

# Rajalakshmi Engineering College

Name: Harish V.H  
Email: 240701175@rajalakshmi.edu.in  
Roll no: 240701175  
Phone: 9080255347  
Branch: REC  
Department: CSE - Section 2  
Batch: 2028  
Degree: B.E - CSE

Scan to verify results



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

### 2028\_REC\_OOPS using Java\_Week 5\_Q4

Attempt : 1  
Total Mark : 10  
Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

You are working as a developer for CityCab, a taxi service company that wants to build a ride fare management system.

Each customer booking has:

A Booking ID (integer) A Customer Name (string) A Distance Travelled in km (double)

The fare calculation rules are:

Base Fare = 50 units (flat charge for every ride). Per km charge = 10 units/km. If the distance is greater than 20 km, a 10% discount is applied on the total fare.

You are required to implement this system using:

A class with attributes for booking details. A constructor to initialize booking details. Setter methods to update details if needed. Getter methods to retrieve details. Objects of the class to represent customer rides.

Finally, display each booking's details and final fare.

### ***Input Format***

The first line of input contains an integer N, representing the number of bookings.

For each booking:

- The next line contains the booking ID (integer).
- The following line contains the customer's name (string).
- The next line contains the distance travelled (double).

### ***Output Format***

For each booking, print the details in the following format:

1. Booking ID: <booking\_id>
2. Customer Name: <customer\_name>
3. Final Fare: <final\_fare> (rounded to one decimal place)

Refer to the sample output for formatting specifications.

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

Input: 1  
1234  
Rahul Sharma  
15

Output: Booking ID: 1234  
Customer Name: Rahul Sharma  
Final Fare: 200.0

### ***Answer***

```
import java.util.Scanner;  
  
class CabBooking {
```

```
private int bookingId;
private String customerName;
private double distance;

// Constructor
public CabBooking(int bookingId, String customerName, double distance) {
    this.bookingId = bookingId;
    this.customerName = customerName;
    this.distance = distance;
}

// Setters
public void setBookingId(int bookingId) {
    this.bookingId = bookingId;
}

public void setCustomerName(String customerName) {
    this.customerName = customerName;
}

public void setDistance(double distance) {
    this.distance = distance;
}

// Getters
public int getBookingId() {
    return bookingId;
}

public String getCustomerName() {
    return customerName;
}

public double getDistance() {
    return distance;
}

// Method to calculate final fare
public double calculateFare() {
    double baseFare = 50;
    double perKmRate = 10;
    double fare = baseFare + (distance * perKmRate);
}
```

```

        // Apply 10% discount if distance > 20 km
        if (distance > 20) {
            fare = fare - (fare * 0.10); // 10% discount
        }

        return fare;
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int N = Integer.parseInt(sc.nextLine()); // Number of bookings
        CabBooking[] bookings = new CabBooking[N];

        for (int i = 0; i < N; i++) {
            int id = Integer.parseInt(sc.nextLine());
            String name = sc.nextLine();
            double distance = Double.parseDouble(sc.nextLine());

            CabBooking booking = new CabBooking(id, name, distance);
            bookings[i] = booking;
        }

        // Output
        for (CabBooking b : bookings) {
            System.out.printf("Booking ID: %d\n", b.getBookingId());
            System.out.printf("Customer Name: %s\n", b.getCustomerName());
            System.out.printf("Final Fare: %.1f\n", b.calculateFare());
        }

        sc.close();
    }
}

```

**Status :** Correct

**Marks :** 10/10