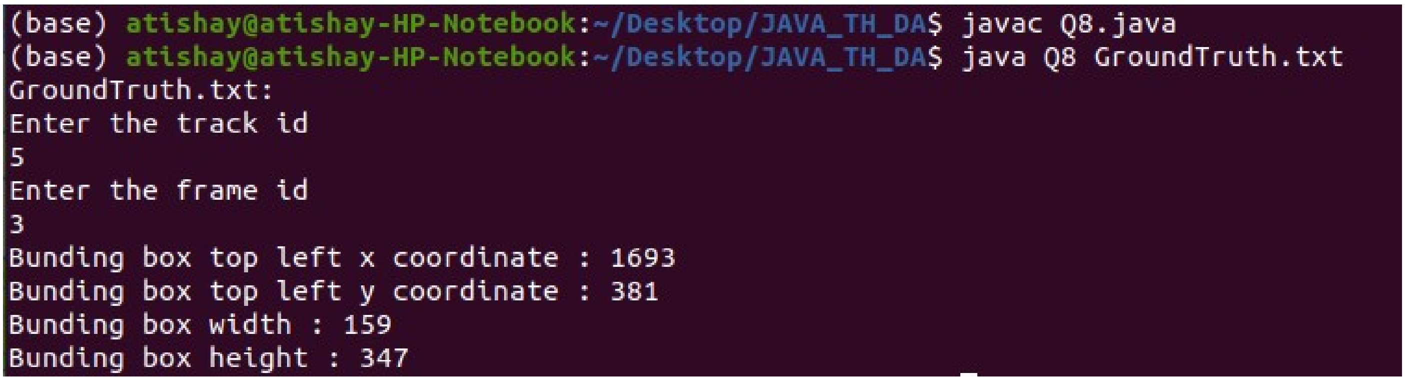
10,4,1269,529,53,131,1,-1,-1,-1

Code :-

Q8.java

```
import java.io.*;
import java.util.*;
public class Q8 {
public static void main(String[] args) {

 if (args.length == 0) {
 System.out.println("Usage: java Q8");
 return;
 }

 for (int i = 0; i < args.length; i++) {
 System.out.print(args[i] + ": \n");
 details(args[i]);
 }

 }

 private static void details(String fileName) {
 Scanner sc = new Scanner(System.in);
 System.out.print("Enter the track id\n");
 int t = sc.nextInt();
 System.out.print("Enter the frame id\n");
 int f = sc.nextInt();
 try(LineNumberReader lnr = new LineNumberReader(new FileReader(fileName))) {
 String line;
 int i = 1;
 while ((line = lnr.readLine()) != null) {
     if (i == f*4 + t){
 String details[] = line.split(",");
 System.out.println("Bunding box top left x coordinate : " + details[2]);
 System.out.println("Bunding box top left y coordinate : " + details[3]);
 System.out.println("Bunding box width : " + details[4]);
 System.out.println("Bunding box height : " + details[5]);
    }
    i = i + 1;
 }
 lnr.close();
 } catch (FileNotFoundException e) {
 System.out.println("Error!!! File Not Found");
 } catch (Exception e) {
 System.out.println("Error!!! Some problem occurred when reading from file");
 }
 }
}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q8.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q8 GroundTruth.txt
GroundTruth.txt:
Enter the track id
5
Enter the frame id
3
Bunding box top left x coordinate : 1693
Bunding box top left y coordinate : 381
Bunding box width : 159
Bunding box height : 347
```

List or Map or Set :-

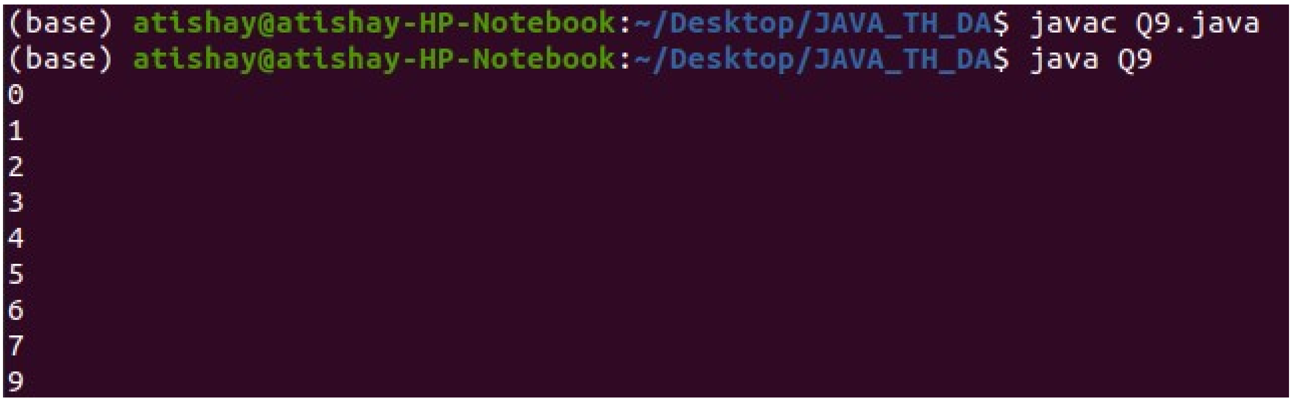
Q9) Take a list and then sort it in such a way that their are no identical elements remaining in the list even though the input string had identical values.

Ans)

Code :-

```
import java.util.*;

public class Q9 {
    public static void main(String[] args) {
        List<Integer> values = new ArrayList<Integer>();
        values.add(4);
        values.add(3);
        values.add(2);
        values.add(5);
        values.add(5);
        values.add(1);
        values.add(6);
        values.add(9);
        values.add(0);
        values.add(7);

        Collections.sort(values); // Sorting list
        Set<Integer> valuesSorted = new LinkedHashSet<Integer>(values); // Removing duplicate
values

        for(Integer o : valuesSorted){
            System.out.println(o);
        }

    }
}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q9.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q9
0
1
2
3
4
5
6
7
9
```

Generic classes and methods :-

Q10) Using method overloading write a function that calculates the area as well as volume of :- i) Cuboid ii) Rectangle iii) Square Create classes for each of the geometric shape which will store its attributes.

Ans)

Code :-

Output :-

GUI programming using JavaFX with event handler and JavaFX menus :-

Q11) You own a grocery store. Make a JavaFX programme that calculates the total bill to be paid by the customer given the number of items purchased by the customer and price of a single product brought by the customer upon clicking the button purchase. Make a nice UI design in which the above mentioned 2 attributes could be entered.

Ans)

Code :-

Output :-

Ploymorphism :-

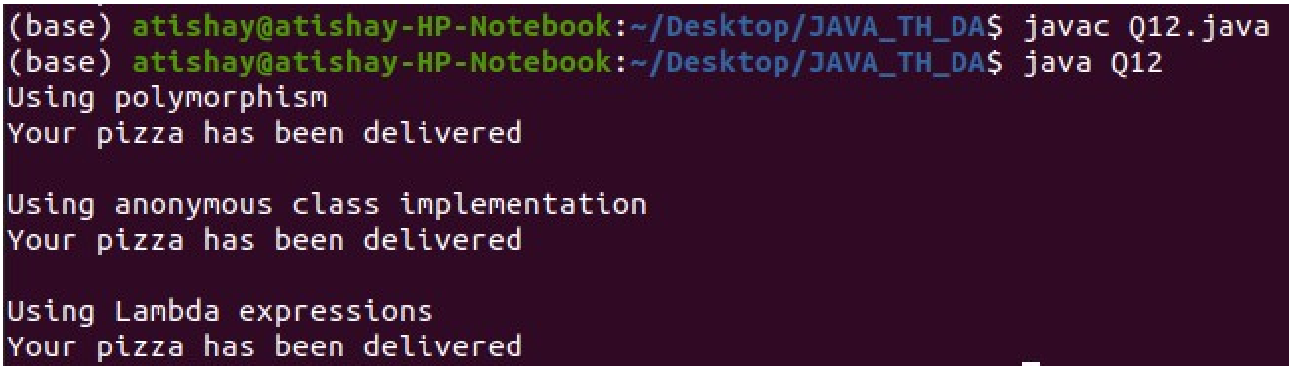
Q12) Write a java code that implements a functional interface that prints out that “Your pizza has been delivered” for your Pizza delivery app using the following 3 ways :- (i) Using Polymorphism (ii) Using anonymous class impelmentation (iii) Using lambda expressions

Ans)

Code :-

```
@FunctionalInterface
interface Pizza{
    void deliveryMessage();
}

class PizzaDelivery implements Pizza{
    public void deliveryMessage(){
        System.out.println("Your pizza has been delivered");
    }
}

public class Q12 {
    public static void main(String[] args) {

        Pizza pizza = new PizzaDelivery(); // Polymorphic statement
        System.out.println("Using polymorphism");
        pizza.deliveryMessage();

        // Anonymous class implementation
        Pizza pizza1 = new Pizza(){
            @Override
            public void deliveryMessage(){
                System.out.println("Your pizza has been delivered");
            }
        };
        System.out.println("\nUsing anonymous class implementation");
        pizza1.deliveryMessage();

        // Using Lambda expressions
        Pizza pizza2 = () -> {
            System.out.println("Your pizza has been delivered");
        };
        System.out.println("\nUsing Lambda expressions");
        pizza2.deliveryMessage();
    }

}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q12.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q12
Using polymorphism
Your pizza has been delivered

Using anonymous class implementation
Your pizza has been delivered

Using Lambda expressions
Your pizza has been delivered
```

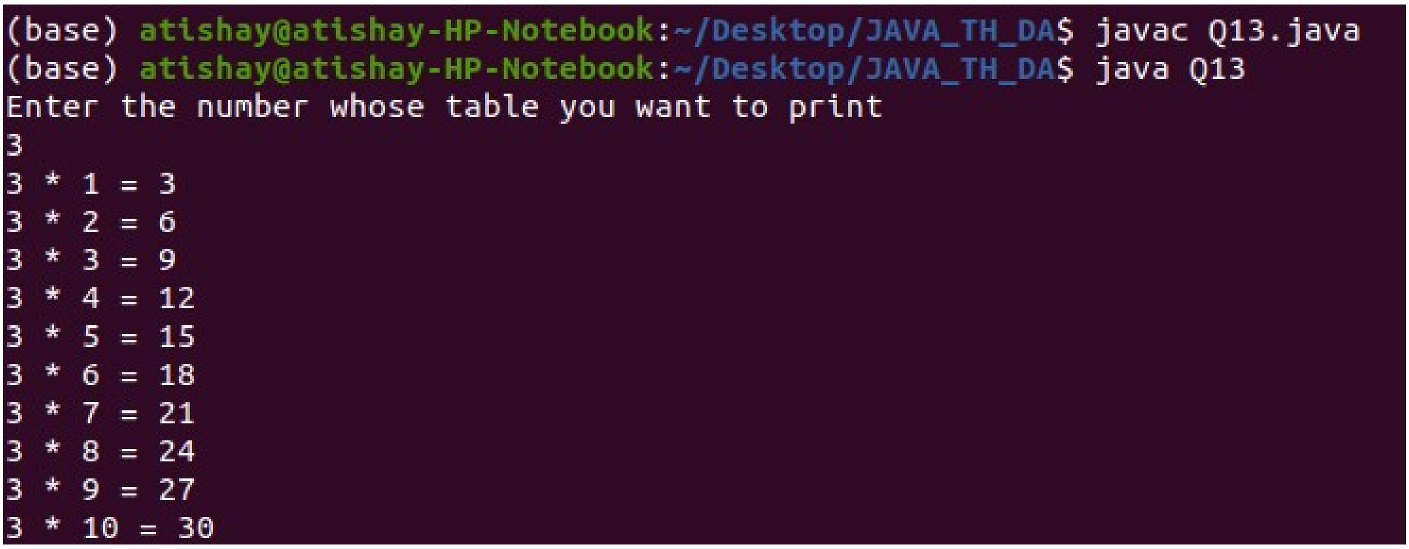
Q13) Create a file that stores the table of a number n taken from user. Create this file using 2 different methods((Libraries DataOutputStream, FileOutputStream) and (Libraries FileOutputStream and DataOutputStream)).

Code :-

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

```
public class Q13 {
    public static void main(String[] args) throws Exception{
    File f = new File("tables.txt");

    FileOutputStream fos = new FileOutputStream(f);
    DataOutputStream dos = new DataOutputStream(fos);
    int i = 0;
    System.out.println("Enter the number whose table you want to print");
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();
    for(i = 1;i <= 10; i++){
        dos.writeUTF(n + " * " + i + " = " + n*i);
    }

    FileInputStream fis = new FileInputStream(f);
    DataInputStream dis = new DataInputStream(fis);
    while(dis.available()>0) {
    String str = dis.readUTF();
    System.out.println(str);
    }
    }
}
```

Output :-



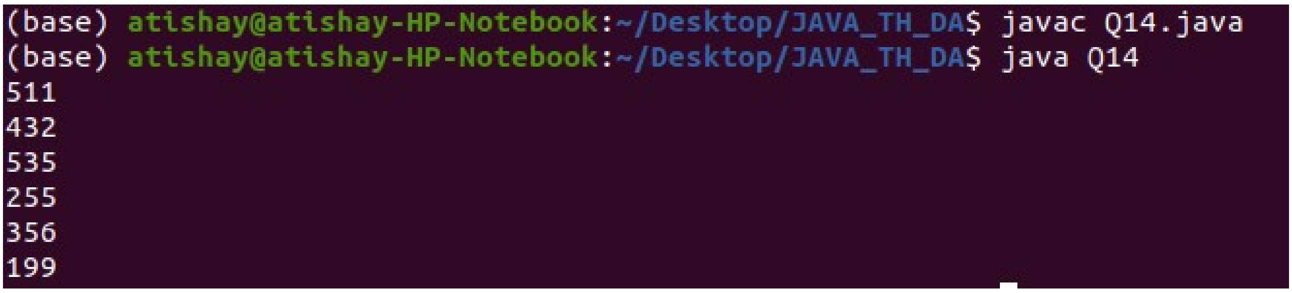
```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q13.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q13
Enter the number whose table you want to print
3
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
3 * 10 = 30
```

Q14) Write a java programme that takes in integers from the user and sorts them according the values of their unit place digit.

Code :-

```
import java.util.*;

public class Q14{
    public static void main(String args[]){
        List<Integer> values = new ArrayList<Integer>();
        values.add(432);
        values.add(356);
        values.add(255);
        values.add(511);
        values.add(535);
        values.add(199);

        Collections.sort(values, (i,j) -> i%10>j%10?1:-1);

        for(Integer o : values){
            System.out.println(o);
        }

    }
}
```

Output :-



```
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ javac Q14.java
(base) atishay@atishay-HP-Notebook:~/Desktop/JAVA_TH_DA$ java Q14
511
432
535
255
356
199
```

Q15) You get n tomatoes on every nth odd day and you have to give m tomatoes on every mth even day. Lets say you start from day 1. Write a java programme using recursion to calculate how many tomatoes you have or you owe at the end of k days. (Write a java programme that calculates the s(n) where s(n) = 1 + (-2) + (3) + (-4) .... (+n or -n) using recursion.)

Code :-

Output :-