



Institute of Computer Engineering Technology



Fashion Shop Coursework

Fashion Shop

Case Study

The Fashion Shop, which was recently start online T-Shirt store, has a large number of transactions every day, Therefore the Fashion shop owner require a system to manage online t-shirt orders. As you are a talented iCET student, they have thought to give you a chance of making a system for them.

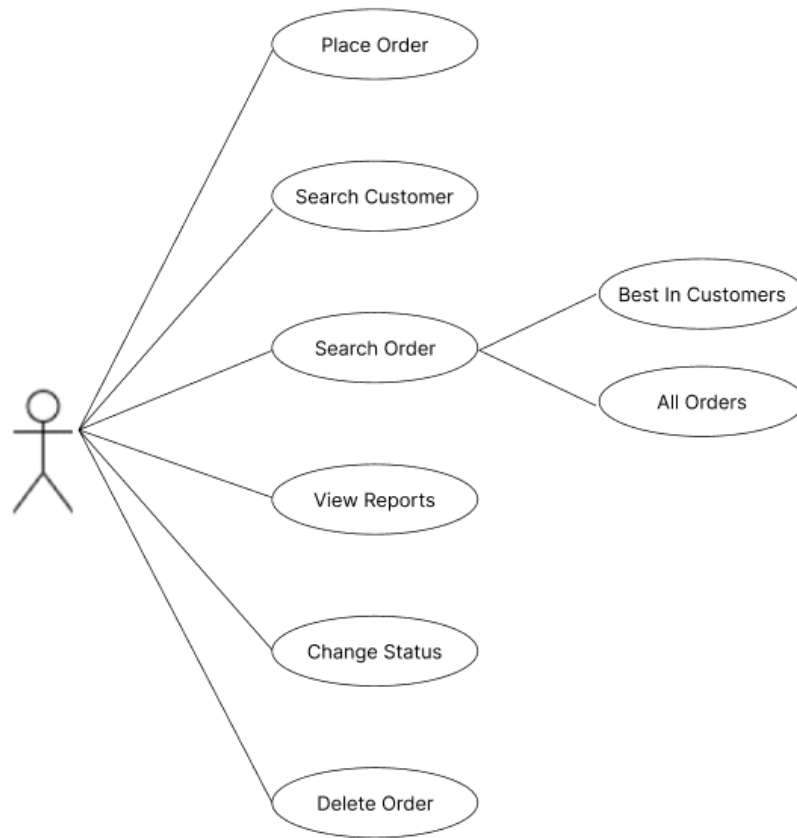


Figure 1 – Use case Diagram

Requirements

You are supposed to create a Java application to manage an online T-shirt store. In the application, you need to implement the following use cases.

When you run the application, you should come up with something similar to the following Command Line Interface (CLI), where the user can enter an option number that they want to execute. This will be the Home Page of the application that you will be developing. If the user inputs a wrong option, the program should not respond.

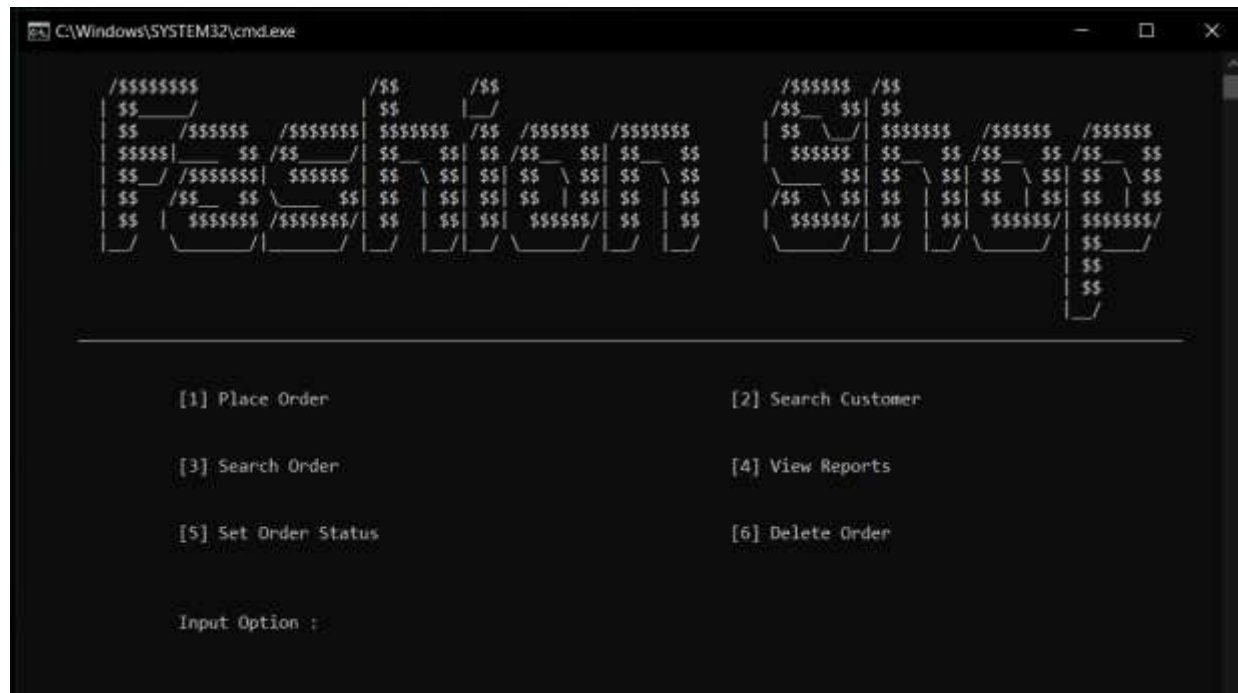


Figure 2 – Home Page of the Fashion Shop

01. Place Order ([Demo](#))

Placing a new order is easy. The system keeps 5 details related to the Order. They are Order ID, Customer Contact Number as the Customer ID, T-Shirt size, Quantity, and Order Status.

Prices of T-Shirts according to sizes are given below and There are constant.

T-Shirt size	Price (LKR)
XS	600.00
S	800.00
M	900.00
L	1000.00
XL	1100.00
XXL	1200.00

- Order ID – The Order ID should be generated by the system and the Order ID should start with 'ODR#' and should have 5 numbers. When the user selects the Place Order option on the home page, the Place Order window should be loaded, the Order ID should be generated by the system and the user should place their Order under that Order ID. The order ID cannot be generated randomly and generate the next Order ID accordingly to the last Order ID. Order ID cannot be repeated.
- Customer Contact Number – When entering the Contact Number, it should be validated. Contact Numbers should start with a "0" and must have 10 numbers. If the user has entered an invalid Contact Number (as an example, start without "0" or the Contact Number has more than 10 numbers), the user should be kept prompted until they enter a valid Contact Number.
- T-Shirt Size – There are 6 T-Shirt sizes (XS/S/M/L/XL/XXL). The user should input What T-Shirt size they want and the customer can only place an order in one size at a time.
- Quantity – The user should input the T-Shirt Quantity. Any value greater than 0 can be entered as quantity.
- Amount – The program should calculate the total amount and display it.

- Order Status – There are 3 order statuses. They are PROCESSING / DELIVERING / DELIVERED. The order status should be declared as a static final variable. Declare that static variable as 0 for PROCESSING, 1 for DELIVERING, and 2 for DELIVERED. While the order is being placed, the system should save PROCESSING as the initial status.

After the order is placed successfully, The program should ask the user “Do you want to place this order”, if the user enters “Y” the order details should be saved to the system, and if the user enters “N” the order details should not be saved to the system.

After placing an order successfully, The program should ask the user “Do you want to place another order (Y/N): ”, if the user enters “Y” the user can place a new order again and if the user enters “N” the user can go to the homepage.



```
C:\Windows\SYSTEM32\cmd.exe

Place Order

Enter Order ID : ODR#00009
Enter Customer Phone Number : 0713859855
Enter T-Shirt Size (XS/S/M/L/XL/XXL) : M
Enter QTY : 3
Amount : 2700.00

Do you want to place this order? (y/n) : Y
      Order Placed..!

Do you want to place another order? (y/n) : S
```

Figure 3 – Add Contact successfully

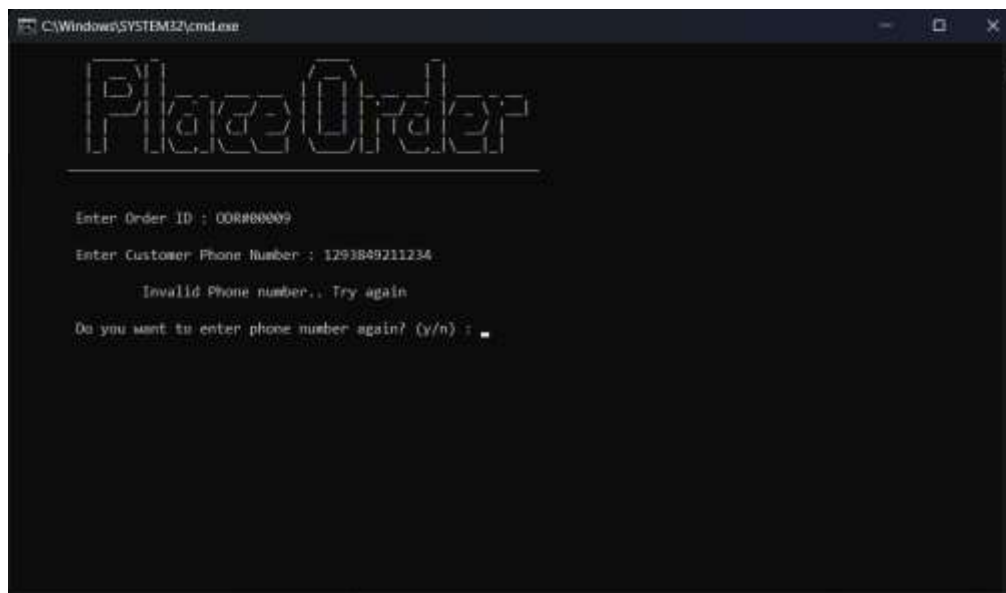


Figure 4 – Invalid phone number

02. Search Customer [Demo](#)

With this option, a customer can be searched from the system. First, the user needs to enter the Customer Contact Number. The contact number should be validated and checked in the system. Otherwise, an error message should be displayed as “Invalid Input”.

If the user enters a contact number that is already in the system, Sizes of all the placed orders with that contact number should be displayed to the user. The sizes should be sorted in the descending order by amount.

Once the information has been displayed, the user should be prompted whether to continue seeking Customer details or return back to the main menu.

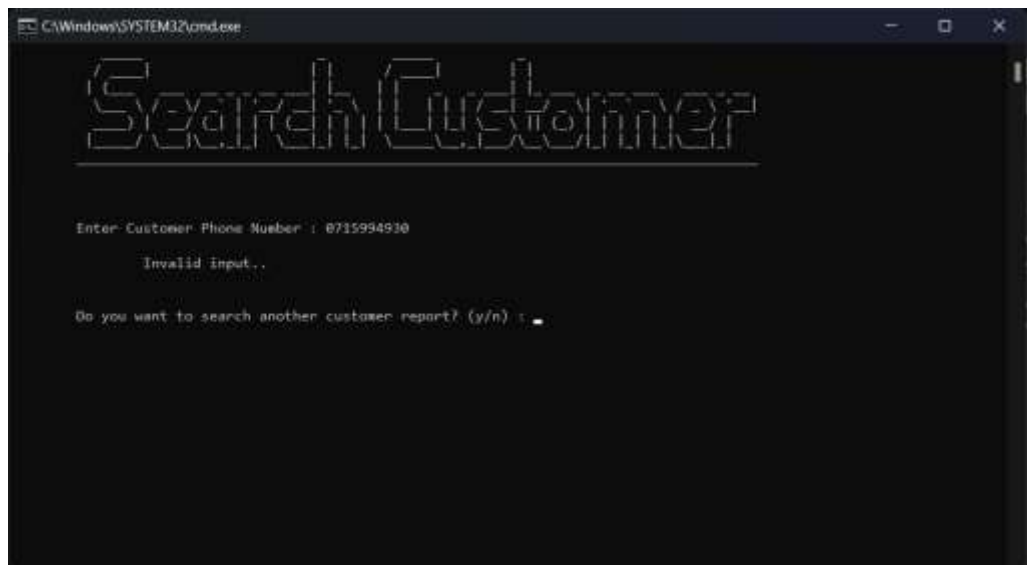


Figure 5 – Invalid Search Contact

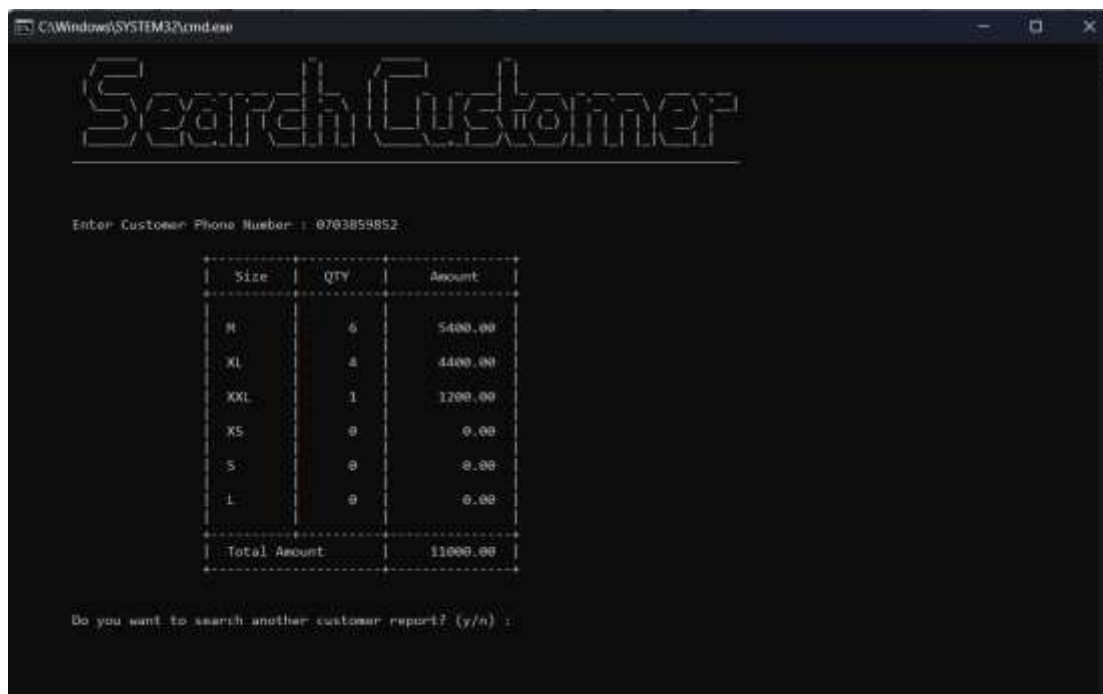


Figure 6 – Search contact successfully

03. Search Order [Demo](#)

With this option, the user can view order details. First, the user needs to enter a valid Order ID. After the user enters the Order ID, the system should search if this Order ID is an existing Order ID or not. If this Order ID hasn't been added yet, then it should be displayed as shown in Figure 7. If the Order ID is correct, display the order details as shown in Figure 8.

Once the information has been displayed, the user should be prompted whether to continue seeking order details or return back to the main menu.

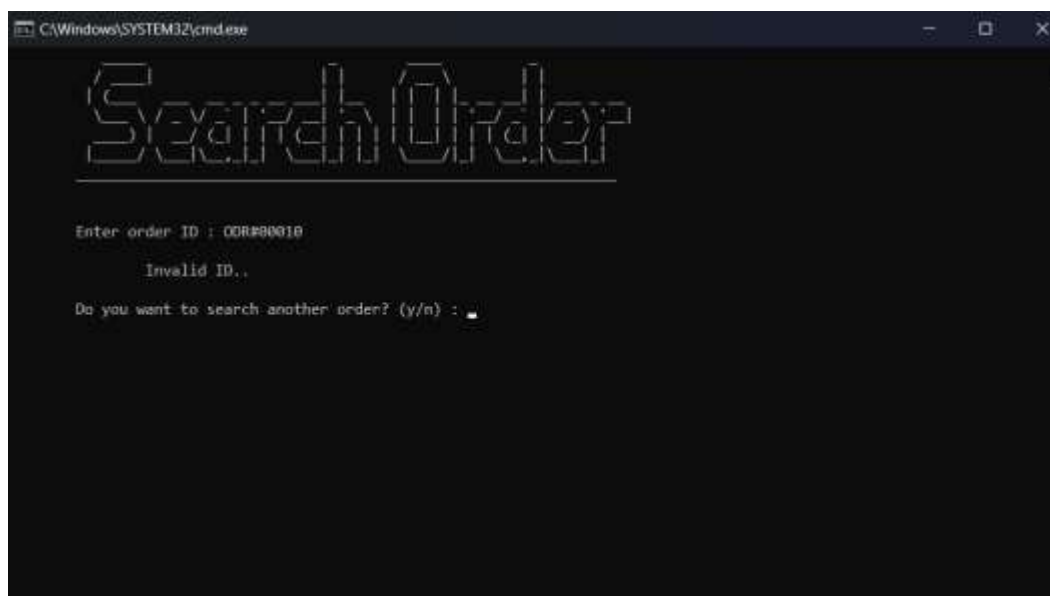




Figure 8 – Search Order (successful)

04. View Reports [Demo](#)

With this option, three other sub options will be displayed for the user to choose from. It should be displayed as follows.

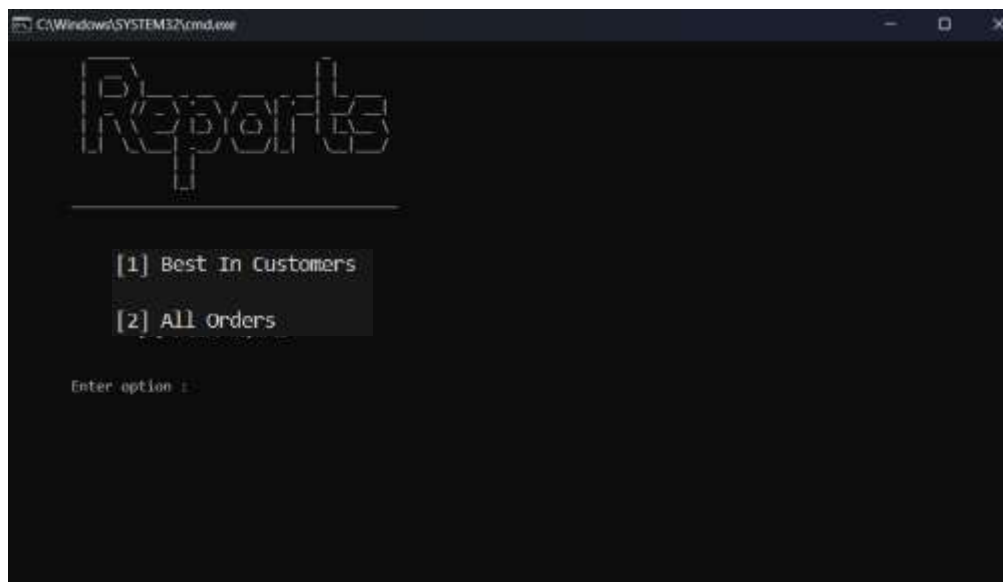


Figure 9 – Report Options

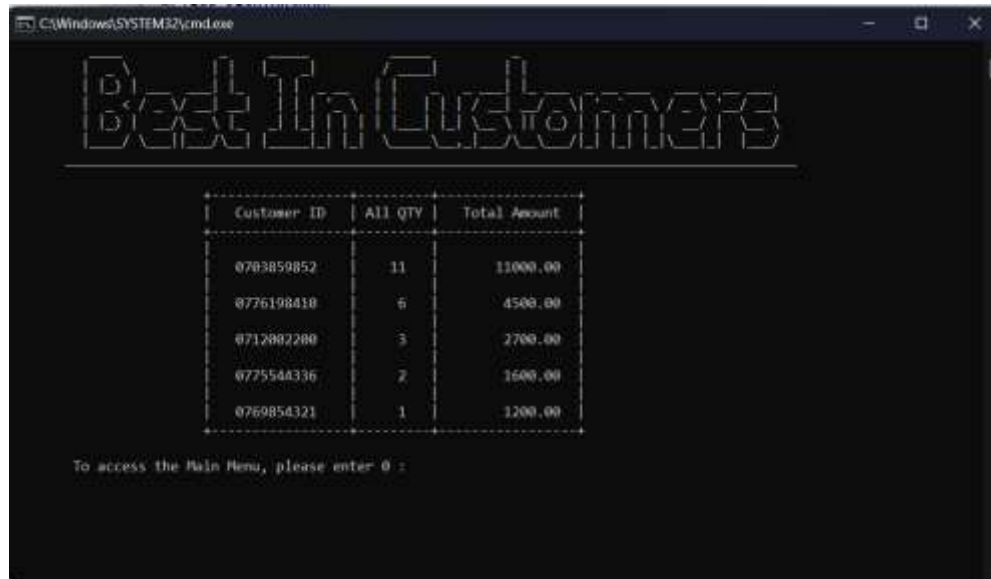
1. Best in Customers

With this option, the program should calculate the customers total amount and print it in descending order.

If the customer has an order that consists of many t-shirt sizes, the program should calculate each amount separately according to the sizes.

The sum of the calculated amounts should be printed as the total amount of the customer.

The Customer ID cannot be repeated in the table.

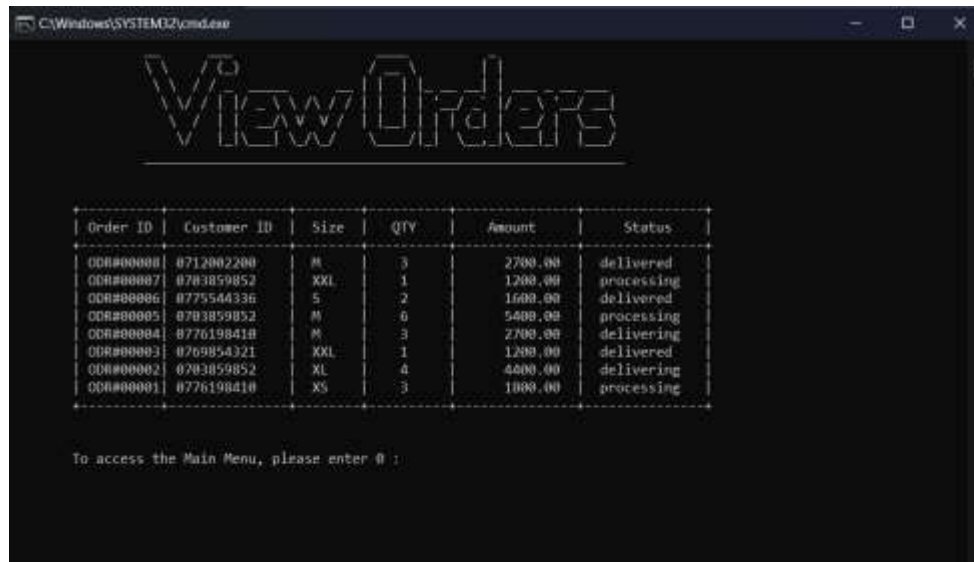


Customer ID	All QTY	Total Amount
0703859852	11	11000.00
0776198410	6	4500.00
0712002200	3	2700.00
0775544336	2	1600.00
0769854321	1	1200.00

To access the Main Menu, please enter 0 :

2. All Orders

In this option, all the orders should be displayed. All orders should be displayed in the descending order by Order ID.



Order ID	Customer ID	Size	QTY	Amount	Status
ODR#000008	0712002200	M	3	2700.00	delivered
ODR#000007	0703859852	XXL	1	1200.00	processing
ODR#000006	0775544336	S	2	1600.00	delivered
ODR#000005	0703859852	M	6	5400.00	processing
ODR#000004	0776198410	M	3	2700.00	delivering
ODR#000003	0769854321	XXL	1	1200.00	delivered
ODR#000002	0703859852	XL	4	4400.00	delivering
ODR#000001	0776198410	XS	3	1800.00	processing

To access the Main Menu, please enter 0 :

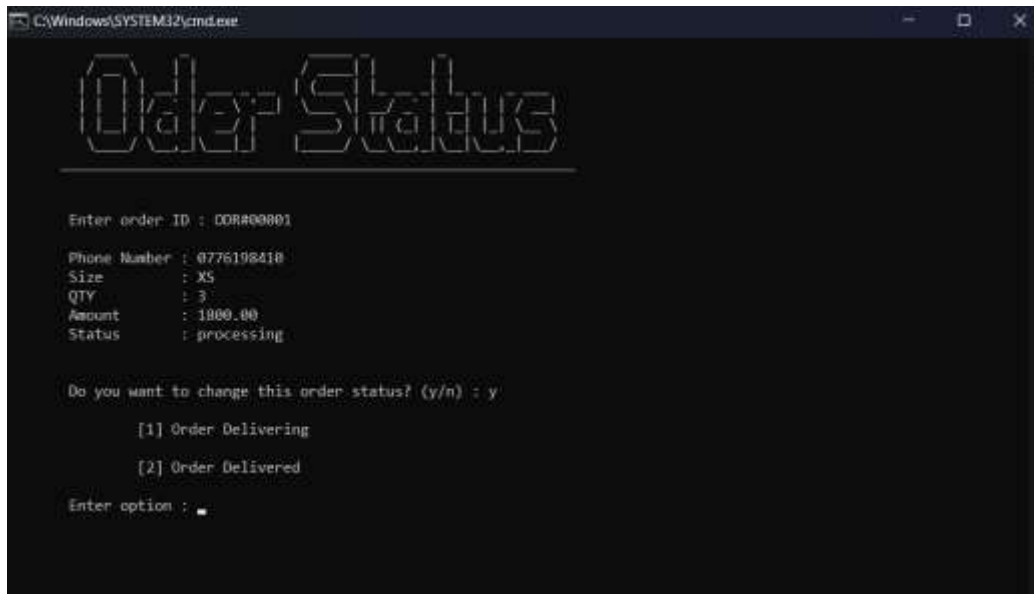
Figure 18 – All Orders

05. Change Order Status [Demo](#)

In this option, the user is required to enter the Order ID.

The option will display the order details.

The user is asked for confirmation to change the order status, if the user proceeds the Order Status will change according to the current status. No change will be made if the current status is “delivered”. Refer the demo video for more information.



```
C:\Windows\SYSTEM32\cmd.exe

Order Status

Enter order ID : DDR#00001

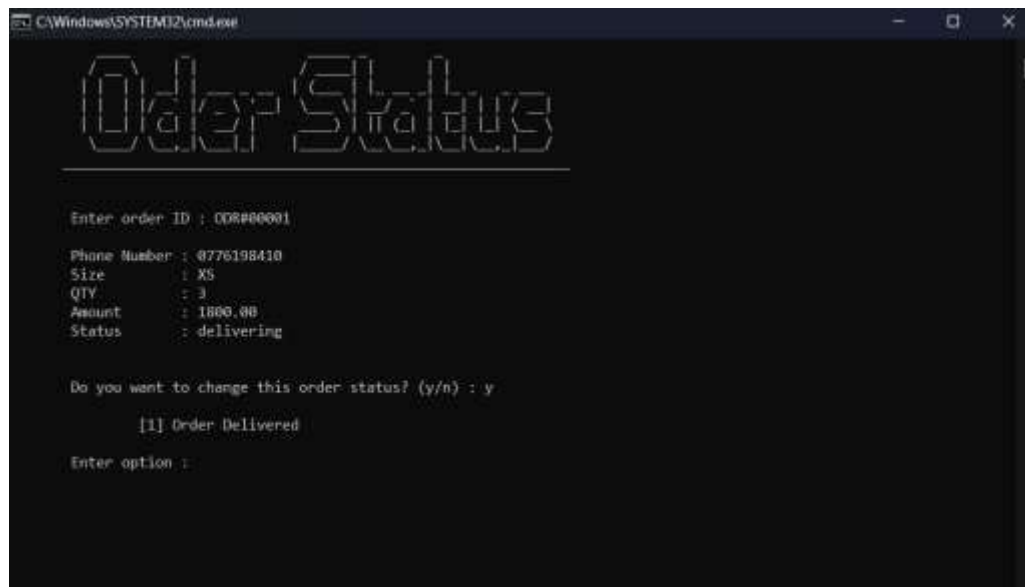
Phone Number : 0776198410
Size : XS
QTY : 3
Amount : 1800.00
Status : processing

Do you want to change this order status? (y/n) : y

[1] Order Delivering
[2] Order Delivered

Enter option : 
```

Figure 20 – Order Status Processing



```
C:\Windows\SYSTEM32\cmd.exe

Order Status

Enter order ID : DDR#00001

Phone Number : 0776198410
Size : XS
QTY : 3
Amount : 1800.00
Status : delivering

Do you want to change this order status? (y/n) : y

[1] Order Delivered

Enter option : 
```

Figure 21 – Order Status Delivering

```
C:\Windows\SYSTEM32\cmd.exe

Order Status

Enter order ID : 008#80001

Phone Number : 0776198410
Size : XS
QTY : 3
Amount : 1800.00
Status : delivered

Can't change this order status, Order already delivered..!

Do you want to change another order status? (y/n) : _
```

Figure 22 – Order Status Delivered

06. Delete Order [Demo](#)

With this option, an order can be deleted from the system. First, the user needs to enter the order ID. The order ID should be validated. otherwise, it should be handled like previously.

If the user has entered a valid order ID but that order still hasn't been added to the system, then it should be notified.

If the user enters an order ID that is already in the system, all the related details of contact should be shown to the user. Before deleting the contact, a confirmation should be taken from the user to delete this order. If the user confirms that this order will be deleted, the order should be successfully deleted. If the user does not confirm, the order should not be deleted.

```
C:\Windows\SYSTEM32\cmd.exe

Delete Order

Enter order ID : 008#80001

Phone Number : 0776198410
Size : XS
QTY : 3
Amount : 1800.00
Status : delivered

Do you want to delete this order? (y/n) : Y

Order Deleted..!

Do you want to delete another order? (y/n) : _
```

Figure 23 – Delete Order

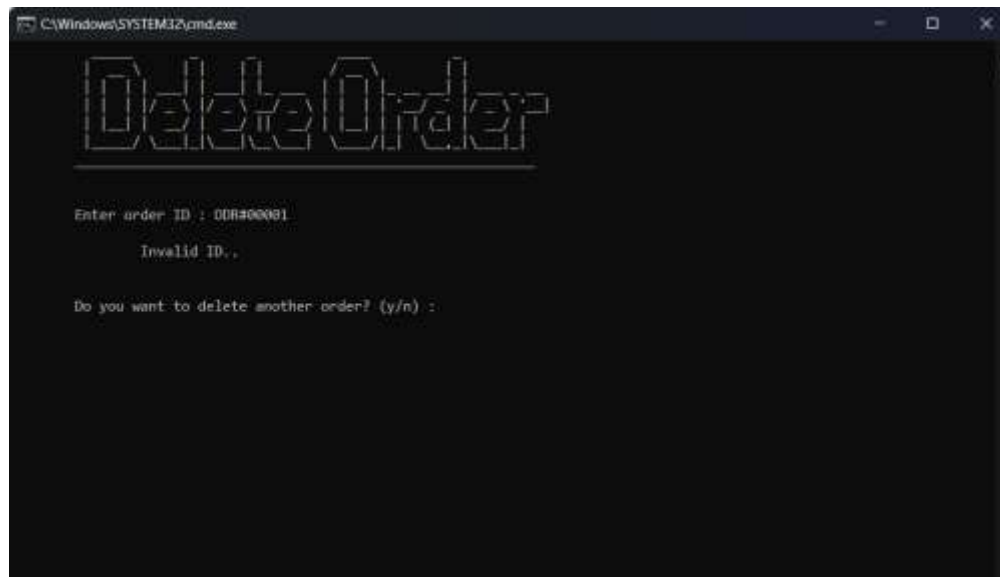


Figure 24 – Invalid Delete Order

Guidelines

- Refer to the Coursework Guidelines at the end to understand the specific guidelines to be followed when developing the project required.
- You should use your knowledge of flow control, loops, methods, and arrays to implement this coursework.
- You can't create classes except for the class that holds the main method and you can create as many methods as you wish and can't use Arrays class method.
- Use the Scanner class to get input from the command-line interface.
- All validations that have been mentioned in this document should be implemented in your program.
- these two lines of code are used to clear the last few lines of output in the console. The first line moves the cursor up lines, and the second line clears those lines and any subsequent lines that were printed after them. You can change the highlighted number for clear lines according to your need.

```
// Move the cursor up five lines
System.out.print("\033[5A");
// Clear the lines
System.out.print("\033[0J");
```

- The code to clear the command line from inside a Java application is as follows. You can use this code when you need to clear the command line.

```
public final static void clearConsole() {
    try {
        final String os = System.getProperty("os.name");
        if (os.contains("Windows")) {
            new ProcessBuilder("cmd", "/c",
"cls").inheritIO().start().waitFor();
        } else {
            System.out.print("\033[H\033[2J");
            System.out.flush();
        }
    } catch (final Exception e) {
        e.printStackTrace();
        // Handle any exceptions.
    }
}
```

- You can create as many methods as you wish in the only class that you have.

- Demo Videos are given at relevant places for you to understand the coursework requirement better and Demo videos may help you to clarify your doubts to some extent.
- Submission - Your work is to be made into a folder with the folder name format [BatchNumber_ Name], and submit on or before the deadline.
- If you still have any questions, feel free to question.