

# Rajalakshmi Engineering College

Name: Shivam Jaiswal

Email: 240701499@rajalakshmi.edu.in

Roll no: 240701499

Phone: 7318545479

Branch: REC

Department: CSE - Section 6

Batch: 2028

Degree: B.E - CSE

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 10\_Q2

Attempt : 1

Total Mark : 10

Marks Obtained : 10

#### **Section 1 : COD**

##### **1. Problem Statement**

John is organizing a fruit festival, and the quantities of various fruits are stored in a HashMap where fruit names are keys and quantities are values.

Help him develop a program to find the total quantity of fruits for the festival by summing up the values in the HashMap.

##### ***Input Format***

The input consists of fruit quantities in the format 'fruitName:quantity', where fruitName is the name of the fruit(a string), and quantity is a double value representing the quantity.

The input is terminated by entering "done".

##### ***Output Format***

The output prints a double value, representing the sum of values in the HashMap, rounded off to two decimal places.

If the value is not numeric, print "Invalid input".

If any special characters other than ':' are entered, print "Invalid format".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: Banana:15.2

Orange:56.3

Mango:47.3

done

Output: 118.80

### ***Answer***

```
// You are using Java
import java.util.*;
import java.text.DecimalFormat;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        HashMap<String, Double> fruits = new HashMap<>();
        boolean invalidFormat = false;
        boolean invalidInput = false;

        while (true) {
            String input = sc.nextLine().trim();

            if (input.equalsIgnoreCase("done")) {
                break;
            }

            // Check format: must contain ':' and no other special character like '-'
            if (!input.contains(":") || input.contains("-") || input.contains("@") ||
                input.contains("#") || input.contains("$")
                || input.contains("!") || input.contains("%") || input.contains("&"))
                ||
```

```
        input.contains("*")) {  
            invalidFormat = true;  
            break;  
    }  
  
    String[] parts = input.split(":");
```

```
    if (parts.length != 2) {  
        invalidFormat = true;  
        break;  
    }
```

```
    String fruit = parts[0];  
    String qtyStr = parts[1];
```

```
    try {  
        double quantity = Double.parseDouble(qtyStr);  
        fruits.put(fruit, quantity);  
    } catch (NumberFormatException e) {  
        invalidInput = true;  
        break;  
    }  
}
```

```
if (invalidFormat) {  
    System.out.println("Invalid format");  
} else if (invalidInput) {  
    System.out.println("Invalid input");  
} else {  
    double total = 0.0;  
    for (double q : fruits.values()) {  
        total += q;  
    }
```

```
    DecimalFormat df = new DecimalFormat("0.00");  
    System.out.println(df.format(total));  
}
```

```
    sc.close();  
}
```

**Status : Correct**

**Marks : 10/10**