



CITY UNIVEERSITY

Faculty of Science and Engineering
Department of Computer Science Engineering
Fall 2020

CSE 312 Database Management System Laboratory

A Project Report on
Restaurant Management System

Supervised By
Md. Al-Mamun Riyadh
Lecturer, Dept. of CSE
City University

Submitted By
Md. Mahmudul Hasan
ID No: 1925102010

Date of Submission: 06/04/2021

Table of Content:

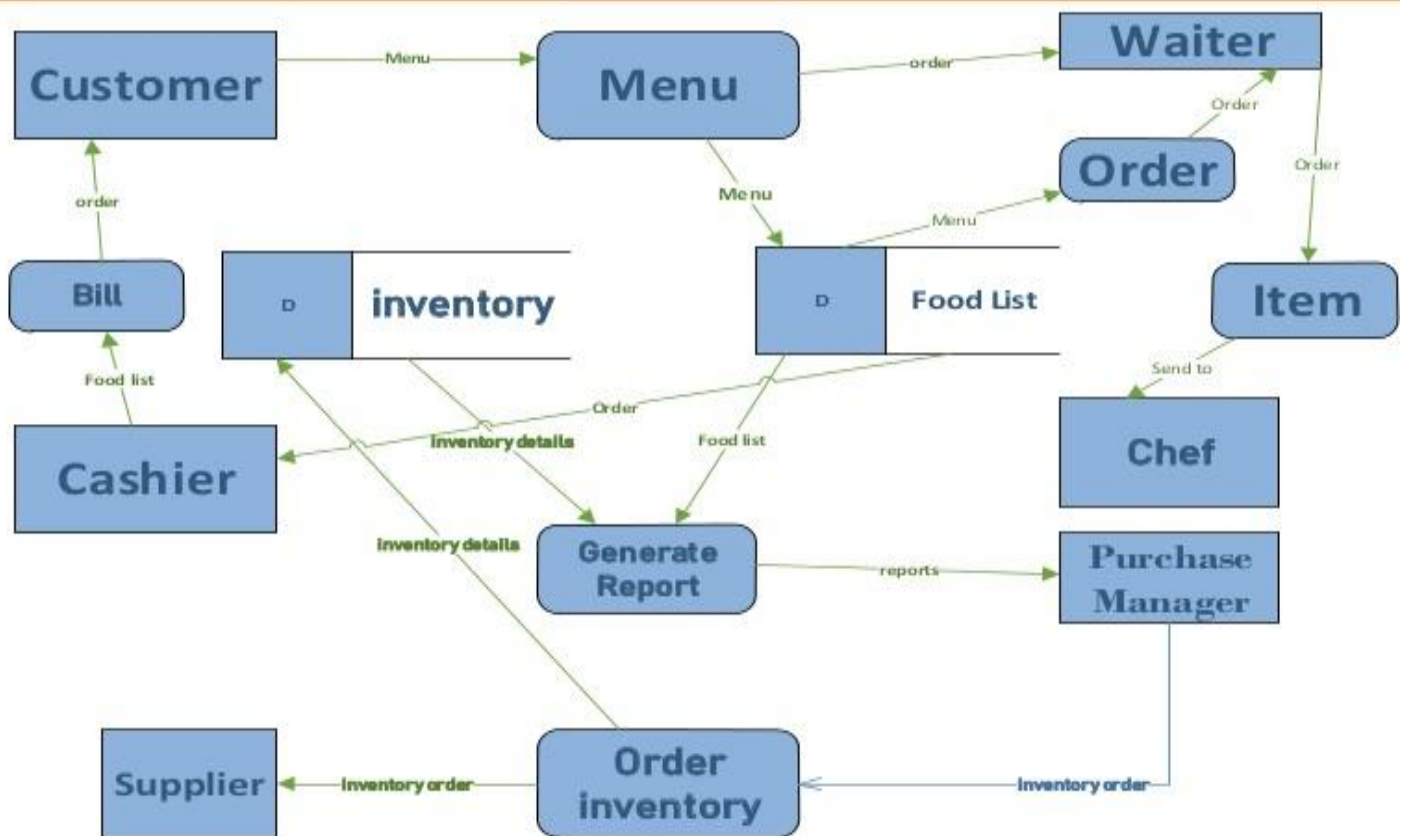
| | |
|---|----|
| Table Name----- | 01 |
| List of Diagram----- | 02 |
| Abstract----- | 03 |
| Introduction----- | 04 |
| Existing System Analysis----- | 05 |
| Proposed System----- | 06 |
| Entity-Relationship Diagram----- | 07 |
| Normalization----- | 08 |
| SQL Queries (Create, Insert, Select)----- | 09 |

Table Name:

1. User Table
2. Restaurant Table
3. Owner Table
4. Bill Table
5. Menu Bill Table
6. Customer Table
7. Manager Table
8. Waiter Table
9. Cashier Table
10. Cook Table
11. Home Delivery Table
12. Delivery Boy Table
13. Delivery Boy2 Table
14. Menu Table
15. Sale Detail Table
16. Supplier Table
17. Tables Table
18. Booking Table
19. Ingredient Table

List of Diagram

Data Flow Diagram of Restaurant Management System



Abstract

This report documents the process of designing, developing and testing a software system to be used in a restaurant; usually given the name restaurant management system. The restaurant management systems there to help communication between all teams within a restaurant by minimizing the probability of human errors.

Introduction

RMS Restaurant Management System are the crucial technology components that enable a single outlet or enterprise to better serve its customers and aid employees with food and beverage transactions and controls.

Online Restaurant Management System is a process of ordering food from a restaurant or food co-operative through a web page or app. Much like ordering consumer goods online, many of these allow customers to keep accounts with them in order to make frequent ordering convenient. A customer will search for a favorite restaurant, usually filtered via type of cuisine and choose from available items, and choose delivery or pick-up.

1. Online User
2. Super Admin and
3. Sub Admin

Existing System Analysis (4 System with Link)

There are many computerized restaurant management systems available but for each system there exist disadvantages or missing features.

The most common type of restaurant management system contains a static order entry computer system usually in the shape of a desktop computer with a touchscreen.

Typically this common approach is adequate to the restaurants requirements but still requires handwritten orders to be relayed to the order entry computer system.

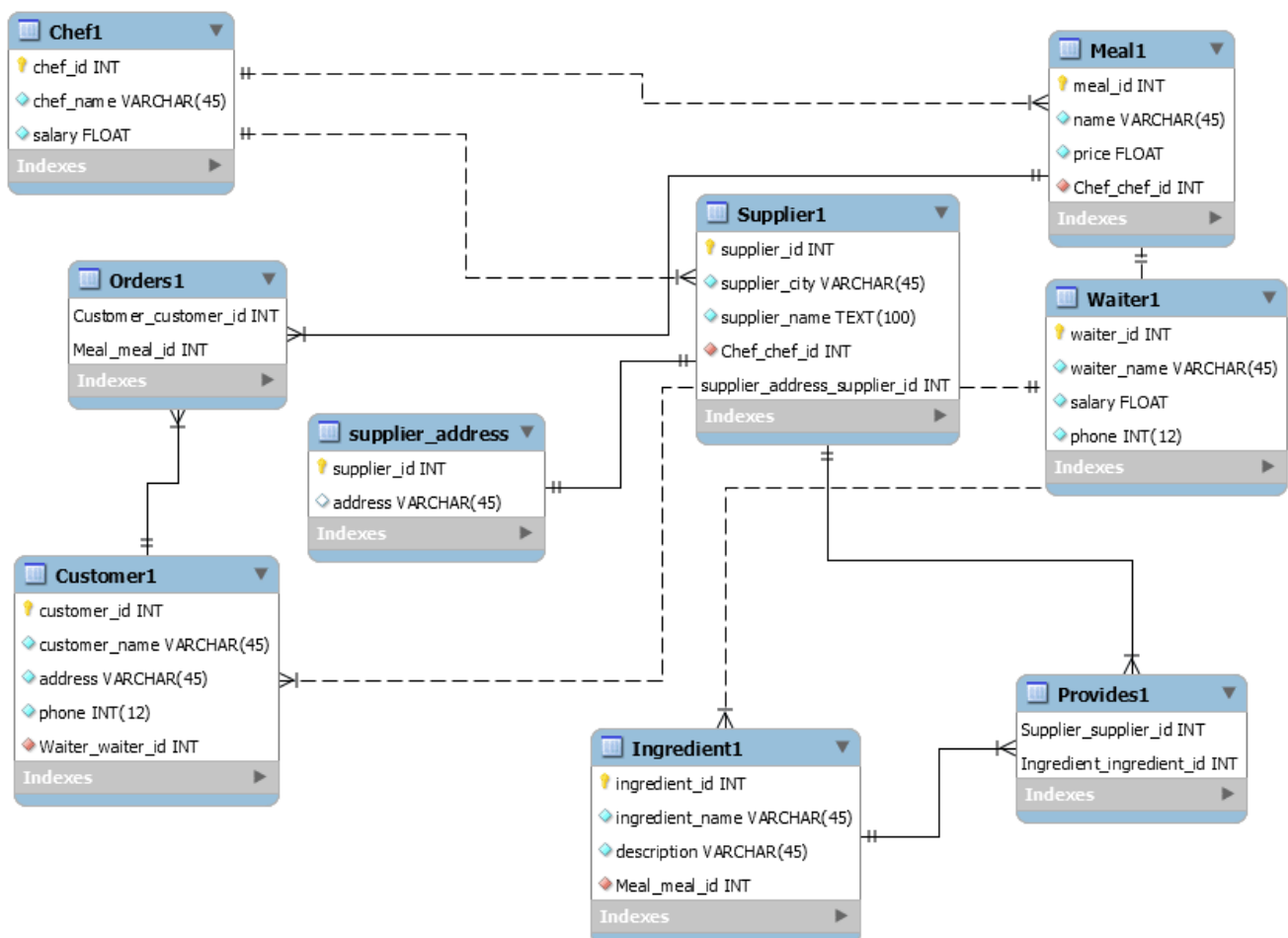
Proposed System

Restaurant is a kind of business that serves people all over world with ready-made food. Currently this industry is going on with lot of flare. People feel more comfortable with lot of variations in the selection and consumption of their food in their busy life. One can see lot more restaurant in the world. Even in Denmark one can see thousands of restaurants with dishes from all over the world like from India, Pakistan, Mexican, etc. fulfilling the needs of people with nourishments and enjoyments. Let's concentrate on booking area in a restaurant. In traditional booking system, a customer has to make a phone call in order to get his meal reserved. If luckily the phone gets connected, then the customer does some formal conversation like hello, hi, etc. Than he demands for today's menu and do some discussion over menu items then he orders and he has to give some of this identification specifications. This process takes 5-8 minutes to complete. On the receiver side there is hardly one phone line and one operator. So he can cover around 15-20 orders maximum in an hour.

For each booking he has to register manually on paper and puts the order in a queue with specific priority according to time and quantity, and then a cook is assigned for the specific order to complete it.

There are lots of areas to be solved for current restaurants using modern IT World. Many areas come like human resource management, accounts management, etc. But our problem lies within domain of end customer and restaurant “Meal Reservation”.

Entity Relationship Diagram



An entity relationship (ER) diagram is a modelling language used to represent a type of semantic data model of a system.

The ER diagrams are often used to represent a relational database and its requirements in a top-down fashion usually defined as the database schema.

The database schema for this database has been split into two ER.

Graphically shows the objects and their relationships that are contained within a meal.

The meal object will be made of at least two ingredients that can be either a normal ingredient

Or a prepared ingredient.

Note, a prepared ingredient is a collection of ingredients used to either group commonly used ingredients or to group optional ingredients.

Each ingredient will have a default and manual measurement with the default measurement entered on input of the ingredient and the manual measurement entered if the meal ingredient link requires a different amount. Also, each ingredient will be part of a generic ingredient object as there are many ingredients that are the same item but packaged in a different way at a different price. This allows the database to be in Boyce-Codd normalized state. An example of this would be the drink Coca-Cola which can be bought by bottle, can or draught, thus are the same item but packaged differently at a different price and amount. Finally each ingredient and prepared ingredient can be part of a category allowing optional ingredients to be interchanged with other ingredients in the same category. Graphically shows the relationships for the menu, order and offer objects. The menu consists of a date time relationship that provides the intervals to when the menu is active and a menu section relationship that contains the color variables and items under that particular menu section. The order consists of one to many suborders with the suborder consisting of one to many items. The order stores all the ingredients within each item and also the replaced ingredient if that optional ingredient was

replaced. The offer consists of a date time relationship that provides the intervals to when the offer is active and an offer section relationship that contains the sets required by the offer

Normalization

Normalization comes in many forms ranging from first normal form to sixth normal form. The normalization of a database is a systematic way to free the database of undesirable characteristics where inserts, updates and deletions of data could lead to the loss of data integrity. The greater the normal form, the greater the data integrity of the database. The database in this system was designed to be in Boyce-Cod normal form which is a slightly stronger version of the third normal form. For the database to be in Boyce-coded normal form, it had to pass for all previous normal forms as well as Boyce-coded normal form. A well designed database will normally abide by the first, second and third normal forms as they are the basics to a well-structured relational database. According to The three normal forms can be defined as:

1. First norm: Every attribute is atomic or single valued therefore there are no repeating fields.2.
2. Second norm: All attributes not part of the primary key must be dependent on the full key.3.
3. Third norm: There must be no transitive determinants, or each attribute that is not part of the key must be determined only by the key

Finally for the database to be in the desired Boyce-coded normal form, all tables must abide by first,

Second and third normal forms and must not have any determinants that are not candidate keys for the table

SQL Queries (Create, Insert, Select)

```
drop database if exists `restaurant`;
```

```
create database `restaurant`;
```

```
use `restaurant`;
```

```
drop table if exists `USER`;
```

```
create table `USER`(  
    `User_Id` int NOT NULL AUTO_INCREMENT,  
    `Fname` varchar(50) NOT NULL,  
    `Lname` varchar(50) NOT NULL,  
    `Password` varchar(50) NOT NULL,  
    PRIMARY KEY (`User_Id`)  
)ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `RESTAURANT`;
```

```
create table `RESTAURANT`(  
    `Name` varchar(100) NOT NULL,  
    `Location` varchar(100) NOT NULL,  
    `Contact` varchar(100) NOT NULL,  
    `Opening_Closing_Time` varchar(100) NOT NULL,  
    `Details` varchar(500) default NULL,  
    PRIMARY KEY (`Name`)  
)ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `OWNER`;
```

```
create table `OWNER` (
```

```
    `Fname` varchar(15) NOT NULL,
```

```
    `Lname` varchar(15) NOT NULL,
```

```
    `Contact` varchar(100) NOT NULL,
```

```
    `Rest_Name` varchar(100) NOT NULL,
```

```
    PRIMARY KEY (`Fname`, `Lname`, `Contact`),
```

```
    FOREIGN KEY (`Rest_Name`) REFERENCES `RESTAURANT` (`Name`)
```

```
    ON UPDATE CASCADE
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `BILL`;
```

```
create table `BILL` (
```

```
    `Order_Id` int NOT NULL AUTO_INCREMENT,
```

```
    `Customer_Fname` varchar(20) NOT NULL,
```

```
    `Customer_Lname` varchar(20) NOT NULL,
```

```
    `Customer_id` int NOT NULL,
```

```
    `Total_Amount` double NOT NULL,
```

```
    PRIMARY KEY (`Order_Id`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `MENU_BILL`;
```

```
create table `MENU_BILL` (
```

```
    `Order_Id` int NOT NULL,
```

```
    `Name` varchar(100) NOT NULL,
```

```
    `Quantity` varchar(20) NOT NULL,
```

```
    `Price` varchar(20) NOT NULL,
```

```
    FOREIGN KEY (`Order_Id`) REFERENCES `BILL` (`Order_Id`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `CUSTOMER`;

create table `CUSTOMER` (
    `Customer_Id` int NOT NULL AUTO_INCREMENT,
    `Fname` varchar(15) NOT NULL,
    `Lname` varchar(15) NOT NULL,
    `Contact` varchar(20) DEFAULT NULL,
    `Email_Id` varchar(50) DEFAULT NULL,
    PRIMARY KEY (`Customer_Id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `MANAGER`;

create table `MANAGER` (
    `Manager_Id` int NOT NULL AUTO_INCREMENT,
    `Fname` varchar(15) NOT NULL,
    `Lname` varchar(15) NOT NULL,
    `Contact` varchar(20) NOT NULL,
    `Address` varchar(30) DEFAULT NULL,
    `Salary` varchar(30) DEFAULT NULL,
    `Sex` char(1) DEFAULT NULL,
    `Bdate` date DEFAULT NULL,
    `Join_Date` date NOT NULL,
    PRIMARY KEY (`Manager_Id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `WAITER`;
```

```
create table `WAITER`(  
    `Waiter_Id` int NOT NULL AUTO_INCREMENT,  
    `Fname` varchar(15) NOT NULL,  
    `Lname` varchar(15) NOT NULL,  
    `Contact` varchar(20) NOT NULL,  
    `Address` varchar(30) DEFAULT NULL,  
    `Salary` varchar(30) DEFAULT NULL,  
    `Sex` char(1) DEFAULT NULL,  
    `Bdate` date DEFAULT NULL,  
    `Join_Date` date NOT NULL,  
    PRIMARY KEY (`Waiter_Id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `CASHIER`;
```

```
create table `CASHIER`(  
    `Cashier_Id` int NOT NULL AUTO_INCREMENT,  
    `Fname` varchar(15) NOT NULL,  
    `Lname` varchar(15) NOT NULL,  
    `Contact` varchar(20) NOT NULL,  
    `Address` varchar(30) DEFAULT NULL,  
    `Salary` varchar(30) DEFAULT NULL,  
    `Sex` char(1) DEFAULT NULL,  
    `Bdate` date DEFAULT NULL,  
    `Join_Date` date NOT NULL,  
    PRIMARY KEY (`Cashier_Id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `COOK`;
```

```
create table `COOK`(  
    `Cook_Id` int NOT NULL AUTO_INCREMENT,  
    `Fname` varchar(15) NOT NULL,  
    `Lname` varchar(15) NOT NULL,  
    `Contact` varchar(20) NOT NULL,  
    `Address` varchar(30) DEFAULT NULL,  
    `Salary` varchar(30) DEFAULT NULL,  
    `Sex` char(1) DEFAULT NULL,  
    `Bdate` date DEFAULT NULL,  
    `Join_Date` date NOT NULL,  
    `Specialization` varchar(50) DEFAULT NULL,  
    PRIMARY KEY (`Cook_Id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `HOME_DELIVERY`;
```

```
create table `HOME_DELIVERY`(  
    `Delivery_Id` int NOT NULL AUTO_INCREMENT,  
    `Address` varchar(50) NOT NULL,  
    `Contact` varchar(20) NOT NULL,  
    `Cust_Id` int NOT NULL,  
    `Order_Id` int NOT NULL,  
    PRIMARY KEY(`Delivery_Id`),  
    FOREIGN KEY (`Cust_Id`) REFERENCES `CUSTOMER`(`Customer_Id`),  
    FOREIGN KEY (`Order_Id`) REFERENCES `BILL`(`Order_Id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `DELIVERY_BOY`;
```

```
create table `DELIVERY_BOY`(  
    `Delivery_Boy_Id` int NOT NULL AUTO_INCREMENT,
```

```

        `Fname` varchar(15) NOT NULL,
        `Lname` varchar(15) NOT NULL,
        `Contact` varchar(20) NOT NULL,
        `Address` varchar(30) DEFAULT NULL,
        `Salary` varchar(30) DEFAULT NULL,
        `Sex` char(1) DEFAULT NULL,
        `Bdate` date DEFAULT NULL,
        `Join_Date` date NOT NULL,
        PRIMARY KEY (`Delivery_Boy_Id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

drop table if exists `DELIVERY_BOY_2`;
create table `DELIVERY_BOY_2`(
        `Boy_Id` int NOT NULL,
        `Delivery_Id` int NOT NULL,
        FOREIGN KEY (`Boy_Id`) REFERENCES
`DELIVERY_BOY`(`Delivery_Boy_Id`),
        FOREIGN KEY (`Delivery_Id`) REFERENCES
`HOME_DELIVERY`(`Delivery_Id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

drop table if exists `MENU`;
create table `MENU`(
        `Menu_Id` int NOT NULL AUTO_INCREMENT,
        `Name` varchar(100) NOT NULL,
        `Price` varchar(20) NOT NULL,
        `Type` varchar(20) DEFAULT NULL,
        `Category` varchar(30) NOT NULL,
        PRIMARY KEY(`Menu_Id`,`Name`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```
drop table if exists `SALE_DETAIL`;
create table `SALE_DETAIL` (
    `Date` date NOT NULL,
    `Daily` int NOT NULL,
    `Weekly` int DEFAULT NULL,
    `Monthly` int DEFAULT NULL,
    `Rname` varchar(30) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `SUPPLIER`;
create table `SUPPLIER` (
    `Fname` varchar(15) NOT NULL,
    `Lname` varchar(15) NOT NULL,
    `Address` varchar(30) NOT NULL,
    `Contact` varchar(20) NOT NULL,
    `Details` varchar(500) DEFAULT NULL,
    PRIMARY KEY (`Fname`, `Lname`, `Address`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
drop table if exists `TABLES`;
create table `TABLES` (
    `Table_Number` varchar(9) NOT NULL,
    `Details` varchar(200) DEFAULT NULL,
    PRIMARY KEY (`Table_Number`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```



```

drop table if exists `BOOKING`;

create table `BOOKING`(
    `Booking_Id` int NOT NULL AUTO_INCREMENT,
    `Table_Num` varchar(30) NOT NULL,
    `Date` varchar(30) NOT NULL,
    `Time` varchar(30) NOT NULL,
    `Cust_Id` int NOT NULL,
    PRIMARY KEY (`Booking_Id`),
    FOREIGN KEY (`Cust_Id`) REFERENCES `CUSTOMER`(`Customer_Id`),
    FOREIGN KEY (`Table_Num`) REFERENCES `TABLES`(`Table_Number`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

drop table if exists `INGREDIENT`;

create table `INGREDIENT`(
    `Ingredient_Id` int NOT NULL AUTO_INCREMENT,
    `Name` varchar(30) NOT NULL,
    `Quantity` varchar(15) NOT NULL,
    `Description` varchar(100) DEFAULT NULL,
    `Supp_Name` varchar(50) NOT NULL,
    PRIMARY KEY (`Ingredient_Id`,`Name`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

insert into
`RESTAURANT`(`Name`,`Location`,`Contact`,`Opening_Closing_Time`,`Details`)
values
("Food Village","Sherpur Bogura","01787964747/01857719067","12:00 AM - 12:00 PM","None");

insert into `OWNER`(`Fname`,`Lname`,`Contact`,`Rest_Name`)
values("MAHMUDUL","HASAN","01857719067","Food Village");

```

```
insert into
`MANAGER`(`Fname`,`Lname`,`Contact`,`Address`,`Salary`,`Sex`,`Bdate`,`Join_Date`)
values
```

```
("Asraful","Islam","01776906190","Omur Gajoghanta , Rangpur","30000","M","1996-09-28","2020-08-01"),
```

```
("Asibul","Asik","01761531774","Omur Gajoghanta , Rangpur","25000","M","1993-11-18","2021-02-01");
```

```
insert into
`WAITER`(`Fname`,`Lname`,`Contact`,`Address`,`Salary`,`Sex`,`Bdate`,`Join_Date`)values
```

```
("Rasel","Islam","123456","E-22 OBH , E-22 OBH ,Sherpur Bogura","8000","U","1993-10-12","2020-08-01"),
```

```
("Ganesh","Das","132244","E-22 OBH ,Sherpur Bogura ","8000","M","1992-12-22","2020-08-01"),
```

```
("Shohag","Islam","213122","E-22 OBH ,Sherpur Bogura","8000","M","1993-05-12","2020-08-01"),
```

```
("Milton","Mallik","121232","E-22 OBH ,Sherpur Bogura ","8000","M","1992-03-12","2020-08-01"),
```

```
("Sofiqul","Islam","213122","DHAKA","9000","M","1986-10-13","2020-08-01"),
```

```
("Sweety","Khatun","323322","DHAKA","9000","F","1982-10-13","2020-08-01");
```

```
insert into
`CASHIER`(`Fname`,`Lname`,`Contact`,`Address`,`Salary`,`Sex`,`Bdate`,`Join_Date`)values
```

```
("Habib","Nazzar","124142","E-19 OBH ,Gajoghanta Rangpur ","12000","M","1993-11-30","2020-08-01"),
```

```
("Abu","Musha","113332","E-19 OBH ,Gajoghanta Rangpur ","12000","M","1993-02-21","2020-08-01"),
```

```
("Bayzid","Hasan","122121","E-19 OBH ,Gajoghanta Rangpur ","12000","M","1992-07-21","2020-08-01");
```

```
insert into
`COOK`(`Fname`,`Lname`,`Contact`,`Address`,`Salary`,`Sex`,`Specialization`,`Bdate`,`Join_Date`)values
```

```
("Biplob","Khan","123211","E-18 OBH , Sherpur Bogura","15000","M","Maggi","1992-08-02","2020-08-01"),
```

```
("Sobuj","Islam","231312","E-19 OBH E-19 OBH , Sherpur Bogura ","15000","M","Samosa","1992-10-12","2020-08-01"),
```

("Rana","Miah","874594"," E-18 OBH ,Sherpur Bogura ","15000","M","Omlette","1992-03-14","2020-08-01"),

("Imran","Khan","587654"," E-26 OBH , Sherpur Bogura ","15000","M","Bonda","1992-01-01","2020-08-01"),

("Rustom","Ali","142355"," E-24 OBH , Sherpur Bogura ","15000","M","Jalebi","1992-02-22","2020-08-01"),

("Priya","Rai","897933","Sherpur Dhaka","15000","F","None","1980-01-02","2020-08-01");

insert into

`DELIVERY_BOY`(`Fname`,`Lname`,`Contact`,`Address`,`Salary`,`Sex`,`Bdate`,`Join_Date`)values

("Sazzad","Miah","133132"," E-17 OBH , Sherpur Bogura ","10000","M","1993-02-21","2020-08-01"),

("Nasir","Soton","657569"," E-17 OBH , Sherpur Bogura ","10000","M","1992-10-21","2020-08-01"),

("Mosiur","Rahman","596509"," E-16 OBH , Sherpur Bogura ","10000","M","1993-02-21","2020-08-01"),

("Milon","Kumar","344244"," E-15 OBH , Sherpur Bogura ","10000","M","1993-10-22","2020-08-01");

insert into `MENU`(`Name`,`Price`,`Type`,`Category`)values

("Vegetable Pakora","3.00","Veg","Starters"),

("Vegetable Samosa","3.00","Veg","Starters"),

("Onion Bhaji ","3.00","Veg","Starters"),

("Potato and Mushroom Chaat","3.00","Veg","Starters"),

("Mushroom Garlic Fry","3.00","Veg","Starters"),

("Chicken Tikka","4.00","Non-Veg","Starters"),

("Tandoori Chicken","4.00","Non-Veg","Starters"),

("Chicken Garlic Fry","4.00","Non-Veg","Starters"),

("Chicken Tikka on Puree","4.00","Non-Veg","Starters"),

("Lamb Tikka","4.00","Non-Veg","Starters"),

("Tandoori King Prawn","6.95","Non-Veg","SeaFood"),

("King Prawn Rosun","5.95","Non-Veg","SeaFood"),

("King Prawn on Puree","5.95","Non-Veg","SeaFood"),

("Prawn Bhuna on Puree", "3.95", "Non-Veg", "SeaFood"),
 ("Prawn Cocktail", "3.95", "Non-Veg", "SeaFood"),
 ("Chi/Lam Sashlik Chi", "9.95", "Non-Veg", "Tandoori Specials"),
 ("Tandoori Deluxe", "12.95", "Non-Veg", "Tandoori Specials"),
 ("Tandoori Chicken Main", "9.95", "Non-Veg", "Tandoori Specials"),
 ("Chicken Tikka", "7.95", "Non-Veg", "Tandoori Specials"),
 ("Lamb Tikka", "9.95", "Non-Veg", "Tandoori Specials"),
 ("Bombay Aloo ", "6.50", "Veg", "Vegetable Dishes"),
 ("Mushroom Bhaji ", "6.50", "Veg", "Vegetable Dishes"),
 ("Saag Dall", "6.50", "Veg", "Vegetable Dishes"),
 ("Mattor Paneer", "6.50", "Veg", "Vegetable Dishes"),
 ("Vegetable Roshun", "6.50", "Veg", "Vegetable Dishes"),
 ("Boiled Rice", "2.50", "Veg", "Side Orders - Rice"),
 ("Keema Pilau Rice", "3.50", "Non-Veg", "Side Orders - Rice"),
 ("Mushroom Rice", "3.20", "Veg", "Side Orders - Rice"),
 ("Garlic Naan", "3.00", "Veg", "Side Orders - Bread"),
 ("Stuffed Naan", "3.50", "Veg", "Side Orders - Bread"),
 ("Chapati", "1.00", "Veg", "Side Orders - Bread"),
 ("Green Salad", "2.00", "Veg", "Side Orders - Sundries"),
 ("Spice Popadum", "0.80", "Veg", "Side Orders - Sundries"),
 ("Chutney", "0.70", "Veg", "Side Orders - Sundries"),
 ("Ras Malai", "2.95", "Veg", "Dessert"),
 ("Ice Cream", "2.95", "Veg", "Dessert"),
 ("Gulab Jamun", "2.95", "Veg", "Dessert"),
 ("Kulfi", "2.95", "Veg", "Dessert"),
 ("Kheer", "2.95", "Veg", "Dessert");

insert into `SUPPLIER`(`Fname`,`Lname`,`Address`,`Contact`,`Details`)
 values

("Mahfuz","Islam"," E-15 OBH , Sherpur Bogura ","123211","Provides Non-Veg Stuff."),

("Monir","Zaman"," E-121 OBH , Sherpur Bogura ","678668","Provides Sea Food."),

("Rustom","Ali"," E-16 OBH , Sherpur Bogura ","856855","Provides Grocery.");

insert into `TABLES`(`Table_Number`,`Details`)values

("1","Capacity 4 People"),

("2","Capacity 4 People Near Window"),

("3","Capacity 3 People"),

("4","Capacity 2 People"),

("5","Capacity 8 People Family Table");

insert into `CUSTOMER`(`Fname`,`Lname`,`Contact`,`Email_Id`)

values

("Arpit","Sharma","938912","arpit.sharma@students.iiit.ac.in"),

("Yash","Shah","289374","yash.shah@students.iiit.ac.in"),

("Darshit","Serna","234322","darshit.serna@students.iiit.ac.in"),

("Aditya","Sharma","778989","aditya.sharma@students.iiit.ac.in"),

("Pallav","Shah","364932","pallav.shah@students.iiit.ac.in"),

("Rajat","Bharadwaj","734277","rajat.bharadwaj@students.iiit.ac.in"),

("Achintya","Madhav","347534","achintya.madhav@students.iiit.ac.in");

```
insert into `USER`(`Fname`,`Lname`,`Password`)
```

```
values
```

```
("admin","admin","admin"),
```

```
("Habib","Nazzar","1234"),
```

```
("Julekha","Akter","4567"),
```

```
("Bayzed","Hasan","7890")
```

