Table of Contents

[1. Name of database: 2](#_Toc468873185)

[2. Name of .sql file: 2](#_Toc468873186)

[3. Explanation: 2](#_Toc468873187)

[4. INSERT element: 3](#_Toc468873188)

[5. UPDATE element: 4](#_Toc468873189)

[6. DELETE element: 6](#_Toc468873190)

[7. SELECT element: 7](#_Toc468873191)

[8. PROCESSING: 9](#_Toc468873192)

[9. View of Data in database 11](#_Toc468873193)

[9.1 Subscribers Table: 11](#_Toc468873194)

[9.2 Cuisines Table: 11](#_Toc468873195)

[9.3 Sittings Table: 12](#_Toc468873196)

[9.4 TableInfos Table: 12](#_Toc468873197)

[9.5 Customers Table: 13](#_Toc468873198)

[9.6 Reservations Table: 13](#_Toc468873199)

Dynamic Web Open Source Approach Project Documentation

## Name of database:

RestaurantSys

## Name of .sql file:

RestaurantDB.sql

## Explanation:

The purpose of this project is to redesign an Asian restaurant’s website which owned by my family and the original website has been referenced in both html and php files.

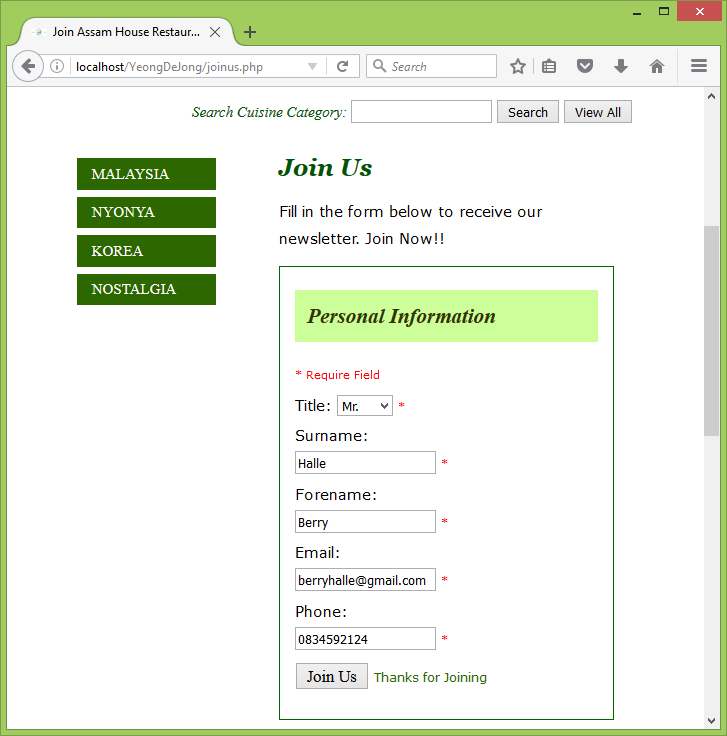
Throughout all web pages, users can search the cuisine category. When users search is invalid, an error message will display to notify users that the category searched is not found. Users can also view all the available cuisines by clicking the ‘view all’ button throughout all web pages. Besides that, users can also receive our recent newsletter/promotion by filling up the ‘Join Us’ form in ‘Join Us’ page and the details are saved in ‘Subscribers’ tables in the database. If users want to update their personal information, users can search their email in ‘Update Personal Information’ form in ‘Join Us’ page and the system will retrieve their personal information from database and display in the textbox which allow users to update their personal information. When users click the ‘update’ button, the latest information will rewrite the current information in the database. If users want to stop subscribing/joining the newsletter, users can unsubscribe by typing in their email in ‘Unsubscribe’ form in ‘Join Us’ page. A confirmation message will be displayed if users unsubscribe successfully.

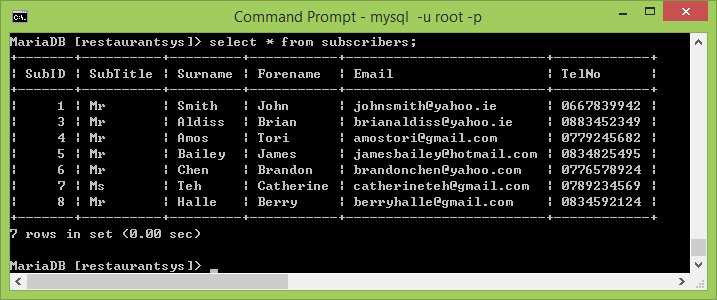
Users can also perform an online reservation to reserve tables for either lunch time from 11.30am to 3.00pm or dinner time 6.00pm to 10.00pm. Reservations that made online will be saved in the database and the reservation status is set by default to ‘EN’ (enquiry) before the reservation is ‘CF’ (confirmed) by the owner through phone calling. Users can only make reservations to the same date and time when the previous booking for that date and time is ‘CC’ (canceled).

Lastly, when owner checks for reserved tables in the database, owner can view the session time reserved by the users by joining both ‘Reservations’ and ‘Sittings’ tables. Owner can also view the customer’s information by joining both ‘Reservations’ and ‘Customers’ tables. Owner can also view the table info’s by joining both ‘Reservations’ and ‘TableInfos’ tables in database. When a reservation has been confirmed, the owner can update the status of the respective customer in the ‘Reservations’ table.

## INSERT Element:

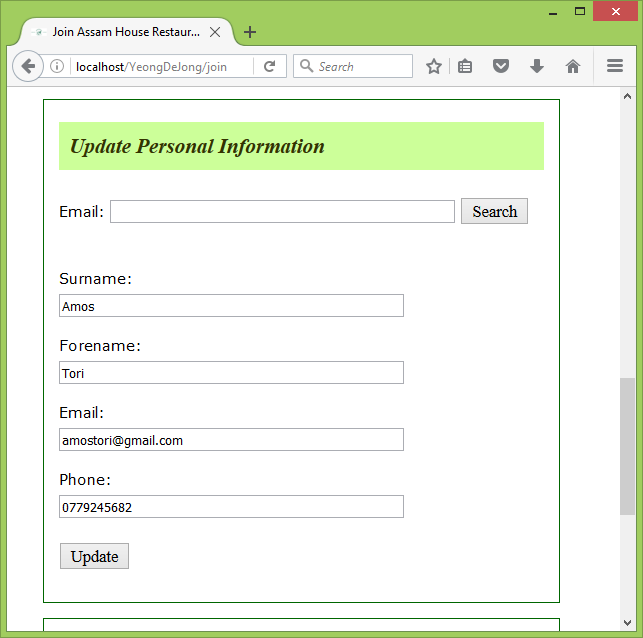
Insert element can be seen in ‘**joinus.php**’ and ‘**reservation.php**’ file. In ‘**joinus.php**’, when user enters their personal information in form, the details that are entered will then fetch to the database in ‘Subscribers’ table. If the user has successfully joins newsletter subscription, it displays ‘Thanks for Joining’ message to indicate that the user has successfully joins the newsletter subscription.



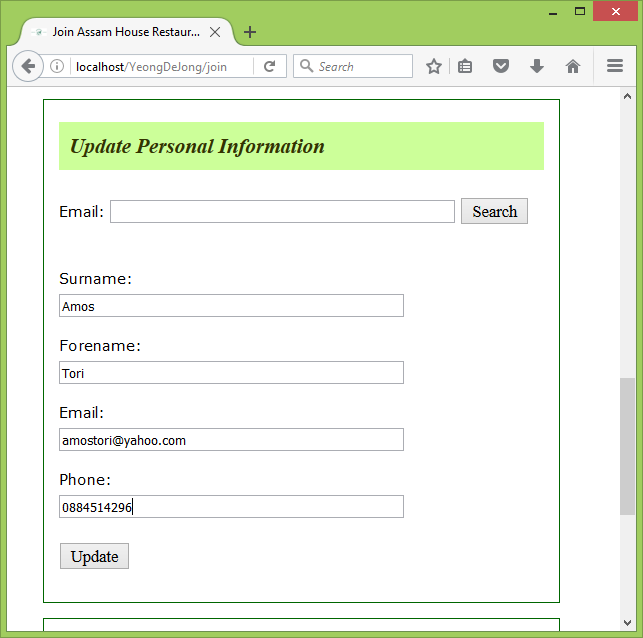


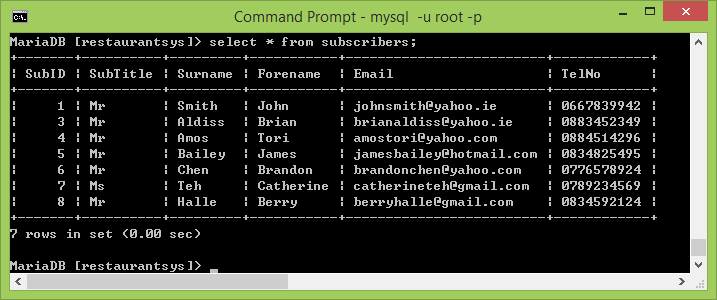
## UPDATE Element:

Update element can be seen in ‘**joinus.php**’ file under ‘Update Personal Information’ form. When user enters their email, and click search, it retrieves details with matching email that are provided by the user and display on textboxes. When user enters an invalid email that does not exist in the database, it displays ‘No such record found’ error message to indicate that the email does not exist in the database.



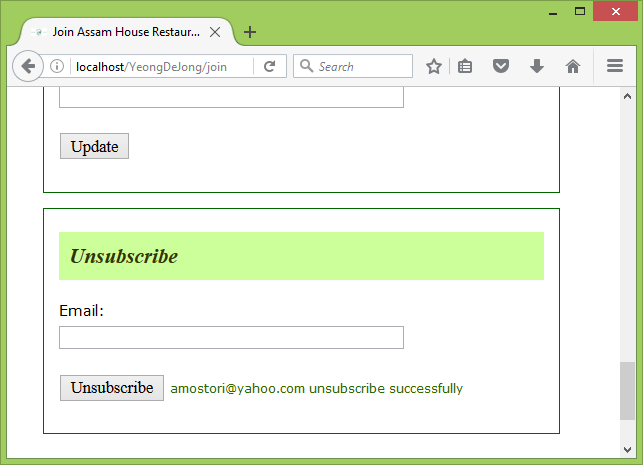
The user updates his/her personal information. When user clicks on the update button, it updates the information of the user and rewrite the previous details that are recorded from database. If no records were updated, it displays ‘No Update’ message to indicate that no records are updated, else it displays ‘Record Updated’ message to indicate that the record for that user has been updated.

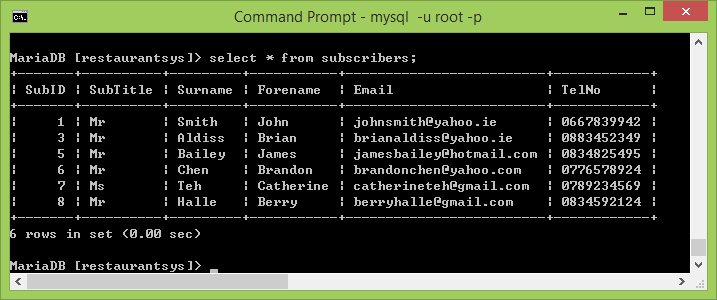




## DELETE Element:

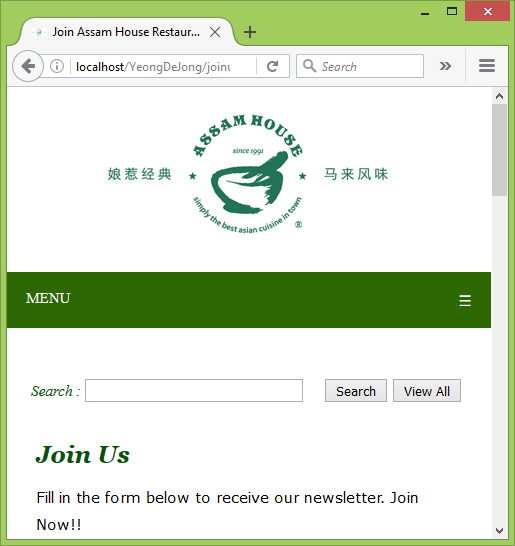
Update element can be seen in ‘**joinus.php**’ file in Unsubscribe form. When user types their email address and clicks ‘Unsubscribe’ button, it searches through the database and delete the details of the user with matching email from database. If user unsubscribe successfully, it displays ‘(email) unsubscribe successfully’ message to indicate that user has unsubscribed successfully. Else it displays ‘no such record deleted’ error message.

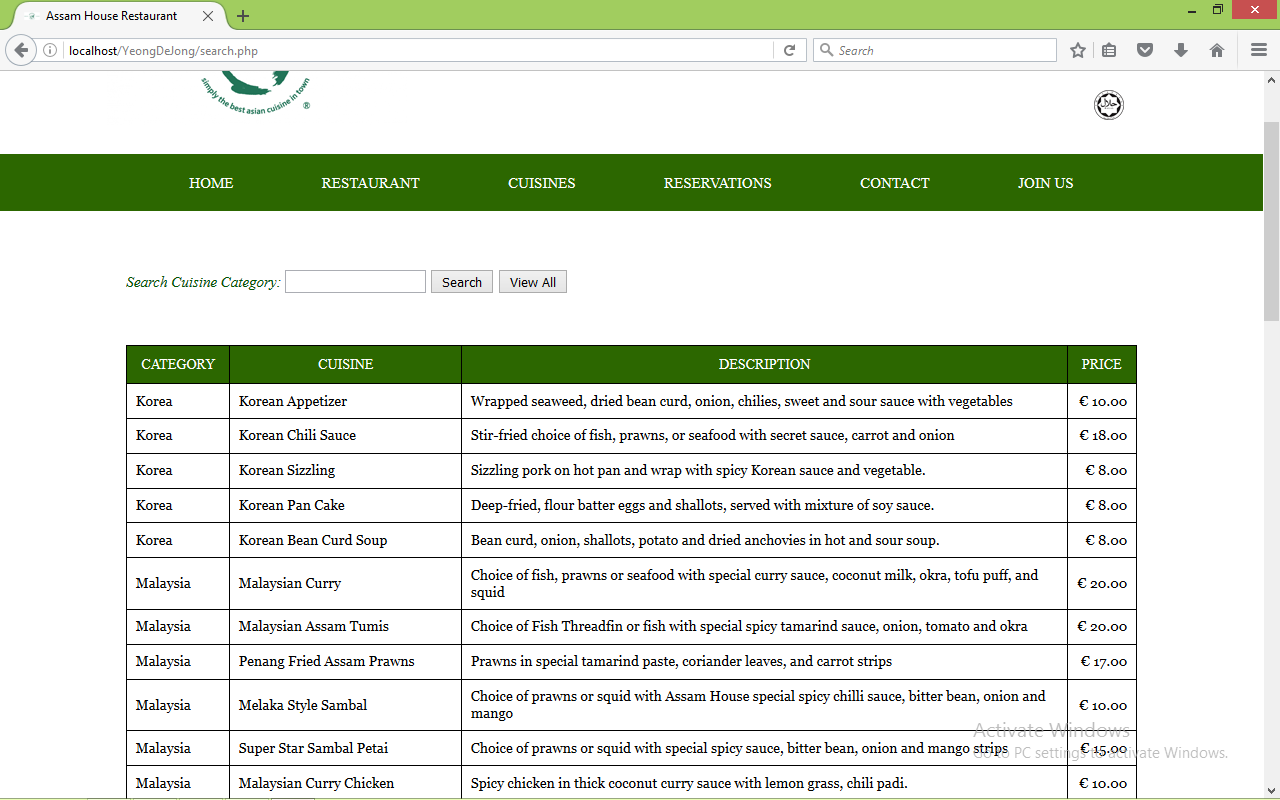




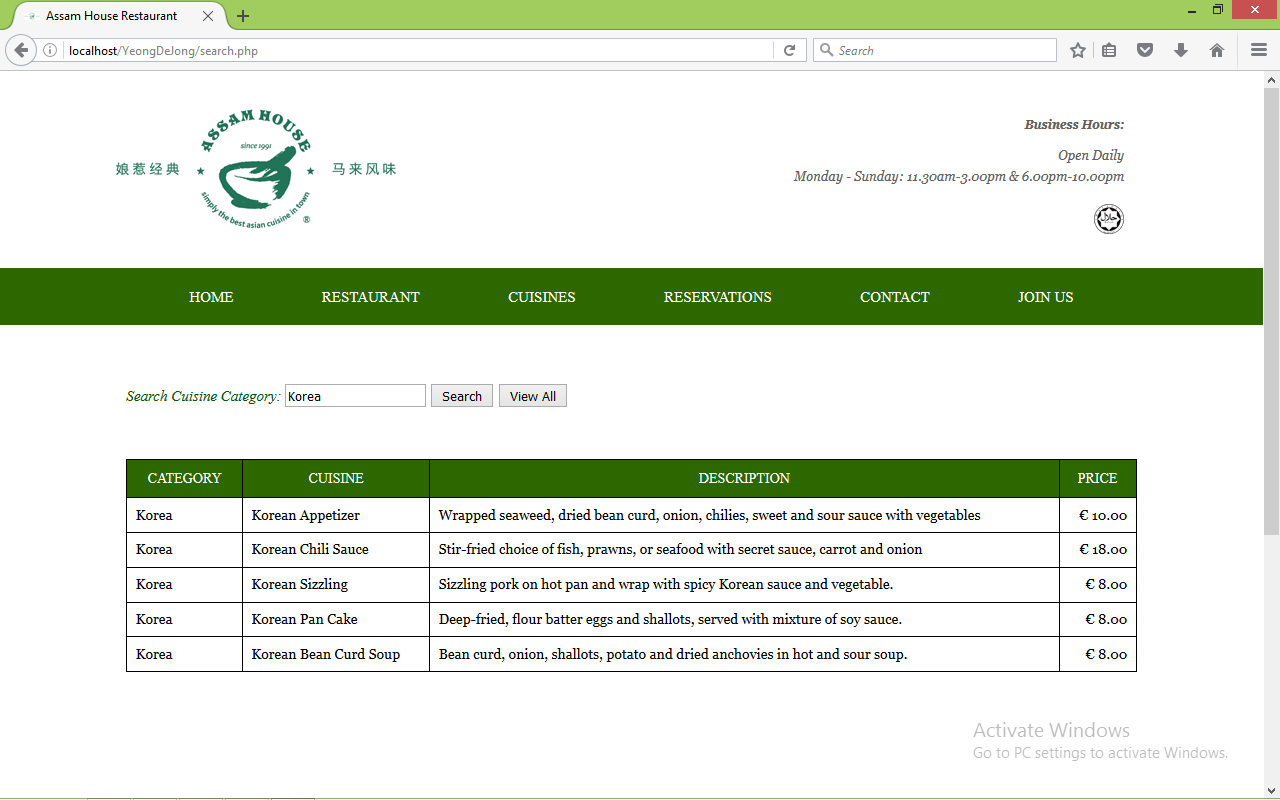
## SELECT Element:

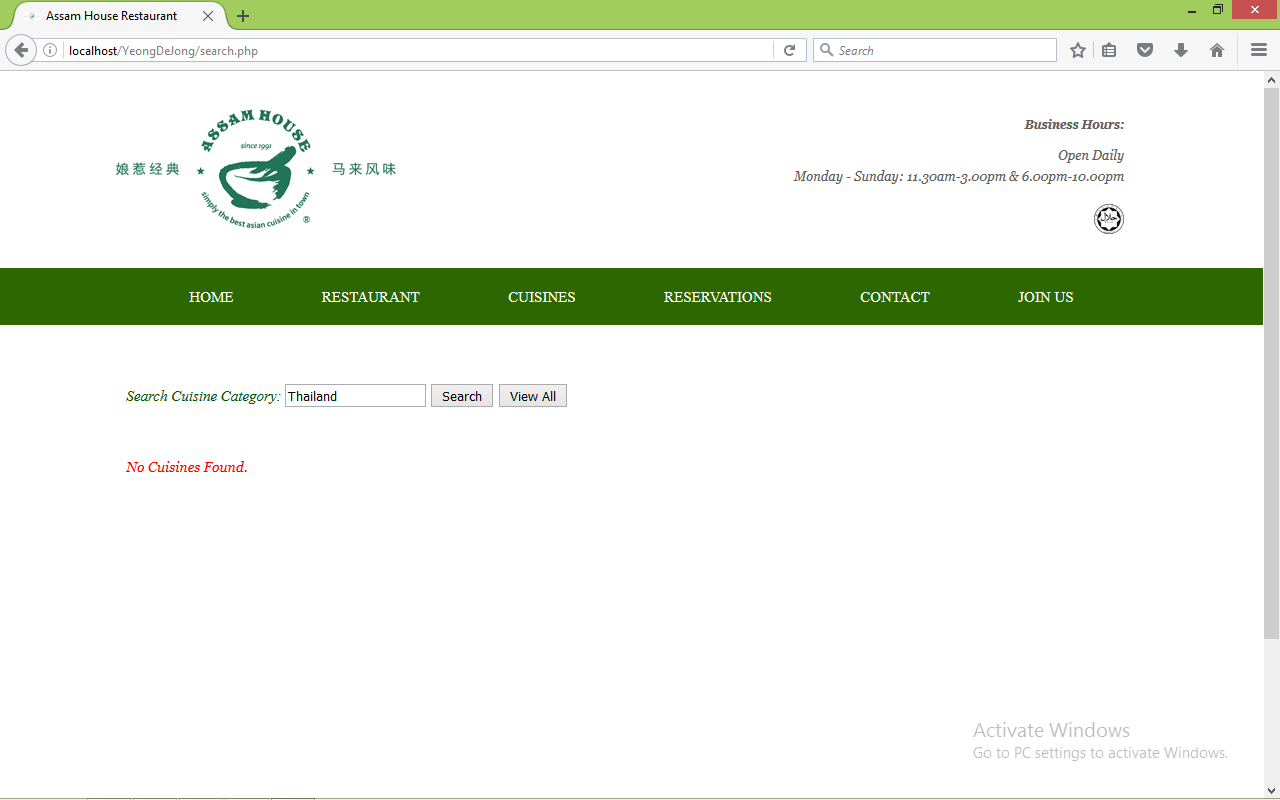
Select element can be seen throughout all the files. When user clicks on ‘View All’ button, it displays a list of cuisines in alphabetical order of category along with the cuisine’s name, descriptions, and the price of that cuisine.





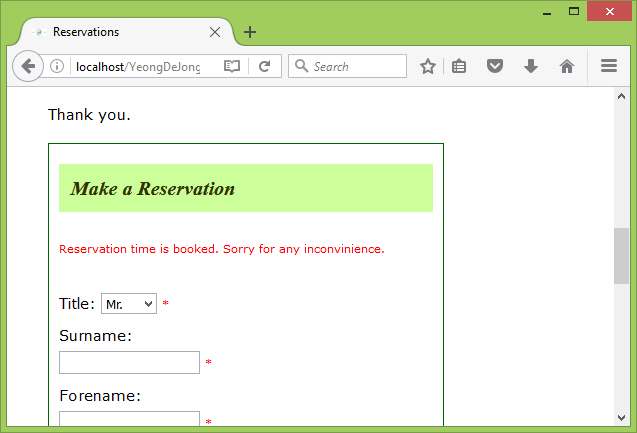
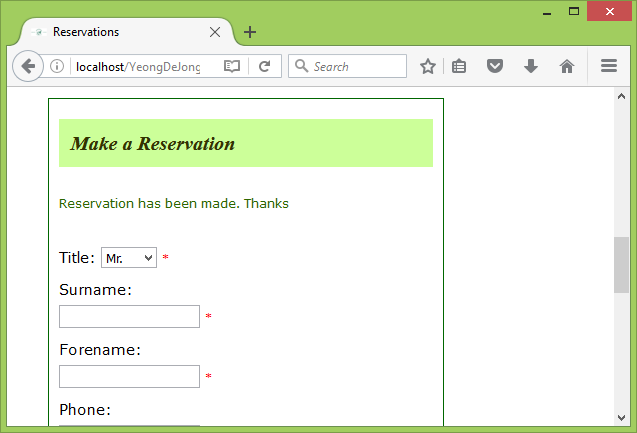
When user searches a cuisine category by typing in the cuisine’s category, it displays the category along with the cuisine’s name, description, and price with matching category from the database. When user enters cuisine category that does not exist in the database, it displays ‘No Cuisine Found’ error message back to user.



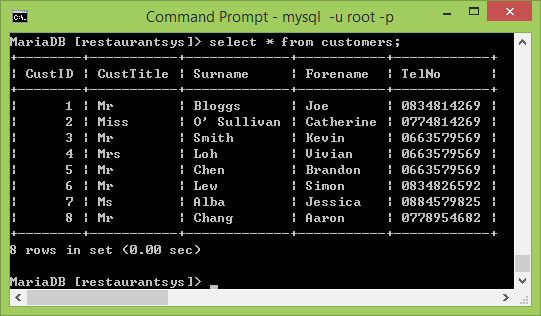


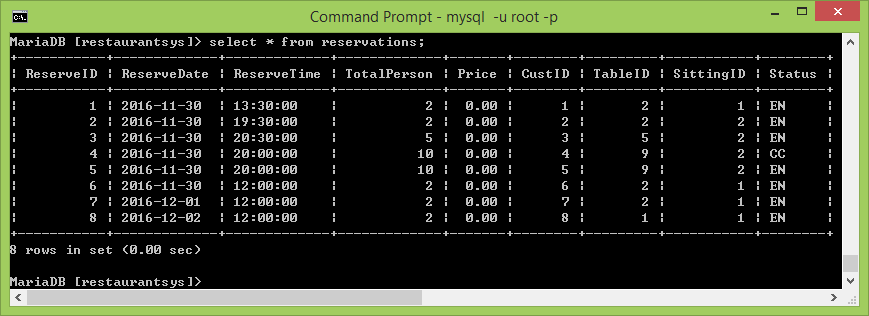
## PROCESSING:

Processing element can be seen in ‘**reservation.php**’ file. User can make a reservation by filling in the forms. When user does not complete the form before clicking the ‘Reserve’ button, it displays ‘You did not complete the form’ to indicate that the user does not complete filling in the form. When user has successfully made a reservation, it displays ‘Reservation has been made. Thanks’ message to user. When user makes a reservation, the system searches through the database to check whether the date, time, and the session has been booked. If there’s a match between the booking details in database, it displays an error message that the reservation time is booked.



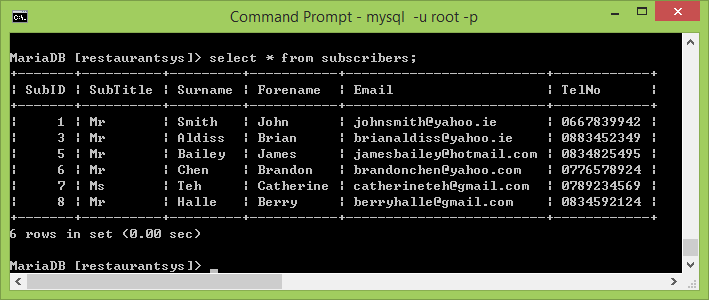
When a reservation has been made, the system first inserts the details of the customer in Customers Table, the last inserted customer id is then takes as a variable and insert into Reservations Table as the customer id in Reservations Table is has a foreign key which references to the Customers Table. ReserveDate, ReserveTime, SittingID can be the same only when the ReserveTime is different and the status of the reservation is either ‘CC’ (cancelled).





## View of Data in database

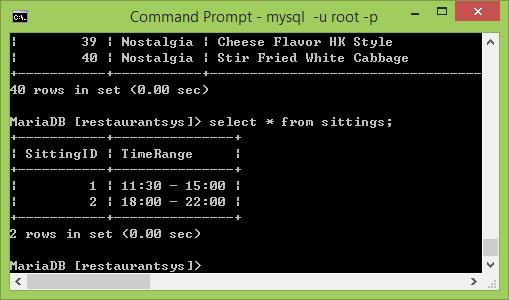
### Subscribers Table:



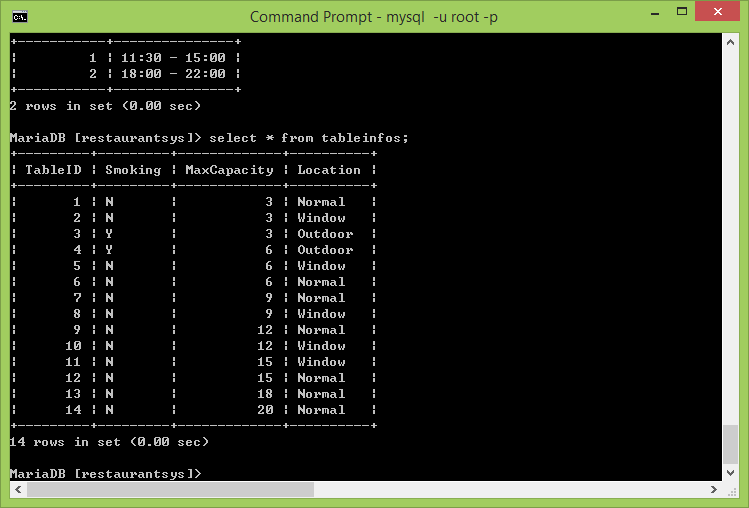
### Cuisines Table:



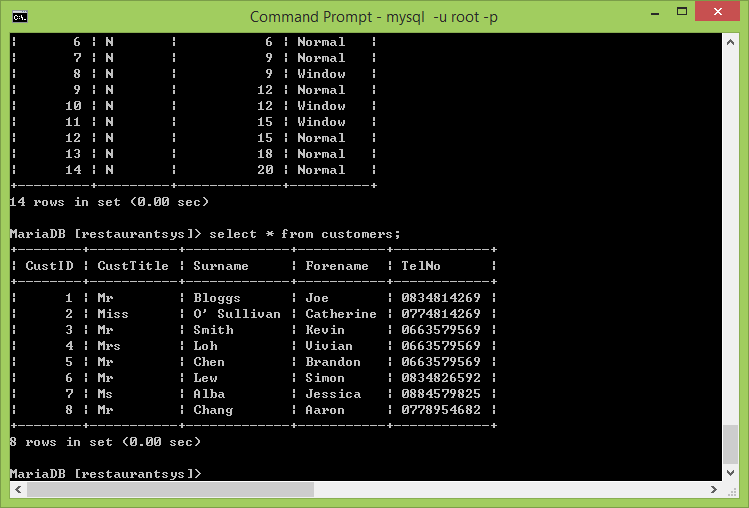
### Sittings Table:



### TableInfos Table:



### Customers Table:



### Reservations Table:

