Payment - Inheritance

**Grade settings**: Maximum grade: 100  
**Disable external file upload, paste and drop external content**: Yes  
**Run**: Yes **Evaluate**: Yes  
**Automatic grade**: Yes **Maximum execution time**: 16 s

**Payment Status**

Roy is a wholesale cloth dealer who sells cloth material to the local tailors on monthly installments. At the end of each month, he collects the installment amount from all his customers. Some of his customers pay by Cheque, some pay by Cash and some by Credit Card. He wants to automate this payment process.

Help him to do this by writing a java program.

**Requirement 1:  Make Payment**

The application needs to verify the payment process and display the status report of payment by getting the inputs like due amount, payment mode and data specific to the payment mode from the user and calculate the balance amount.

**Component Specification: Payment** **Class**(Parent Class)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Component Name** | **Type(Class)** | **Attributes** | **Methods** | **Responsibilities** |
| Make payment for EMI amount | Payment | int dueAmount | Include a public getter and setter method |  |
| Make payment for EMI amount | Payment |  | public boolean payAmount() | The boolean payAmount() method should return true if there is no due to be paid, else return false. |

**Note:**

·        The attributes of Payment class should be private.

·        The payment can be of three types: Cheque, Cash, Credit Card.

**Component Specification: Cheque** **class** (Needs to be a child of Payment class)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Component Name** | **Type(Class)** | **Attributes** | **Methods** | **Responsibilities** |
|  | Cheque | String chequeNo  int chequeAmount  Date dateOfIssue | Include a public getter and setter method for all the attributes. |  |
| Make payment for EMI amount | Cheque |  | public boolean payAmount() | This is an overridden method of the parent class. It should return true if the cheque is valid and the amount is valid. Else return false. |

**Note:**

·        The cheque is valid for 6 months from the date of issue.

·        Assume the current date is 01-01-2020 in dd-MM-yyyy format.

·        The chequeAmount is valid if it is greater than or equal to the dueAmount.

**Component Specification: Cash** **class** (Needs to be a child of Payment class)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Component Name** | **Type(Class)** | **Attributes** | **Methods** | **Responsibilities** |
| Make payment for EMI amount | Cash | int cashAmount | Include a public getter and setter method for the attribute. |  |
| Make payment for EMI amount | Cash |  | public boolean payAmount() | This is an overridden method of the parent class. It should return true if the cashAmount is greater than or equal to the dueAmount. Else return false. |

**Component Specification: Credit** **class** (Needs to be a child of Payment class)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Component Name** | **Type (Class)** | **Attributes** | **Methods** | **Responsibilities** |
| Make payment for EMI amount | Credit | int creditCardNo    String cardType    int creditCardAmount | Include a public getter and setter method for all the attributes. |  |
| Make payment for EMI amount | Credit |  | public boolean payAmount() | This is an overridden method of the parent class. It should deduct the   dueAmount and service tax from the creditCardAmount and return true if the credit card payment was done successfully. Else return false. |

**Note:**

·        The payment can be done if the credit card amount is greater than or equal to the sum of due amount and service tax. Else payment cannot be made.

·        The cardType can be “silver” or “gold” or “platinum”. Set the creditCardAmount based on the cardType.

·        Also service tax is calculated on dueAmount based on cardType.

|  |  |  |
| --- | --- | --- |
| **Credit Card Type** | **Credit Card Amount** | **Service Tax** |
| silver | 10000 | 2% of the due amount |
| gold | 50000 | 5% of the due amount |
| platinum | 100000 | 10% of the due amount |

·        The boolean  payAmount() method should deduct the due amount and the service tax amount from a credit card. If the creditCardAmount is less than the dueAmount+serviceTax, then the payment cannot be made.

·        The balance in credit card amount after a successful payment should be updated in the creditCardAmount by deducting the sum of dueAmount and serviceTax from creditCardAmount itself.

**Component Specification: Bill** **class**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Component Name** | **Type(Class)** | **Attributes** | **Methods** | **Responsibilities** |
| Payment Status Report | Bill |  | public String processPayment (Payment obj) | This method should return a message based on the status of the payment made. |

**Note:**

·        If the payment is successful, processPayment method should return a message “Payment done successfully via cash” or “Payment done successfully via cheque” or “Payment done successfully via creditcard. Remaining amount in your <<cardType>> card is <<balance in CreditCardAmount>>”

·        If the payment is a failure, then return a message “Payment not done and your due amount is <<dueAmount>>”

Create a **public class Main** with the main method to test the application.

**Note:**

·        Assume the current date as 01-01-2020 in dd-MM-yyyy format.

·        In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the rest of the text represents the output.

·        Ensure to follow the object oriented specifications provided in the question.

·        Ensure to provide the names for classes, attributes and methods as specified in the question.

·        Adhere to the code template, if provided.

·        Adhere to the sample input and output.

**Sample Input 1:**

Enter the due amount:

**3000**  
Enter the mode of payment(cheque/cash/credit):

**cash**  
Enter the cash amount:

**2000**

**Sample Output 1:**

Payment not done and your due amount is 3000

**Sample Input 2:**

Enter the due amount:

**3000**  
Enter the mode of payment(cheque/cash/credit):

**cash**  
Enter the cash amount:

**3000**

**Sample Output 2:**

Payment done successfully via cash

**Sample Input 3:**

Enter the due amount:

**3000**  
Enter the mode of payment(cheque/cash/credit):

**cheque**  
Enter the cheque number:

**123**  
Enter the cheque amount:

**3000**  
Enter the date of issue:

**21-08-2019**

**Sample Output 3:**

Payment done successfully via cheque

**Sample Input 4:**

Enter the due amount:

**3000**  
Enter the mode of payment(cheque/cash/credit):

**credit**  
Enter the credit card number:

**234**  
Enter the card type(silver,gold,platinum):

**silver**

**Sample Output 4:**

Payment done successfully via credit card. Remaining amount in your silver card is 6940