**SynCart**

**E-Commerce Application for Buying Consumer Electronics Products**

------------------------------------------------------------------------------------------------------------------------------------------

Need to prepare a console application for the E-Commerce Application.

The application should show the following menu option to select the operation to be performed:

1. **CustomerRegistration**
2. **Login**
3. **Exit**

## **Customer Registration**

1. If the customer selects 1 from the Main Menu, **CustomerRegistration** will be selected.
2. Get the below details from the customer to create a new customer account and add it to the list.
   1. CustomerName
   2. City
   3. MobileNumber
   4. WalletBalance
   5. EmailID
   6. CustomerID (Auto-generated)
3. Once the new account is created, need to **show the CustomerID** to the **current registered Customer.**

## **Login**

1. If the customer selects 2 from the main menu, the Login option will be selected.
2. Get the CustomerID and validate the CustomerID.
3. If the CustomerID doesn’t match, show “**Invalid customerID**”. Else, the following submenu option “**a”** to **“e**” is shown.
   1. **Purchase**
   2. **OrderHistory**
   3. **CancelOrder**
   4. **WalletBalance**
   5. **WalletRecharge**
   6. **Exit**

## Purchase:

1. Once the Customer logged in show the list of Products. Ask the customer to select a product using the Product ID.
2. Validate ProductID; if it is invalid, show **“Invalid ProductID.”**.
3. If it is valid, then ask for the count he wishes to purchase.
4. If the required count is not available in the product’s stock, then it will show **“Required count not available. Current availability is {**product’s stock count**}.”.**
5. If the count is available, calculate the total amount with the below formula.

Delivery charge is Rs 50.

Total Amount = (required count \* price per quantity) + Delivery Charge

1. Check the current logged-in customer’s wallet balance to ensure he has enough balance to purchase by comparing the total price.
2. If the wallet balance is insufficient for this order, then display “Insufficient Wallet Balance. Please recharge your wallet and do the purchase again.”.
3. If the wallet has sufficient balance, then.
   1. Deduct the total amount from the wallet balance of the current logged-in customer.
   2. Deduct the count from the stock availability of the product.
4. Create an order with the available details, and make its status as Ordered, add it to the order list and show “**Order Placed Successfully. Order ID: OID1001**”.
5. Show the delivery date of the order by making a calculation based on the purchase date and shipping duration of the product, like “**Order placed successfully. Your order will be delivered on {**Order date +shipping duration of the product**}**.

## OrderHistory:

Show all the information about the orders that the current logged-in customer has made.

## CancelOrder:

1. Show all orders placed by the current logged-in customer whose order status is **Ordered**.
2. Ask the customer to select an order to be cancelled by the **OrderID**.
3. Validate orderID and show “Invalid OrderID” if there is no such order.
4. If it is valid, then pick the order based on the provided **OrderID**.
5. Increase the available stock quantity by the count of products purchased in the current order to be cancelled.
6. Refund the amount to the customer’s wallet balance.
7. Change the order status to “Cancelled” and finally show “**Order:{OrderID} cancelled successfully.”.**

## WalletBalance:

Show the current available WalletBalance of the current logged-in customer.

## WalletRecharge:

1. Ask the customer whether he wishes to recharge the wallet.
2. If “Yes” then ask for the amount to be recharged and update the amount in the wallet and display the updated wallet balance.
3. Else, go to the submenu.

## Exit:

1. If Customer select “Exit” option, then need to exit sub menu and have to show the main menu.

## **Exit:**

1. If Customer selects option 3 in main menu “Exit” option will be selected and need to close the application.

**Classes:**

1. **Order Details:**

Properties:

* OrderID (Auto Increment – OID1001)
* CustomerID
* ProductID
* TotalPrice
* PurchaseDate
* Quantity
* OrderStatus – (Enum- Default, Ordered, Cancelled)

1. **Product Details**

Properties:

* ProductID (Auto Increment – PID2001)
* ProductName
* Stock
* Price
* ShippingDuration

1. **Customer Details:**

Properties:

* CustomerID (Auto Increment -CID3001)
* CustomerName
* City
* MobileNumber
* WalletBalance
* EmailID

**Methods:**

* + - **WalletRecharge:** Get the recharge amount through parameters and update wallet balance.
    - **DeductBalance:** Get the deducted amount through parameters and update wallet balance.

For validation purpose need to add the below objects.

Default Customer Details:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CustomerID** | **CustomerName** | **City** | **MobileNumber** | **WalletBalance** | **EmailID** |
| CID3001 | Ravi | Chennai | 9885858588 | 50000 | [ravi@mail.com](mailto:ravi@mail.com) |
| CID3002 | Baskaran | Chennai | 9888475757 | 60000 | [baskaran@mail.com](mailto:baskaran@mail.com) |

**Default product Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ProductID** | **ProductName** | **Stock** | **Price** | **ShippingDuration** |
| PID2001 | Mobile (Samsung) | 10 | 10000 | 3 |
| PID2002 | Tablet (Lenovo) | 5 | 15000 | 2 |
| PID2003 | Camara (Sony) | 3 | 20000 | 4 |
| PID2004 | iPhone | 5 | 50000 | 6 |
| PID2005 | Laptop (Lenovo I3) | 3 | 40000 | 3 |
| PID2006 | HeadPhone (Boat) | 5 | 1000 | 2 |
| PID2007 | Speakers (Boat) | 4 | 500 | 2 |

Default Order Details:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **OrderID** | **CustomerID** | **ProductID** | **TotalPrice** | **PurchaseDate** | **Quantity** | **OrderStatus** |
| OID1001 | CID3001 | PID2001 | 20000 | DateTime.Now | 2 | Ordered |
| OID1002 | CID3002 | PID2003 | 40000 | DateTime.Now | 2 | Ordered |