Online Takeaway system

Prepared and submitted by SIBI AKASH M

2018103594

N BATCH

**Objective:**

Online food takeaway system for a particular restaurant provide customers the liberty of placing orders online so that they could save the time of waiting at the restaurant.

**Functionalities:**

* Maintains the data of:
  + Food items present in the restaurant menu.
  + Customer Details.
  + Order details of a particular order.
  + Orders taken by the restaurant.
* Provides an interface that allows to place orders.

**Project Details:**

**Front End:** Html, css

**Back End: Nodejs, Express.js**

**SQL: MySql 8.0 server**

**Sql tools: MySql** Command **Line 8.0**

**RELATIONS UNDER THE DATABASE:**

**1.USER LOGIN:**

This table maintains the data about the user login portal of this delivery system.

Attributes:

* + - **Mobile\_No (Primary Key)**
    - Password
    - Customer\_FName

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**2.CUSTOMER DETAILS:**

This table maintains the data about the customer referenced by unique UserId.

Attributes:

* + - **Mobile\_No (Primary Key)**
    - Customer\_FName
    - Customer\_LName
    - Bdate

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**3.FOOD ITEM:**

This table maintains the data about the food items in the menu.

Attributes:

* + - Price
    - **Item\_Name (Primary key)**
    - Category

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**4.ORDERS TAKEN:**

This table maintains the data the orders received in the database.

Attributes:

* + - **OrderID (Primary Key)**
    - Customer\_FName
    - Customer\_LName
    - Status\_Id
    - Order\_date
    - Order\_time
    - Total\_amount

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**5.STATUS:**

This table maintains the data about the status of the orders received in the database.

Attributes:

* + - **Status\_Id (Primary Key)**
    - Status

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**6.ORDER DETAILS:**

This table maintains the data about order invoice contents received in the database.

Attributes:

* + - **OrderID (Primary Key)**
    - **Item\_Name (Primary key)**
    - Quantity

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form.**

**7.DELIVERY ADDRESS:**

This table maintains the data about the delivery address of customers.

Attributes:

* + - **Mobile\_No (Primary Key)**
    - Address\_type
    - House\_No
    - Street
    - Area
    - Pincode
    - Landmark

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**8.MEMBERSHIP:**

This table maintains the data about the membership status of customers

Attributes:

* + - **Mobile\_No (Primary Key)**
    - Membership\_status

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form**

**9.FEEDBACK:**

This table maintains the data about the feedback given by the customers who have got their food delivered.

Attributes:

* + - **OrderID (Primary Key)**
    - Delivery\_feedback
    - Food\_feedback
    - Suggestions

All attributes are non-transitively determined by the primary Key (Super key). Hence it is in **Boyce-Codd Normal Form.**

**10.Promo:**

This table maintains the order ids which have availed promo code.

Attributes:

* + - **OrderID (Primary Key)**

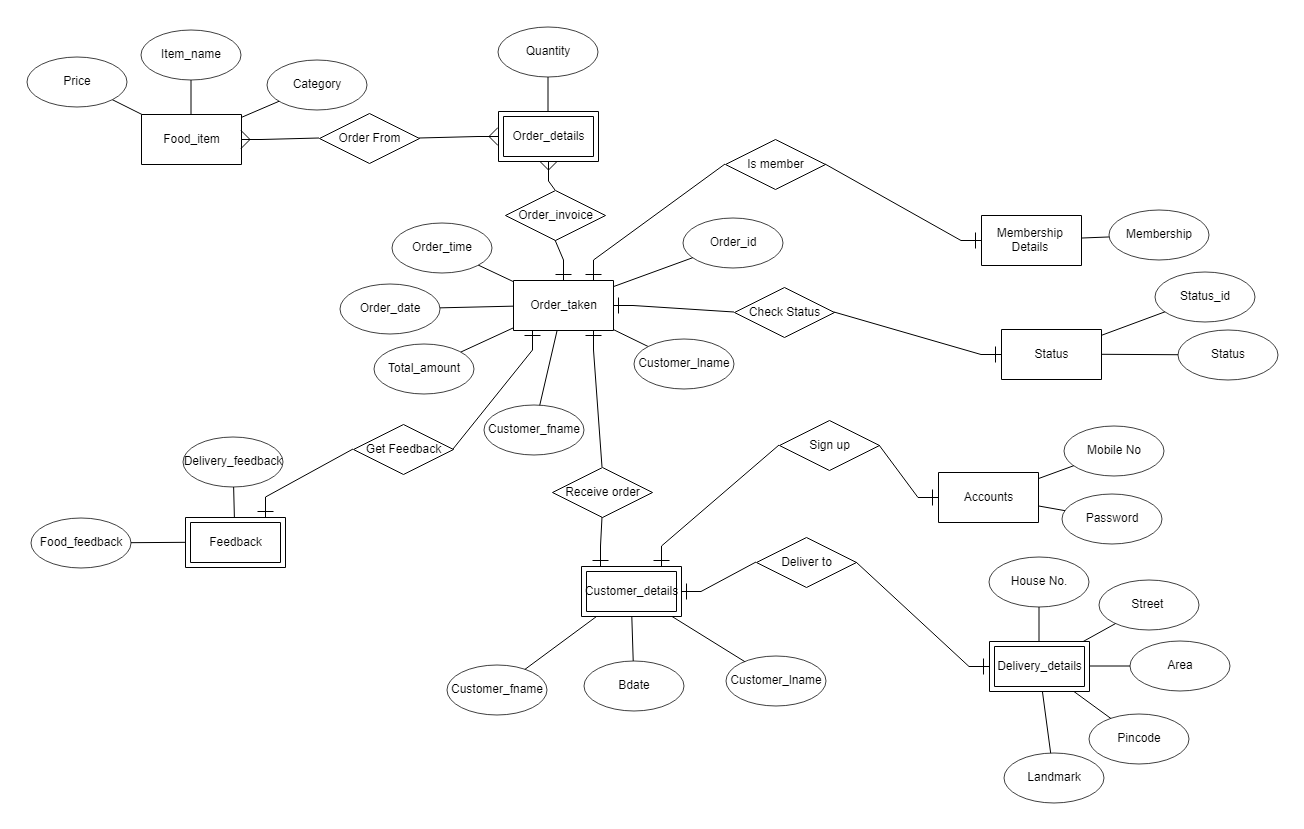
**11.Membership avail:**

This table maintains all order ids and membership users will be granted a 40% discount with a suitable trigger.

Attributes:

* + - **OrderID (Primary Key)**

**ER-DIAGRAM:**



**RELATIONAL SCHEMA:**



**Triggers used:**

**1. Cancel order Trigger: //Deletes order details from order\_details table**

CREATE TRIGGER CANCEL\_ORDER BEFORE INSERT ON cancel1

FOR EACH ROW

BEGIN

DELETE FROM ORDER\_DETAILS WHERE order\_details.ORDER\_ID = NEW.ORDER\_ID;

END;

/

**2.Set status Cancelled: // Sets the status of the cancelled order as ‘canc’**

CREATE TRIGGER SET\_CANCEL BEFORE INSERT ON cancel1

FOR EACH ROW

BEGIN

UPDATE ORDER\_TAKEN SET STATUS\_ID='canc' WHERE ORDER\_TAKEN.ORDER\_ID = NEW.ORDER\_ID;

END;

/

**3.Membership offer avail : // Applies an instant 40% discount on the total bill amount for membership orders.**

CREATE TRIGGER APPLY\_OFF BEFORE INSERT ON mem\_avail

FOR EACH ROW

BEGIN

DECLARE amount mediumint;

DECLARE x varchar(4);

select mem\_status into x from mem\_status where mobileno = (SELECT mobileno from order\_taken where ORDER\_TAKEN.ORDER\_ID = NEW.ORDER\_ID);

IF x = 'yes' THEN

SET amount = mem\_reward(NEW.ORDER\_ID);

CALL UPD\_AMT(NEW.ORDER\_ID,amount);

END IF;

END;

/

**Functions used:**

**Function used to apply membership discount:**

CREATE FUNCTION mem\_reward(O\_ID varchar(4))

RETURNS mediumint

DETERMINISTIC

BEGIN

DECLARE amount mediumint;

SELECT TOT\_AMT INTO amount FROM ORDER\_TAKEN WHERE ORDER\_ID= O\_ID;

RETURN 0.6\*amount;

END;

**Procedures used:**

**1.Procedure to apply promo discount on the total amount:**

CREATE PROCEDURE promo(IN O\_ID varchar(4),IN TOT\_AMT mediumint)

BEGIN

DECLARE new\_amt mediumint;

SET new\_amt = TOT\_AMT - 0.3\*TOT\_AMT;

UPDATE ORDER\_TAKEN SET TOT\_AMT = new\_amt WHERE ORDER\_ID = O\_ID;

END;

//

**2.Procedure to select total amount of an order:**

CREATE PROCEDURE UPD\_AMT(IN O\_ID varchar(4),IN new\_amt mediumint)

BEGIN

UPDATE ORDER\_TAKEN SET TOT\_AMT = new\_amt WHERE ORDER\_ID = O\_ID;

END;

//

**3.Procedure to select total amount of an order:**

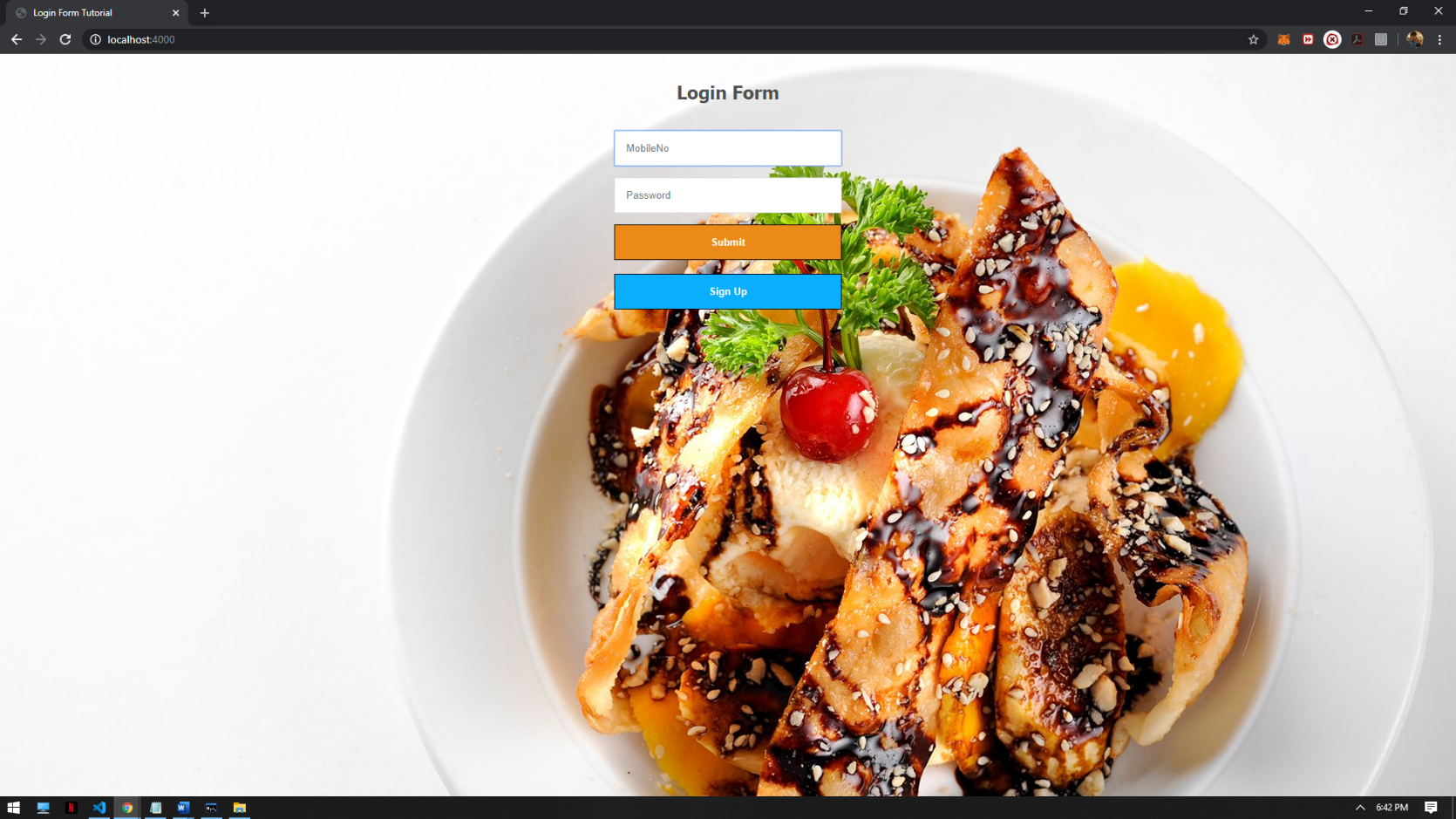
CREATE PROCEDURE SELECT\_AMT(IN O\_ID varchar(4))

BEGIN

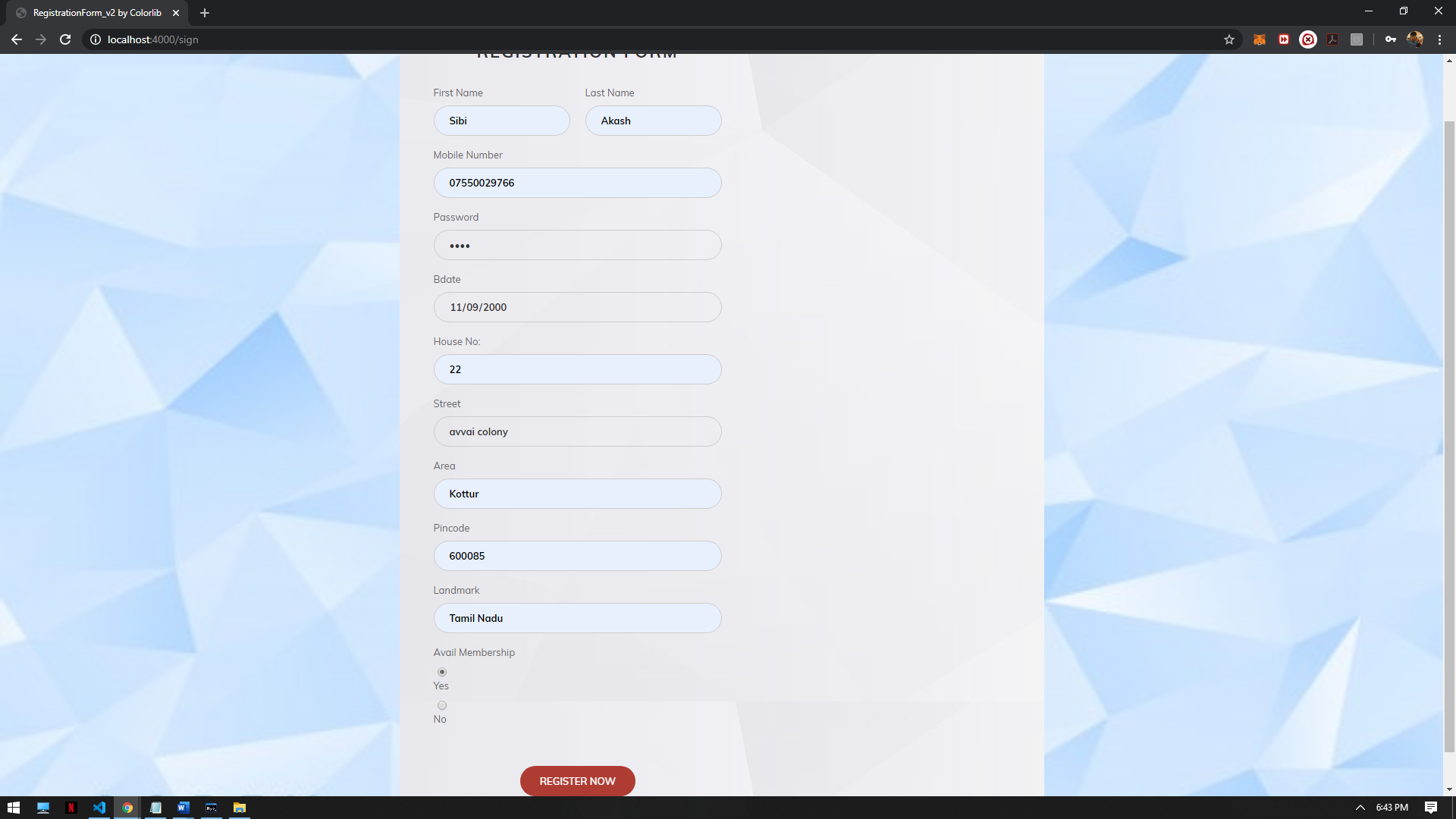
SELECT TOT\_AMT FROM ORDER\_TAKEN WHERE ORDER\_ID = O\_ID;

END;

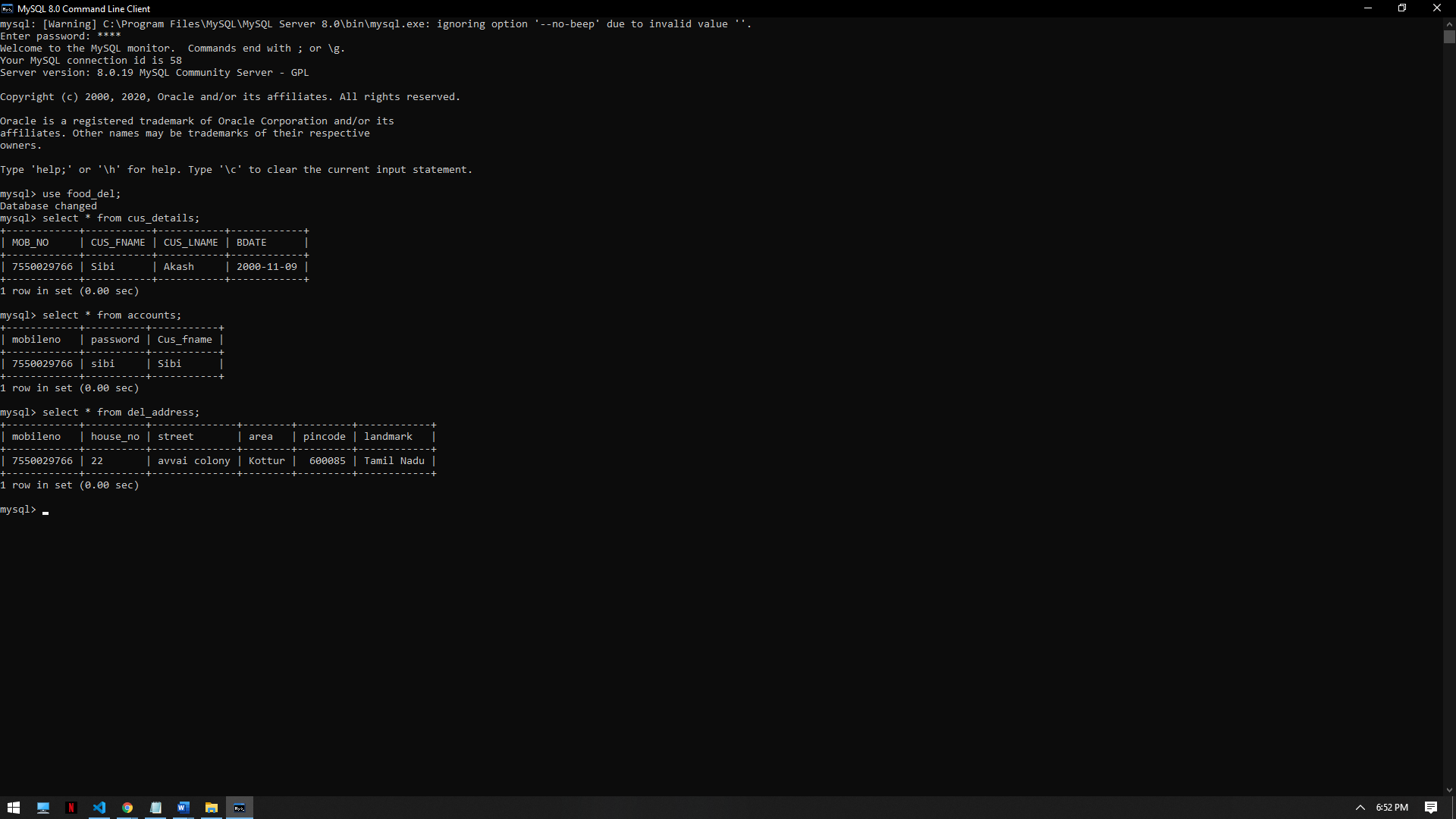
**Screenshots:**



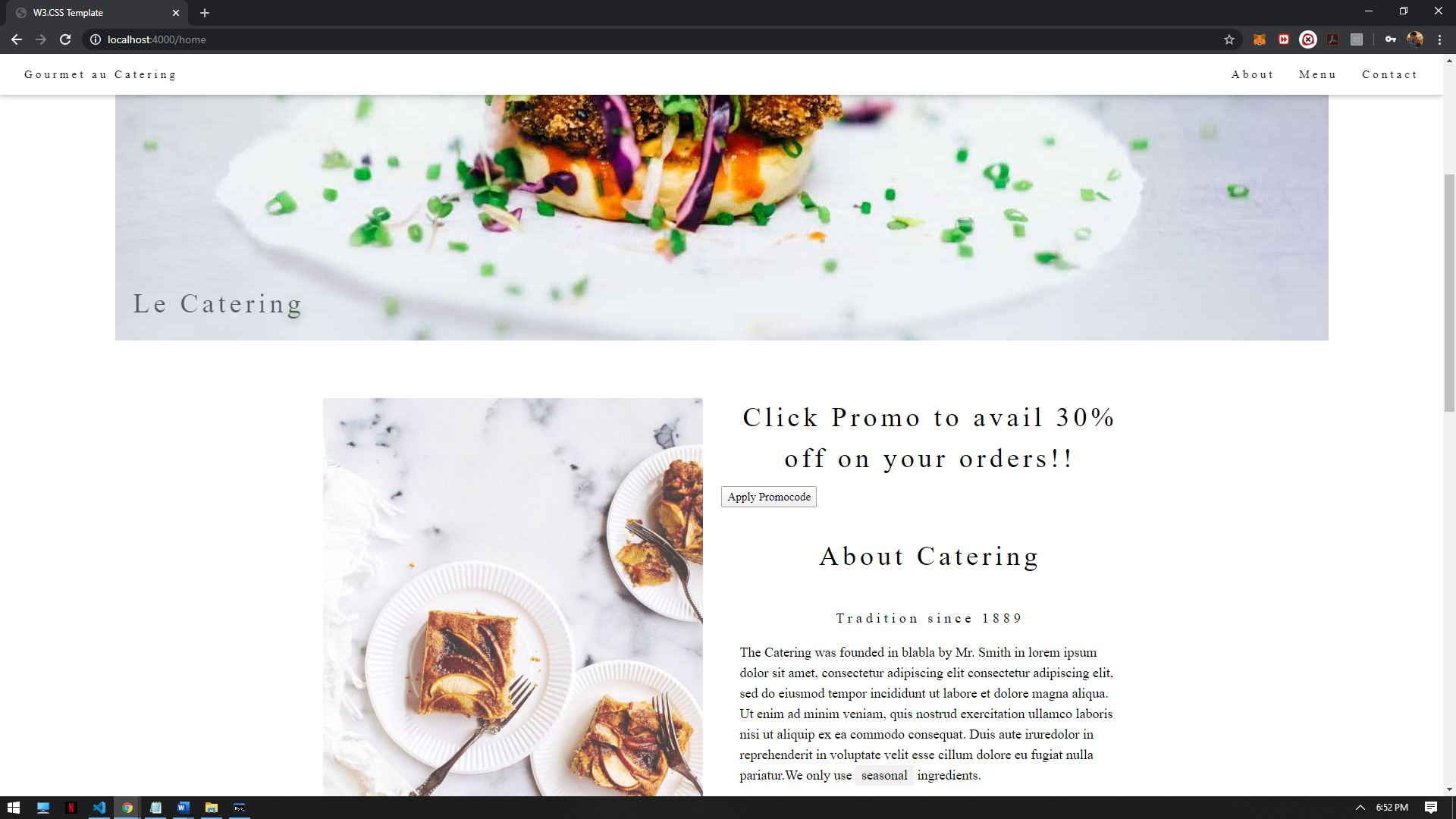
This is the login Screen , We are going to sign up and enter our details.



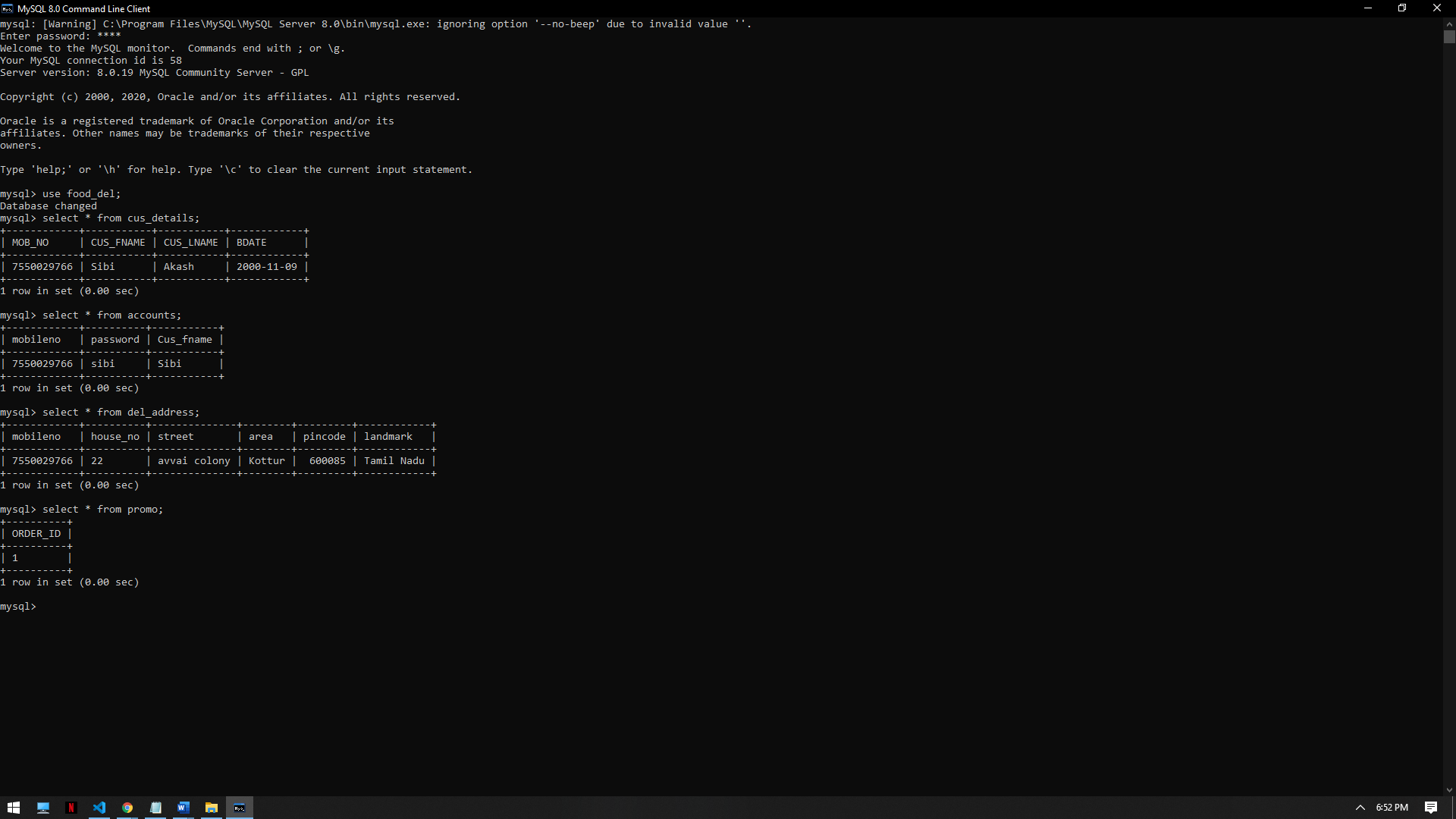
In this registration page , we enter our details so that we can login to this site.



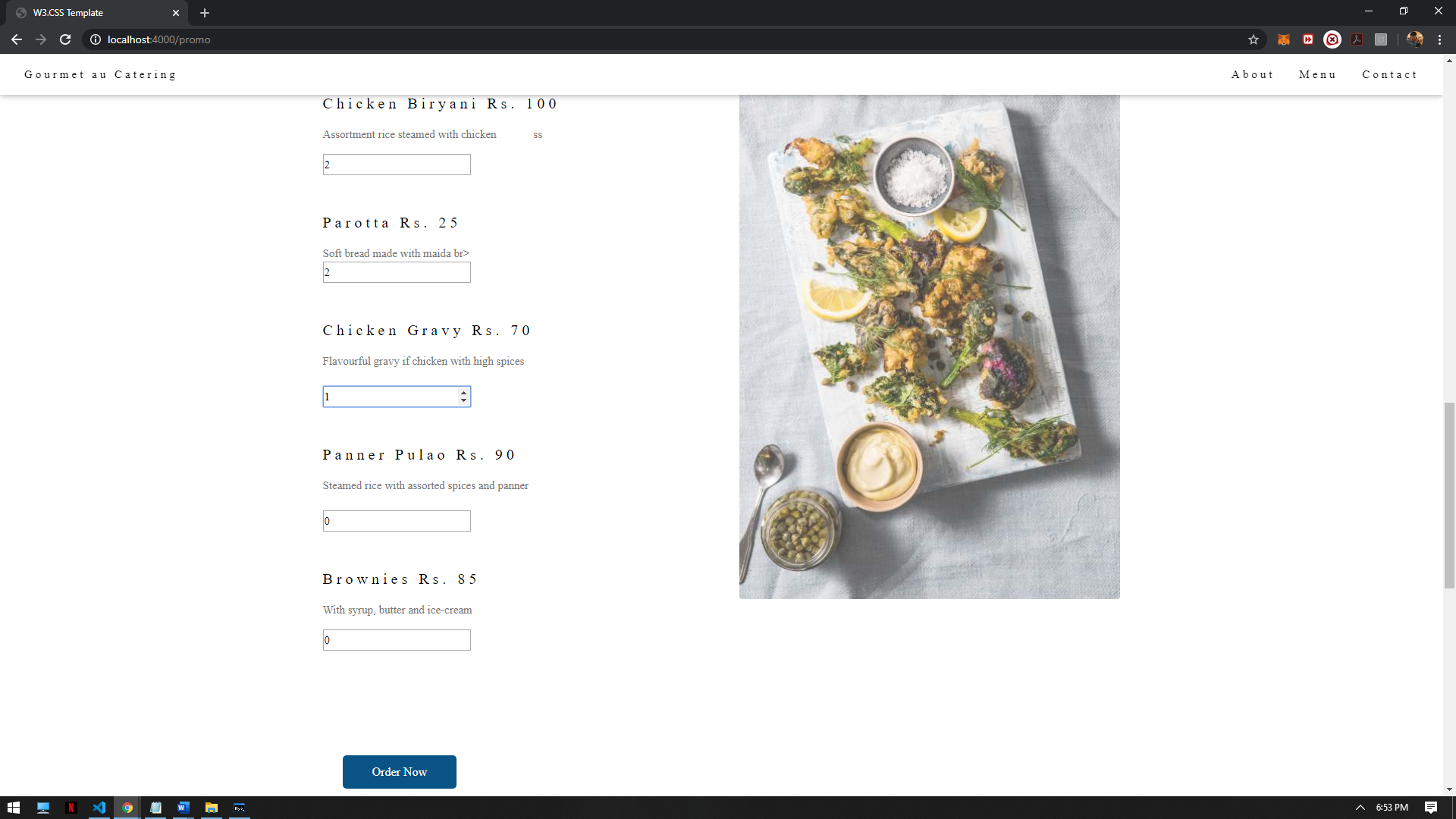
As you can see , the details enterd have been inserted into the respective tables



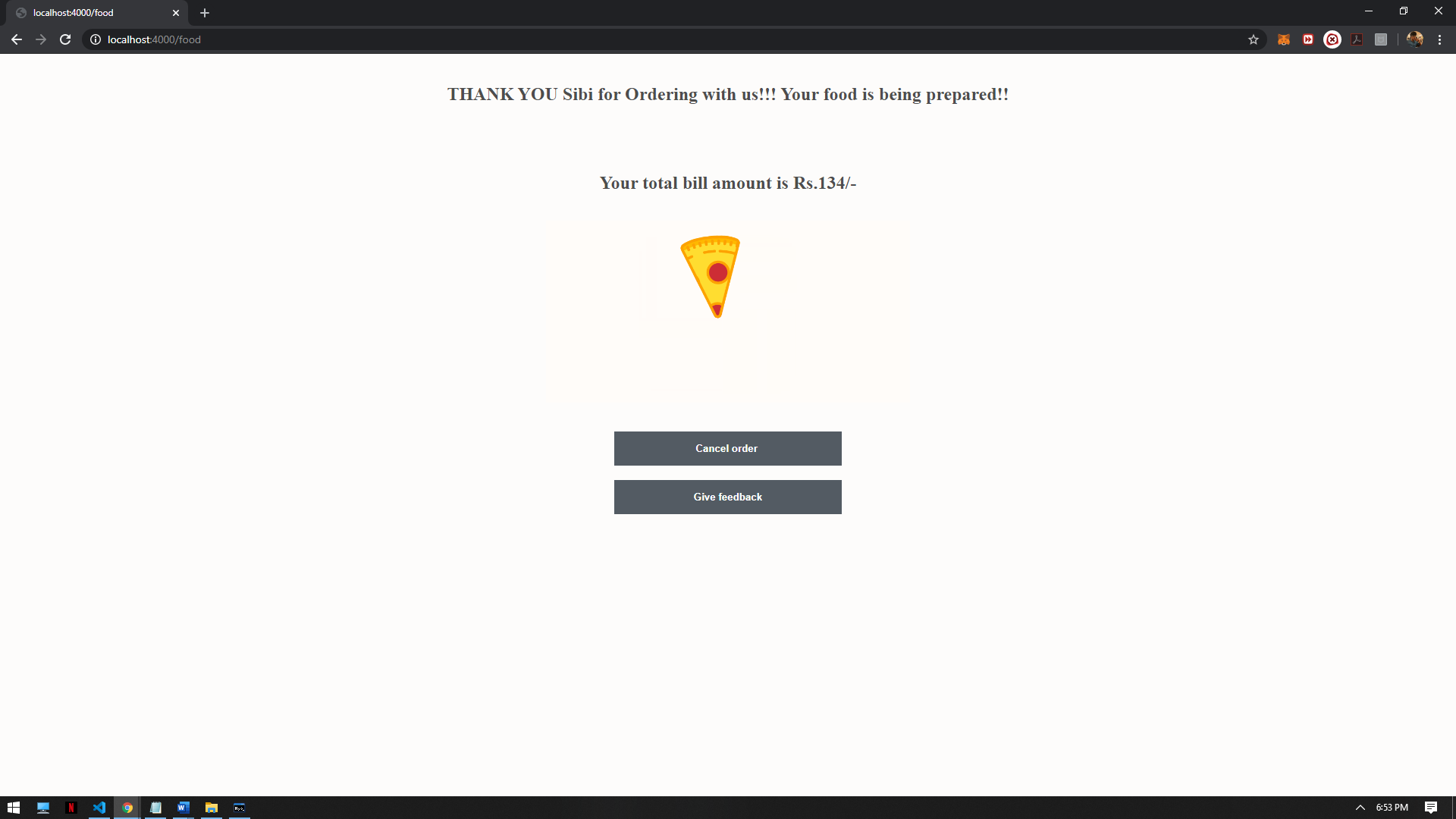
We see this page after login, we can click the Apply promocode to apply 30% offer.



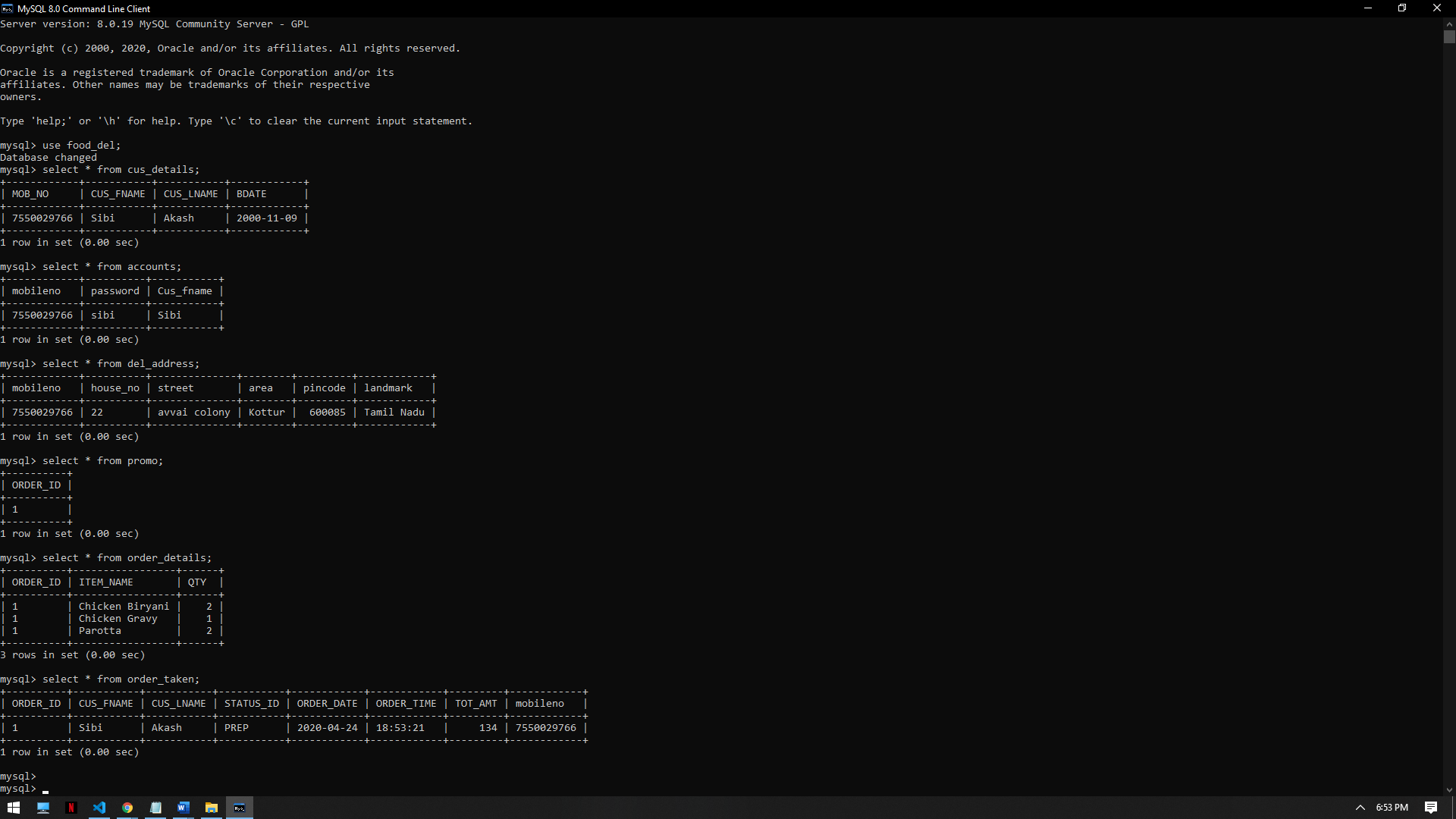
This order\_id has now got inserted into Promo table on clicking the button.



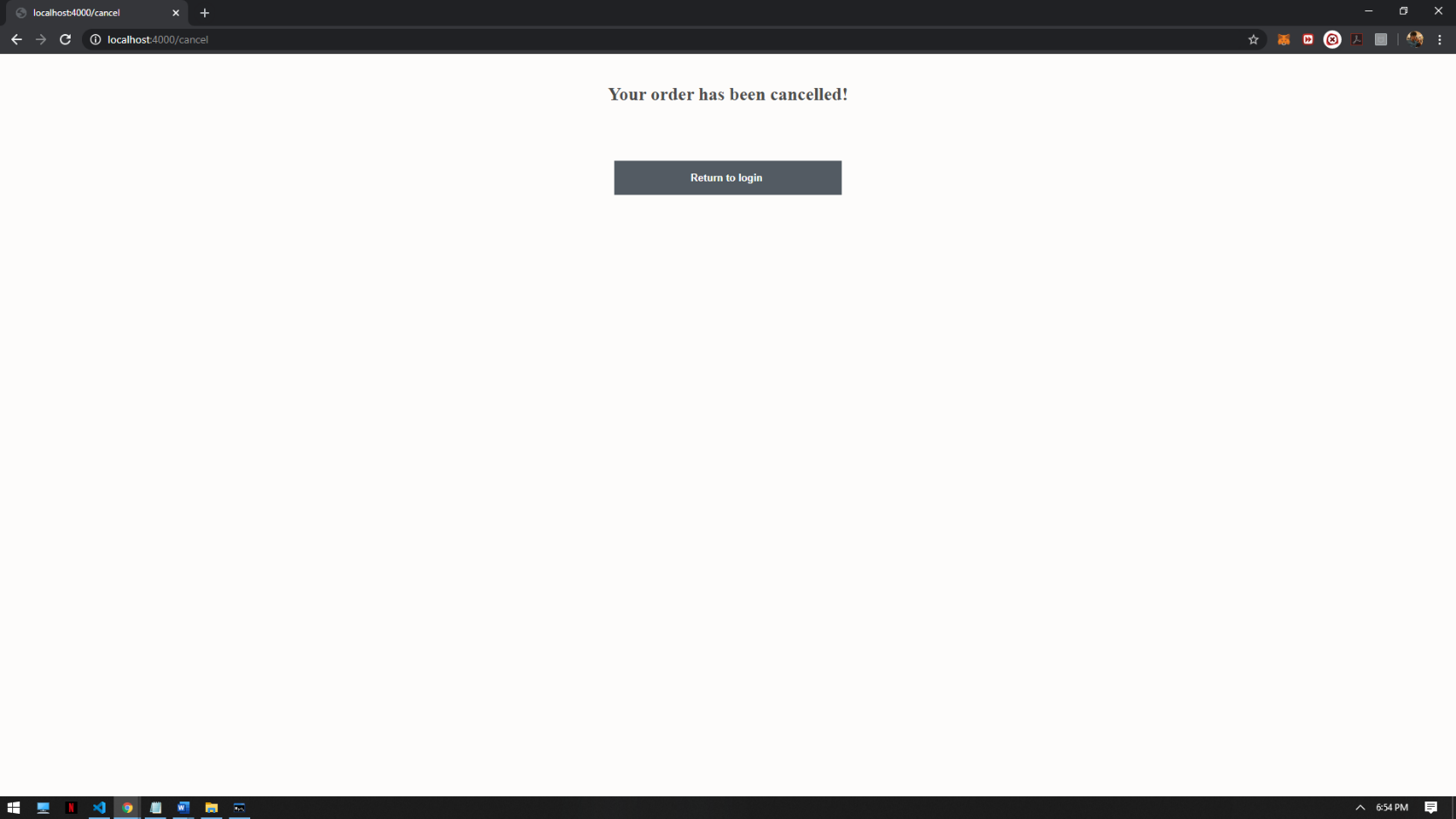
We can now enter the order details by choosing the menu.



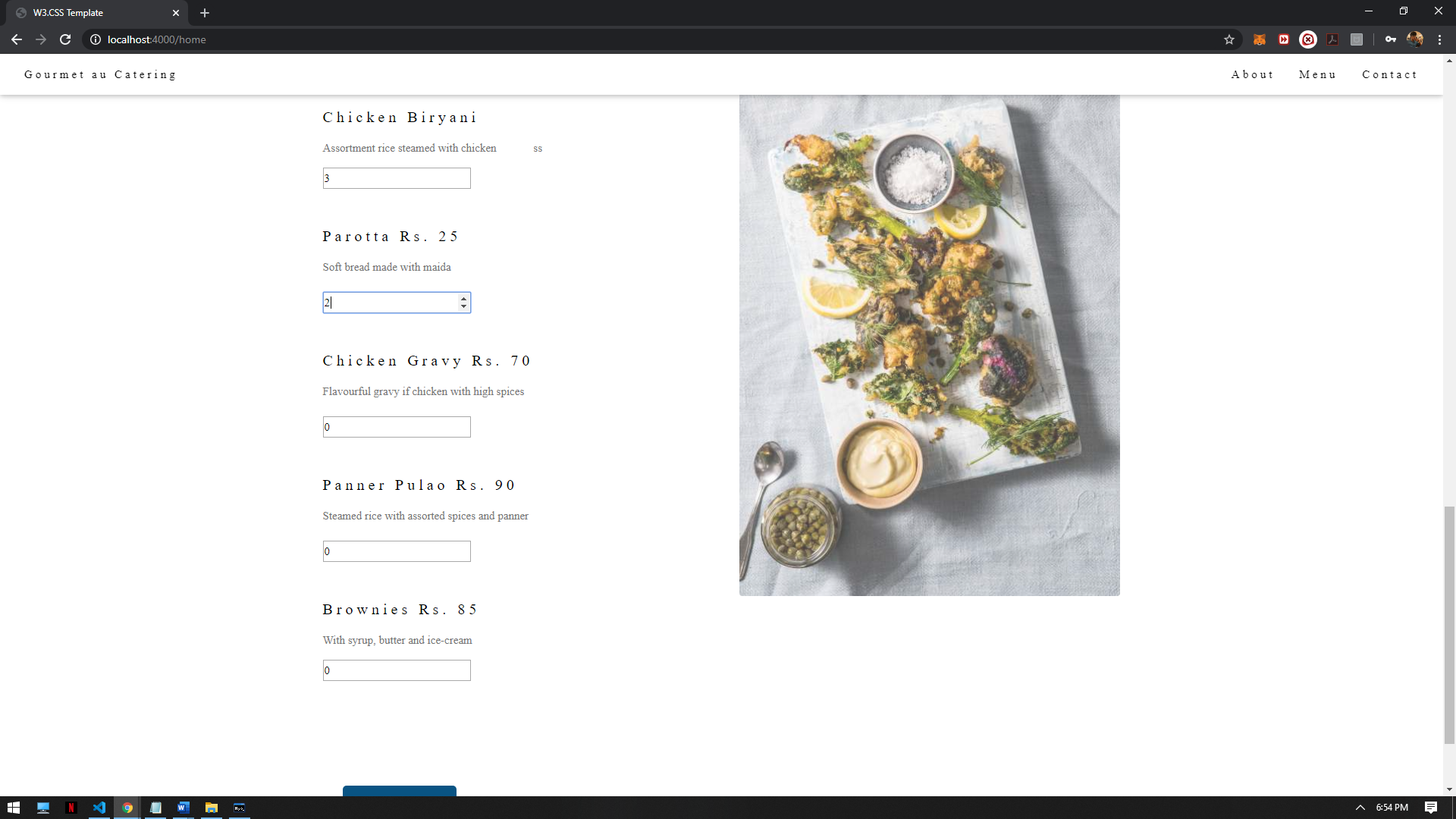
After submission , we enter this page where we can see that the order is accepted.



We can see the order details being inserted to tables. We can also see that promo and membership triggers working thus reducing the total price.

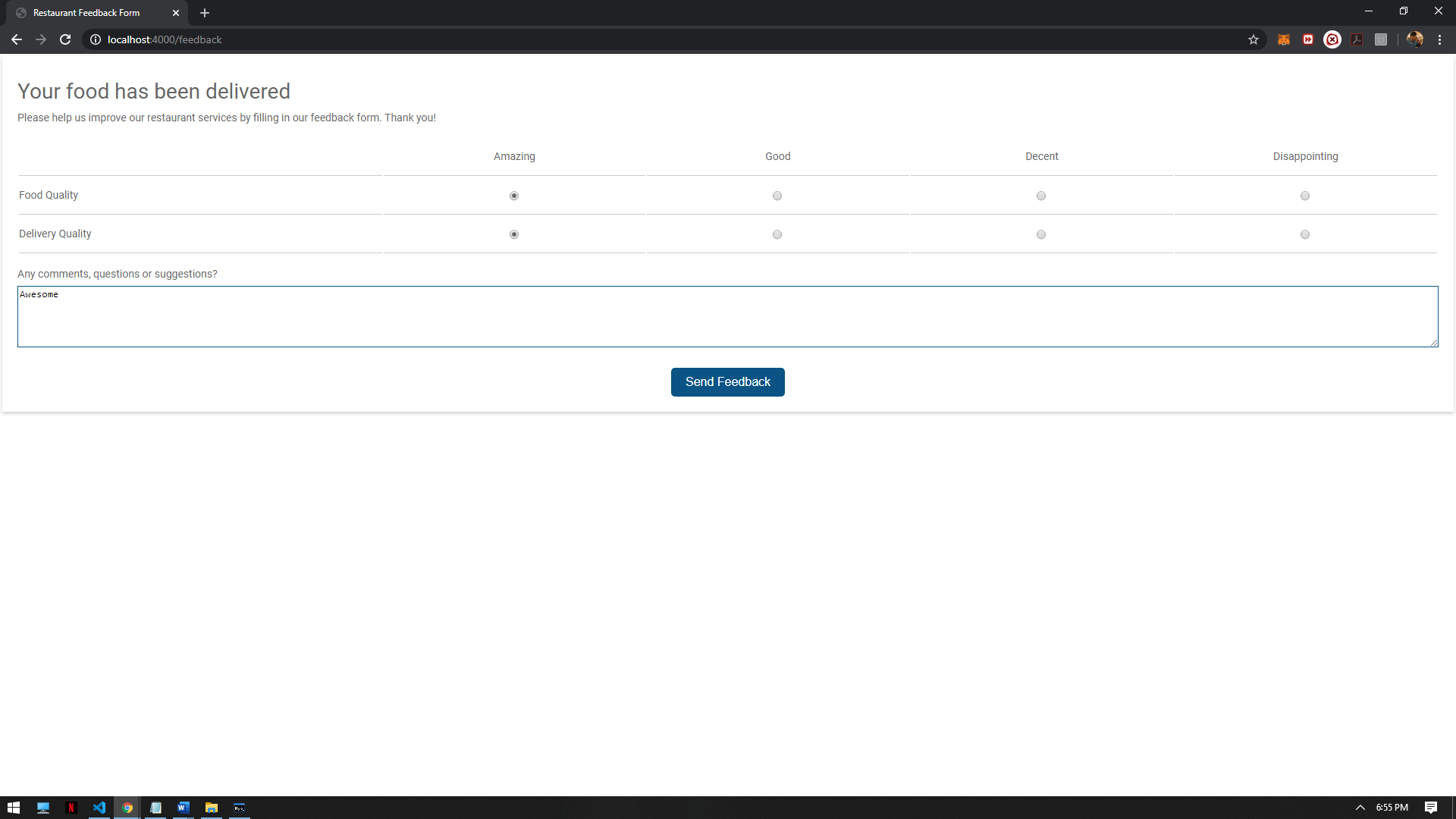


Now we click cancel Button to see that the order has been cancelled.

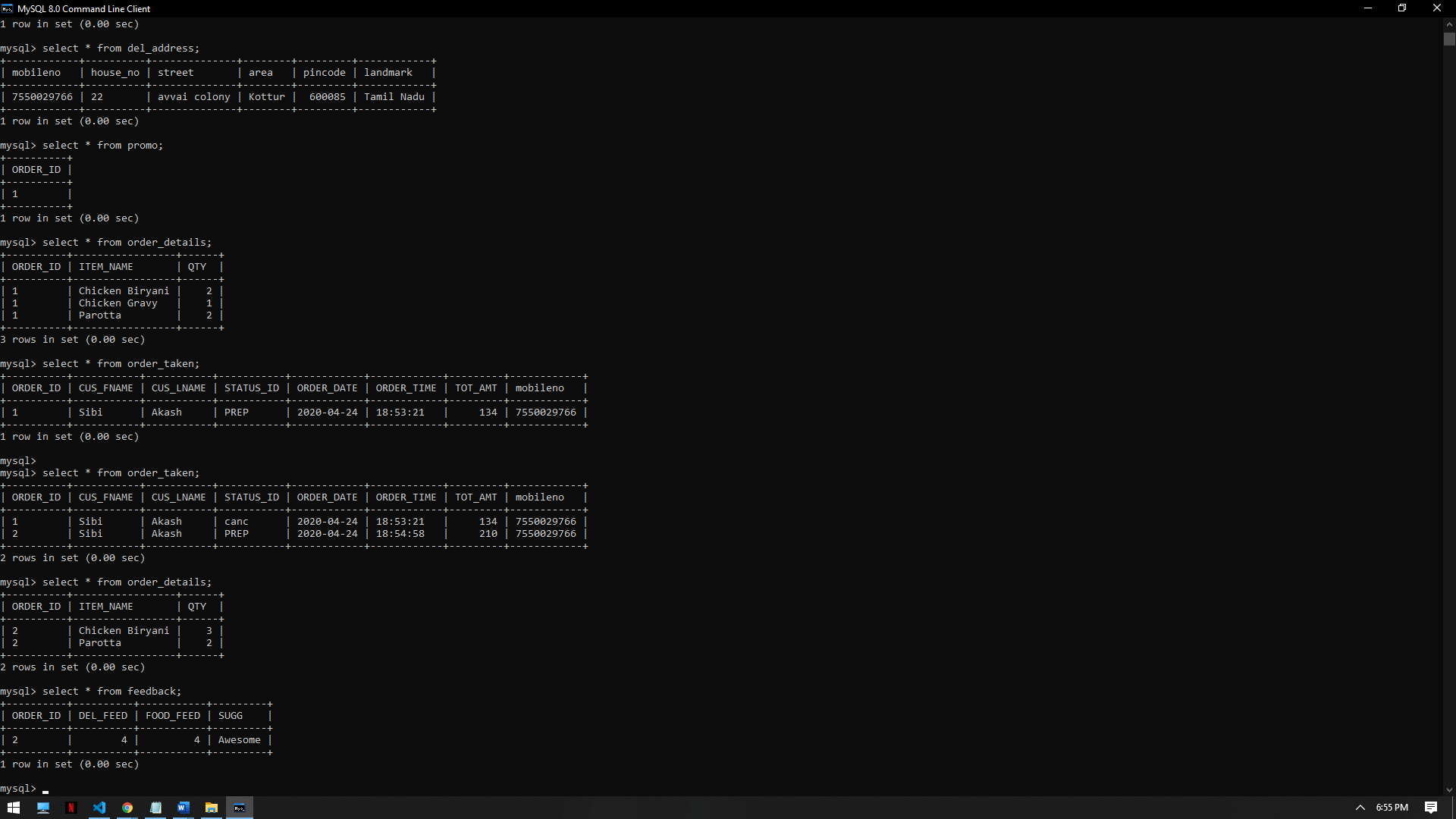


Now we order some new items with another Order\_id and time we are ready to give feedback which is the final part of this webage.





We enter the feedback and then on submission we are redirected to the login page.



Here you can see the 1st order being cancelled and then 1st order details deleted . We can also see that that feedbackof 2nd order being inserted in feedback table.