

Assignment #21

Customer Table Booking - Requirement 1

Your friend has opened a new restaurant and he finds it difficult to keep track of the tables booked by customers. Being an aspirant programmer help your friend with a small application that would help him keep track of the tables booked. There are three major domains **Table**, **Customer**, and **Booking**. The Table and Customer domain are used to store table and Customer details respectively. The third domain Booking is used maintain the booking details of a Customer corresponding to a table.

Requirement 1:

Let's start off by creating Customer objects and check whether two objects are equal by overriding equals method.

1. Create a **Customer** Class with the following private attributes:

Member Field Name	Type
id	Long
name	String
mobileNumber	String
birthdate	java.util.Date
averageSpendAmount	Double
totalAmount	Double
dateEnrolled	java.util.Date
rating	Double

2. Mark all the attributes as private
3. Create / Generate appropriate Getters & Setters
4. Add a default constructor and a parameterized constructor to take in all attributes in the given order: Customer(Long id, String name, String mobileNumber, java.util.Date birthdate, Double averageSpendAmount, Double totalAmount, java.util.Date dateEnrolled, Double rating)
5. When the "customer" object is printed, it should display the following details: [Override the toString method]

Print format:

Id:"id"

Name:"name"

Mobile Number:"mob num"

Date of Birth:"dob"

Average spent amount:"avg spent amount"

Total amount:"total amount"

Date Enrolled:"date enrolled"

Rating:"rating"

6. Two customers are considered same if they have the same name, mobileNumber, and birthdate. Implement the logic in the appropriate function. (Case – Insensitive) [Override the equals method]

The return type of Equals method is bool(either true or false), If it returns true, then print "Customer 1 is same as Customer 2", else print "Customer 1 and Customer 2 are different", That print statement will be present in the main method.

7. The input format consists of customer details separated by comma in the below order, (id, name, mobileNumber, birthdate, averageSpendAmount, totalAmount, dateEnrolled, rating)

The Input to your program would be details of two customers, you need to display their details as given in "5th point(refer above)" and compare the two customers and display if the Customers are same or different.

Create a class named as Main, which contains the main method, all the input, and output operations are performed in this method(main).

Note: There is an empty line between display statements. The empty lines are displayed in the Main method.

Sample INPUT & OUTPUT 1:

Enter the details of Customer 1:

1,John,9876543210,12-12-1990,5000,25000,12-12-2012,3

Enter the details of Customer 2:

2,James,9876543201,12-12-1991,6000,35000,12-12-2013,4

Details of customer 1:

Id:1

Name:John

Mobile Number:9876543210

Date of Birth:12-12-1990

Average spent amount:5000.0

Total amount:25000.0

Date Enrolled:12-12-2012

Rating:3.0

Details of customer 2: Id:2

Name:James

Mobile Number:9876543201

Date of Birth:12-12-1991

Average spent amount:6000.0

Total amount:35000.0

Date Enrolled:12-12-2013

Rating:4.0

Customer 1 and Customer 2 are different

Sample INPUT & OUTPUT 2:

Enter the details of Customer 1:

1, John, 9876543210, 12-121990, 5000, 25000, 12-12-2012, 3

Enter the details of Customer 2: **2, John, 9876543210, 12-121990, 8000, 28000, 12-11-2012, 3.5**

Details of customer 1:

Id: 1

Name: John

Mobile Number: 9876543210

Date of Birth: 12-12-1990

Average spent amount: 5000.0

Total amount: 25000.0

Date Enrolled: 12-12-2012

Rating: 3.0

Details of customer 2: Id: 2

Name: John

Mobile Number: 9876543210

Date of Birth: 12-12-1990

Average spent amount: 8000.0

Total amount: 28000.0

Date Enrolled: 12-11-2012

Rating: 3.5

Customer 1 is same as Customer 2