

✕

2 : 5 : 55

[10 Marks]



◀

◀

Exam Data Submit Help 2 : 5 : 34

- Do not include any extra instance/static variables and instance/static methods in the given classes
- Case-insensitive comparison is to be done if not explicitly mentioned
- Do not change any value or case of the given variables
- Read notes and examples for better understanding of the logic
- In the derived classes, the order of passing arguments to the constructor would be- base class variables followed by derived class variables

Implementation details:

Class Name	Implementation Details
Caterer	Partially implemented
WeddingCaterer	Partially implemented
BookCatering	Partially implemented

BookCatering class:

availableCuisineArr:

- This is a static array (String[]) which denotes the list of available cuisines (String) for catering

Activate Windows
Go to Settings to activate Windows.

Exam Data Submit Help

2 : 5 : 20

- It is initialized as below:

availableCuisineArr	["indian", "italian", "chinese"]
---------------------	----------------------------------

Note:

- This array is supplied and hence, no need to code
- Do not change the CASE of the elements in the array

validateCuisine():

- This method checks if the cuisine (String) requested by the customer is present in the availableCuisineArr
- If it is present, return true
- Otherwise, return false

Note: Perform case-insensitive string comparison

Example: If cuisine is "indian", then the above method returns true

Caterer class:

generateBookingID():

- This method auto-generates and sets the bookingID (String)
- The bookingID must be prefixed by "r" in lowercase followed by the auto-generated value starting from 1001

Activate Windows
Go to Settings to activate Windows.

Exam Data Submit Help

2 : 5 : 20

- It is initialized as below:

availableCuisineArr	["indian", "italian", "chinese"]
---------------------	----------------------------------

Note:

- This array is supplied and hence, no need to code
- Do not change the CASE of the elements in the array

validateCuisine():

- This method checks if the cuisine (String) requested by the customer is present in the availableCuisineArr
- If it is present, return true
- Otherwise, return false

Note: Perform case-insensitive string comparison

Example: If cuisine is "indian", then the above method returns true

Caterer class:

generateBookingID():

- This method auto-generates and sets the bookingID (String)
- The bookingID must be prefixed by "r" in lowercase followed by the auto-generated value starting from 1001

Activate Windows
Go to Settings to activate Windows.

HandsOn Client - ruchita.biradar

• The bookingID must be prefixed by "C" in uppercase followed by the auto-generated value starting from 1001

• Subsequent bookingID must be incremented by 1

• Use static variable counter appropriately to implement the auto-generation logic

Example: The first bookingID would be "C1001", the second would be "C1002" and so on

findCostPerPlate():

• This method finds and returns the costPerPlate (int) based on the noOfPlates (int) requested by the customer

• Refer the below table to identify the costPerPlate

noOfPlates	costPerPlate
Less than 25	-1
Between 25 and 100 (both inclusive)	300
Between 101 and 200 (both inclusive)	200
Greater than 200	150

Example: If the customer requests for 280 noOfPlates, the costPerPlate would be 150 currency

WeddingCaterer class:

Exam Data Submit Help2:5:5

- The **bookingID** must be prefixed by "C" in uppercase followed by the auto-generated value starting from 1001
- Subsequent **bookingID** must be incremented by 1
- Use static variable **counter** appropriately to implement the auto-generation logic

Example: The first **bookingID** would be "C1001", the second would be "C1002" and so on

findCostPerPlate():

- This method finds and returns the *costPerPlate* (int) based on the **noOfPlates** (int) requested by the customer
- Refer the below table to identify the *costPerPlate*

noOfPlates	costPerPlate
Less than 25	-1
Between 25 and 100 (both inclusive)	300
Between 101 and 200 (both inclusive)	200
Greater than 200	150

Example: If the customer requests for 280 **noOfPlates**, the *costPerPlate* would be 150 currency

WeddingCaterer class:

Activate Windows
Go to Settings to activate Windows.

```
9         this.cuisine = cuisine;
10     }
11
12     public boolean validateCustomer() {
13         if (this.customerName.length() >= 3) {
14             return true;
15         }
16         else {
17             return false;
18         }
19     }
20
21     //To Trainee
22     public Boolean validateCuisine() {
23         //Implement your logic here
24         for(int i=0; i<availableCuisineArr.length ; i++){
25             if(availableCuisineArr[i].equalsIgnoreCase(cuisine)){
26                 return true;
27             }
28         }
29
30         //Change the return statement accordingly
31         return false;
32     }
33
34 }
35
```

```

51
52 //To Trainee
53 public void generateBookingID() {
54     //Implement your logic here
55     this.bookingID = "C" + Integer.toString(++Caterer.counter);
56 }
57
58 //To Trainee
59 public int findCostPerPlate() {
60     int costPerPlate = 0;
61
62     //Implement your logic here
63     if(noOfPlates <25){
64         costPerPlate =-1;
65     }else if (noOfPlates>=25 && noOfPlates <=100){
66         costPerPlate =300;
67     }else if (noOfPlates >=101 && noOfPlates <=200){
68         costPerPlate = 200;
69     }else{
70         costPerPlate = 150;
71     }
72     return costPerPlate;
73 }
74
75 public abstract void calculateBillAmount();
76 }
77

```



```

12
13 //To Trainee
14 @Override
15 public void calculateBillAmount() {
16     //Implement your logic here
17     boolean isCustomer = getBookCatering().validateCustomer();
18     boolean isCuisine = getBookCatering().validateCuisine();
19     if(isCustomer == true && isCuisine == true && noOfDays >=1){
20         int costPerPlate = findCostPerPlate();
21         if(costPerPlate != -1){
22             generateBookingID();
23             double basicCost = (costPerPlate) *( getNoOfPlates());
24             int discountPercentage = identifyDiscountPercentage(basicCost);
25             basicCost -= (basicCost*discountPercentage/100);
26             if(serviceType.equals("Buffet")){
27                 basicCost += 5000.0;
28             }
29             setBillAmount(basicCost*noOfDays);
30         }else{
31             setBookingID("-1");
32             setBillAmount(-1.0);
33         }
34     }else{
35         setBookingID("-1");
36         setBillAmount(-1.0);
37     }
38 }
39 }
40 }
41

```