VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANASANGAMA, BELAGAVI - 590018

Mini Project
Report On

Food Delivery Website (Food Express)

Submitted in partial fulfillment for the award of degree of

Bachelor Of Engineering
In
Computer Science and Engineering

Submitted By:

Karthik Kombrenje

Kumaraswamy (1BG19CS043)

Vidyayâmruthamashnuthe

B.N.M. Institute of Technology

Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.

All UG branches – CSE, ECE, EEE, ISE & Mech.E accredited by NBA for academic year

2018-19 to 2021-22 & valid upto 30.06.2022

Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070,
INDIA Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www. bnmit.org

Department of Computer Science and Engineering

2021 – 2022

B.N.M. Institute of Technology

Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.

All UG branches – CSE, ECE, EEE, ISE & Mech. E accredited by NBA for academic year 2018-19 to 2021-22 & valid upto 30.06.2022. Post box no. 7087, 27th cross, 12th Main, Banshankari 2nd Stage, Bengaluru – 560070, INDIA Ph: 91-80-26711780/81/82. Email: principle@bnmit.in , www.bnmit.org

Department Of Computer Science and Engineering

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the Mini Project entitled **Food Express** by **Mr Karthik Kombrenje Kumaraswamy USN 1BG19CS043 a bonafide student of V Semester B.E., B.N.M Institute of Technology** in partial fulfilment for the Bachelor of Engineering in **COMPUTER SCIENCE AND ENGINEERING of the Visvesvaraya Technological University, Belagavi** during the year **2021-22**. It is certified that all corrections /suggestions indicated for Internal Assessment have been incorporated in the report. The Project report has been approved as it satisfies the academic requirements in respect of Database Management System with Mini Project Laboratory prescribed for the said Degree.

Dr. Sreevidya R C	**Dr. Sahana D. Gowda**
Associate Professor and Lab Incharge	**Professor and HOD**
Department of CSE	**Department of CSE**
BNMIT, Bengaluru	**BNMIT, Bengaluru**

Name & Signature

Examiner 1:

Examiner 2:

ABSTARCT

With people practicing social distancing, self-quarantine, and staying indoors to stop the spread of the virus, food-delivery startups are in demand more than ever. They cater to more orders than ever. Food delivery companies can actually help people stay indoors and fight this pandemic. Change is a decent and obvious thing in the world of technology. So, we can expect changes in the food delivery market for a better future. According to the report from McKinsey & Company, "The worldwide market for food delivery was valued at €83 billion in 2016". In the current times, the food delivery market has already matured in most of the countries with an overall growth rate of approximately 3.5% for the next five years.

The online food-delivery has already seen much growth in the past few years, however, this market has akeady matured due to the overall level of funding and technology penetration. However, things are not ended here. There is another key catalyst is to increase convenience and transparency for both the ends i.e. customers as well as merchants.

To get extraordinary results from your business, it is important to get a customized website for your on-demand food delivery business. Being an owner of the food delivery business, if you want to retain your customers for a long time, then you need to focus on developing a website along with unique features and functionalities.

A website for food delivery can do a lot for your business. From enticing your website users to order their first food order to make use of coupon offers. Customers usually find it interesting and place orders. So, to get extraordinary results, it is important to consider a customized website to easily target the right audience.

ACKNOWLEDGEMENT

The completion of this project brings with a sense of satisfaction, but it is never complete without thanking the persons responsible for its successful completion.

I take this opportunity to express our profound gratitude to Shri. Narayan Rao R Maanay, Secretary, BNMIT, Bengaluru for his constant support and encouragement. I would like to express my special thanks to Prof. T. J. Rama Murthy, director, BNMIT, Bengaluru and S Y Kulkarni, director, BNMIT, Bengaluru for their constant guidance towards our goals and professions.

I extend my deep sense of sincere gratitude to Dr. Krishnamurthy G.N, Principal, BNMIT, Bengaluru, for providing us facilities required for the project.

I would also like to thank Prof. Eishwar Maanay, Dean Administration, BNMIT, Bengaluru, for providing us useful suggestions required for the project.

I express my in-depth, heartfelt, sincere gratitude to Dr. Sahana D. Gowda, Professor and H.O.D, Department of Computer Science and Engineering, BNMIT, Bengaluru, for her valuable suggestions and support.

I extend my heartfelt, sincere gratitude to Dr Srividhya , Associate Professor, Department of Computer Science and Engineering, BNMIT, Bengaluru, for completion of the project.

Finally, I would like to thank all the faculty members of Department of Computer Science and Engineering, BNMIT, Bengaluru, for their support. I would like to thank our family and friends for their unfailing moral support and encouragement.

Karthik Kombrenje

Kumaraswamy

1BG19CS043

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Overview of Database Management System

A Database is a collection of related data organized in a way that data can be easily accessed, managed and updated. Any piece of information can be a data, for example name of your school. Database is actually a place where related piece of information is stored and various operations can be performed on it. A DBMS is a software that allows creation, definition and manipulation of database. DBMS is actually a tool used to perform any kind of operation on data in database. DBMS also provides protection and security to database. It maintains data consistency in case of multiple users. Here are some examples of popular DBMS, Sql, Oracle, Sybase, Microsoft Access and IBM DB2.

The database system can be divided into four components:

- The database system can be divided into System developer and End users.
- Database application: Database application may be Personal, Departmental, Enterprise and Internal
- DBMS: Software that allow users to define, create and manages database access, Ex: Sql, Oracle etc.
- Database: Collection of logical data.

Functions of database management system:

- Provides Recovery services
- Provides utility
- Provides data Independence
- Provides a clear and logical view of the process that manipulates data.

Advantages of DBMS:

- Segregation of application program
- Minimal data duplicity
- Reduced development time and maintenance need
- Easy retrieval of data.
- Seamless integration into the application programming languages which makes it very easier to add a database to almost any application or website.

Disadvantages of DBMS:

- It's Complexity
- Except MySQL, which is open source, licensed DBMSs are generally costly.
- They are large in size.

1.2 Problem Statement

To design a food Delivery website using Flutter Web, MYSQL ,PHP

The objective of this project is to:

- Have collaboration between the customers, drivers, restaurant with a simple database
- Getting delivery done faster, cheaper and better by applying their common knowledge through the notes.
- Bringing together a selection of resources and attainments in a project.

1.3 Dataset Description

Food express provides a Registeration page for registering users and Login page for logging in the users. The details of the users are maintained in a User table in the database.

We have made use of 6 tables

1.Drivers

2.Customers

3.Restaurant

4.Food

5.Reviews

6. Order details

Users table: Contains the registered users and their passwords

Driver table:-Contains the driver details Has Driver id(Did),Driver Name(Dname),Driver Age(Dage),

Customers table: This contains the Customer id(Cid),Customer Name(Cname),Customer Address(Cadd),the type of Customer(Regular , Occasional , New Customer)

Restaurant table:-Has the restaurant details in it.This contains the Restaurant id(Rid),Name of Restaurant(Rname),Location of restaurant(Loc)

Food table: It stores the details of food available in different restaurants.This contains Restaurant id (Rid),Food id (Fid),Food Name(Fname),Cost of Food(Cost).

Reviews module: This contains all reviews given by the customer.It has Customer id(Cid),Restaurant id(Rid),Rating given by the customer from 1-5(Rating)

Order Details: Contains the order details of different customers. It has Rid,Cid,Date of Billing(Date),Bill amount(Bill)

The tables along with attributes are:

1) Users (Username,Password}
2) Drivers(Did,Dname,Age}
3) Customers{ Cid,Cname,Cadd }

4) Restaurant {Rid,Rname,Location}
5) Food(Rid,Fid,Fname,Cost}
6) Review (Cid,Rid,Rating }
7) OrderDetails {Rid,Cid,Date,Bill}

Chapter 2

SYSTEM REQUIREMENTS

2.1 Software and Hardware

Software Configuration:

- Operating System: Windows 8, 64 bit
- Front end: Flutter
- Server side language: PHP
- Frameworks used: Android Studio
- Back end: MySQL
- Web Server: Apache 2.0
- Browser: Chrome
- Application Software: XAMPP

Hardware Configuration:

- Processor: Intel Core i5
- RAM: 8GB
- Hard disk: 1TB

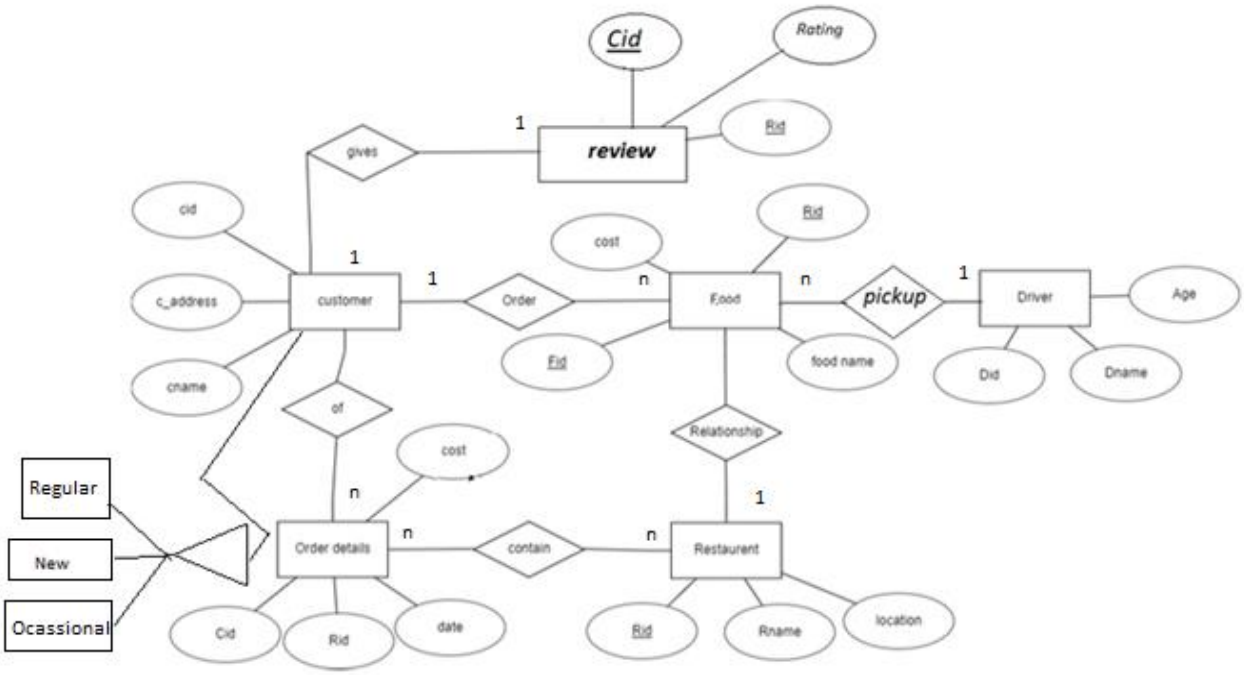
Chapter 3

SYSTEM DESIGN

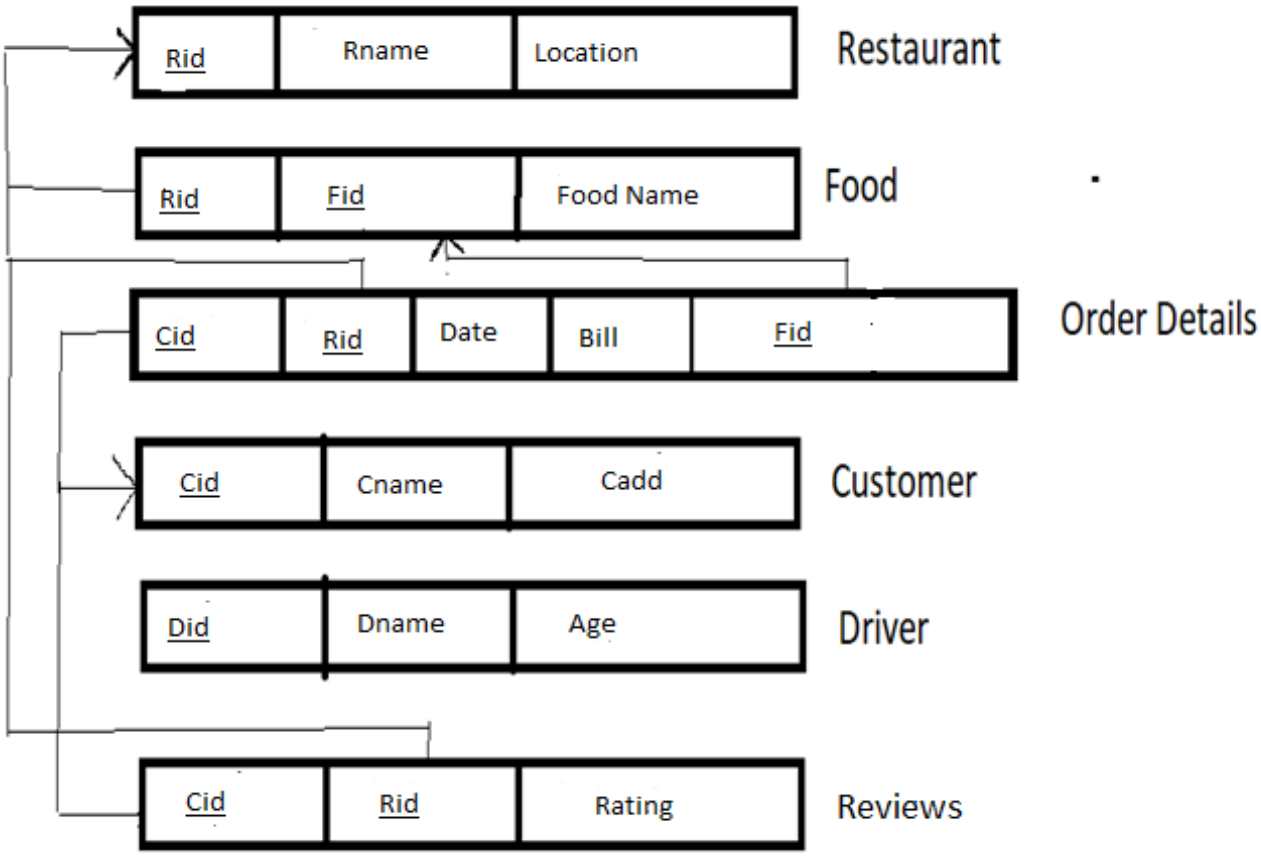
3.1 E R Diagram

An entity-relationship diagram(ERD) is a data modelling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD contains different symbols and connectors that visualize two important information: The major entities within the system scope and the inter relationships among these entities.

The above below illustrates ER diagram for the Food Express .



3.2 Schema Diagram



A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and the relations among them are associated. It formulates all the constraints that are to be applied on data. A database schema defines its entities and relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams.

The figure 3.2.1 shows the schema diagram for Food Express. It shows the various relations, references between entities

3.3 Overview of GUI

GUI is a program interface that takes advantage of the computer's graphics capabilities to make the program easier to use. Well-designed graphical user interfaces can free the user from learning complex command languages. On the other hand, many users find that they work more effectively with a command-driven interface, especially if they already know the command language.

Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, Web platform, and the web from a single codebase.

The Flutter framework consists of both a software development kit (SDK) and their widget-based UI library.

This library consists of various reusable UI elements, such as sliders, buttons, and text inputs.

Developers building mobile applications with the Flutter framework will do so using a programming language called Dart. With a syntax like JavaScript, Dart is a typed object programming language that focuses on front-end development.

3.4 Normalization

Normalization is a process of analyzing the given relation schema based on their functional dependencies and primary key to achieve desirable properties of minimizing redundancy and minimizing insert, delete, update anomaly. The normalization process takes a relation schema through a series of tests to certify whether it satisfies a certain normal form. The normal form of a relation refers to the highest normal form condition that it meets, and hence the degree to which it has been normalized.

Normalization rule are divided into following normal form.

1. First Normal Form
2. Second Normal Form
3. Third Normal Form
4. Boyce-codd Normal Form

3.4.1 First Normal Form

First normal form states that the domain of an attribute must include only atomic (simple, individual) values and that the value of any attribute in a tuple must be a single value from the domain of attribute.

Considering the relations of Food Express, all the relations are in lNF as they have neither any multivalued attributes nor composite attributes. Hence the relations are said to be in lNF.

3.4.1 Second Normal Form

Second normal from is based on the concept of full functional dependency. A functional dependency X -> Y is a full functional dependency if removal of any attribute A from X means that the dependency does not hold anymore. A relation schema R is in 2NF if every nonprime attribute A in R is fully functionally dependent on the primary key of R.

3.4.3 Third Normal Form

Third normal form is based on the concept of transitive dependency. A relation schema R is in 3NF if it satisfies 2NF and no nonprime attribute of R is transitively dependent on the primary key. A relation schema R is in 3NF if every nonprime attribute of R meets both of these following conditions:

- It is fully functionally dependent on every key of R.
- It is non transitively dependent on every key of R.

The relations used in this database are fully functionally dependent on its key attribute and does not hold any transitive dependencies. Hence all the relations are in 3NF.

Chapter 4

IMPLEMENTATION

4.1 Table Creation

```
CREATE TABLE users (
   Username VARCHAR(128) PRIMARY KEY,
   password  VARCHAR(128)
   NOT NULL);


CREATE TABLE driver (
   Did INT PRIMARY KEY,
   Dname VARCHAR(128)
   NOT NULL, Age int(128)
   NOT NULL);

CREATE TABLE customer (
   Cid INT PRIMARY KEY, Cname
   VARCHAR(128) NOT NULL,Cadd
   VARCHAR(256) NOT NULL,Ctype
   VARCHAR(256) NOT NULL);


CREATE TABLE restaurant (
   Rid INT PRIMARY KEY,
   Rname VARCHAR(128)
   NOT NULL, Location
   Varchar2(256)NOT NULL);
```

```
CREATE TABLE food(
    FOREIGN KEY (Rid)
    REFERENCES
    restaurant(Rid), Fid INT
    PRIMARY KEY,
    FoodName
    VARCHAR2(128) NOT
    NULL, Cost int(128) NOT
    NULL);


CREATE TABLE orders(
    FOREIGN KEY (Rid)
    REFERENCES
    restaurant(Rid), FOREIGN
    KEY (Cid) REFERENCES
    customers(Cid),DateVARCH
    AR2(128) NOT NULL, Bill
    int(128) NOT NULL);



CREATE TABLE orders(
    FOREIGN KEY (Rid)
    REFERENCES
    restaurant(Rid), FOREIGN
    KEY (Cid) REFERENCES
    customers(Cid),Rating
    INT(128) NOT NULL);
```

4.2 Description of Table

desc user;

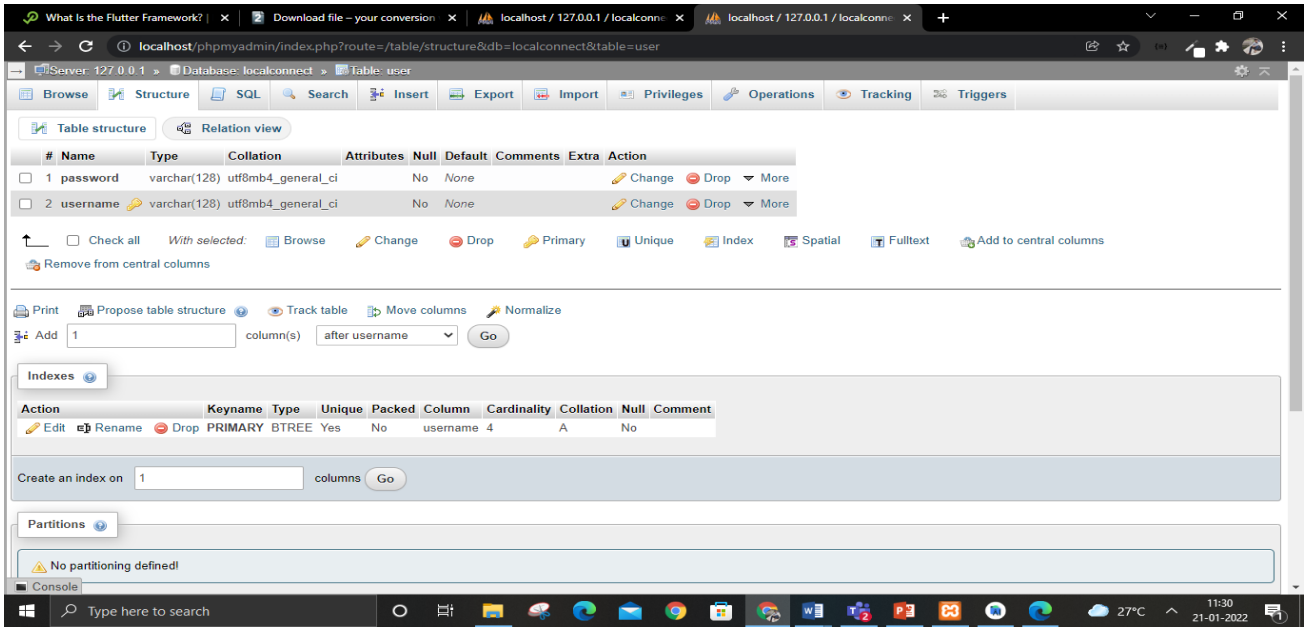
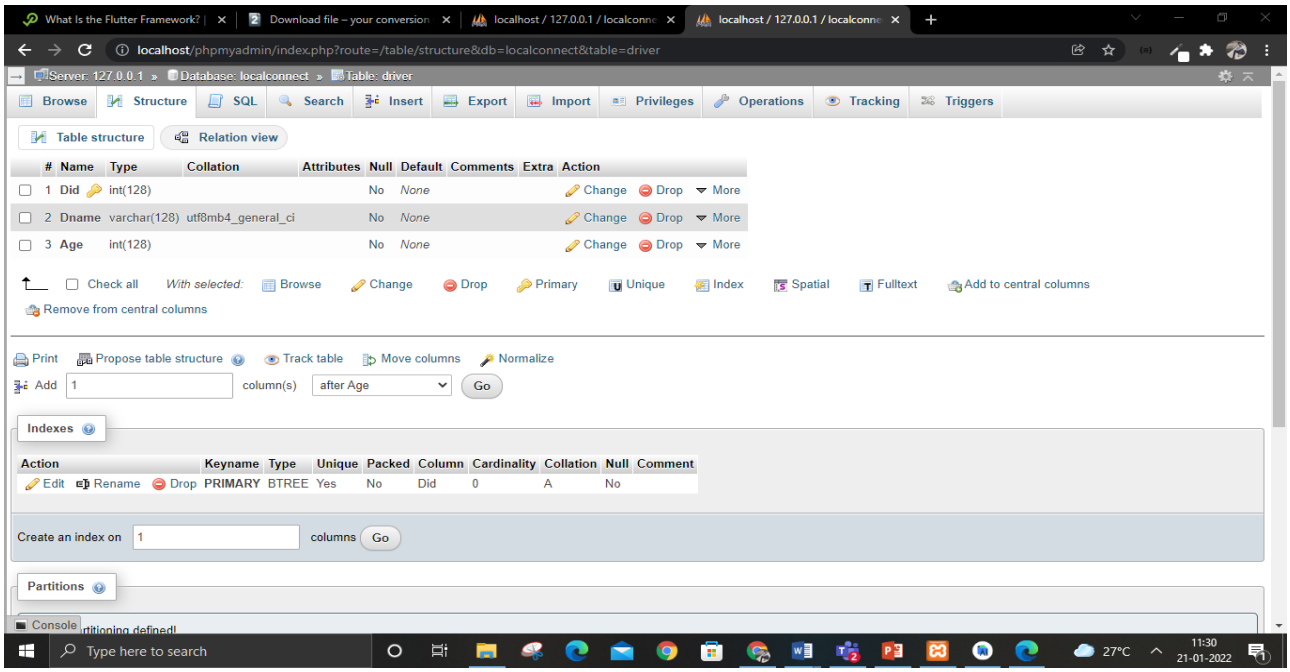


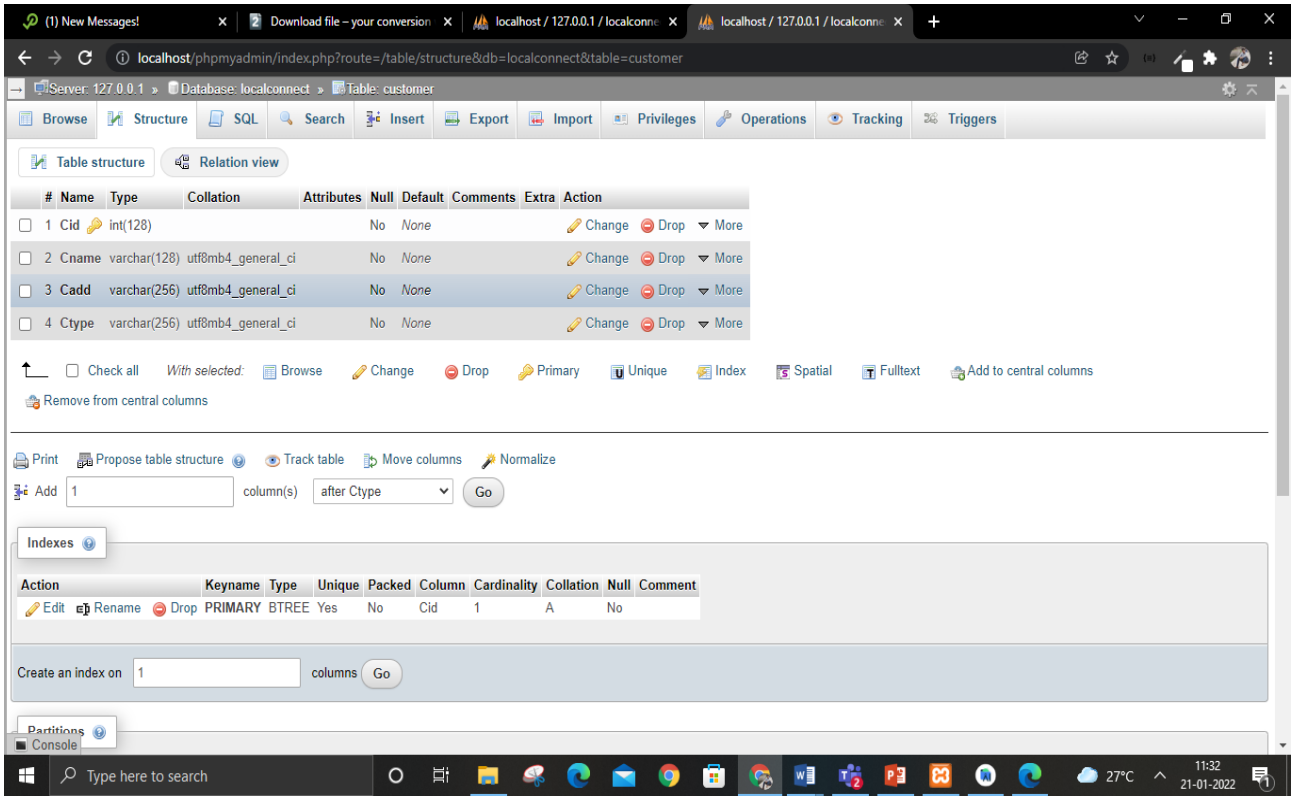
Figure 4.2.1 Description of users table

desc driver;

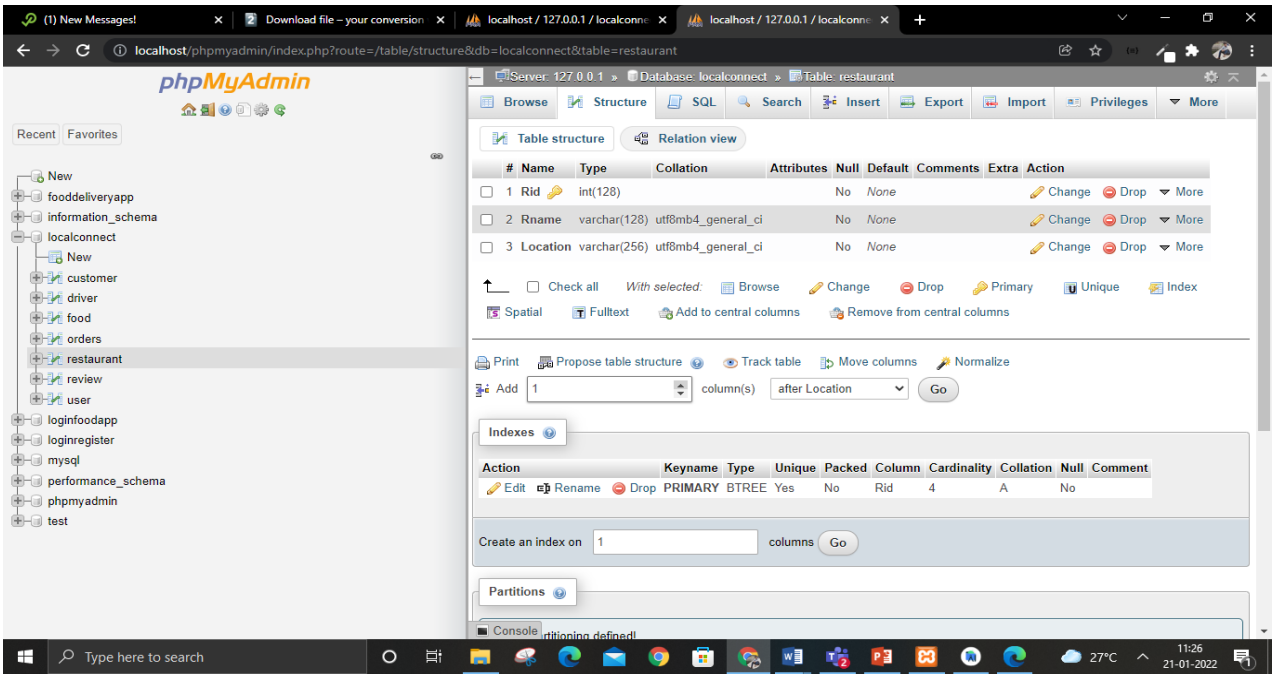
bb



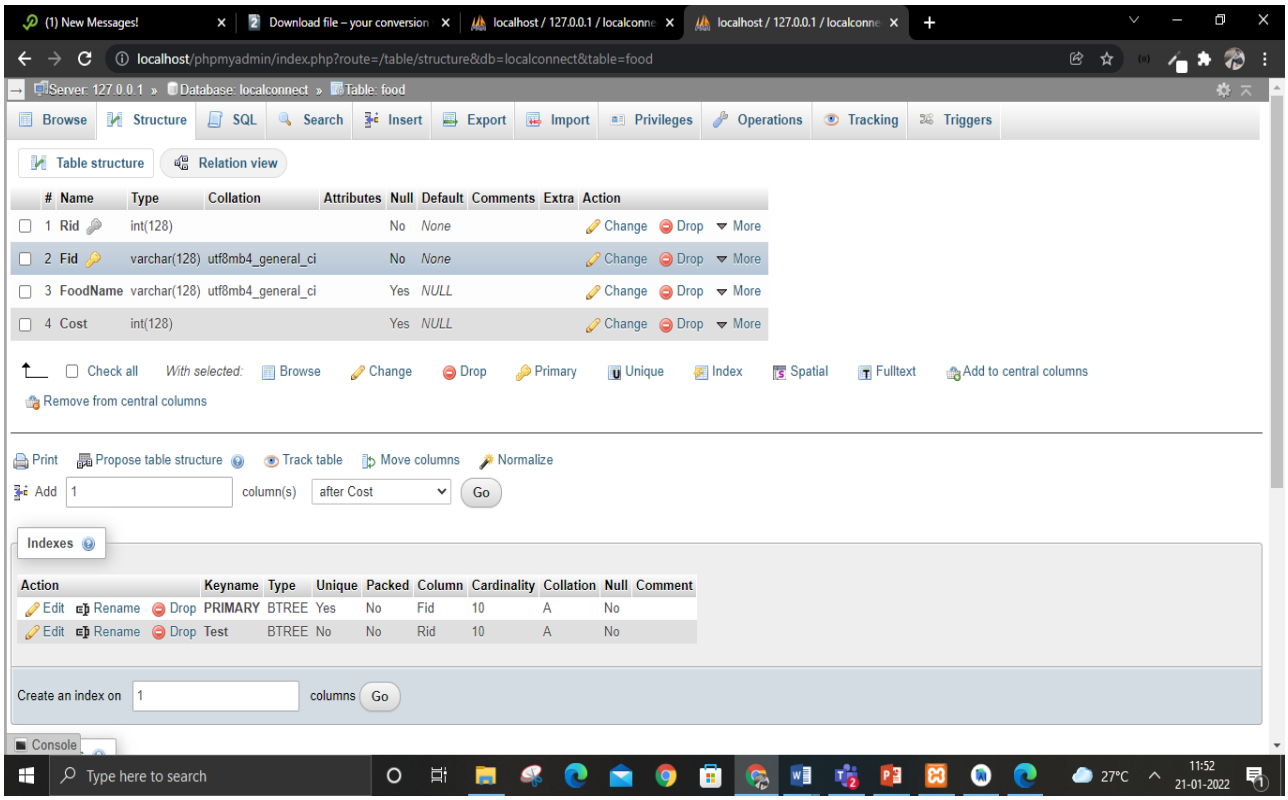
desc customer;



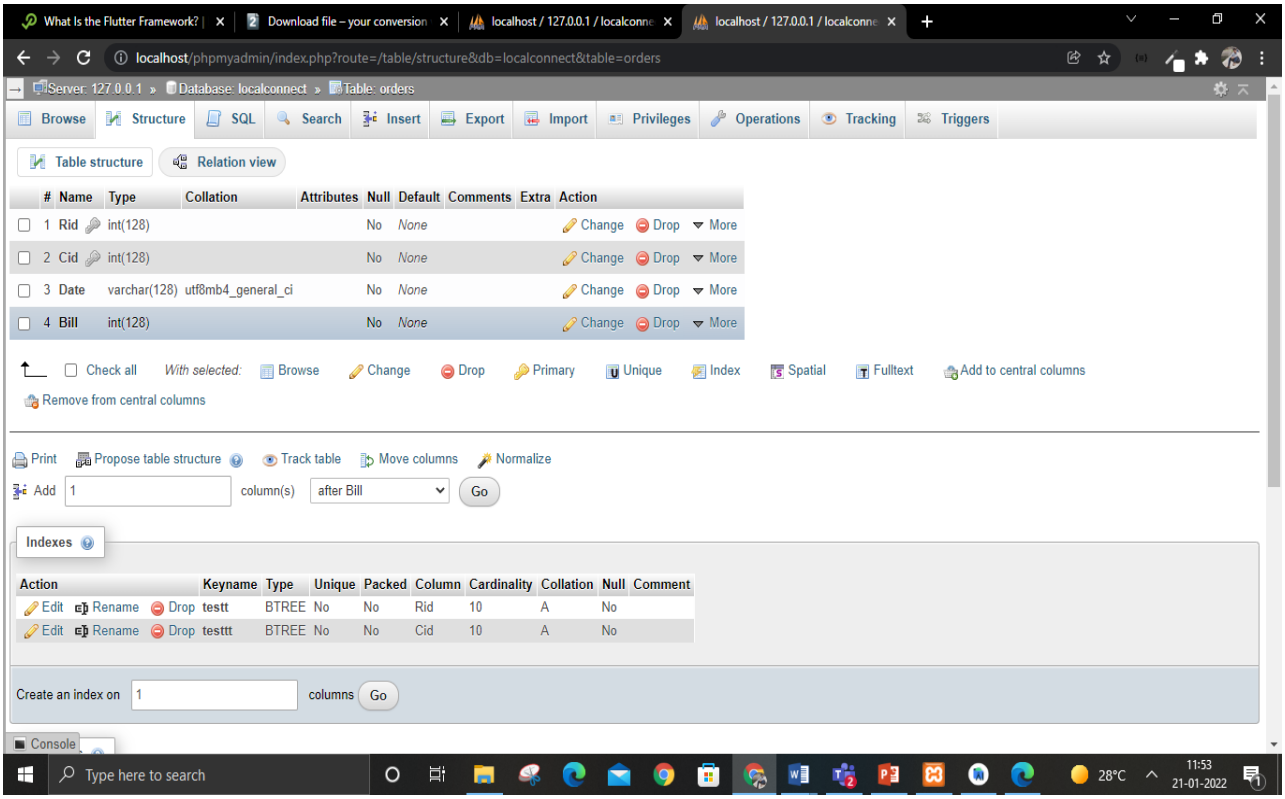
desc restaurant;



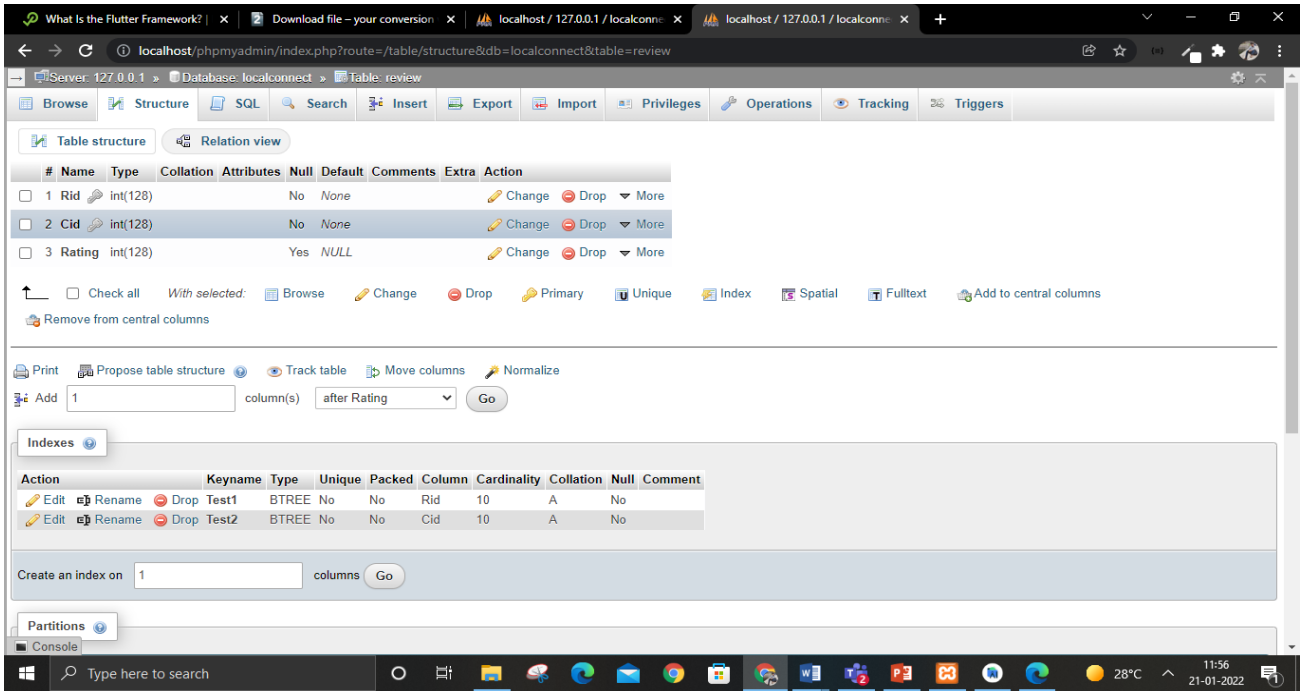
desc food;



desc orders;

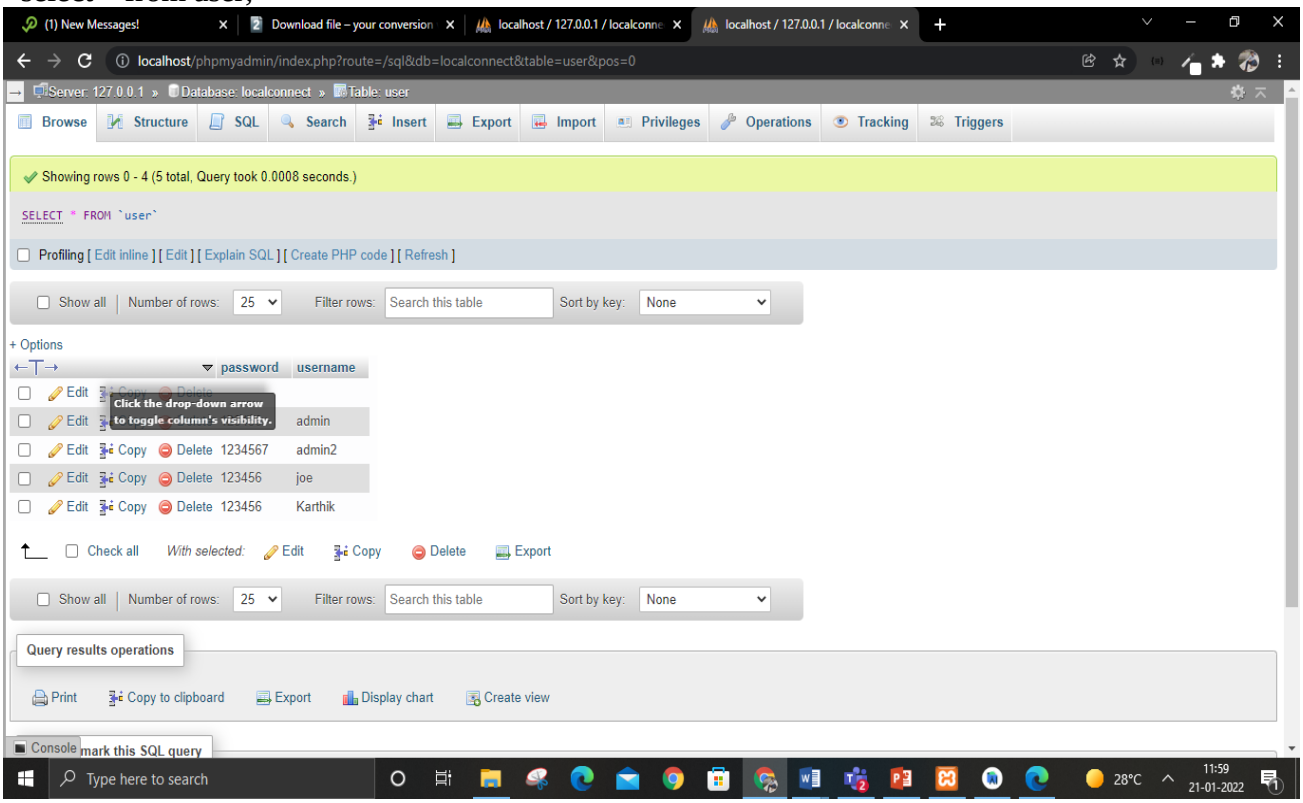


desc reviews;

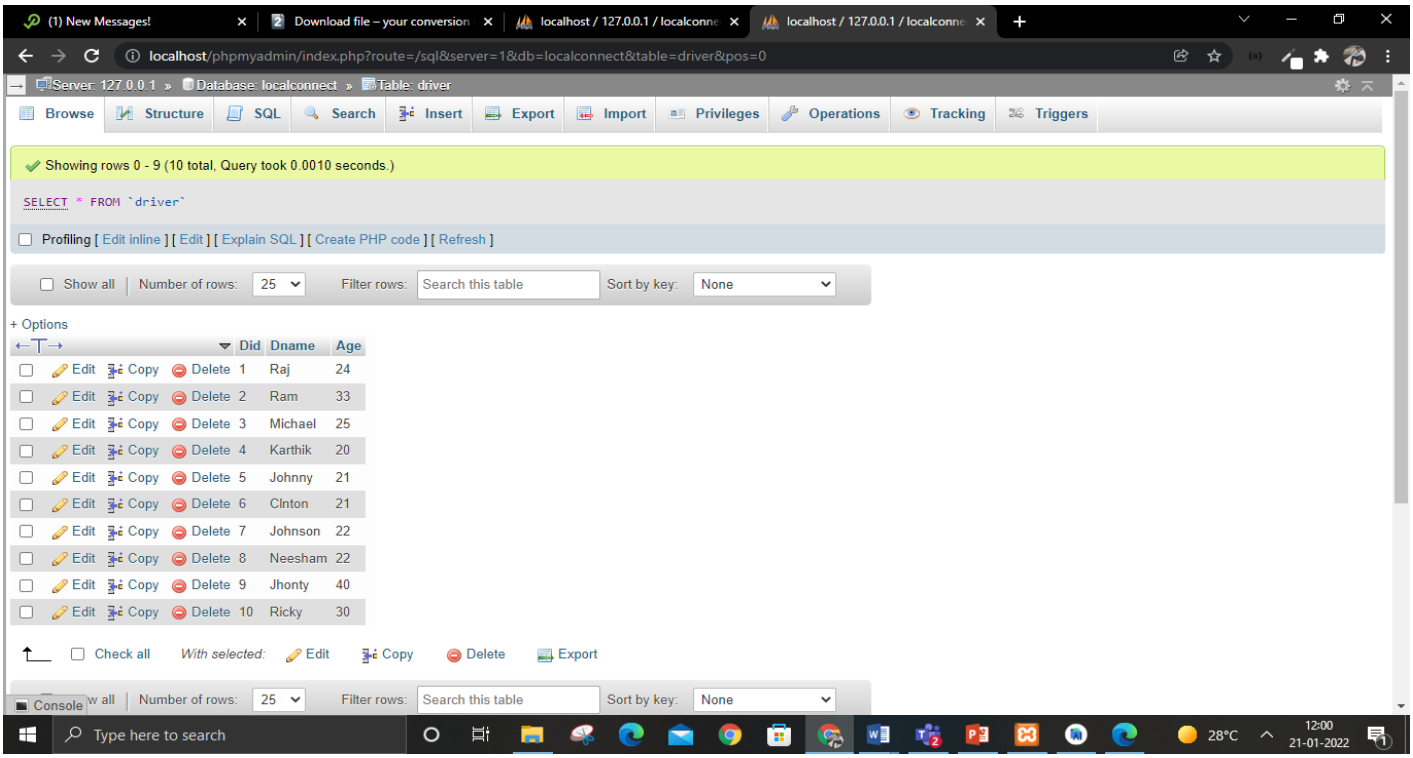


4.3.Populated Tables

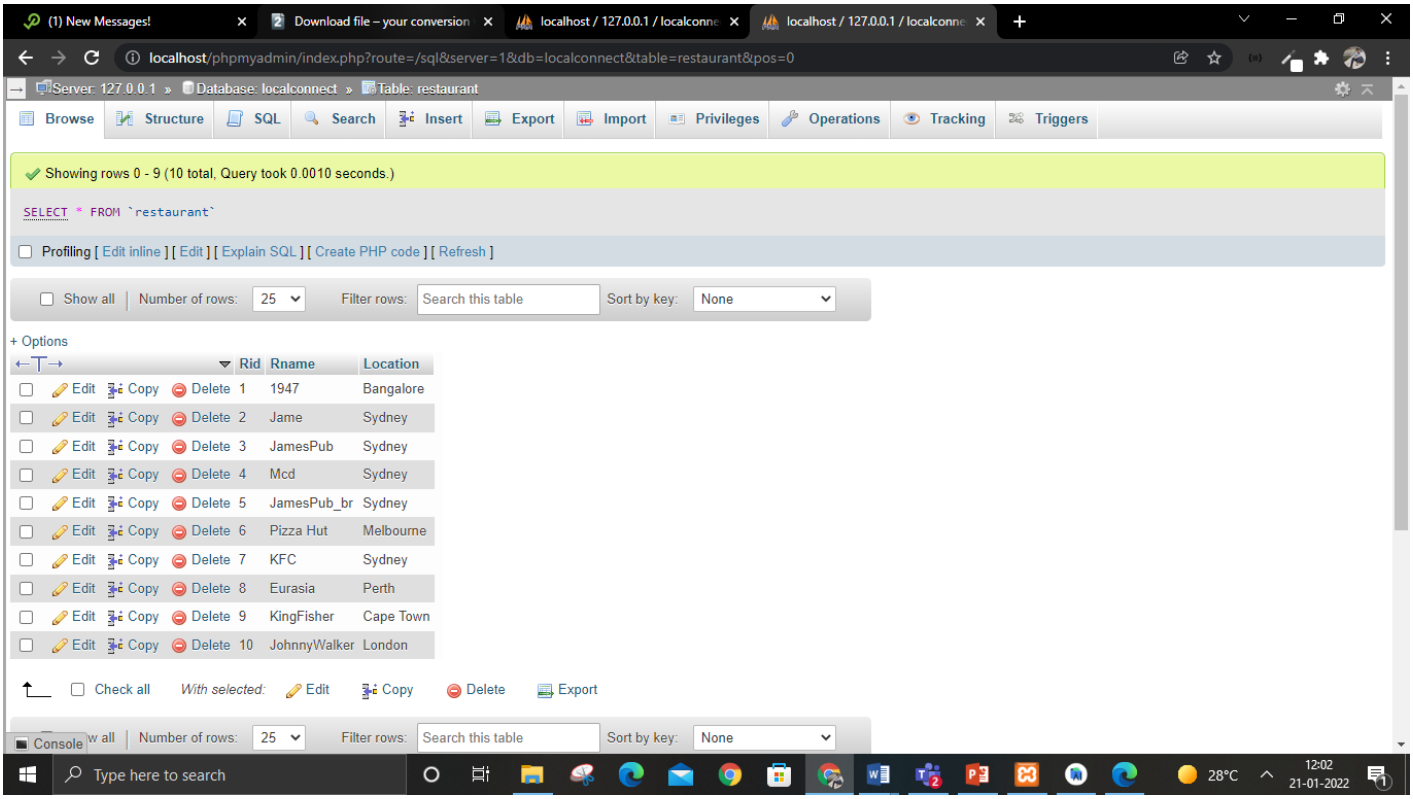
select * from user;



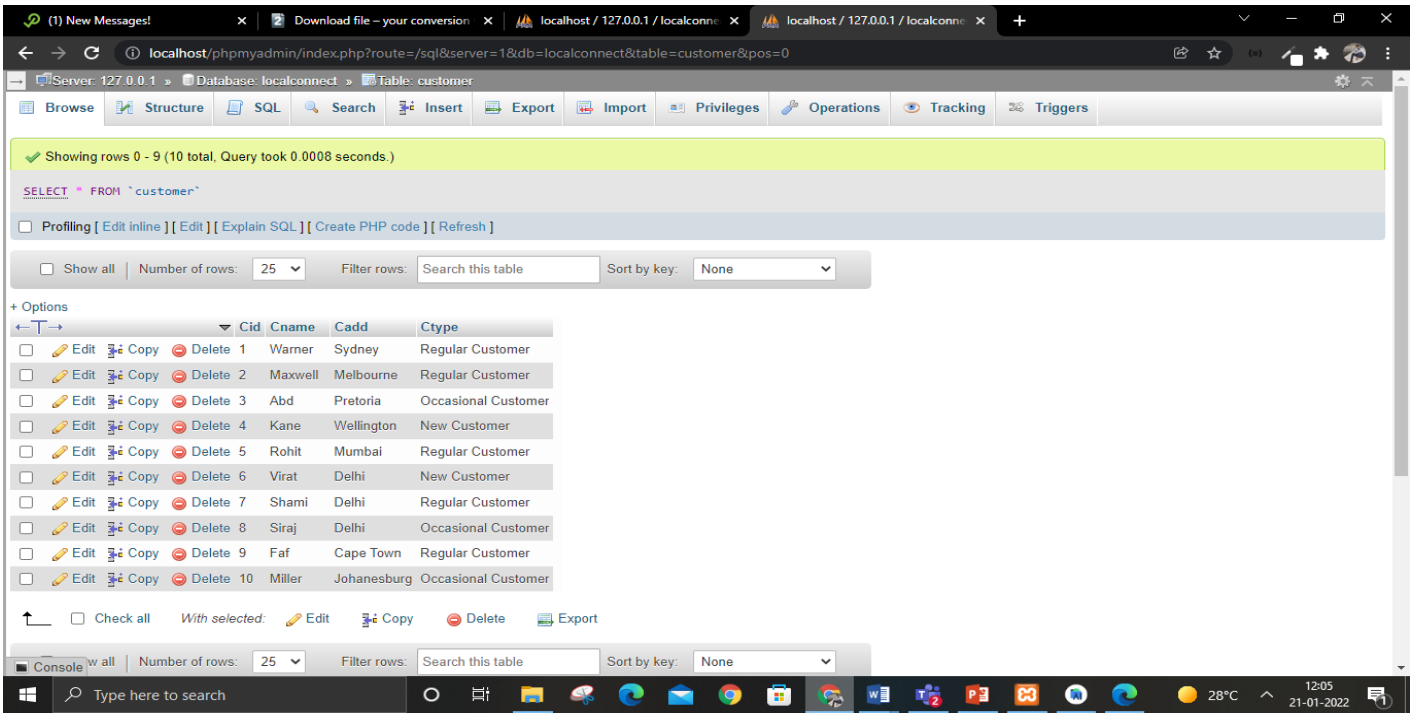
select * from driver;



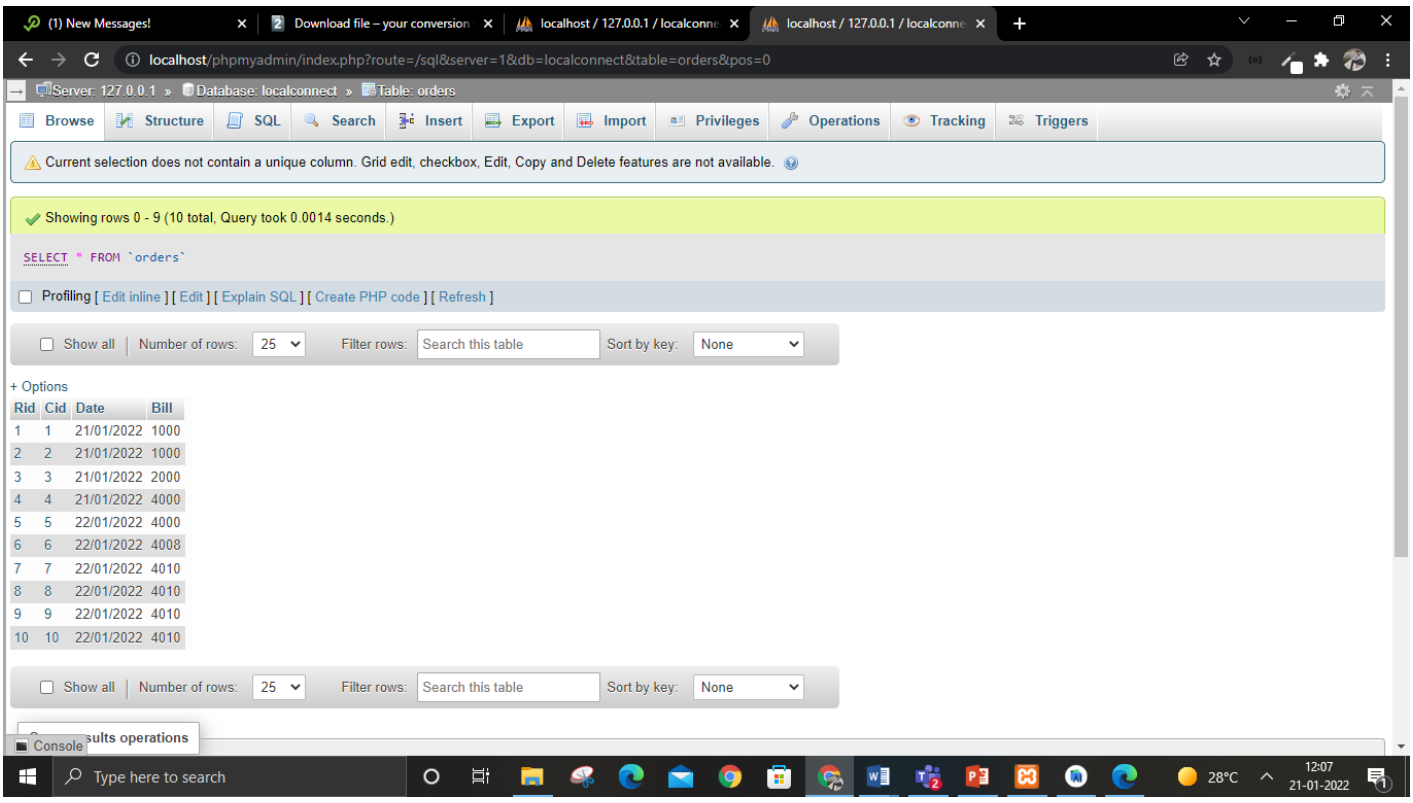
select * from restaurant;



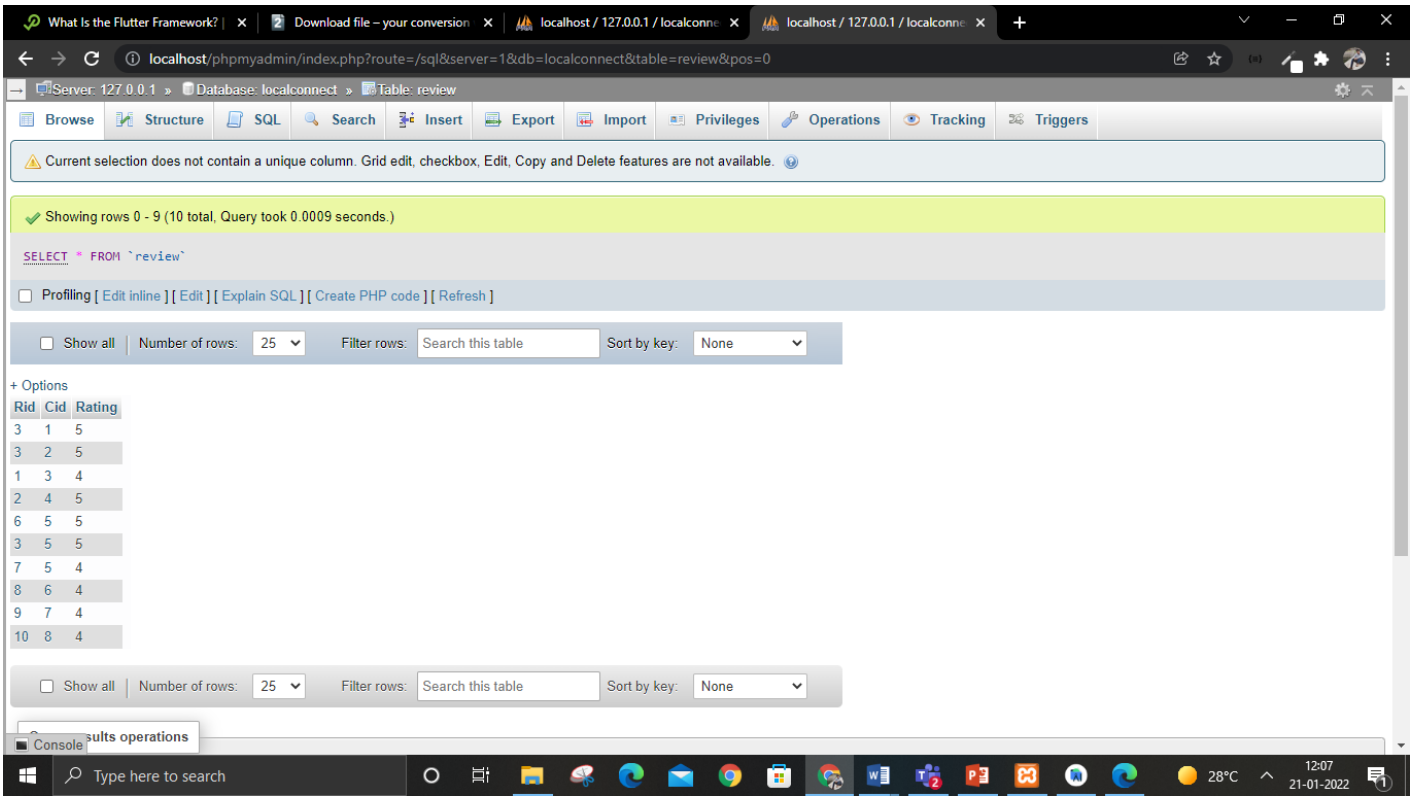
Select * from customer;



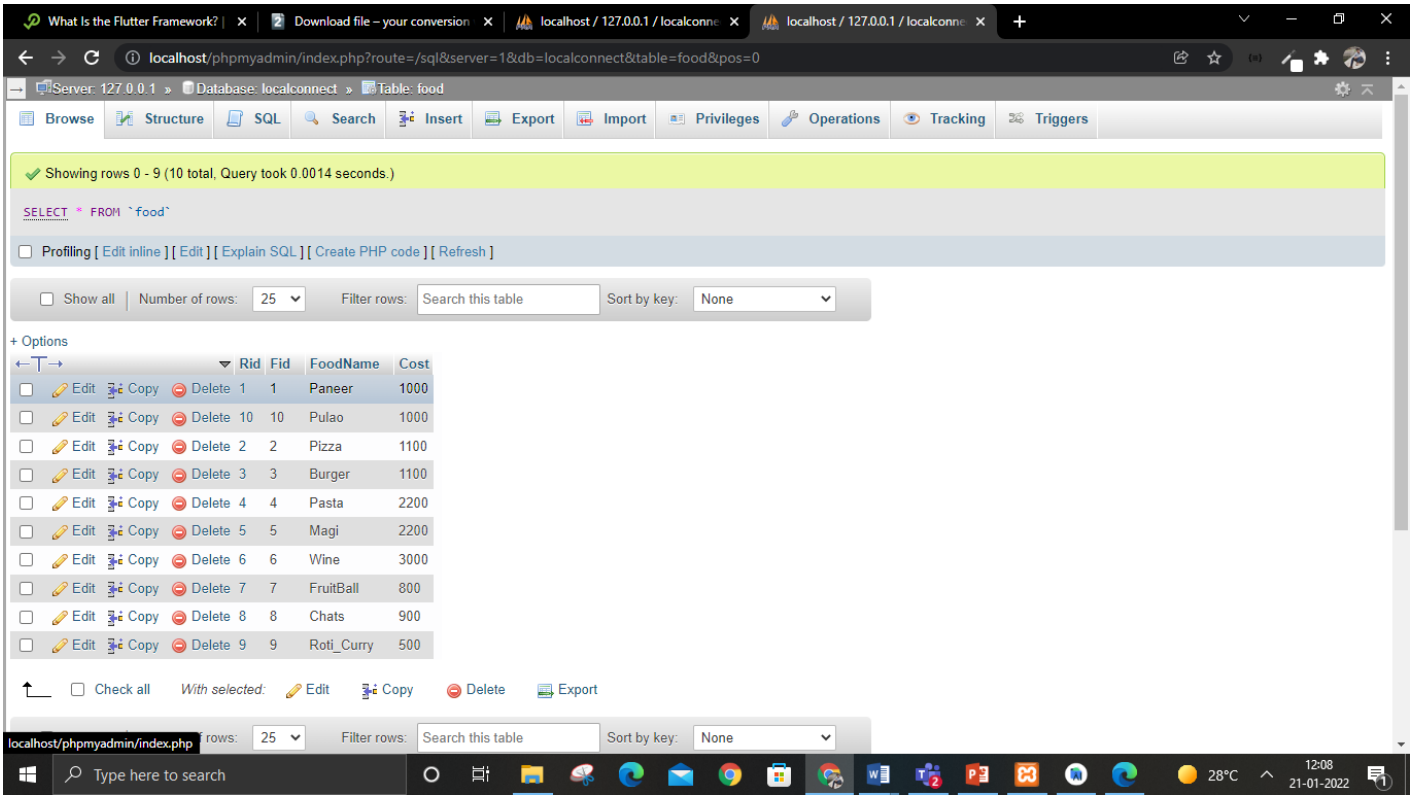
Select * from orders;



Select * from review;



Select * from food;



4.4 SQL Triggers and Stored Procedures

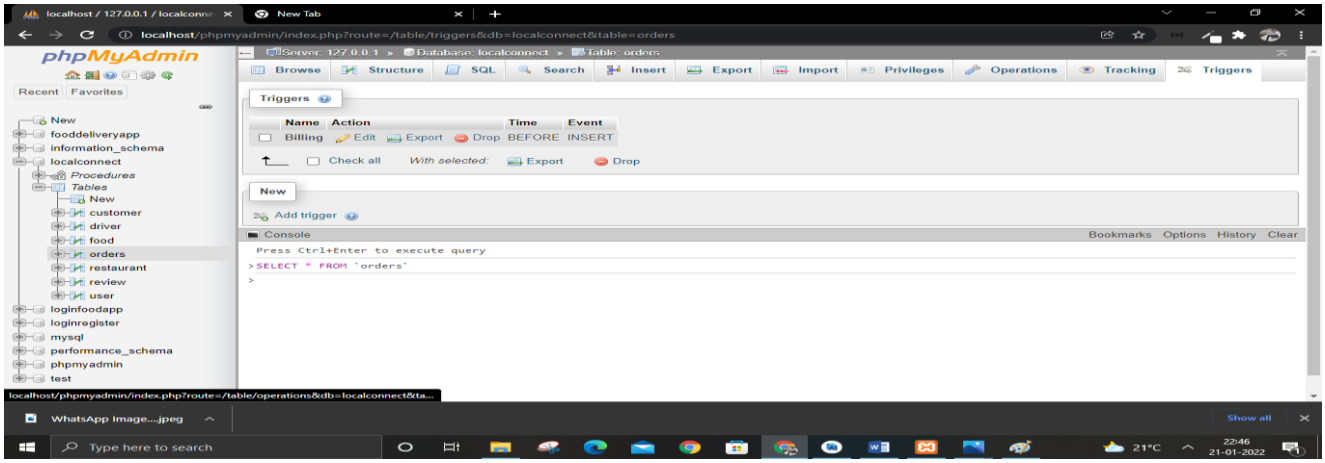
4.4.1 Trigger

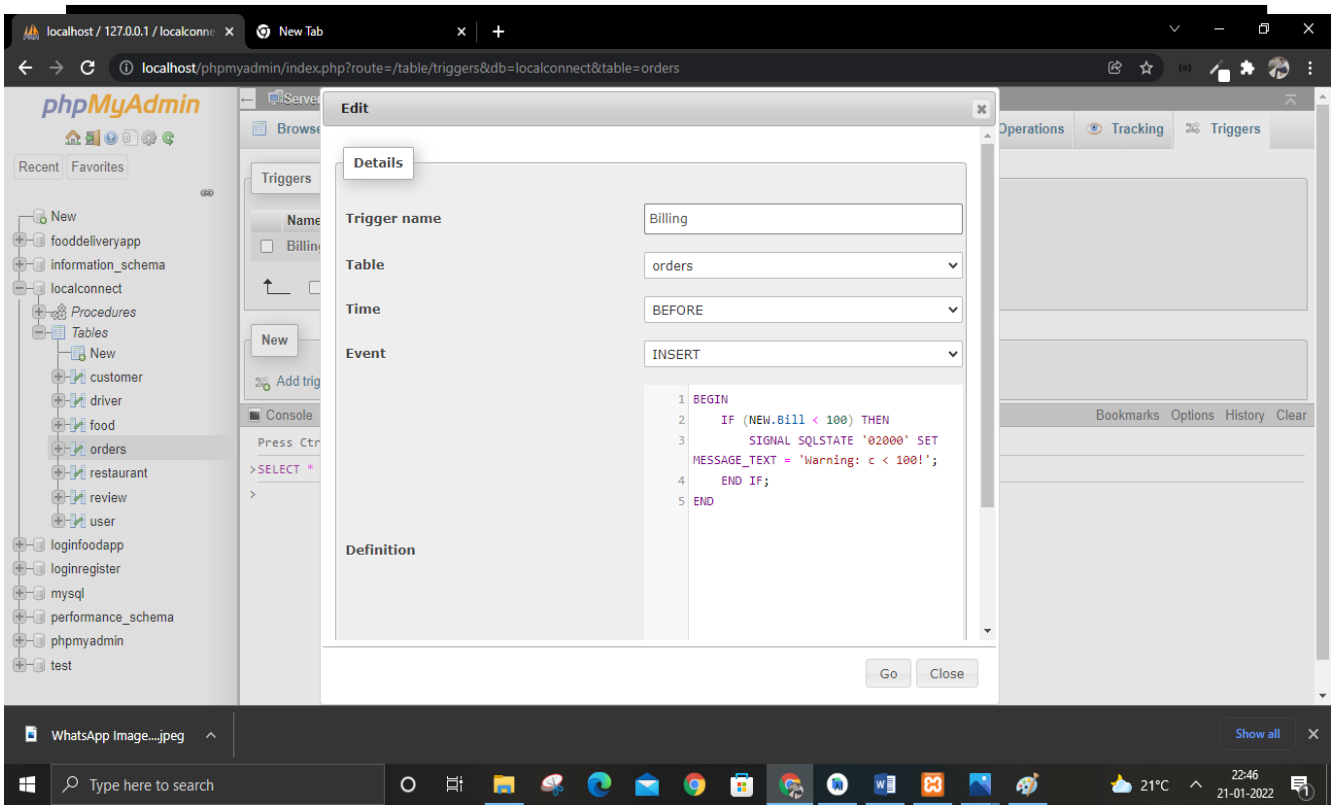
A database trigger is procedural code that is automatically executed in response to certain events on a particular table or view in a database. The trigger is mostly used for maintaining the integrity of the information on the database. Triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view.

Trigger I have used is to make sure the Bill in order table is greater than 100

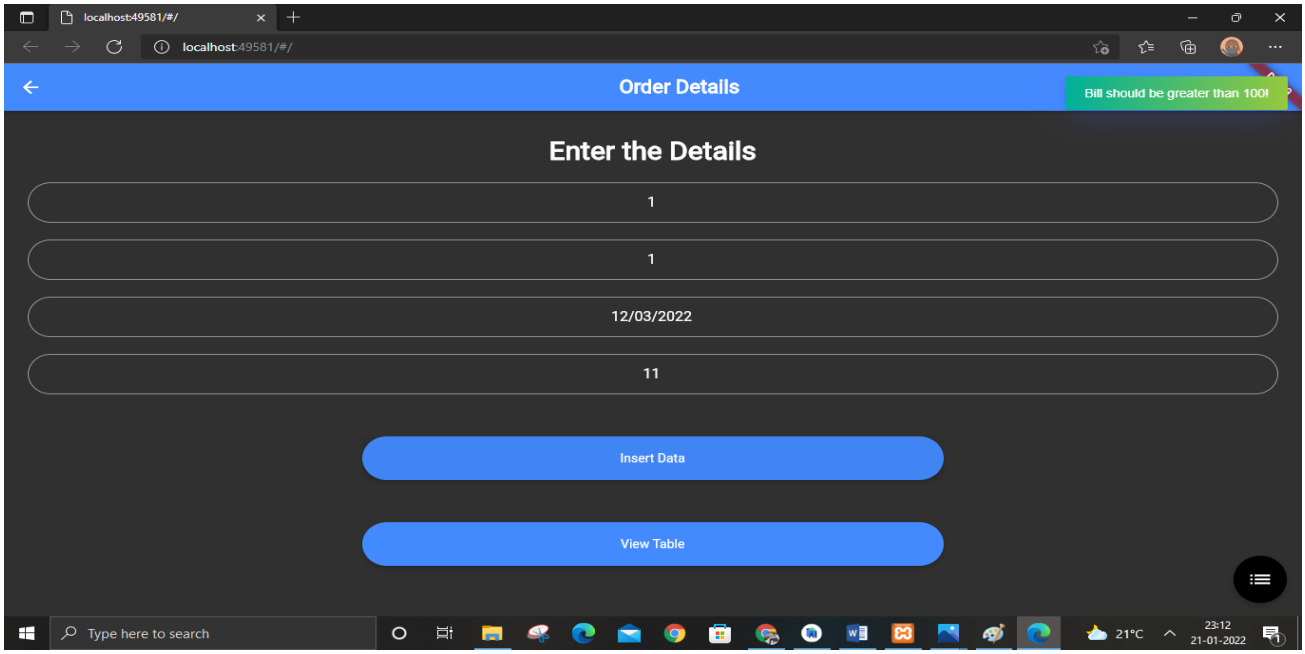
Here is the Query

```
DELIMITER $$
CREATE TRIGGER `Billing` BEFORE INSERT ON `orders`
 FOR EACH ROW BEGIN
   IF (NEW.Bill < 100) THEN
      SIGNAL SQLSTATE '02000' SET MESSAGE_TEXT = 'Warning: c < 100!';
   END IF;
END$$
DELIMITER ;
```





Because of this trigger we are not allowed to insert Bill less than 100 as shown in figure Below

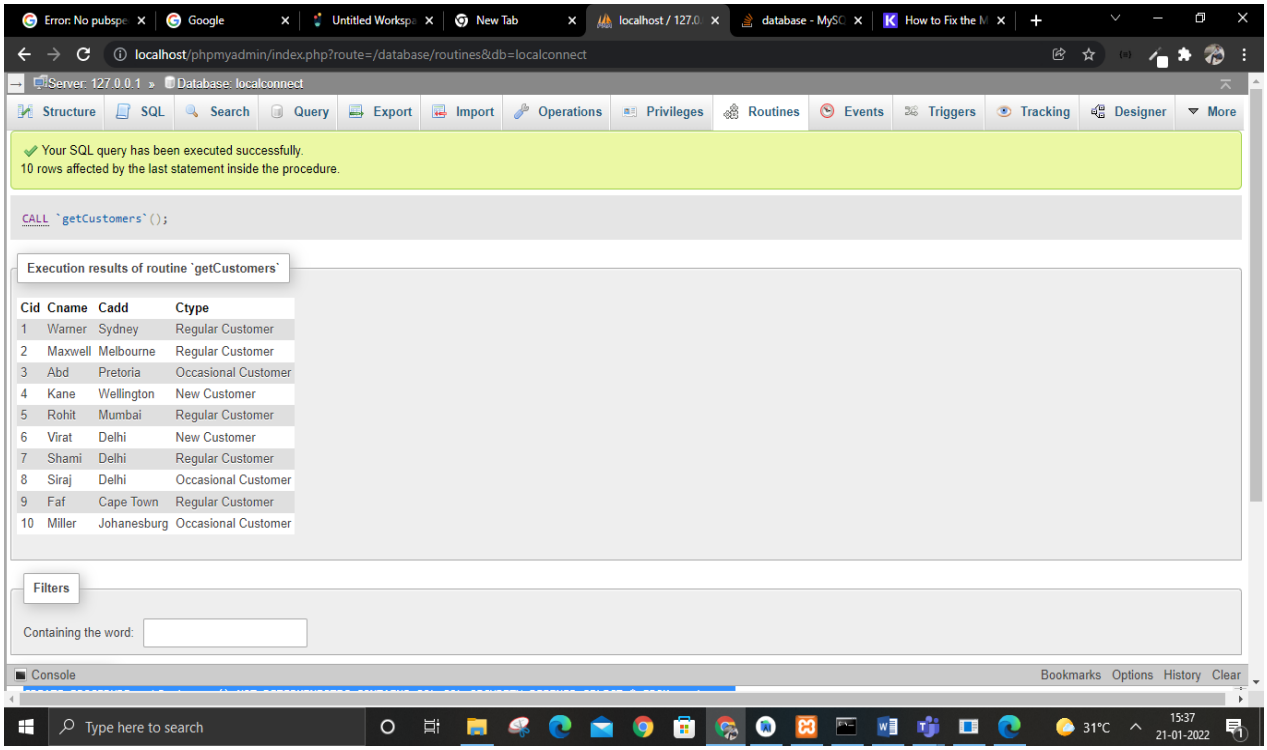


4.4.2 Stored Procedure

A stored procedure is a set of Structured Query Language (SQL) statements with an assigned name, which are stored in a relational database management system as a group. So, if a query has to be written over and over again, instead of having to write that query each time, it can be saved as a stored product and can be executed just by calling the procedure. In addition, parameters can also be passed to the stored procedure. So, depending on the need, the stored procedure can act accordingly.

Stored Procedure I have created is shown below:-

```
CREATE PROCEDURE getCustomers() NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER
SELECT * FROM customer;
```



\

4.5 Database Connectivity

A Database connection is a facility in computer science that allows client software to talk to database server software, whether on the same machine or not. A connection is required to send commands and receive answers, usually in the form of a result set. PHP has a pretty straight forward method to working with MySQL databases.

There are five steps to make PHP database interaction

1. Create a connection
2. Select database
3. Perform database query
4. Use return data
5. Close connection

Php Code to insert data into Database

```
<?php
header("Access-Control-Allow-Origin: *");
$db = mysqli_connect("localhost","Username","Password","DatabaseName");
if(!$db){
echo "Database connection failed";
}
$var1 = $_POST['Attribute name'];
$var2 = $_POST[' Attribute name''];
$var3 = $_POST[' Attribute name''];
//echo "User name $username $password";
$sql = "INSERT INTO tablename(Attribute name', Attribute name', Attribute
name') VALUES('".$var1."','".$var2."','".$var3."')";
$query = mysqli_query($db, $sql);
if($query)
{
echo ("success");
}
?>
```

Php Code to retrieve data from DataBase

```
$connection = mysqli_connect("localhost","Username","Password","DatabaseName") or
die("Error " . mysqli_error($connection));
//fetch table rows from mysql db
$sql = "select * from food";
$result = mysqli_query($connection, $sql) or die("Error in Selecting " .
mysqli_error($connection));
```

```
    //create an array
    $emparray = array();
    while($row =mysqli_fetch_assoc($result))
    {
        $emparray[] = $row;
    }
    echo json_encode($emparray);

    //close the db connection
    mysqli_close($connection);
?>
```

PhpCode for Updation and deletion

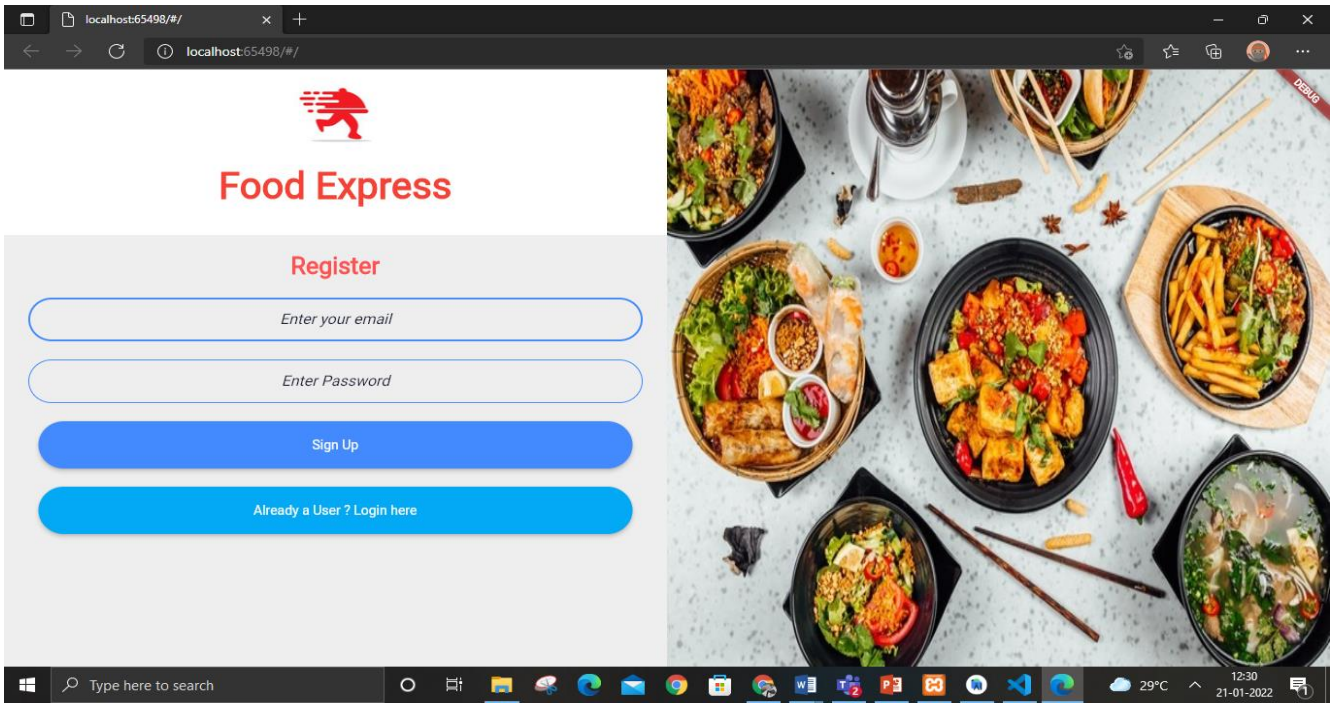
(Note I am writing query in my Ui to update and delete.So my Php file looks like this)

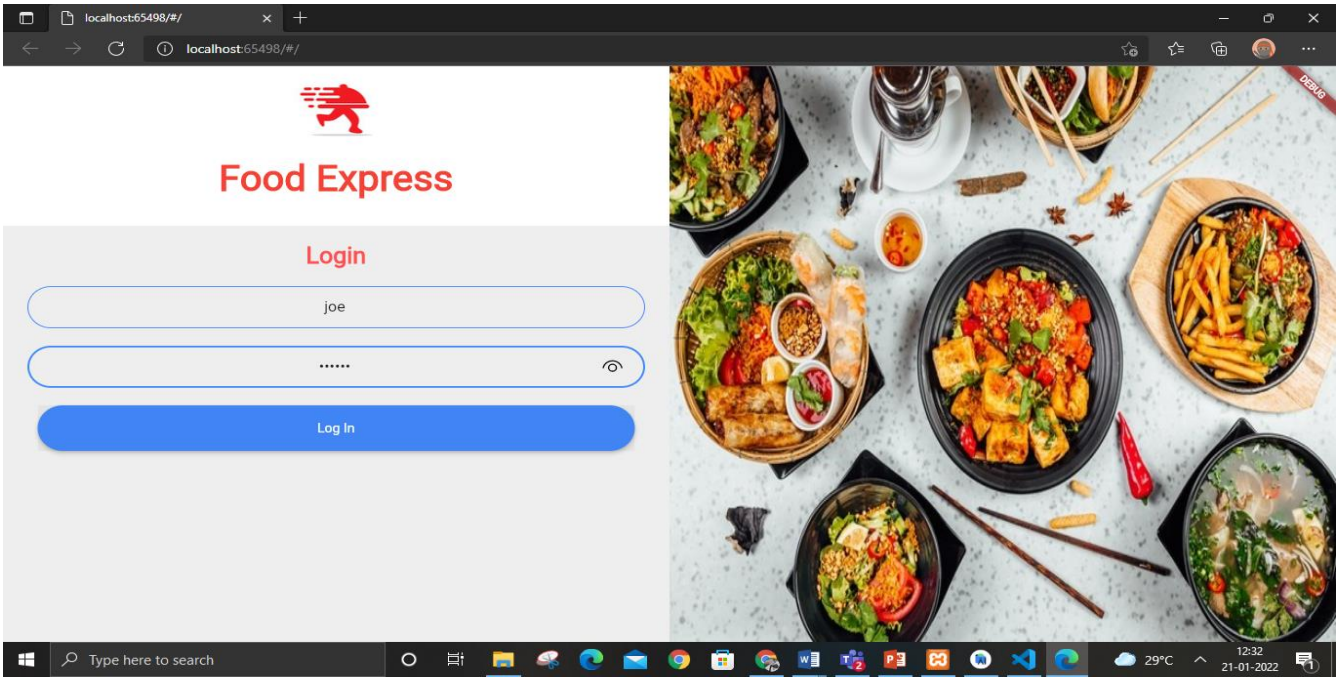
```
    <?php
    header("Access-Control-Allow-Origin: *");
    $db = mysqli_connect('localhost','root','Pitac123','localconnect');

    if(!$db){
      echo "Database connection failed";
    }
    $upd = $_POST['upd'];
    //echo "User name $username $password";
    $sql = "$upd";
    $query = mysqli_query($db, $sql);
    if($query)
   {
    echo ("success");
   }
   ?>
```

Chapter 5

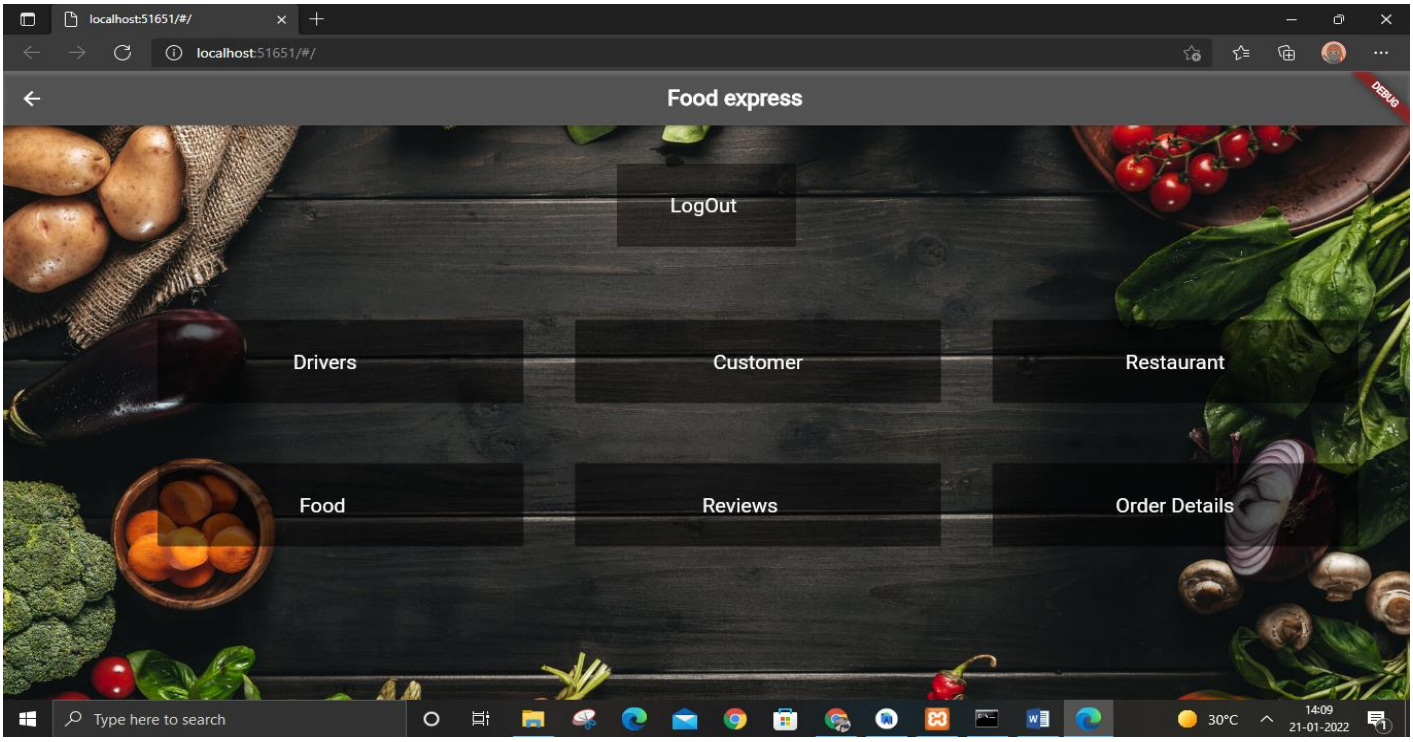
RESULT

Login page which contains views for users registration and Login

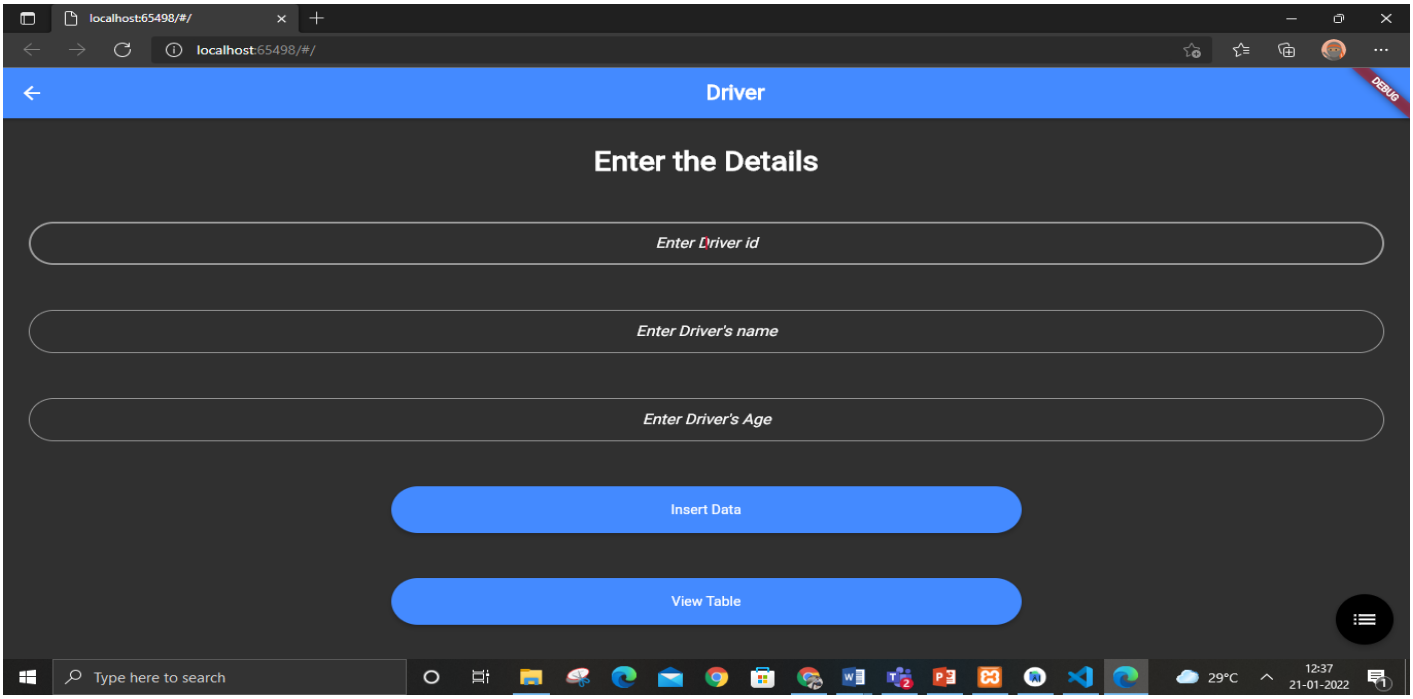


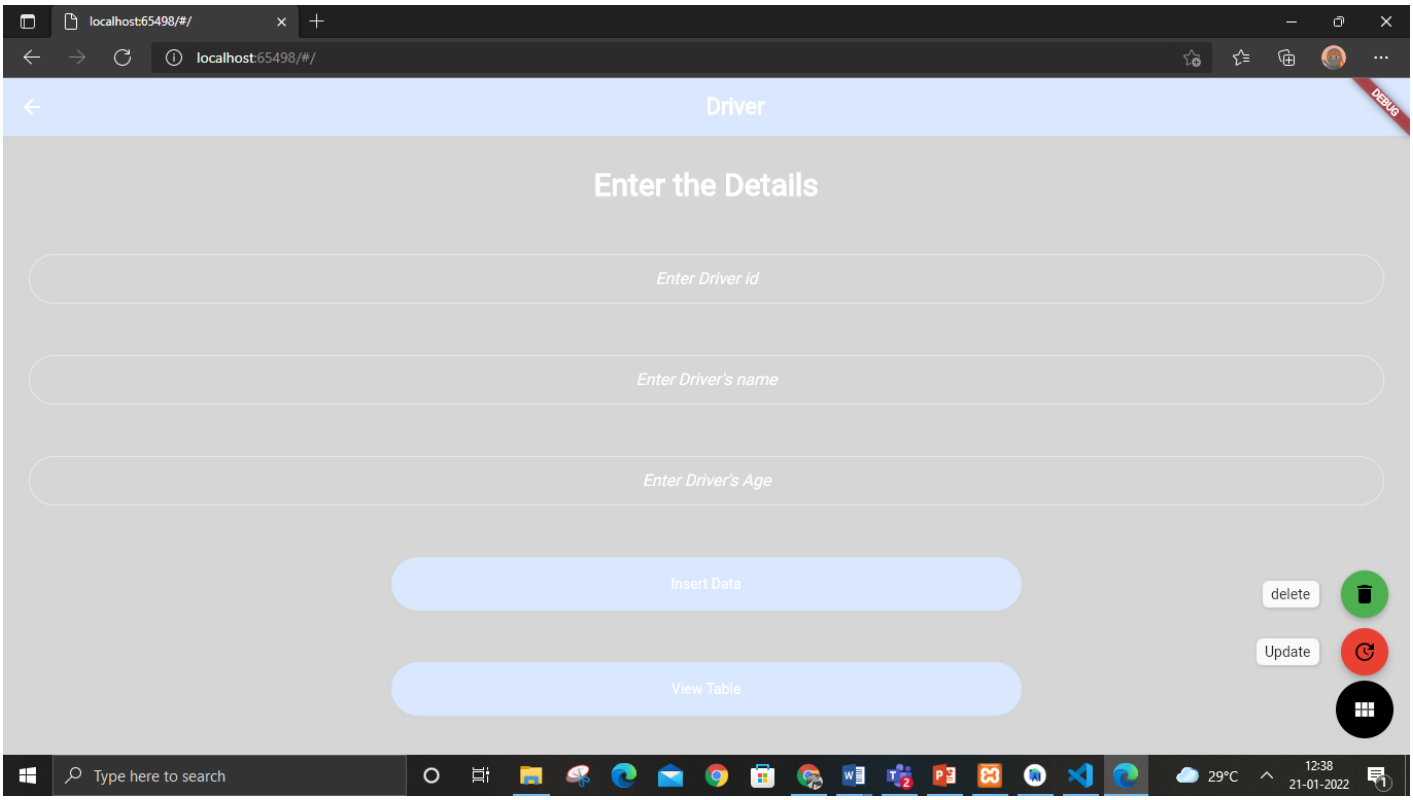


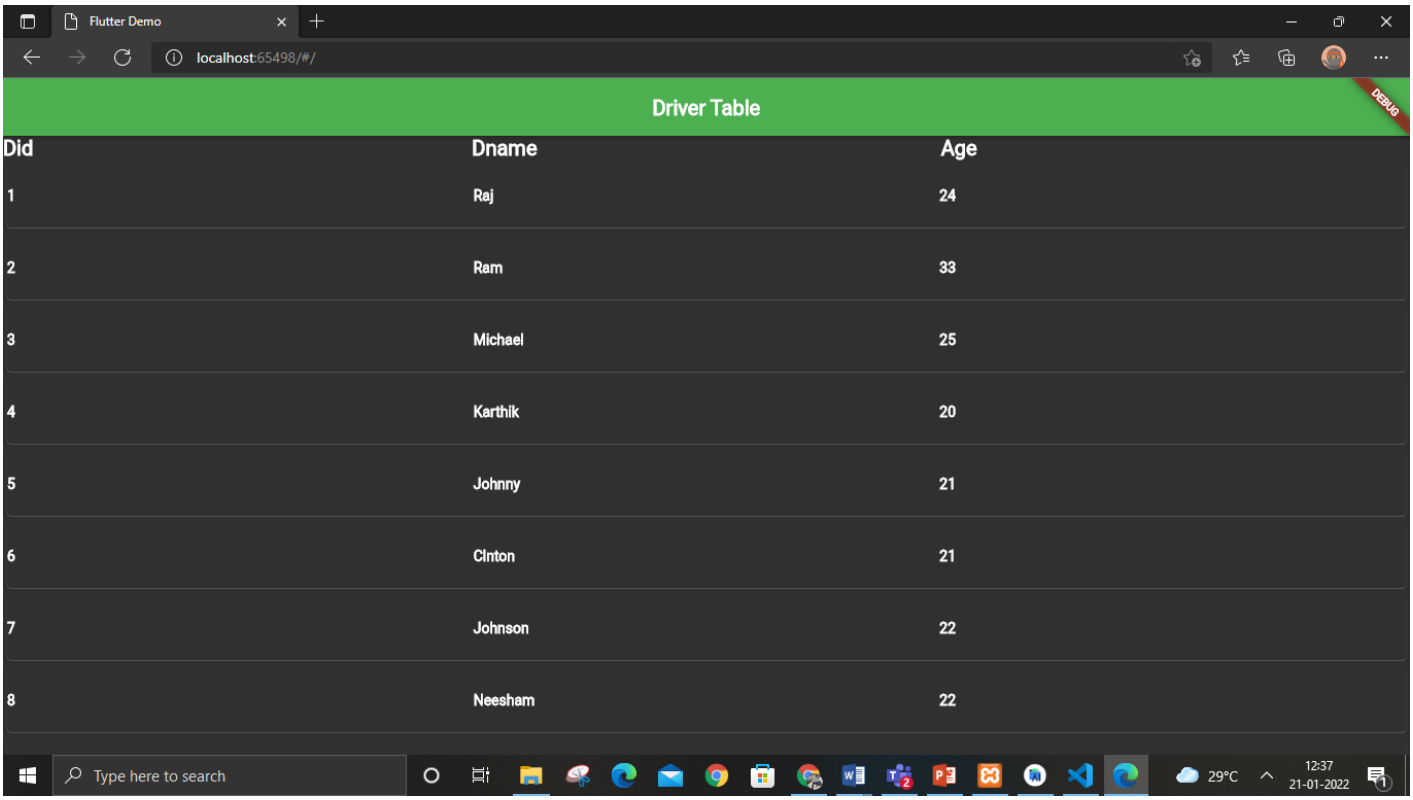
Home Page contains the different tables and Logout option

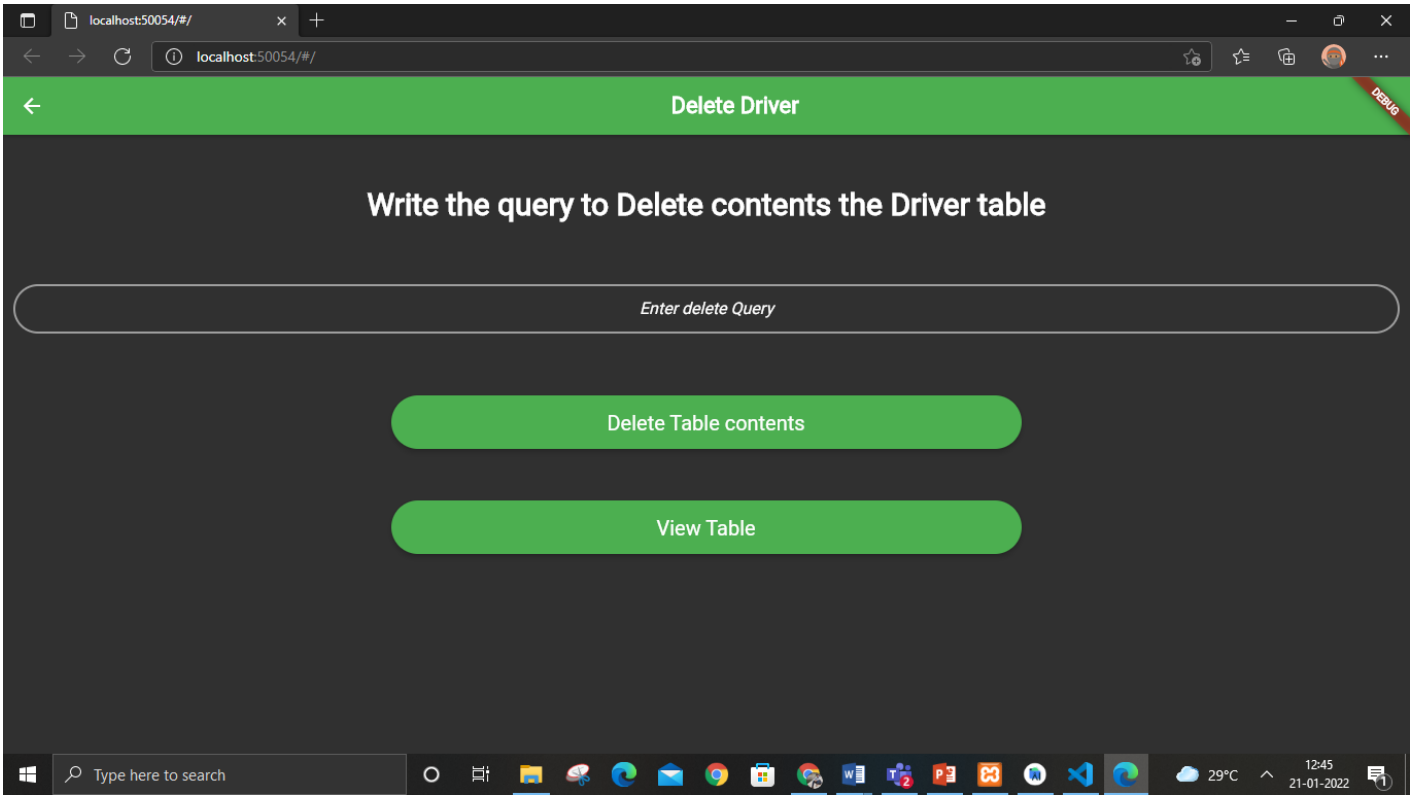


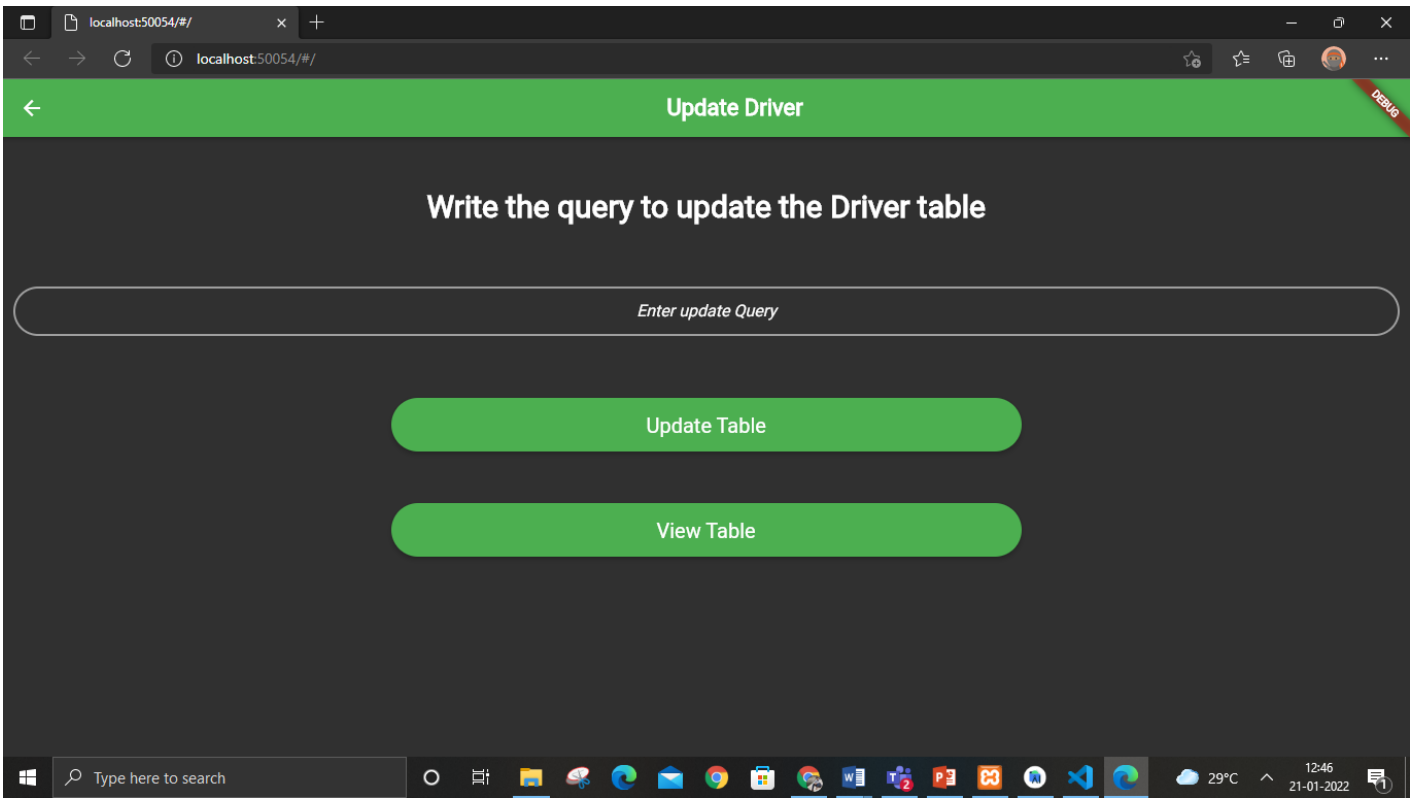
Drivers Table



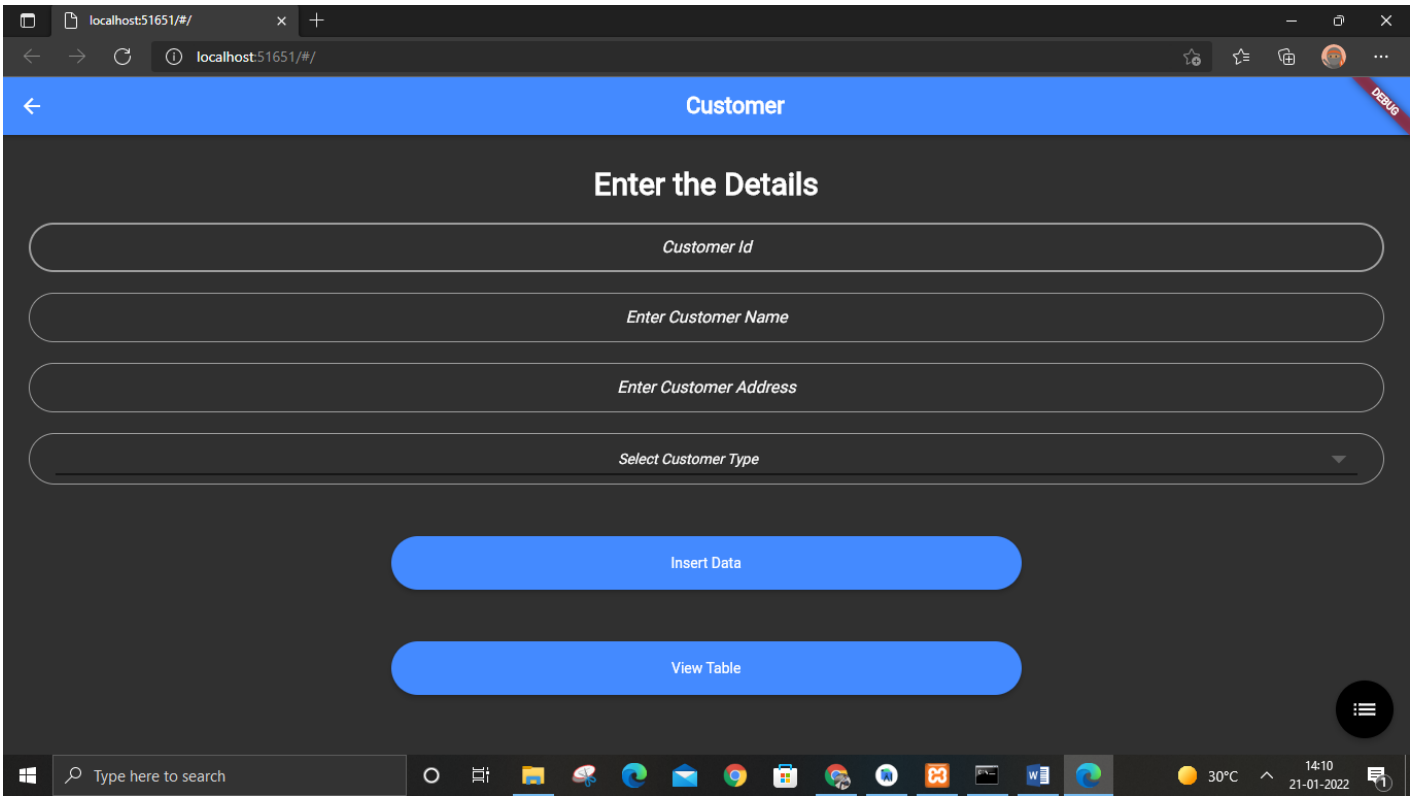

localhost:65498/#/
Driver
DEBUG
Enter the Details
Enter Driver id
Enter Driver's name
Enter Driver's Age
Insert Data
View Table
delete
Update
Type here to search
29°C
12:38
21-01-2022

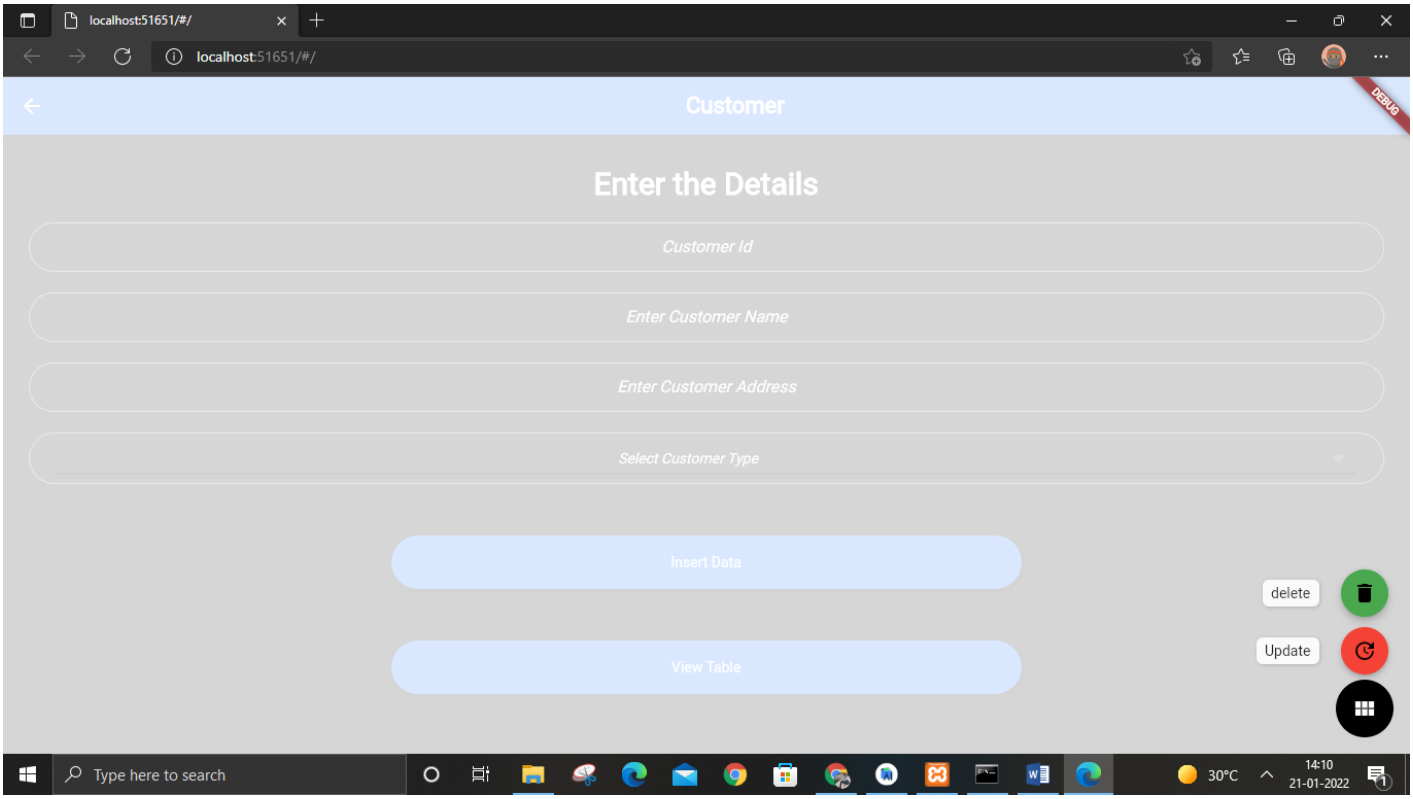

Flutter Demo
localhost:65498/#/
Driver Table
DEBUG
Did
Dname
Age
1 Raj 24
2 Ram 33
3 Michael 25
4 Karthik 20
5 Johnny 21
6 Clnton 21
7 Johnson 22
8 Neesham 22
Type here to search
29°C
12:37
21-01-2022

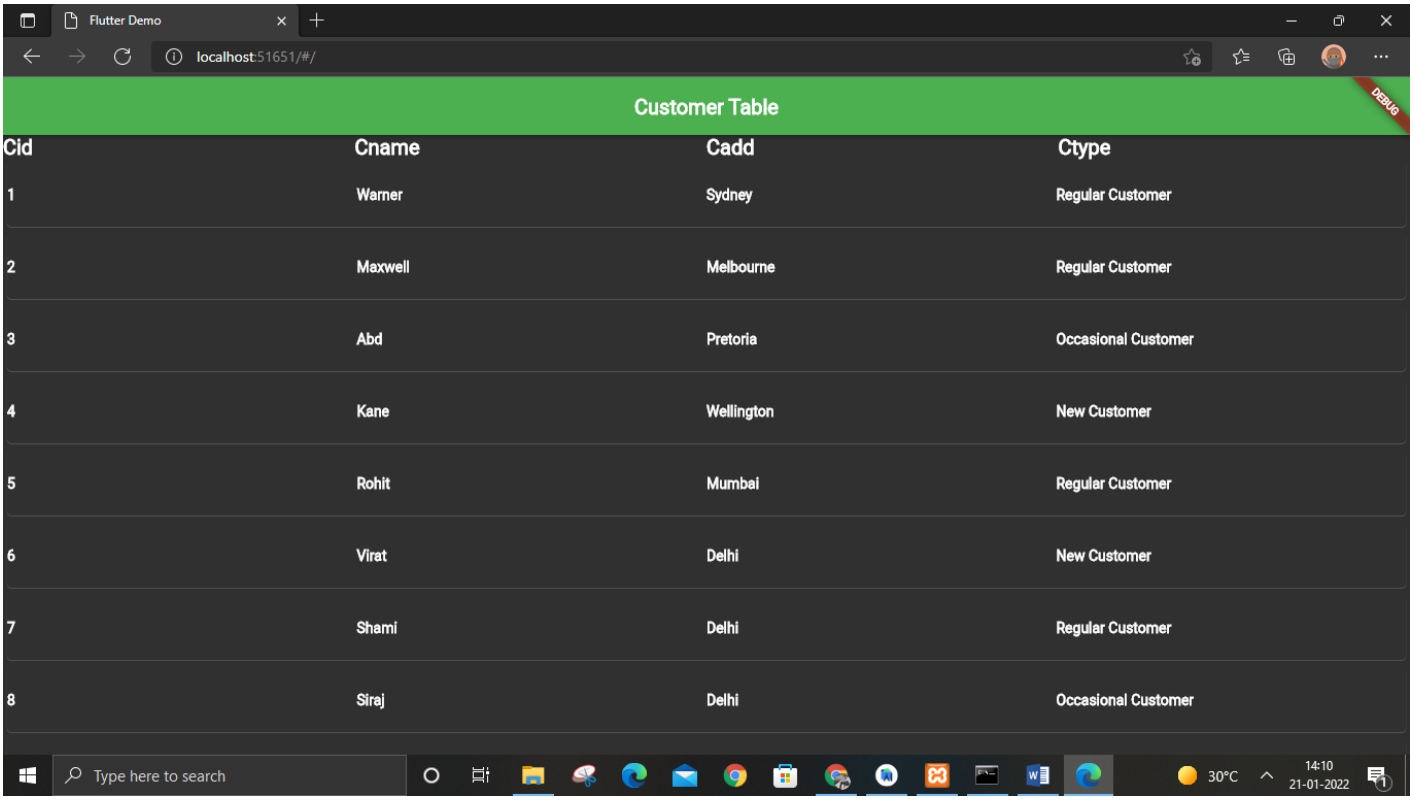

localhost:50054/#/
Delete Driver
DEBUG
Write the query to Delete contents the Driver table
Enter delete Query
Delete Table contents
View Table
Type here to search
29°C
12:45
21-01-2022

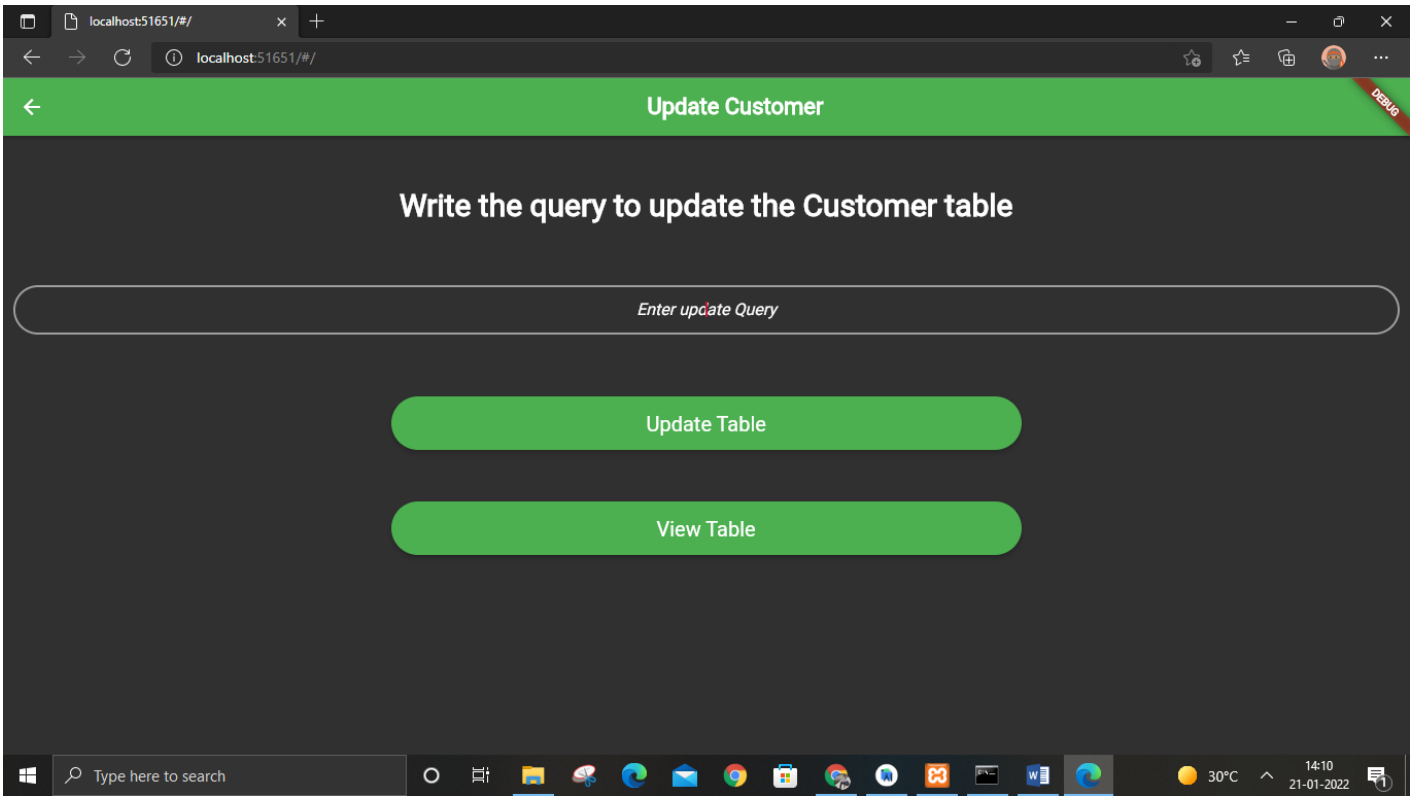

localhost:50054/#/
Update Driver
DEBUG
Write the query to update the Driver table
Enter update Query
Update Table
View Table
Type here to search
29°C
12:46
21-01-2022

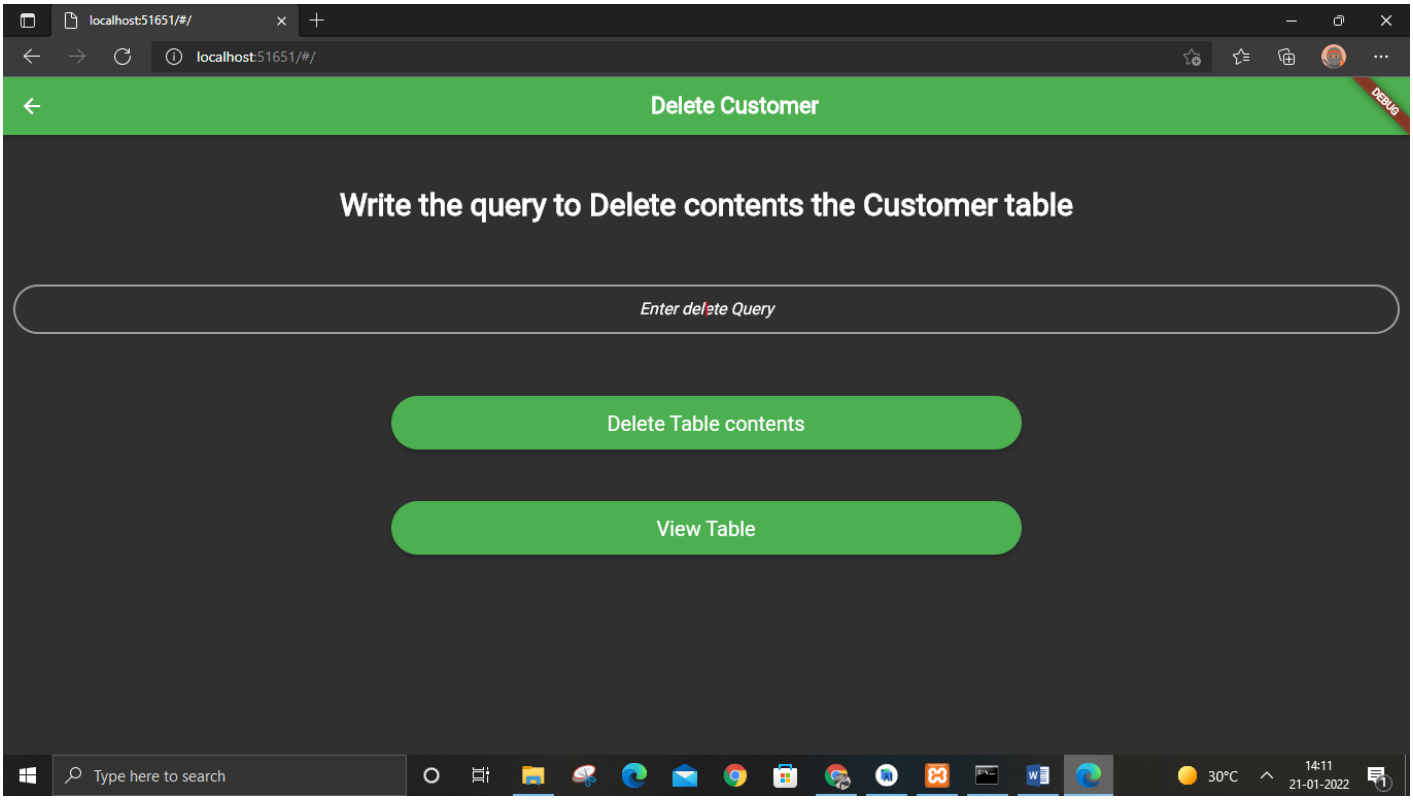
Customer Table



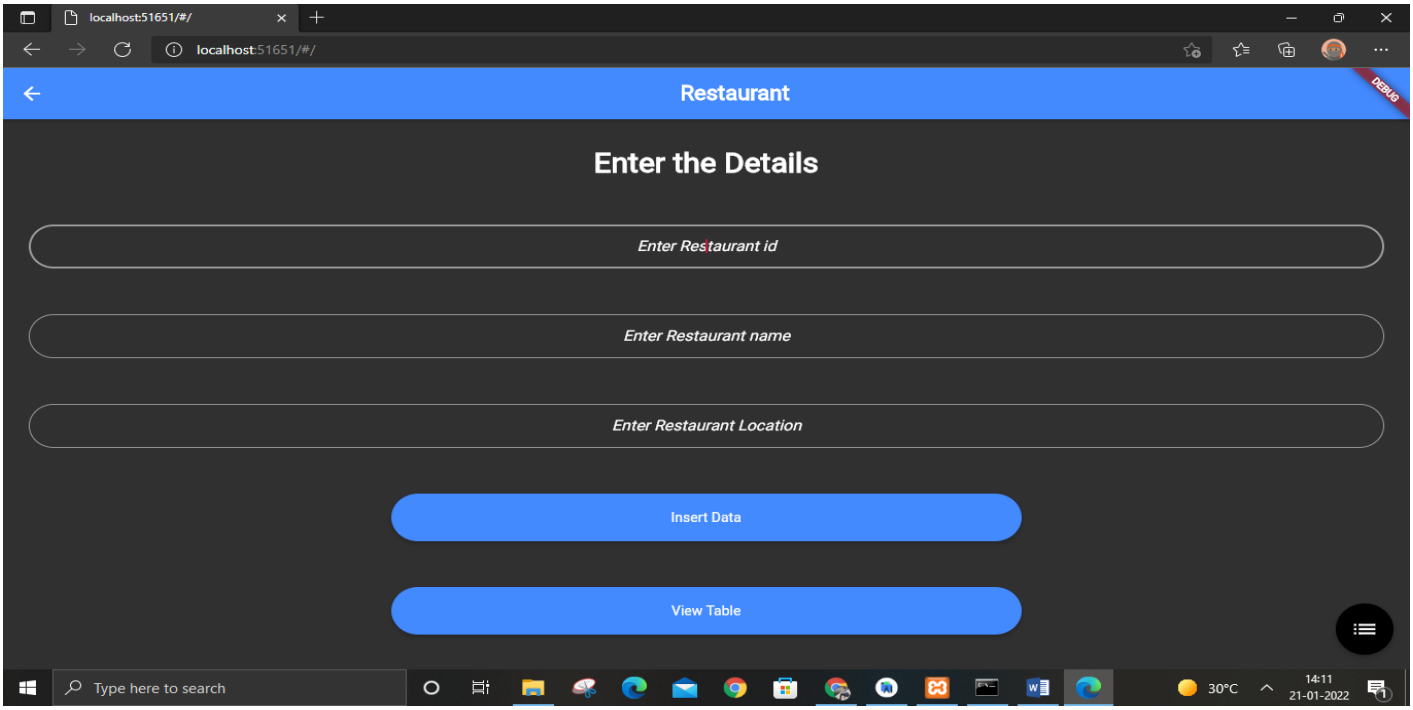


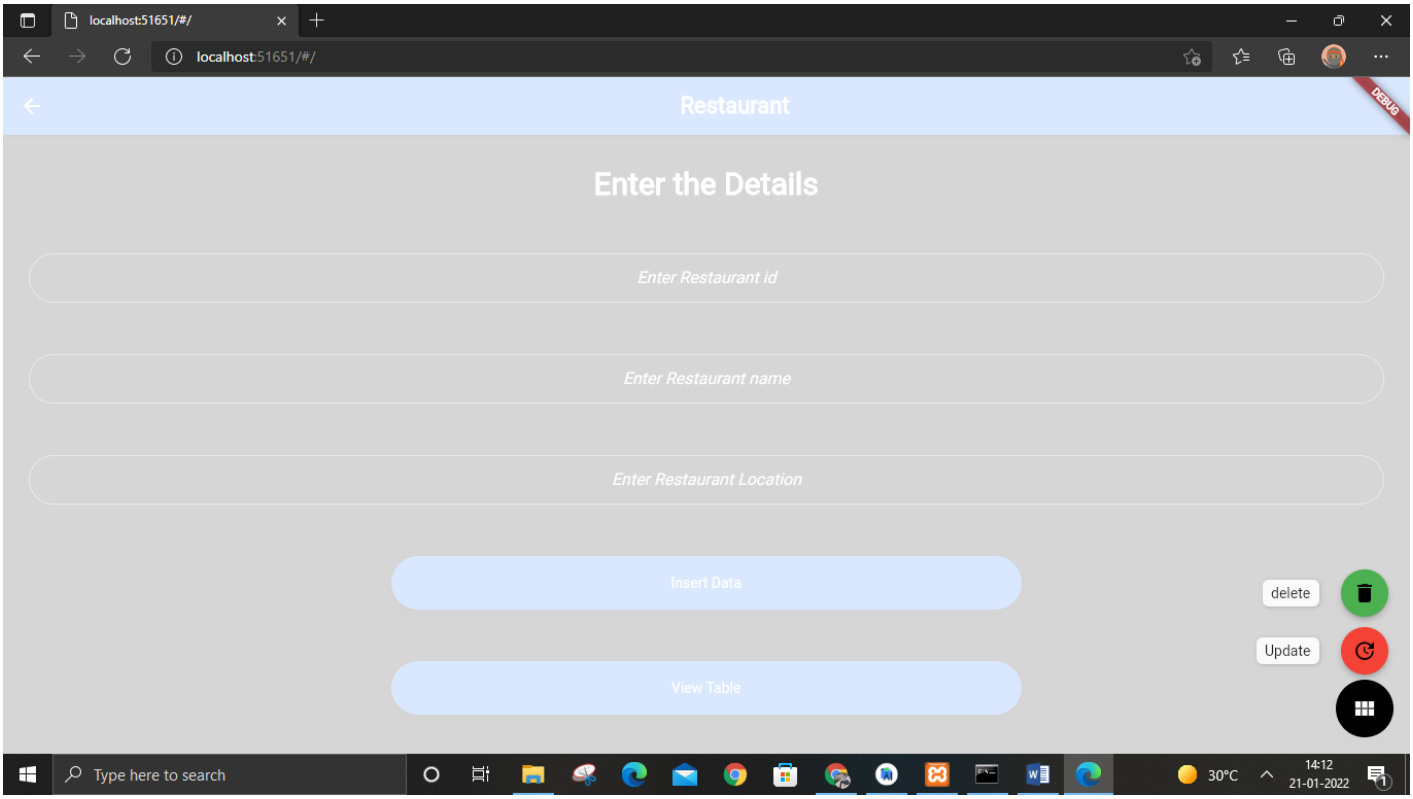

Flutter Demo
localhost:51651/#/
Customer Table
DEBUG
Cid
Cname
Cadd
Ctype
1
Warner
Sydney
Regular Customer
2
Maxwell
Melbourne
Regular Customer
3
Abd
Pretoria
Occasional Customer
4
Kane
Wellington
New Customer
5
Rohit
Mumbai
Regular Customer
6
Virat
Delhi
New Customer
7
Shami
Delhi
Regular Customer
8
Siraj
Delhi
Occasional Customer
Type here to search
30°C
14:10
21-01-2022

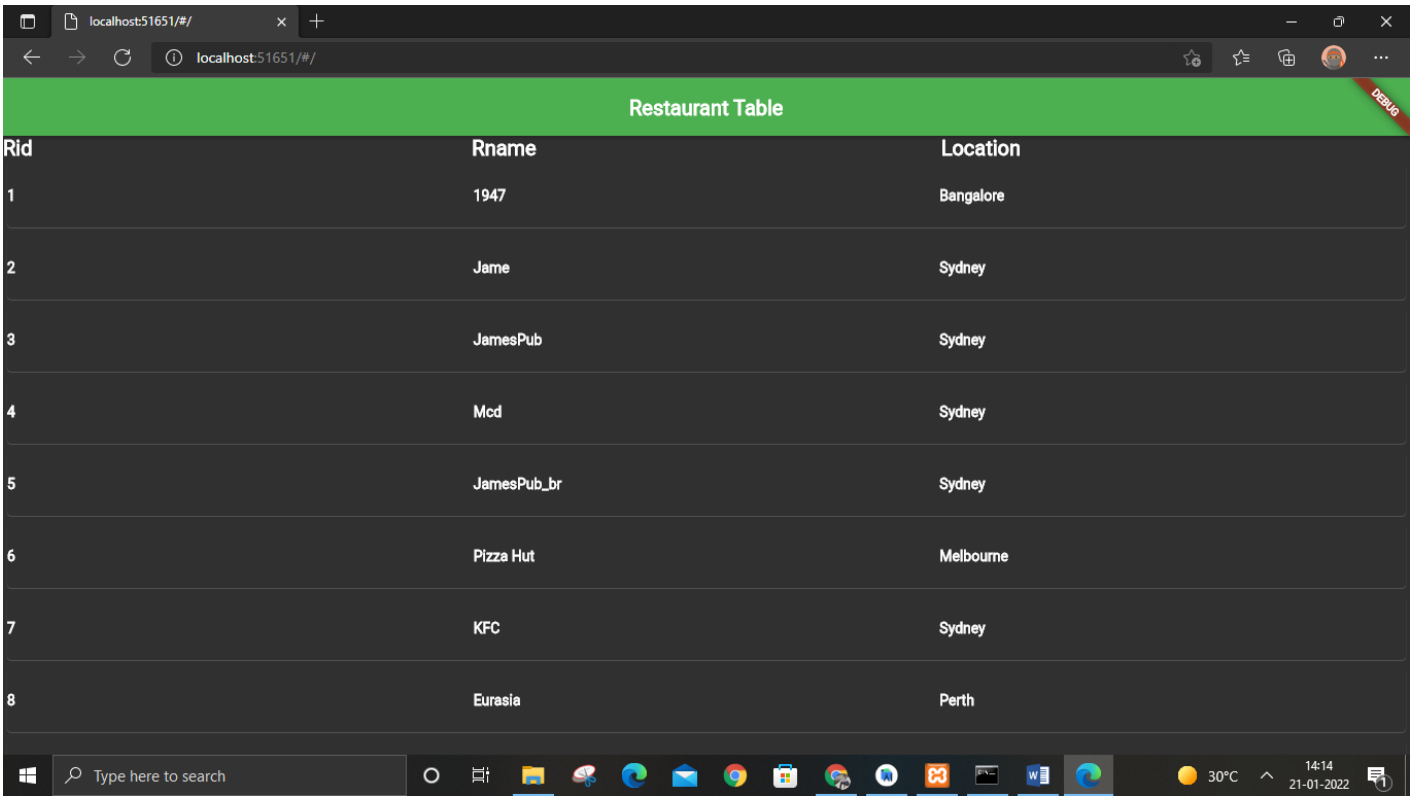

localhost:51651/#/
localhost:51651/#/
Update Customer
DEBUG
Write the query to update the Customer table
Enter update Query
Update Table
View Table
Type here to search
30°C
14:10
21-01-2022

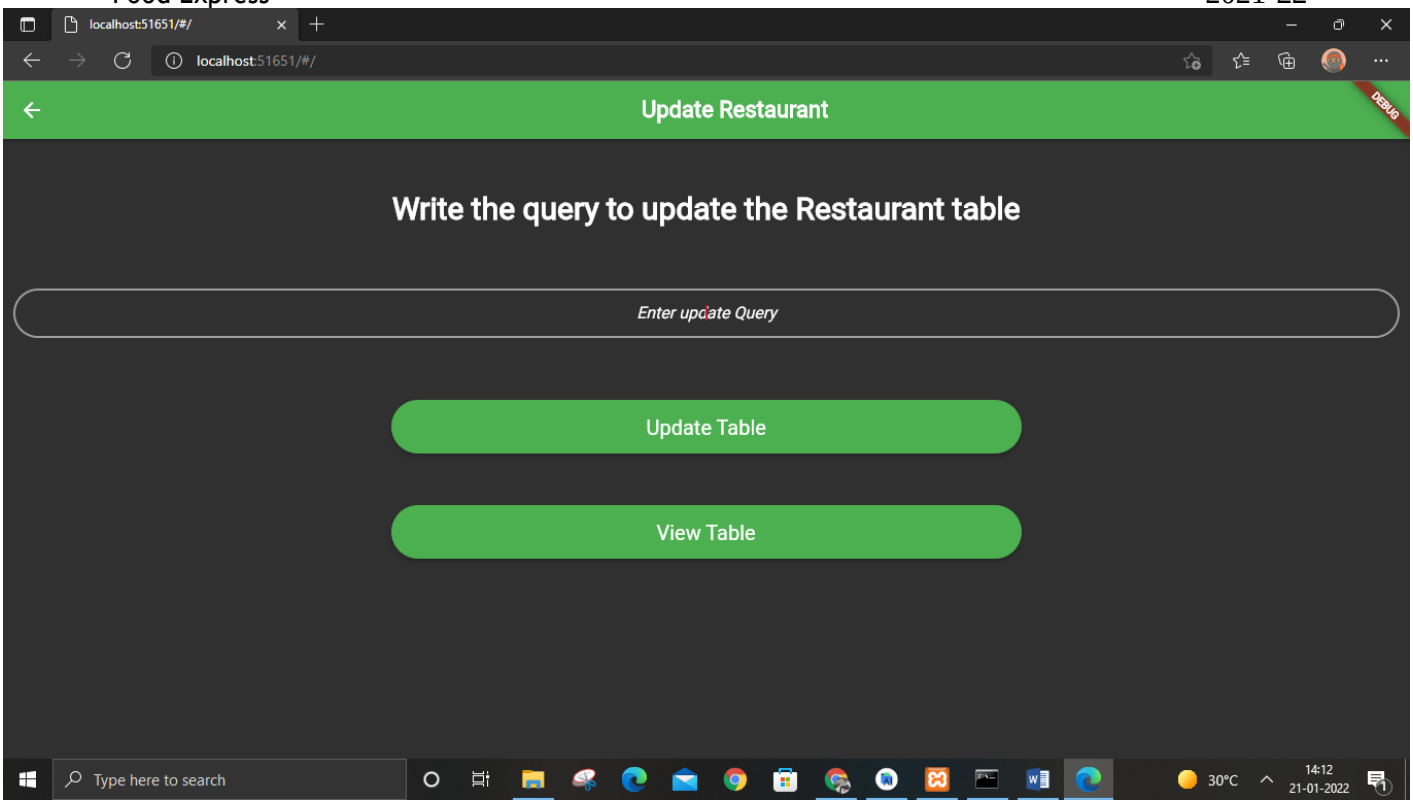


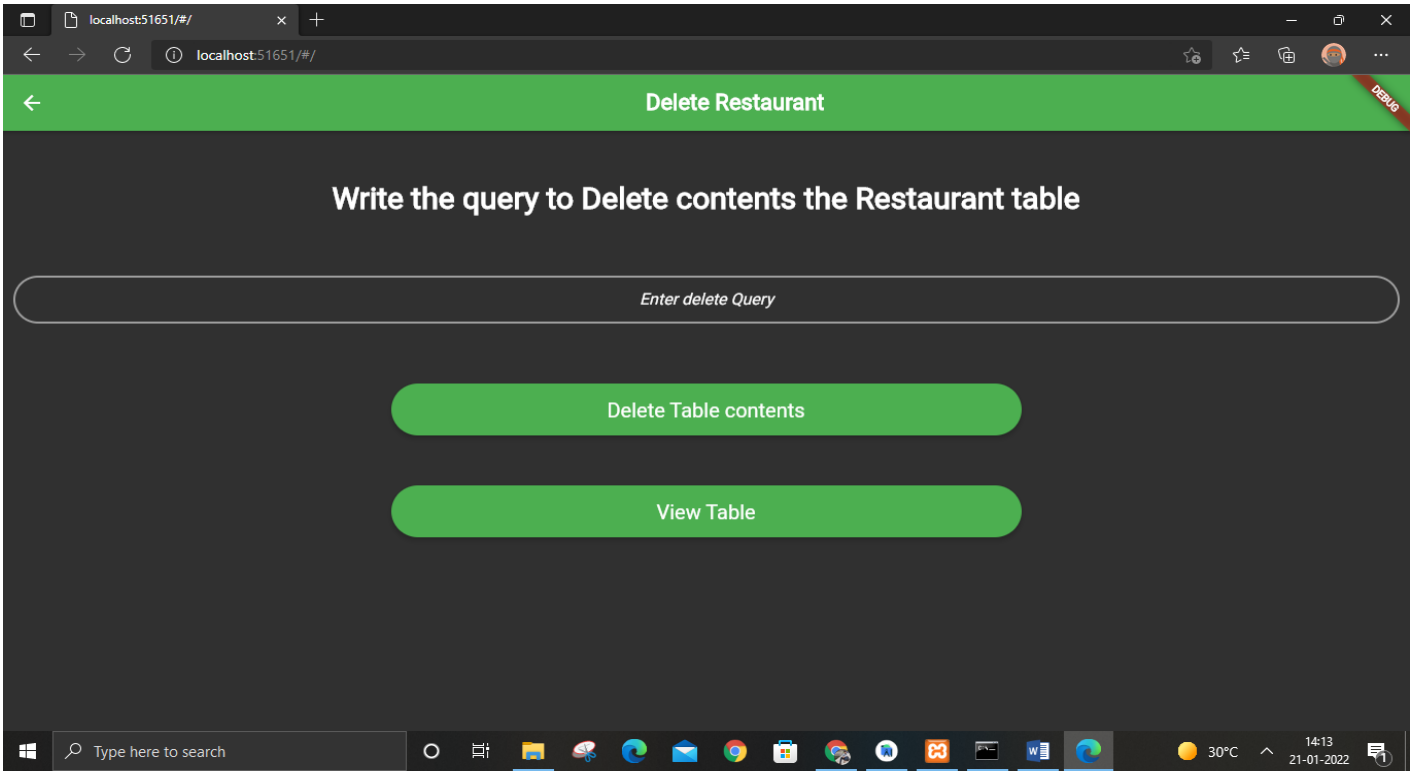
Restaurant table



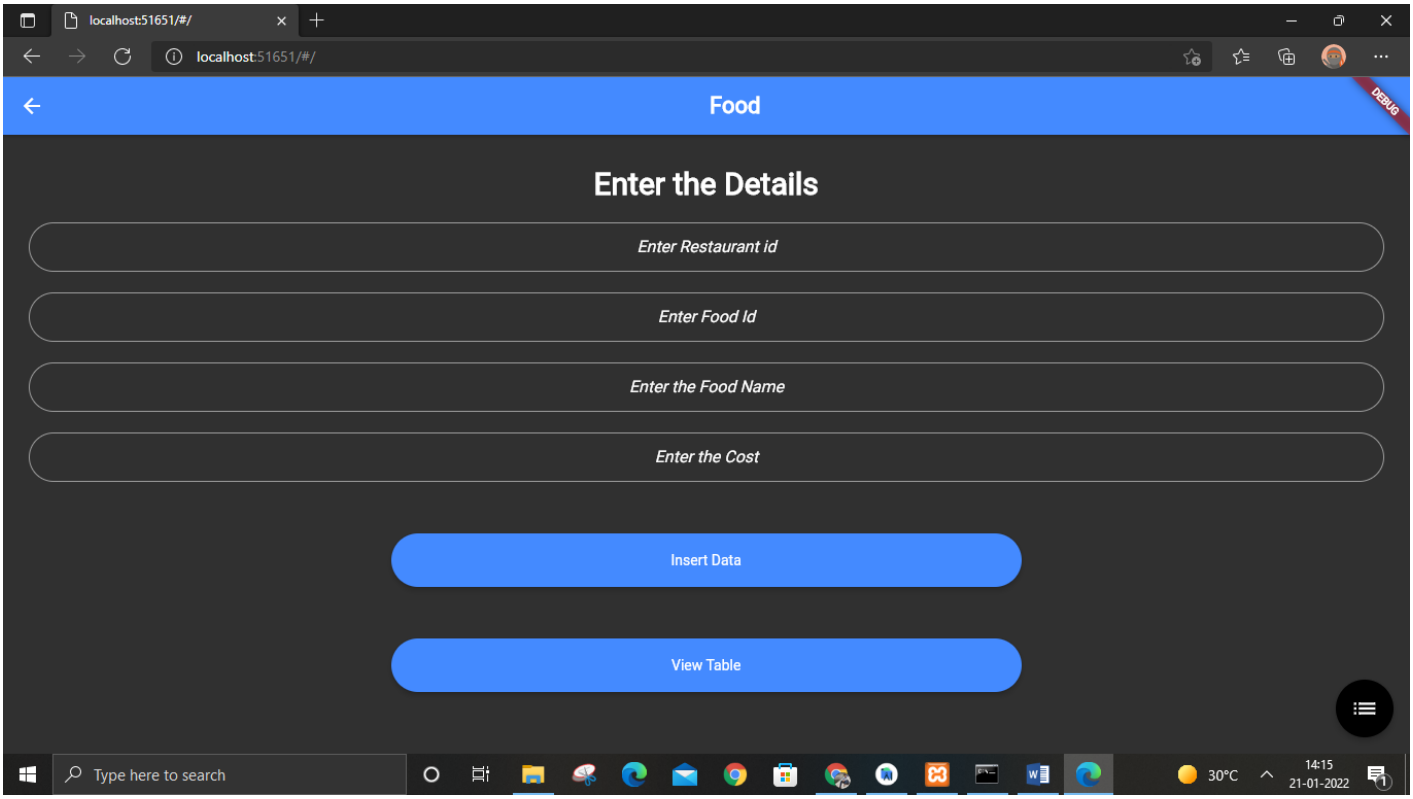

localhost:51651/#/
Restaurant
DEBUG
Enter the Details
Enter Restaurant id
Enter Restaurant name
Enter Restaurant Location
Insert Data
View Table
delete
Update
Type here to search
30°C
14:12
21-01-2022

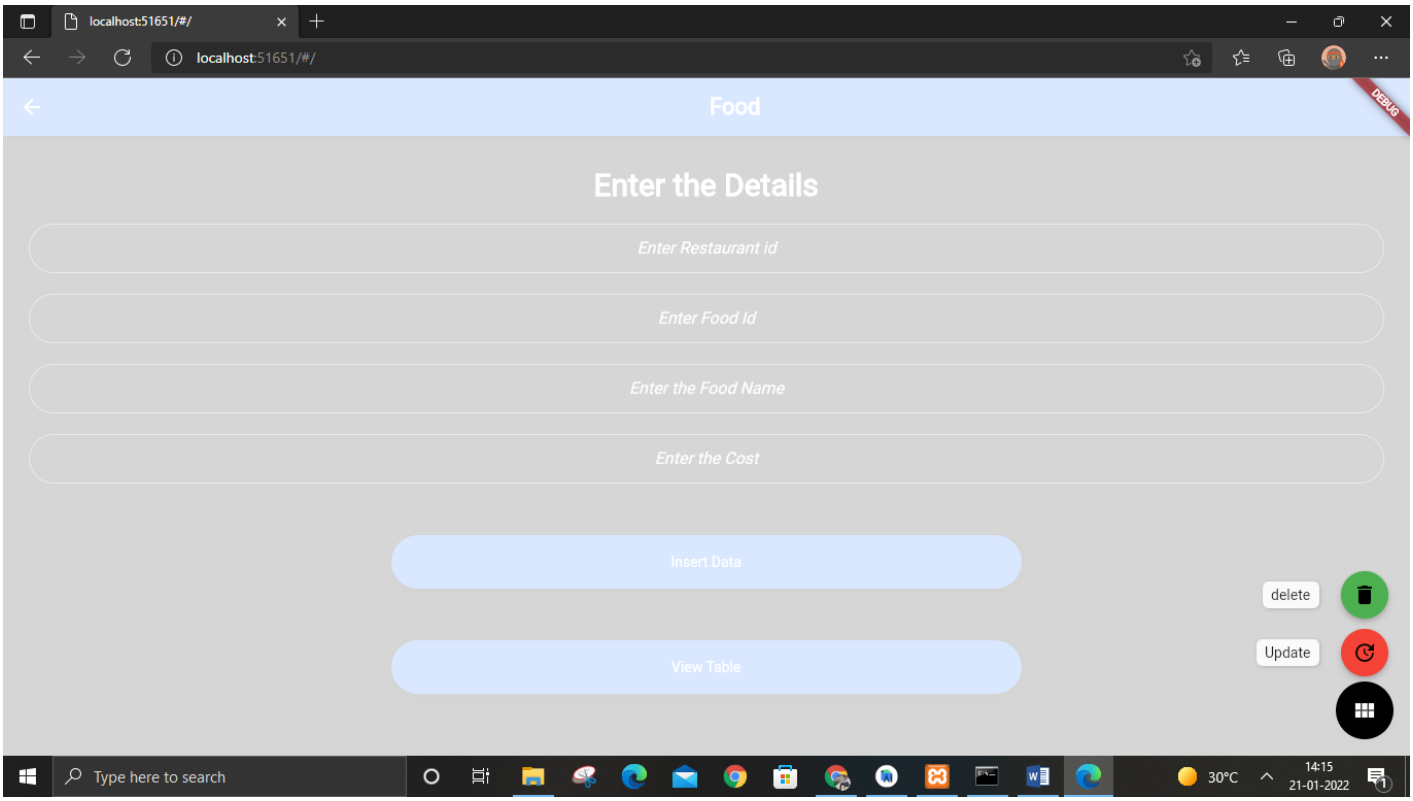

localhost:51651/#/
Restaurant Table
DEBUG
Rid
Rname
Location
1 1947 Bangalore
2 Jame Sydney
3 JamesPub Sydney
4 Mcd Sydney
5 JamesPub_br Sydney
6 Pizza Hut Melbourne
7 KFC Sydney
8 Eurasia Perth
Type here to search
30°C
14:14
21-01-2022

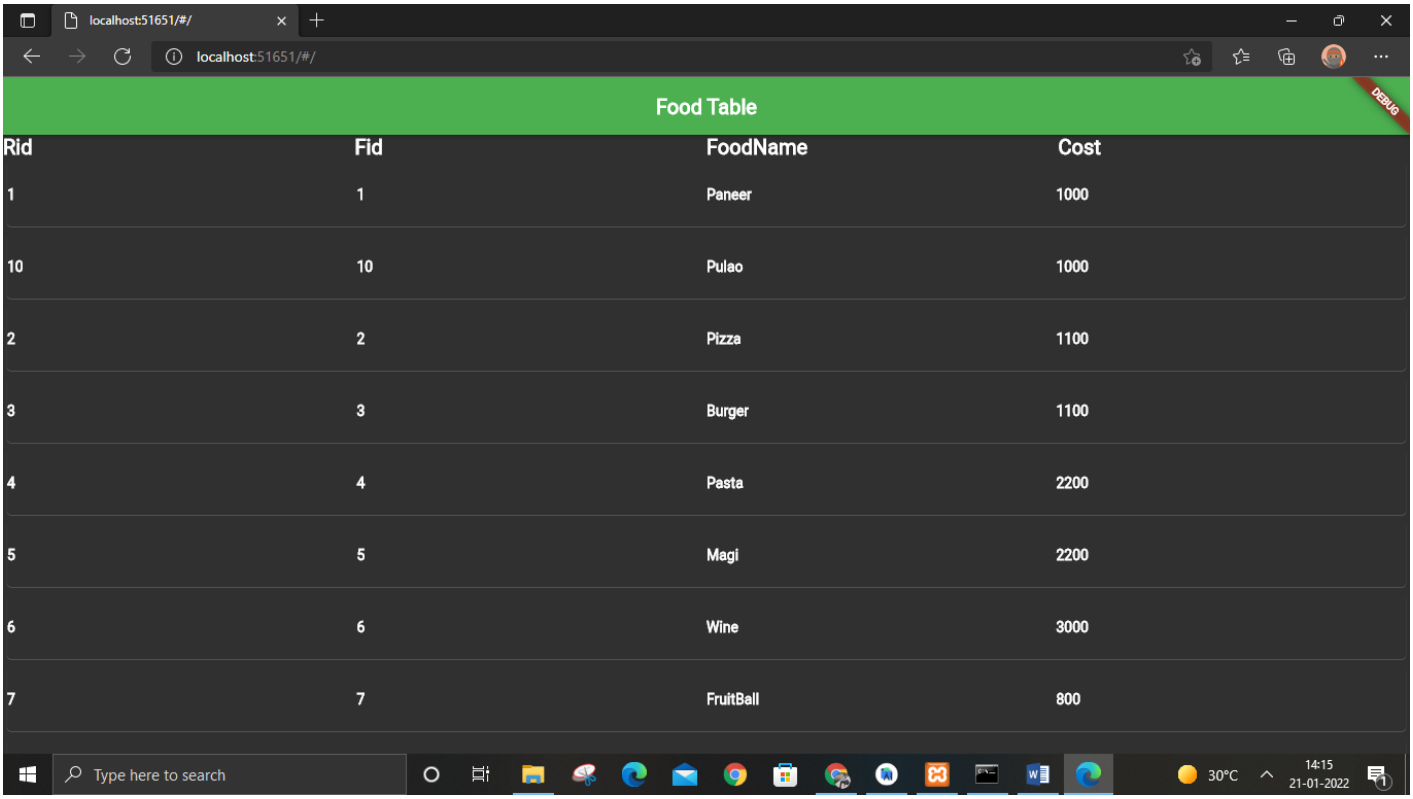

localhost:51651/#/
Update Restaurant
DEBUG
Write the query to update the Restaurant table
Enter update Query
Update Table
View Table
Type here to search
30°C
14:12
21-01-2022

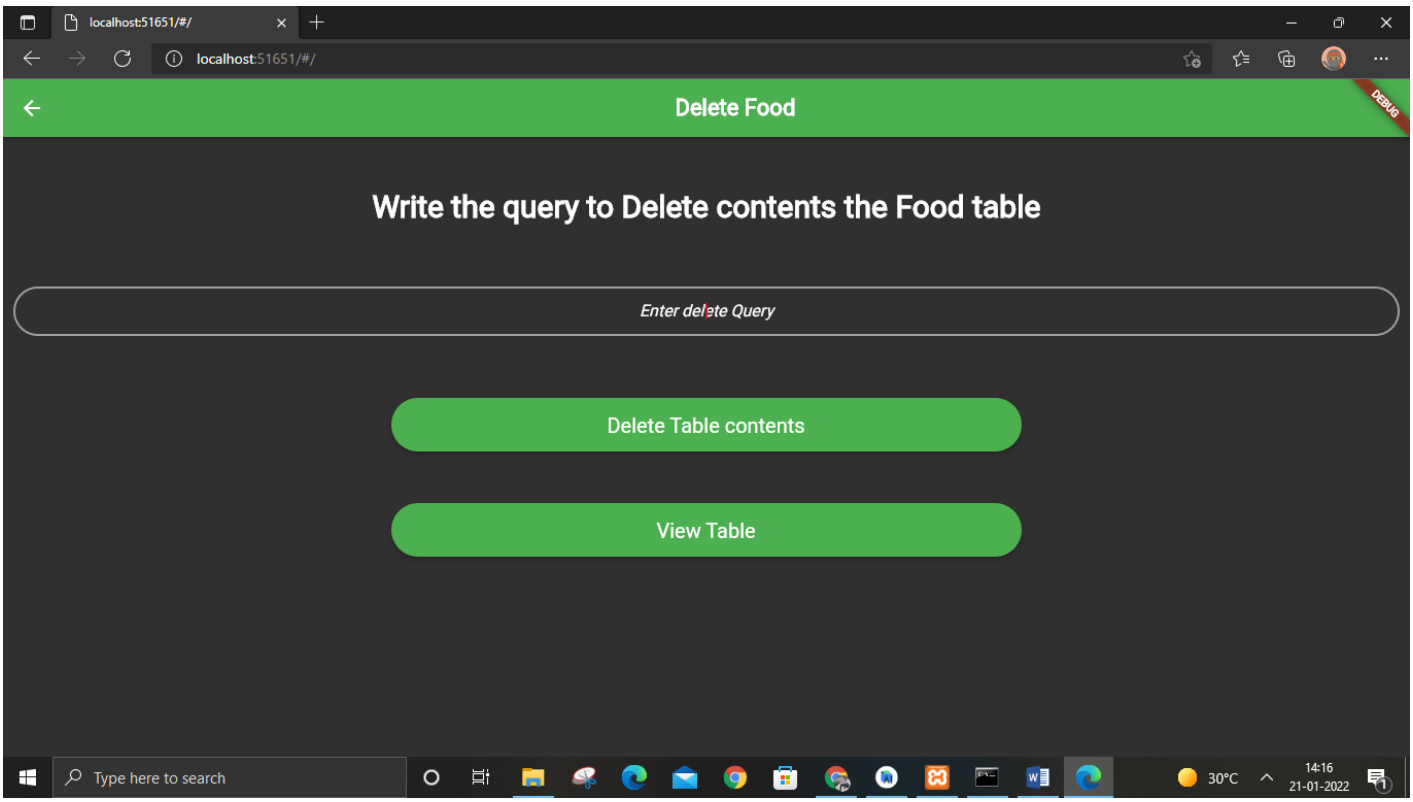

localhost:51651/#/
Delete Restaurant
DEBUG
Write the query to Delete contents the Restaurant table
Enter delete Query
Delete Table contents
View Table
Type here to search
30°C
14:13
21-01-2022

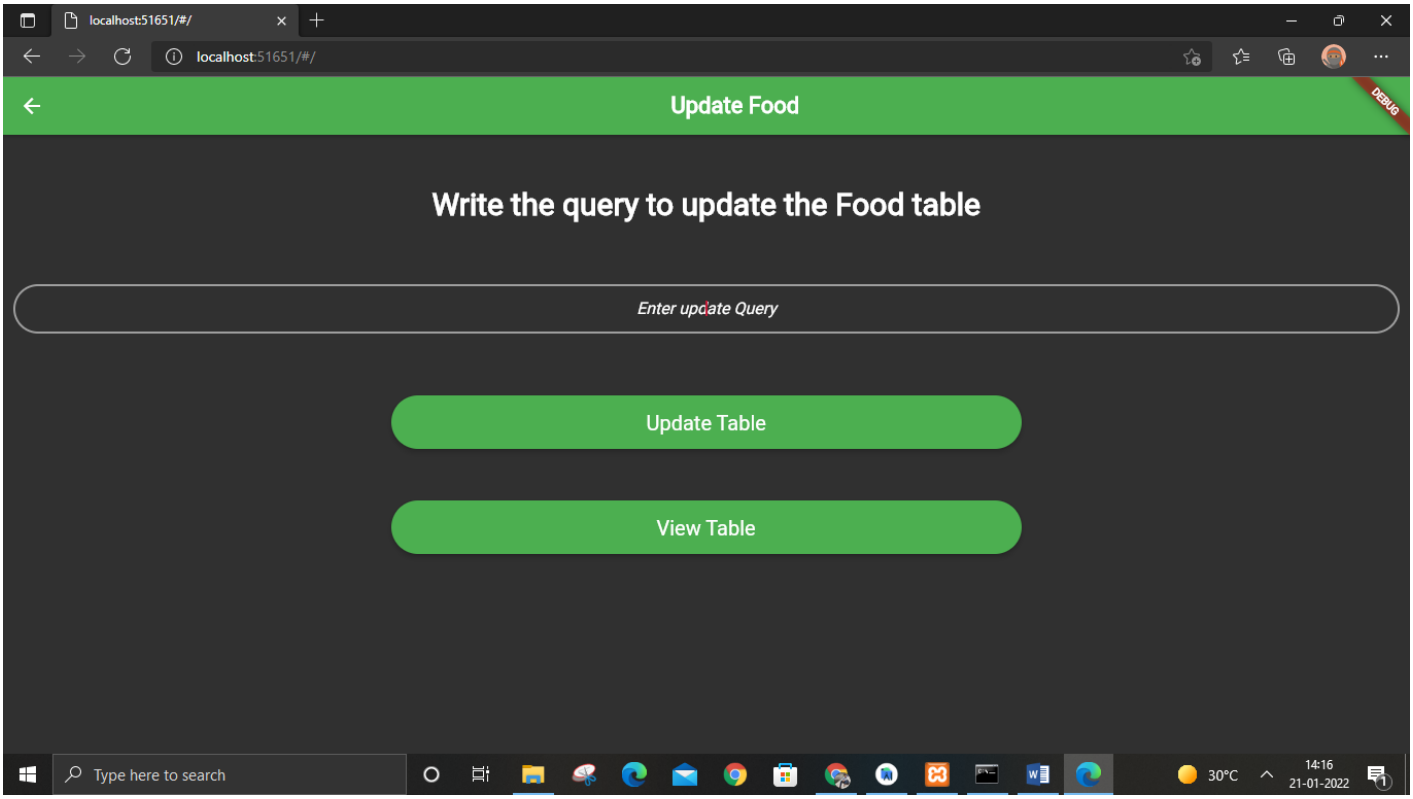
Food Table


localhost:51651/#/
Food
DEBUG
Enter the Details
Enter Restaurant id
Enter Food Id
Enter the Food Name
Enter the Cost
Insert Data
View Table
Type here to search
30°C
14:15
21-01-2022

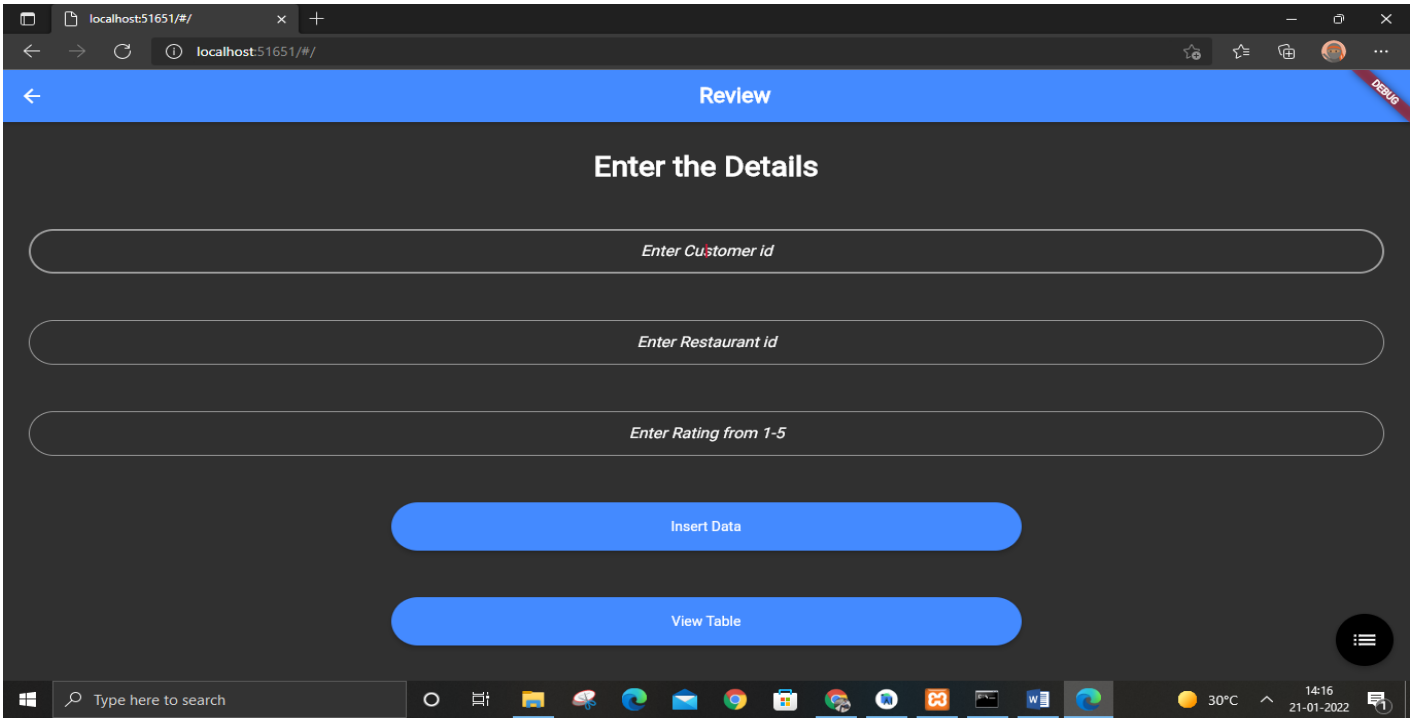

localhost:51651/#/
Food
DEBUG
Enter the Details
Enter Restaurant id
Enter Food Id
Enter the Food Name
Enter the Cost
Insert Data
View Table
delete
Update
Type here to search
30°C
14:15
21-01-2022

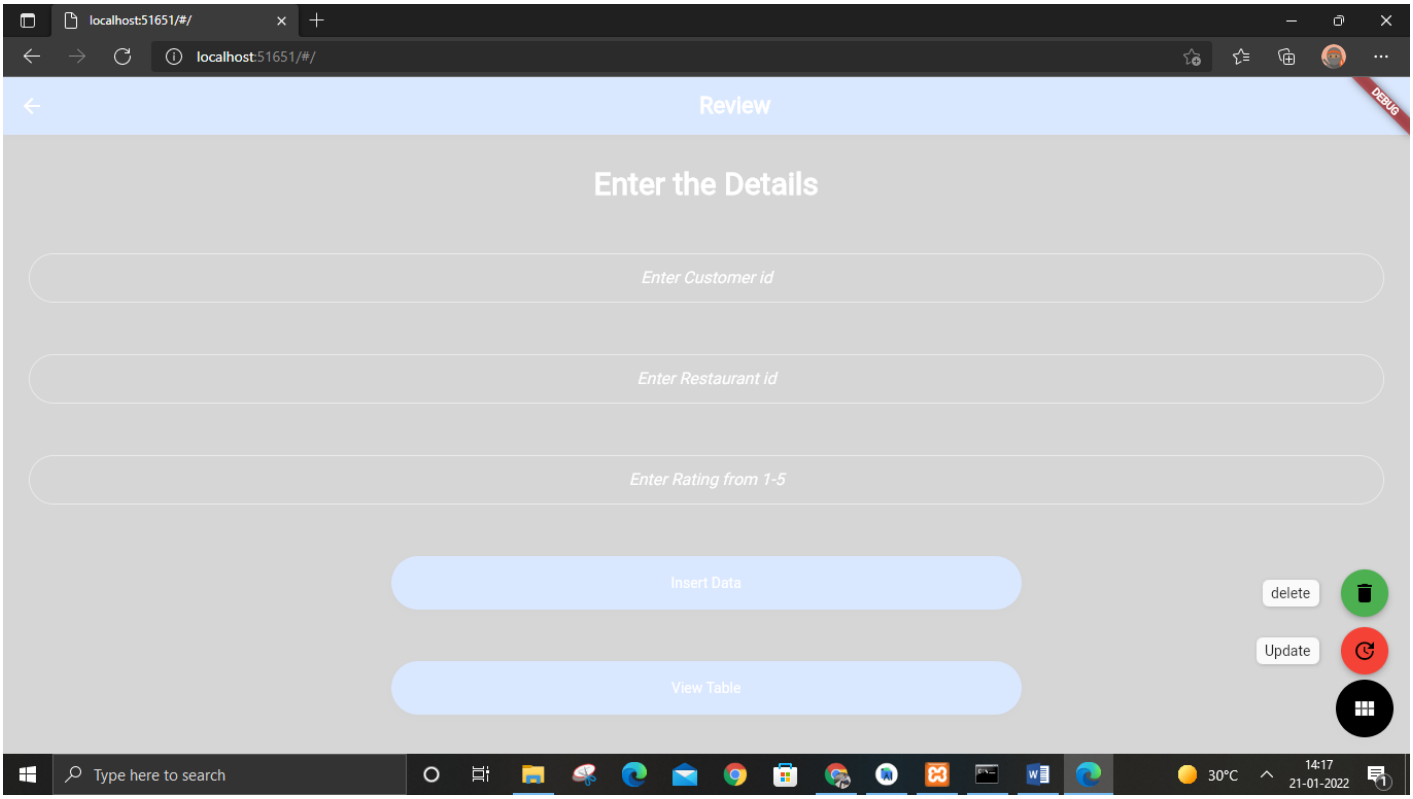

localhost:51651/#/
Food Table
DEBUG
Rid Fid FoodName Cost
1 1 Paneer 1000
10 10 Pulao 1000
2 2 Pizza 1100
3 3 Burger 1100
4 4 Pasta 2200
5 5 Magi 2200
6 6 Wine 3000
7 7 FruitBall 800
Type here to search
30°C
14:15
21-01-2022

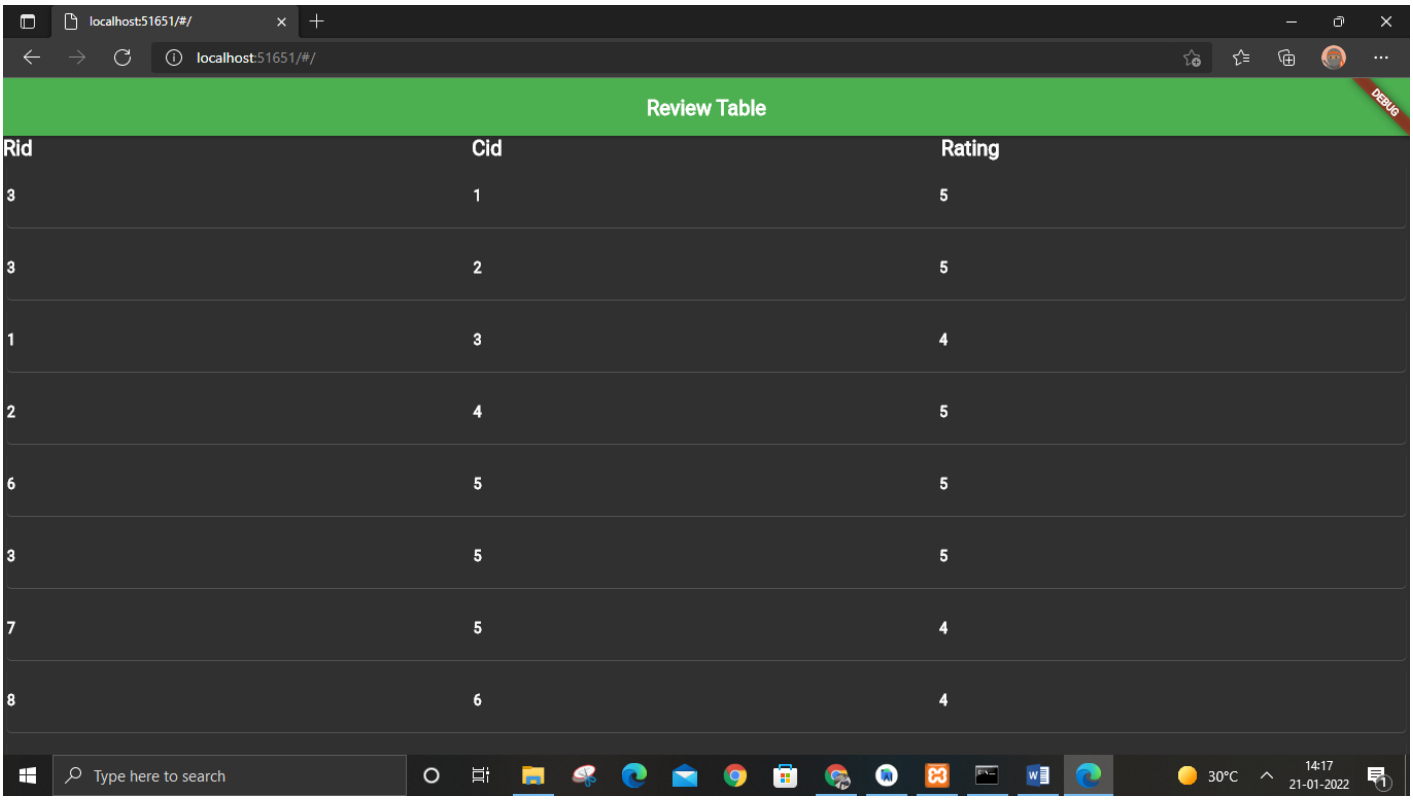

localhost:51651/#/
Delete Food
DEBUG
Write the query to Delete contents the Food table
Enter delete Query
Delete Table contents
View Table
Type here to search
30°C
14:16
21-01-2022

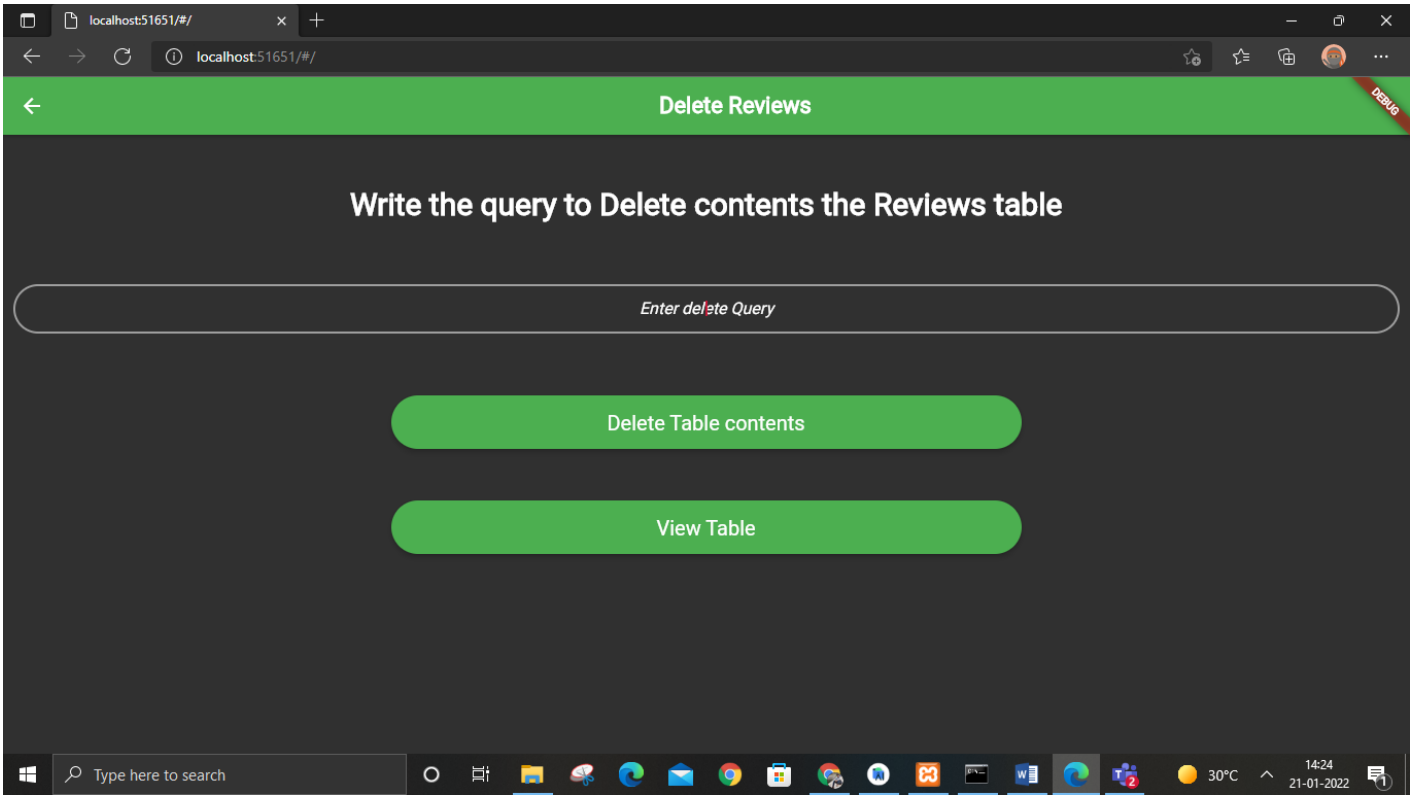


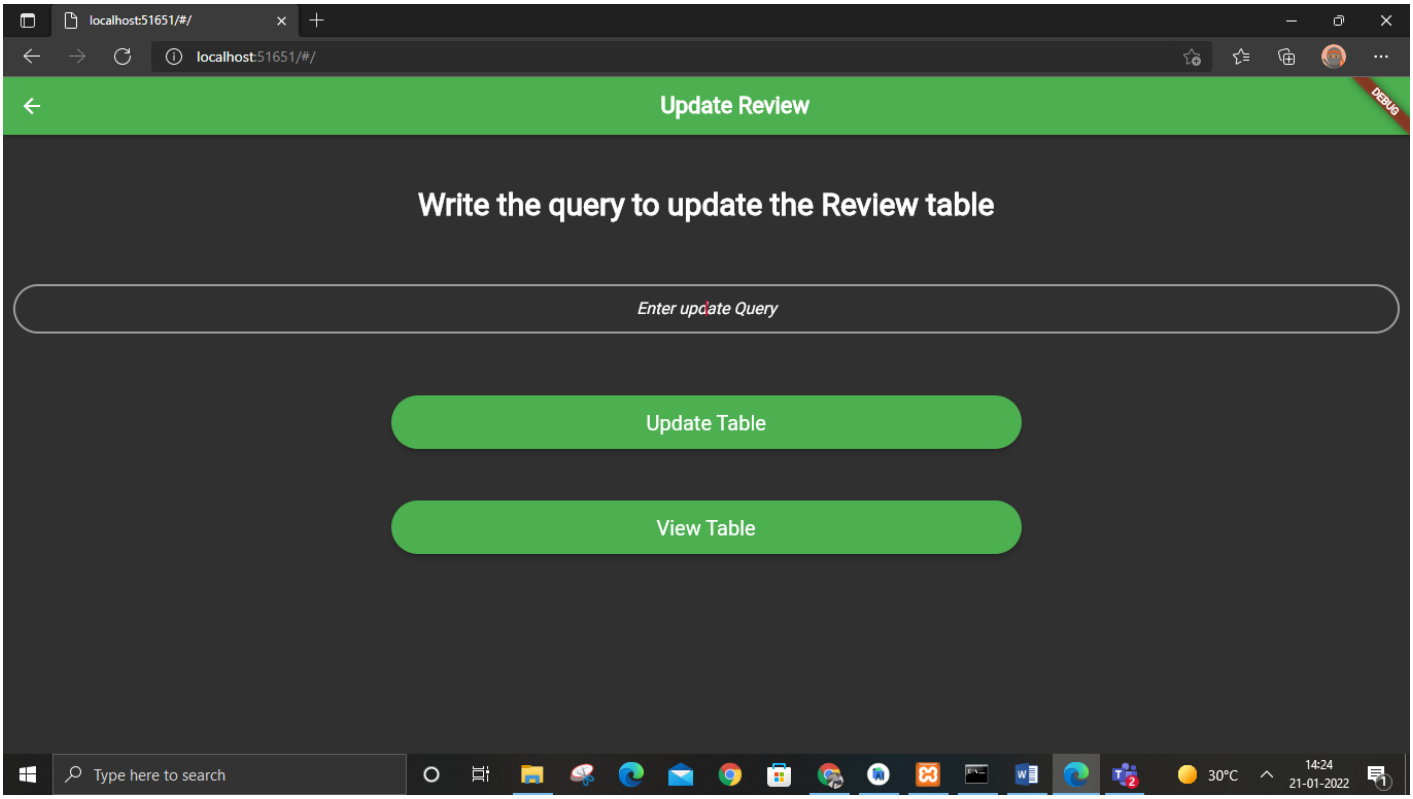
Reviews Table



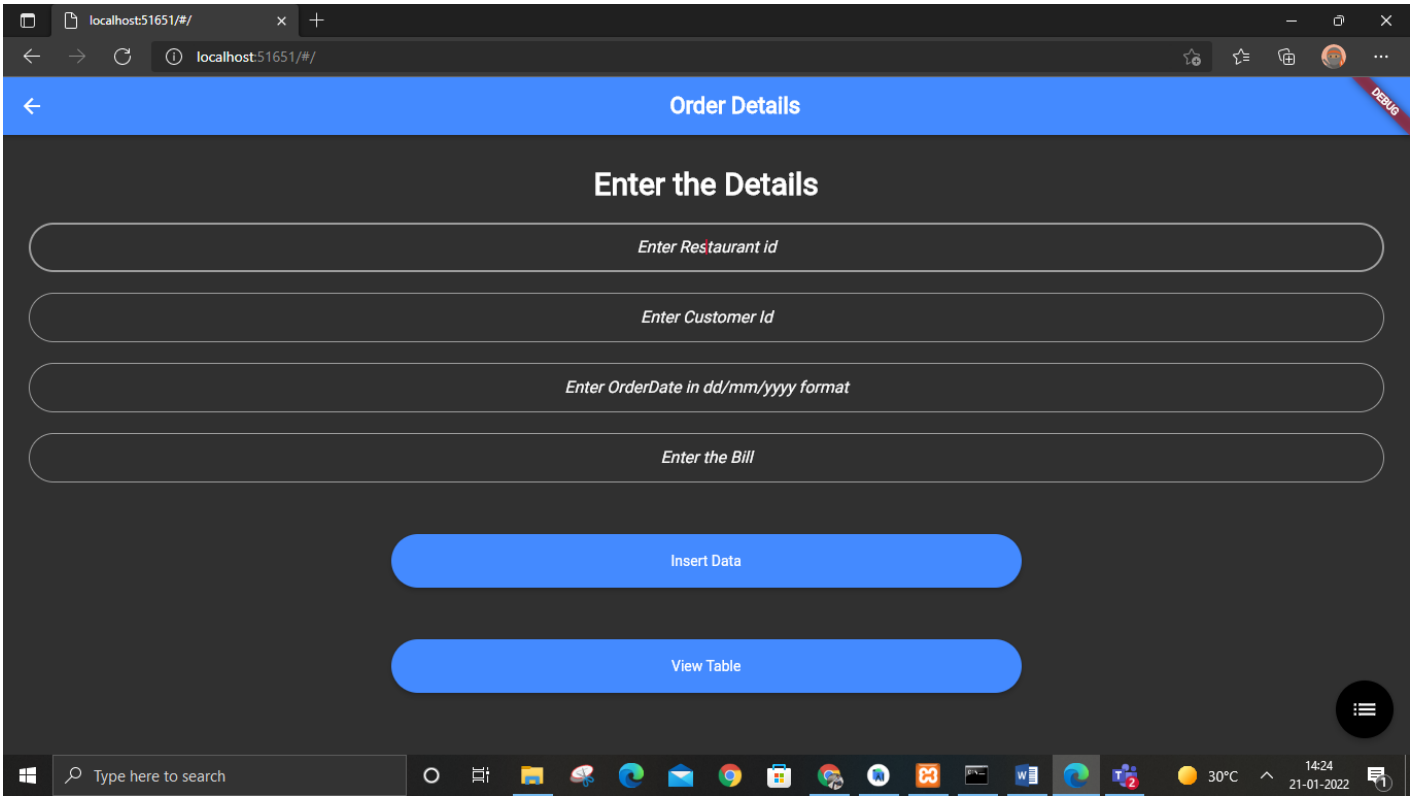

localhost:51651/#/
Review
DEBUG
Enter the Details
Enter Customer id
Enter Restaurant id
Enter Rating from 1-5
Insert Data
View Table
delete
Update
Type here to search
30°C
14:17
21-01-2022

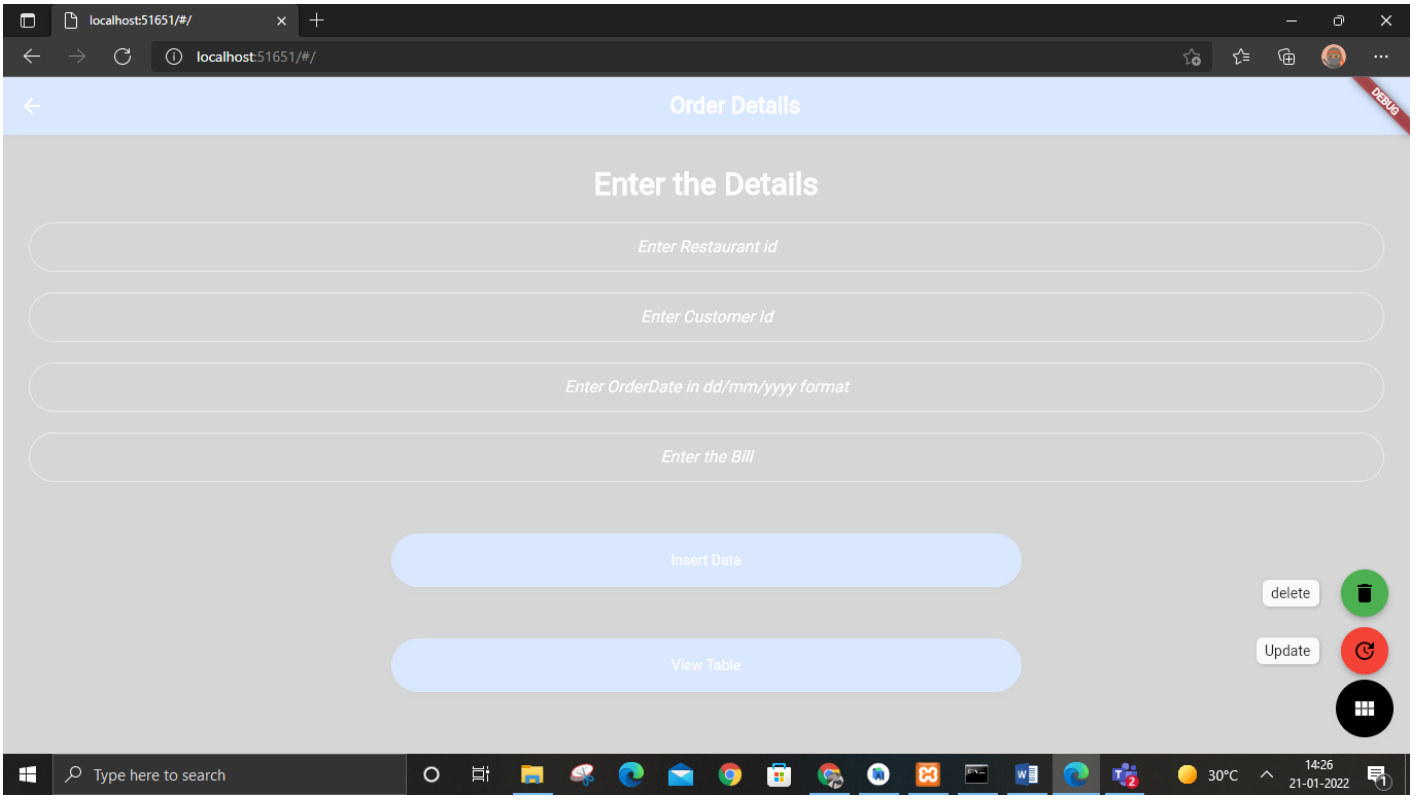

localhost:51651/#/
Review Table
DEBUG
Rid Cid Rating
3 1 5
3 2 5
1 3 4
2 4 5
6 5 5
3 5 5
7 5 4
8 6 4
Type here to search
30°C
14:17
21-01-2022

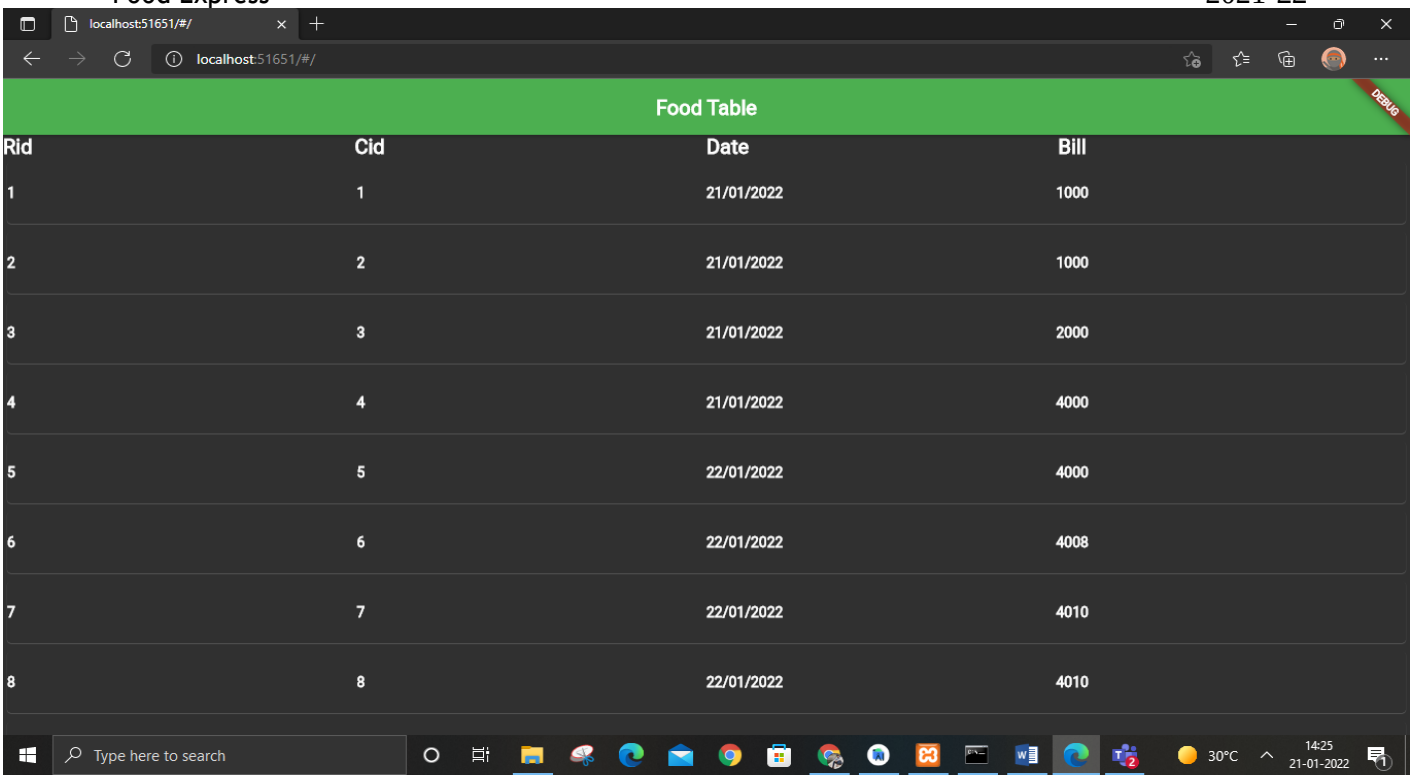

localhost:51651/#/
Delete Reviews
DEBUG
Write the query to Delete contents the Reviews table
Enter delete Query
Delete Table contents
View Table
Type here to search
30°C
14:24
21-01-2022

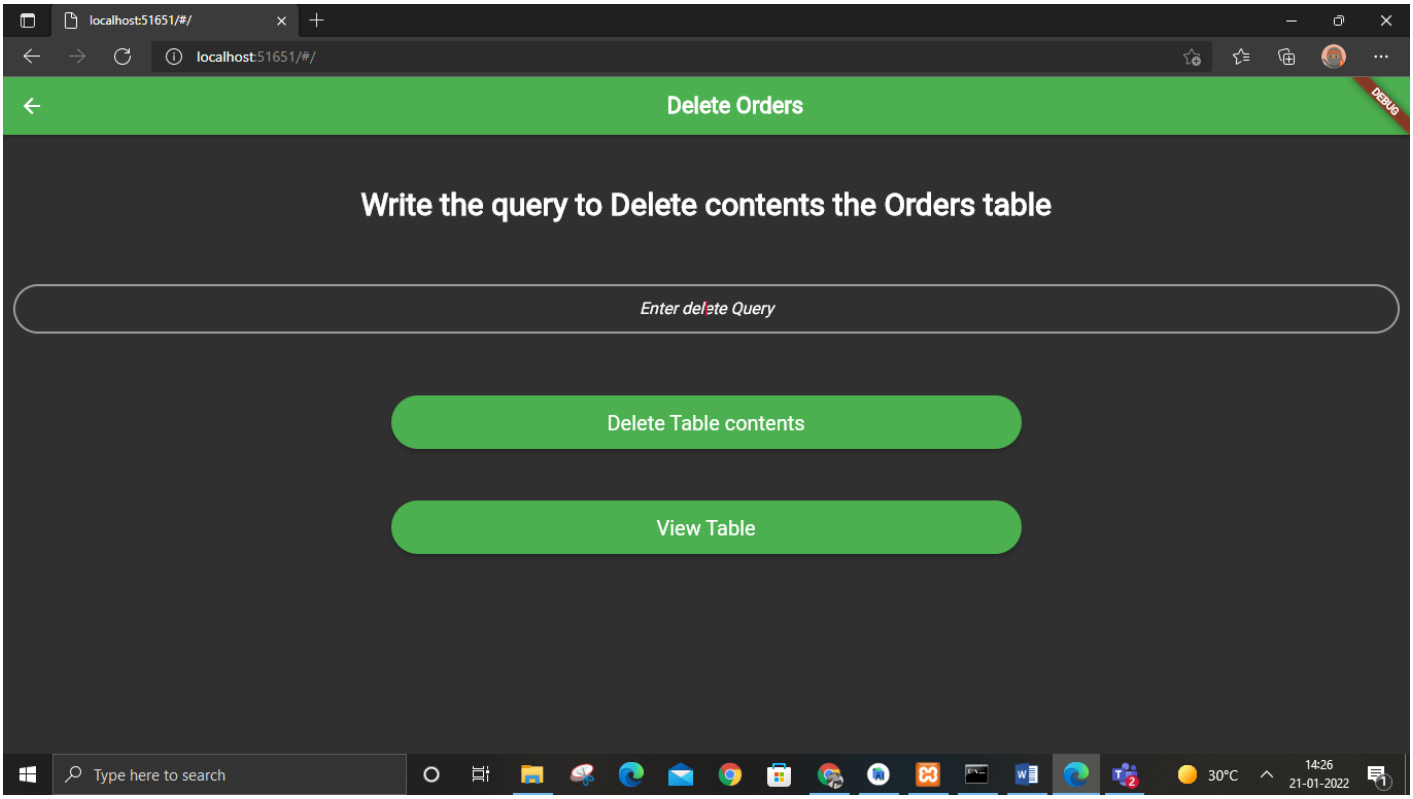

localhost:51651/#/
Update Review
DEBUG
Write the query to update the Review table
Enter update Query
Update Table
View Table
Type here to search
30°C
14:24
21-01-2022

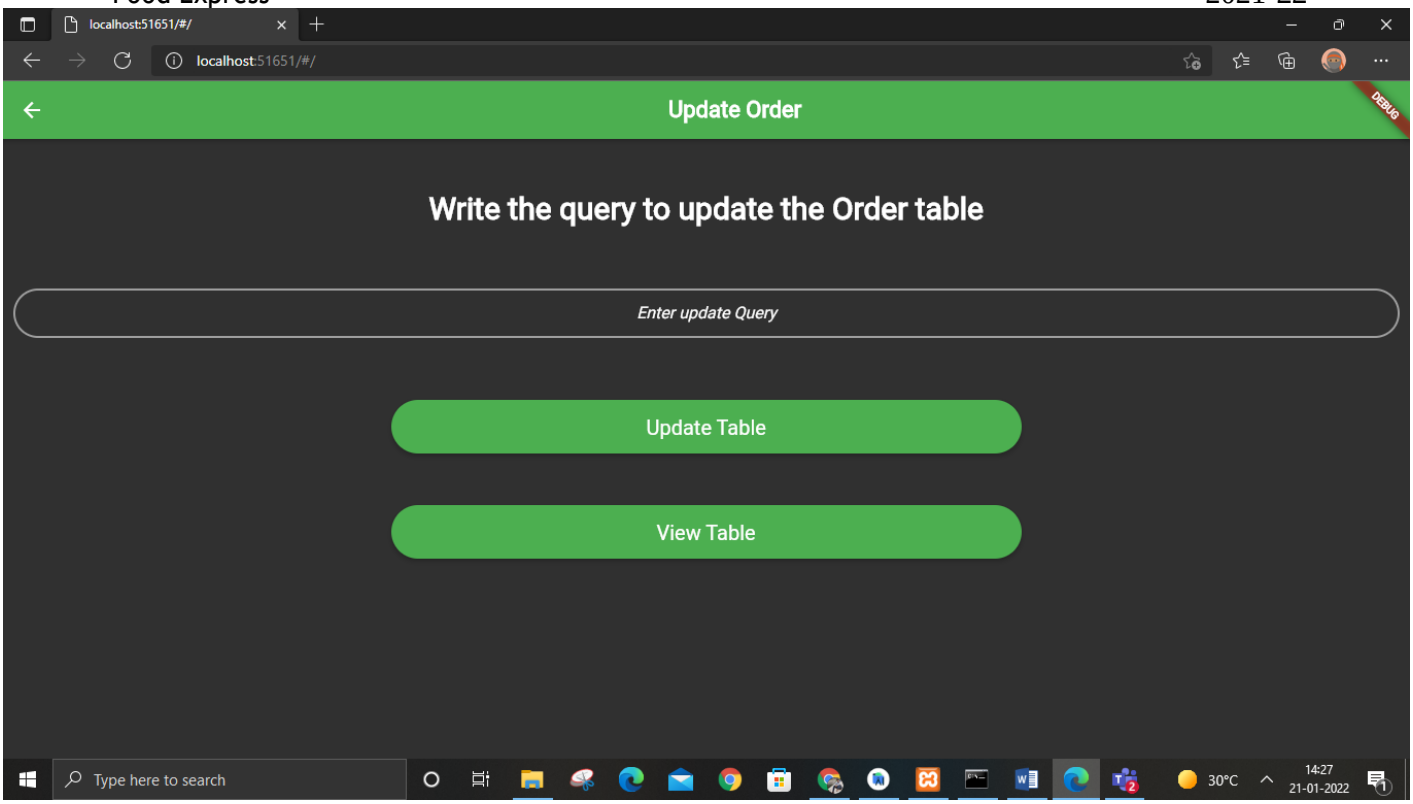
Orders Table



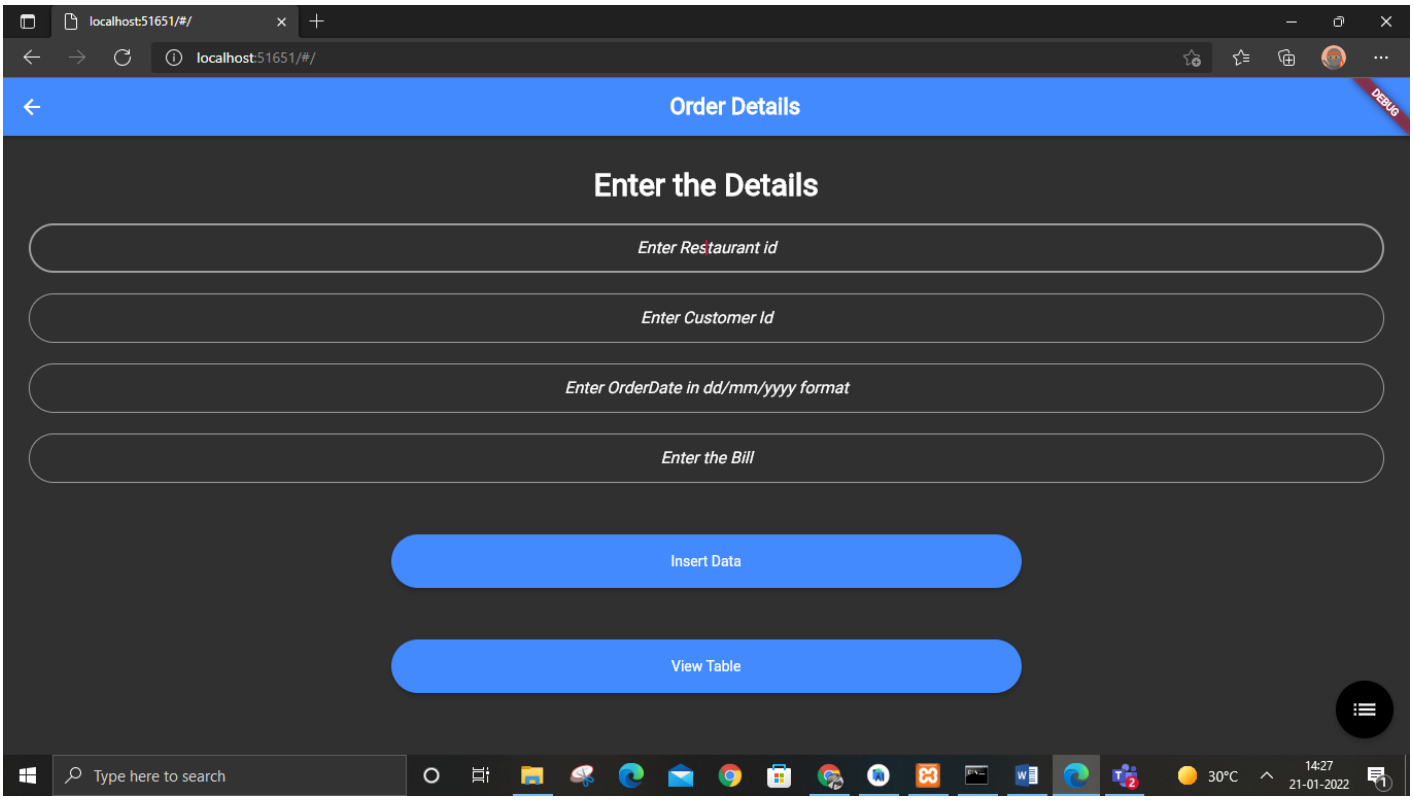


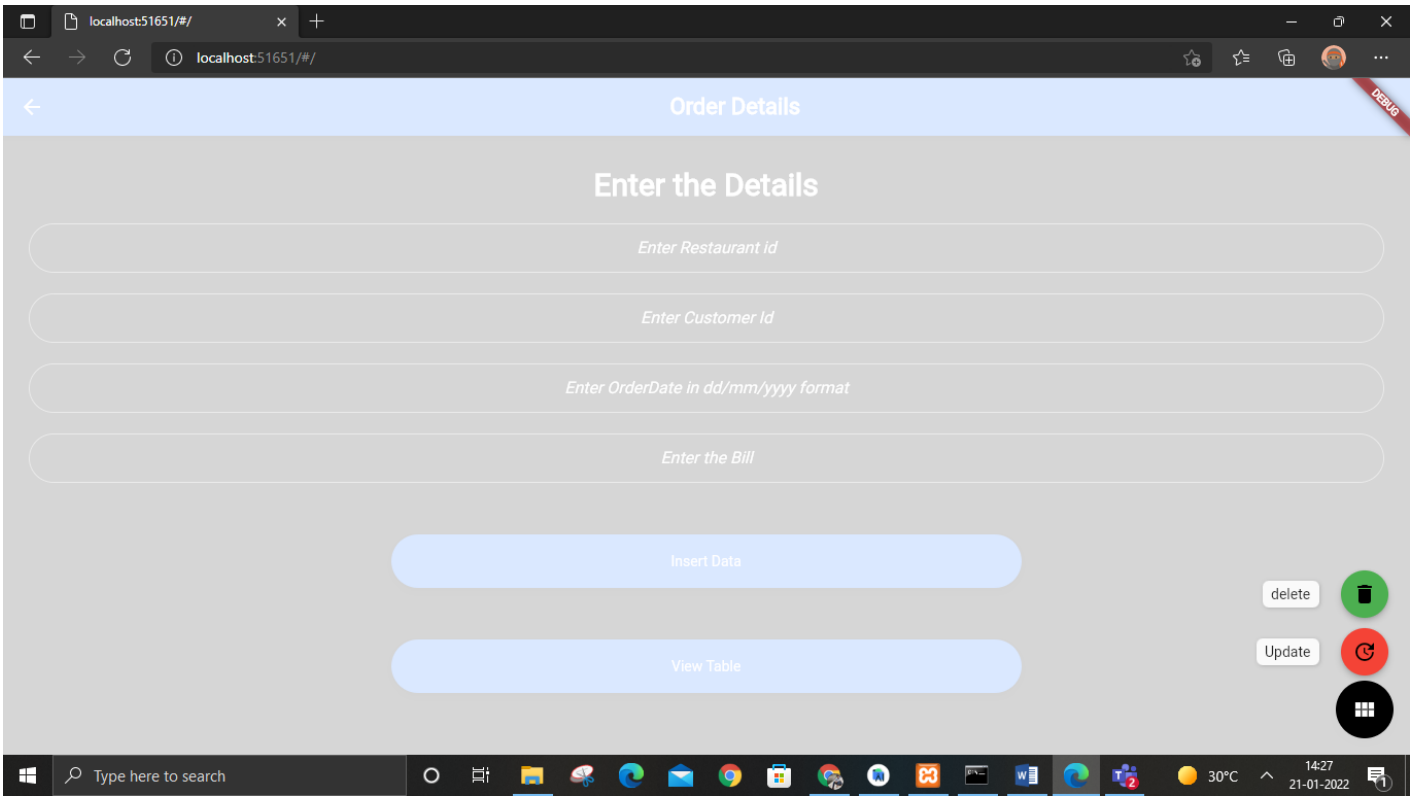

localhost:51651/#/
Food Table
DEBUG
Rid Cid Date Bill
1 1 21/01/2022 1000
2 2 21/01/2022 1000
3 3 21/01/2022 2000
4 4 21/01/2022 4000
5 5 22/01/2022 4000
6 6 22/01/2022 4008
7 7 22/01/2022 4010
8 8 22/01/2022 4010
Type here to search
30°C
14:25
21-01-2022

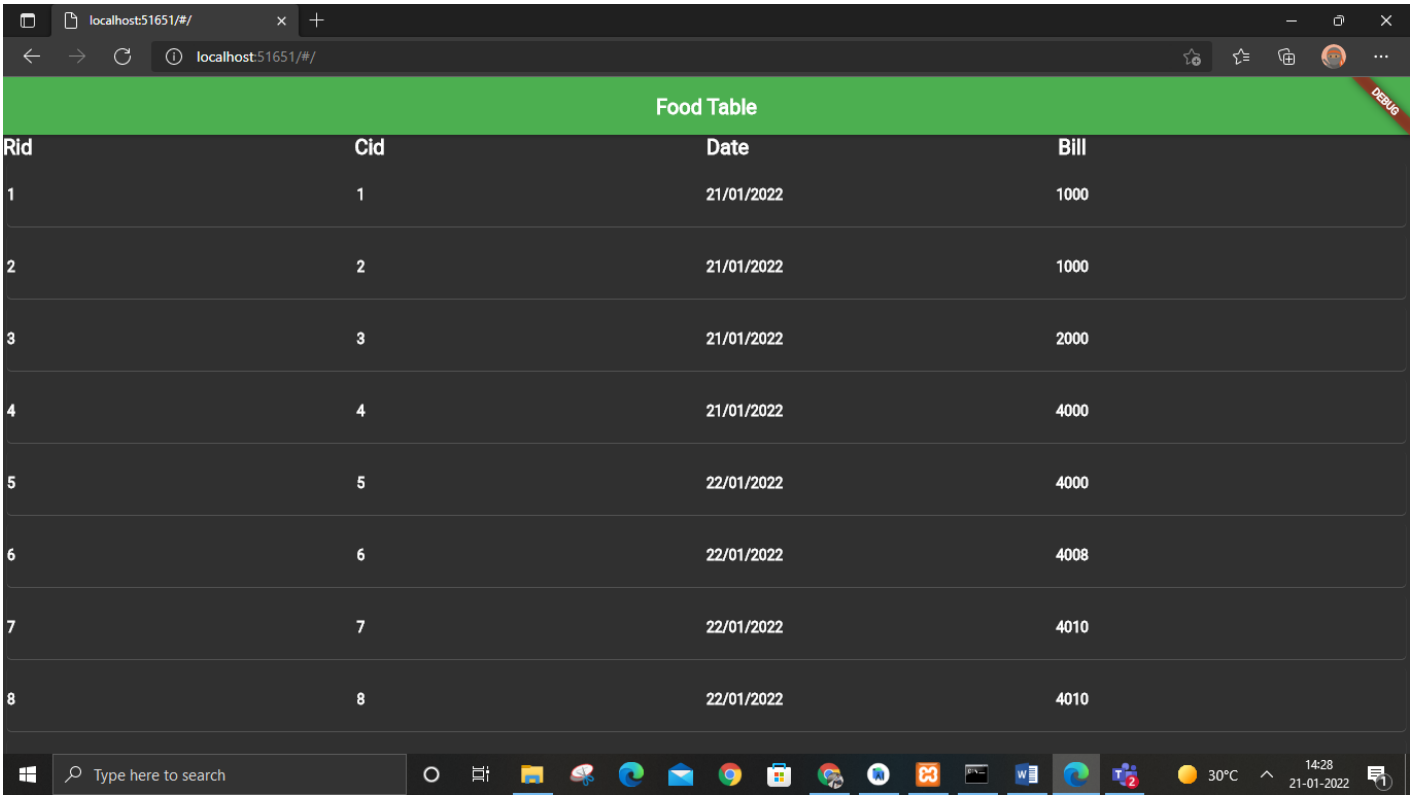

localhost:51651/#/
Delete Orders
DEBUG
Write the query to Delete contents the Orders table
Enter delete Query
Delete Table contents
View Table
Type here to search
30°C
14:26
21-01-2022

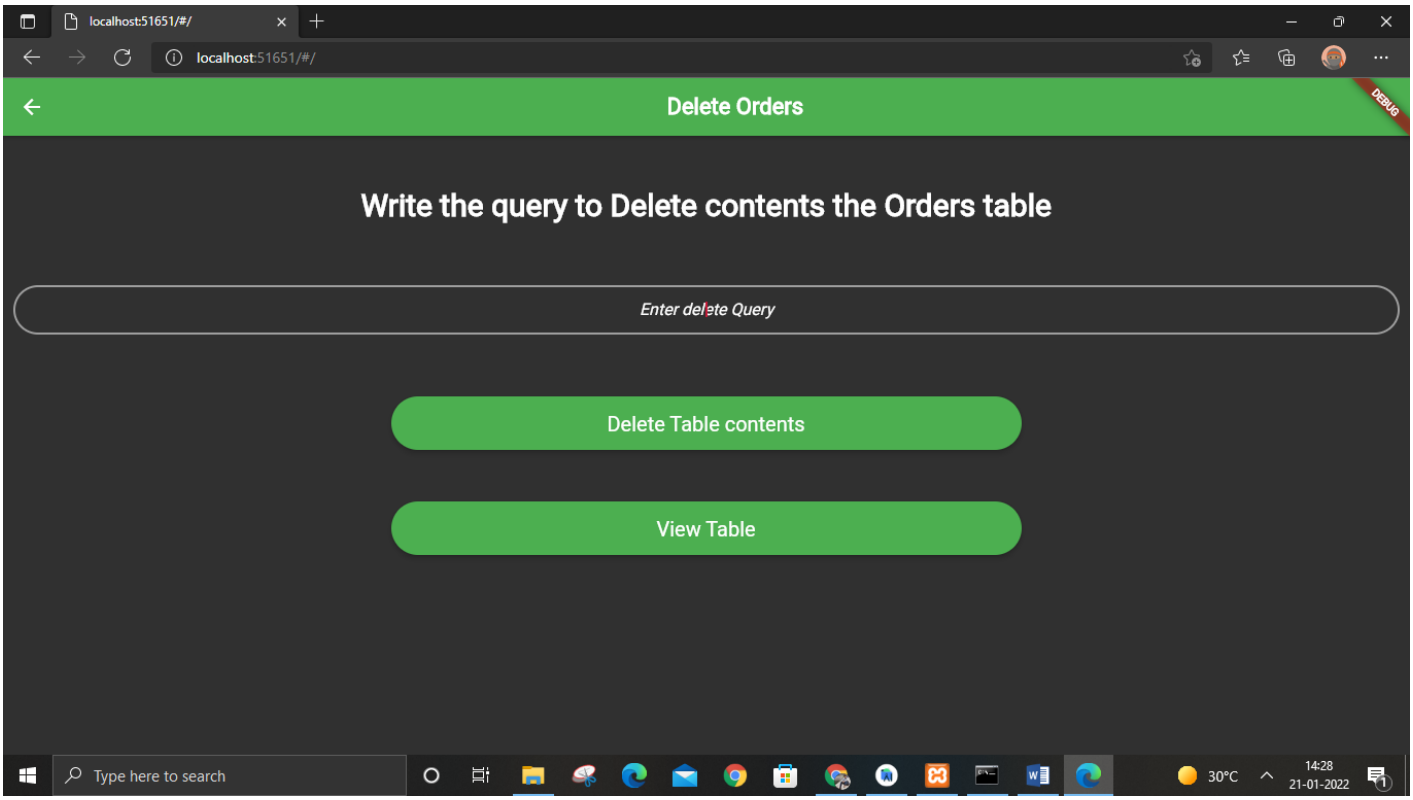


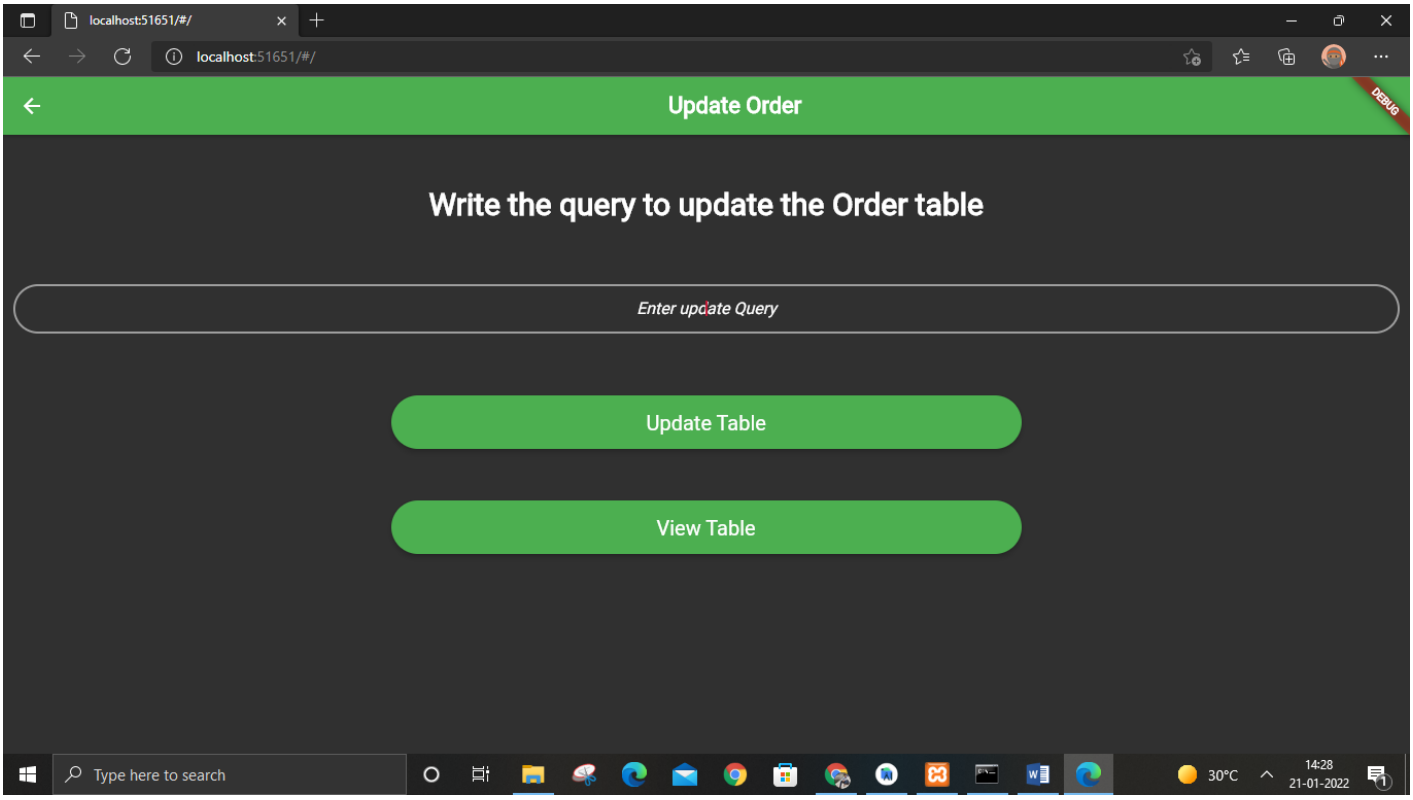
Order Details Table




localhost:51651/#/
Order Details
DEBUG
Enter the Details
Enter Restaurant id
Enter Customer Id
Enter OrderDate in dd/mm/yyyy format
Enter the Bill
Insert Data
View Table
delete
Update
Type here to search
30°C
14:27
21-01-2022


localhost:51651/#/
Food Table
DEBUG
Rid Cid Date Bill
1 1 21/01/2022 1000
2 2 21/01/2022 1000
3 3 21/01/2022 2000
4 4 21/01/2022 4000
5 5 22/01/2022 4000
6 6 22/01/2022 4008
7 7 22/01/2022 4010
8 8 22/01/2022 4010
Type here to search
30°C
14:28
21-01-2022


localhost:51651/#/
Delete Orders
DEBUG
Write the query to Delete contents the Orders table
Enter delete Query
Delete Table contents
View Table
Type here to search
30°C
14:28
21-01-2022


localhost:51651/#/
Update Order
DEBUG
Write the query to update the Order table
Enter update Query
Update Table
View Table
Type here to search
30°C
14:28
21-01-2022

Chapter 6

CONCLUSION & FUTURE ENHANCEMENTS

6.1 Conclusion

- The give Food delivery website helps in fast delivery of food from different restaurants to the customers .It allots drivers to deliver the food from restaurants to customers.There is a provision for reviews aswell for further improvement of restaurants.
- Overall it is a very simple handy website that can u=be used by restaurant owners and Customers

6.2 Future Enhancement

This system is designed in such a way that provisions can be given for further enhancements without affecting the system presently developed. The enhancements that can be incorporated are:

- There can be a module to check the if driver or Restaurant workers are sanitized
- Ui can be made more attractive by including animations
- Inserting User and Admin views separately during login
- Adding a cart option for all the sales items.