**Database Management Systems LAB**

**(3+1 Credit Hours)**

**CSL-220**

**FOOD DELIVERY SYSTEM**

**Project Report**



**Course Instructor:** **Mr. Malik M. Ali**  
 **Lab Instructor:** **Ms. Hafsa Munawar  
 Semester: Fall’23  
 Class: BSCS 4B**

**Group Members:**

|  |  |
| --- | --- |
| **Enrollment** | **Name** |
| 02-134221-045 | Muhammad Isam |
| 02-134221-019 | Mahira |
| 02-134221-028 | Muskaan Adil |

**DEPARTMENT OF COMPUTER SCIENCE**

BAHRIA UNIVERSITY, KARACHI, PAKISTAN

## Table of Contents

[Table of Contents 2](#_Toc155695195)

[Analysis 3](#_Toc155695196)

[a) Summary 3](#_Toc155695197)

[b) Objectives 3](#_Toc155695198)

[c) Screenshots of main modules to be replicated 3](#_Toc155695199)

[d) Scope of the project 7](#_Toc155695200)

[Methodology 8](#_Toc155695201)

[a) Business Rules to be applied in the DBMS 8](#_Toc155695202)

[b) Entity Relationship Diagram 9](#_Toc155695203)

[Implementation 10](#_Toc155695204)

[a) Conceptual to Logical Mapping 10](#_Toc155695205)

[b) Normalized Tables up to BCNF (SQL Server Schema Diagram) 11](#_Toc155695206)

[c) Code snippets 12](#_Toc155695207)

[Conclusion 53](#_Toc155695208)

[a) Evaluation of the project's success in meeting its objectives 53](#_Toc155695209)

[b) Screenshots of major modules’ outputs 53](#_Toc155695210)

## Analysis

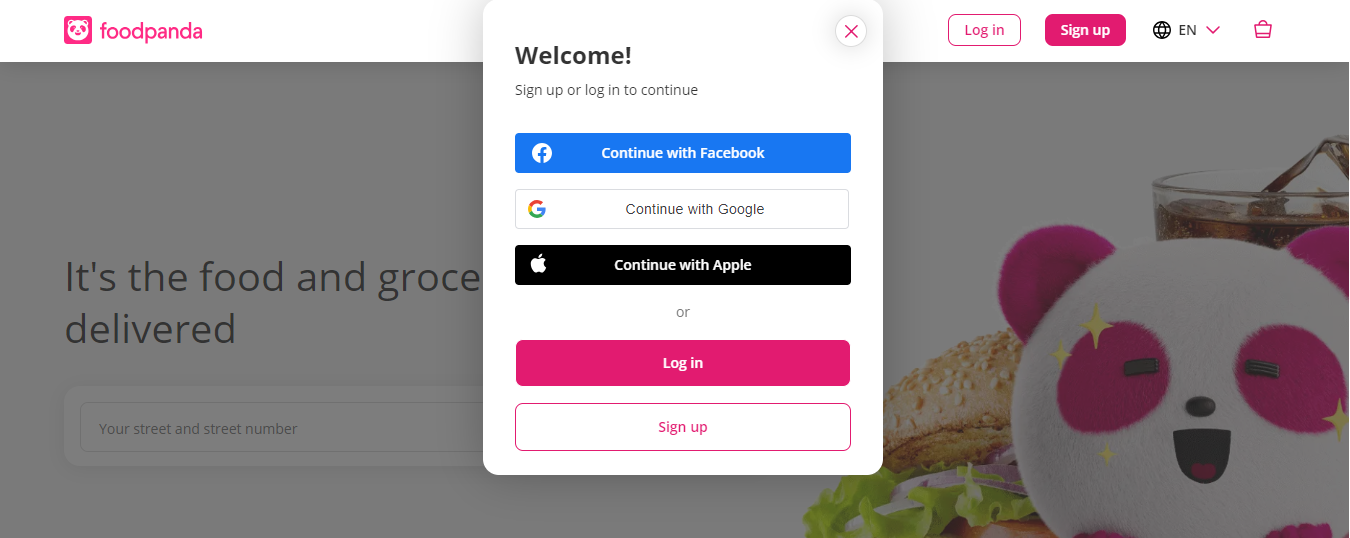
### Summary

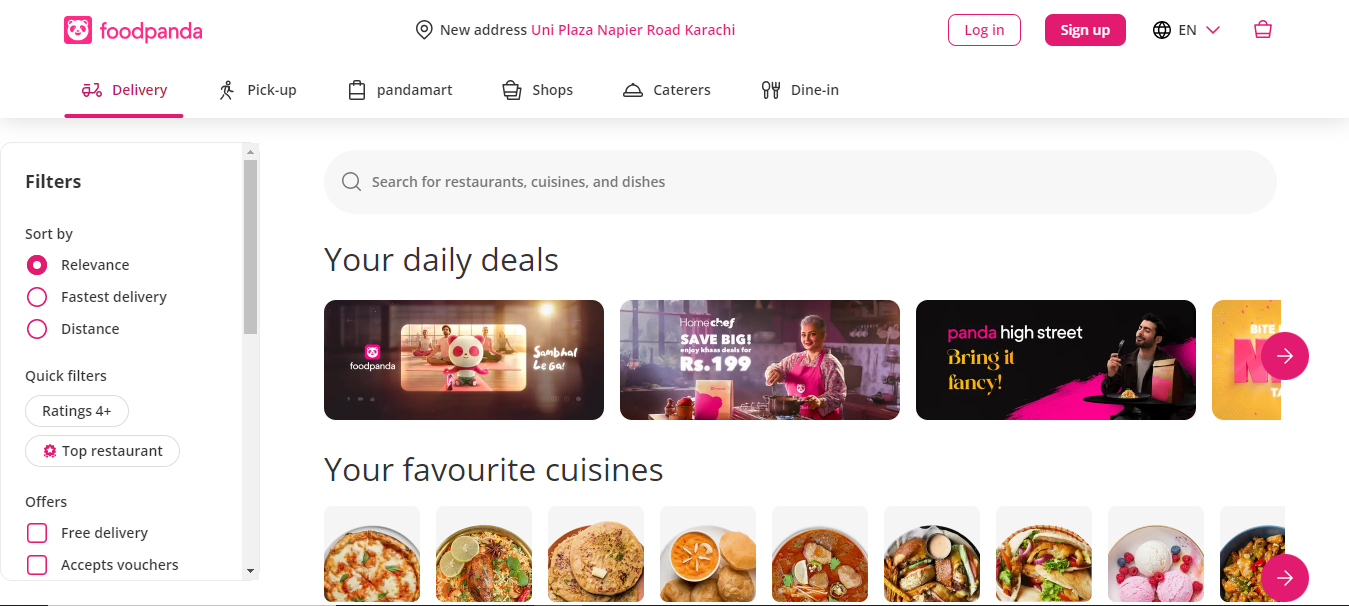
The Food Delivery Database System aims to streamline and enhance the operations of a food delivery service by implementing a robust and efficient database solution. In response to the growing demand for online food delivery, this project focuses on creating a centralized database that can manage and organize information related to restaurants, menu items, orders, and delivery logistics. The system aims to improve the overall customer experience, optimize order processing, and facilitate seamless communication between customers, restaurants, and delivery personnel.

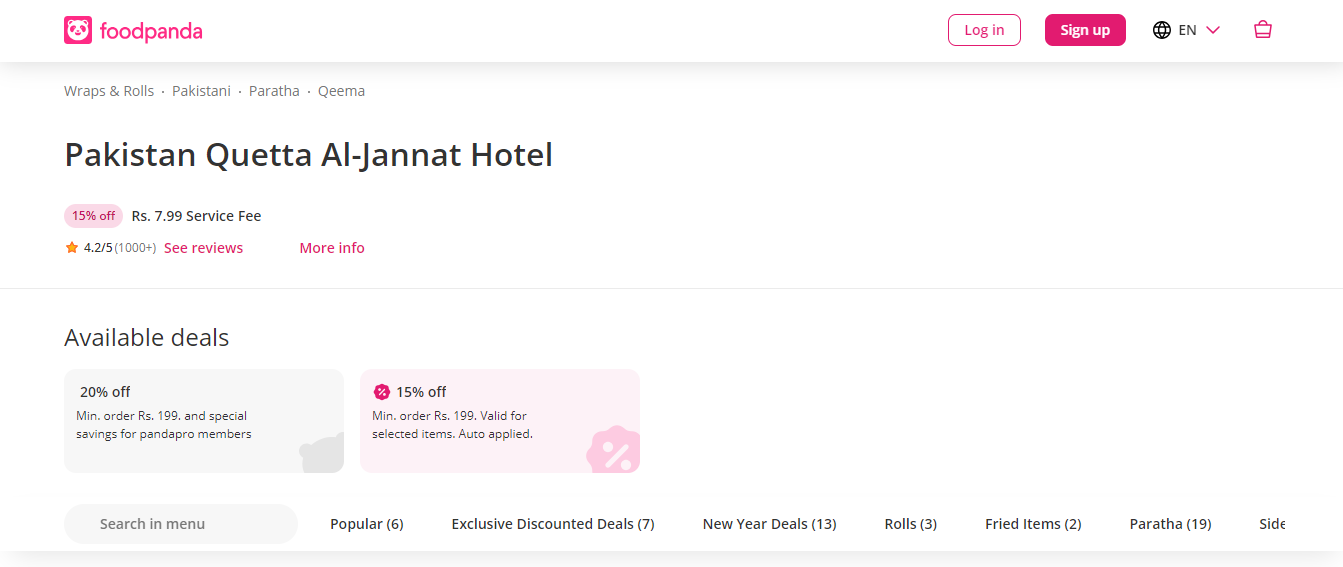
### Objectives

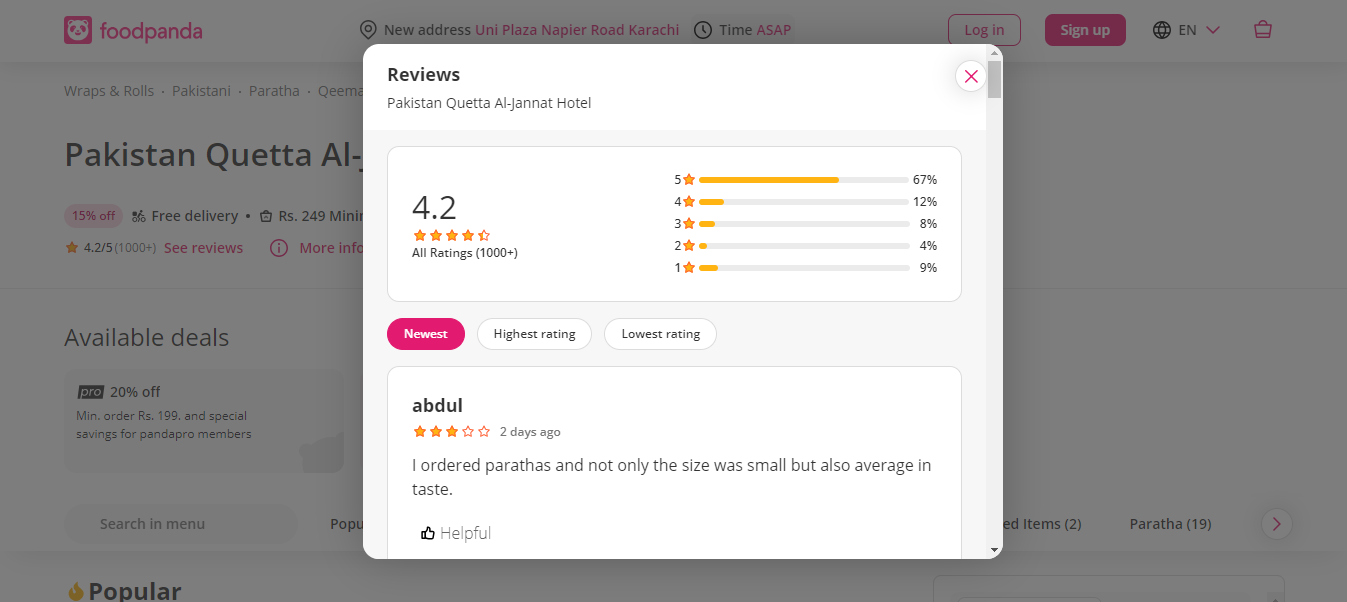
* **Efficient Order Management:** The primary objective was to develop a database system that could efficiently manage and process customer orders. This includes recording order details, managing menu items, and ensuring timely delivery.
* **Menu and Restaurant Information**: The system will provide a comprehensive repository for menu details and restaurant information. This includes menu items, prices, categories, operating hours, and special coupons, allowing for easy updates and maintenance.
* **User-Friendly Interface:** The project aims to create an intuitive and user-friendly interface for both customers and delivery staff. This involves a seamless order placement process and a clear display of menu options.
* **Scalability:** The system is designed to handle the potential growth of the food delivery service. This includes accommodating an increasing number of restaurants, menu items, and customer orders while maintaining optimal performance.

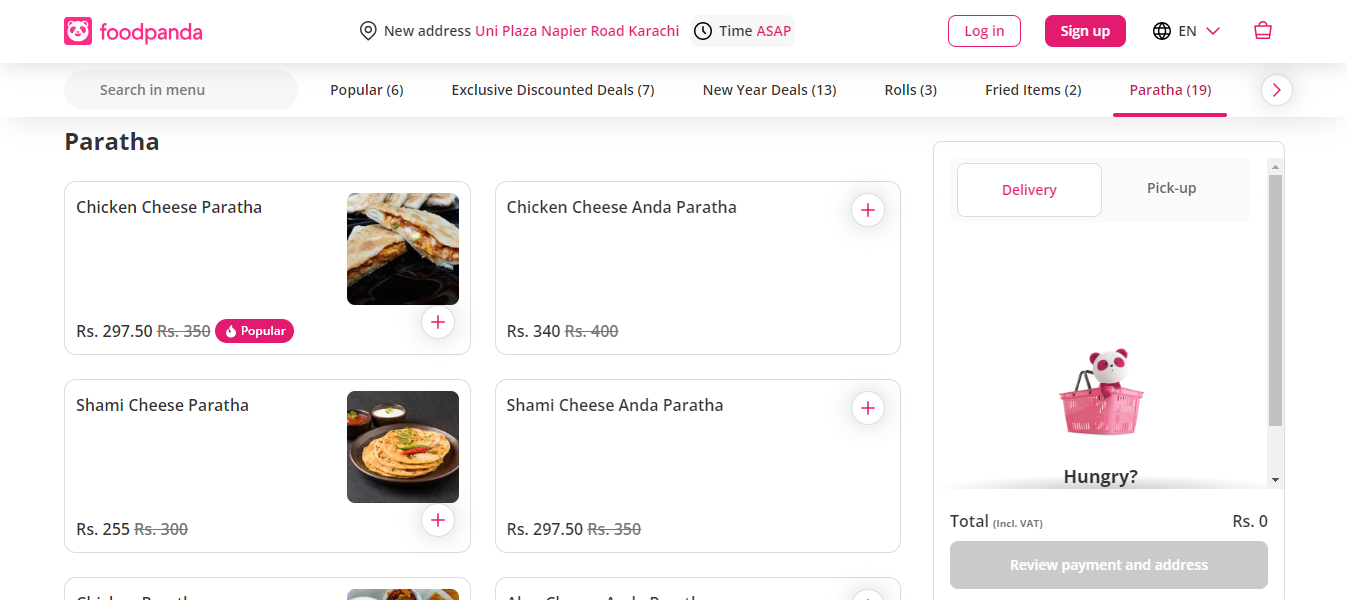
### Screenshots of main modules to be replicated

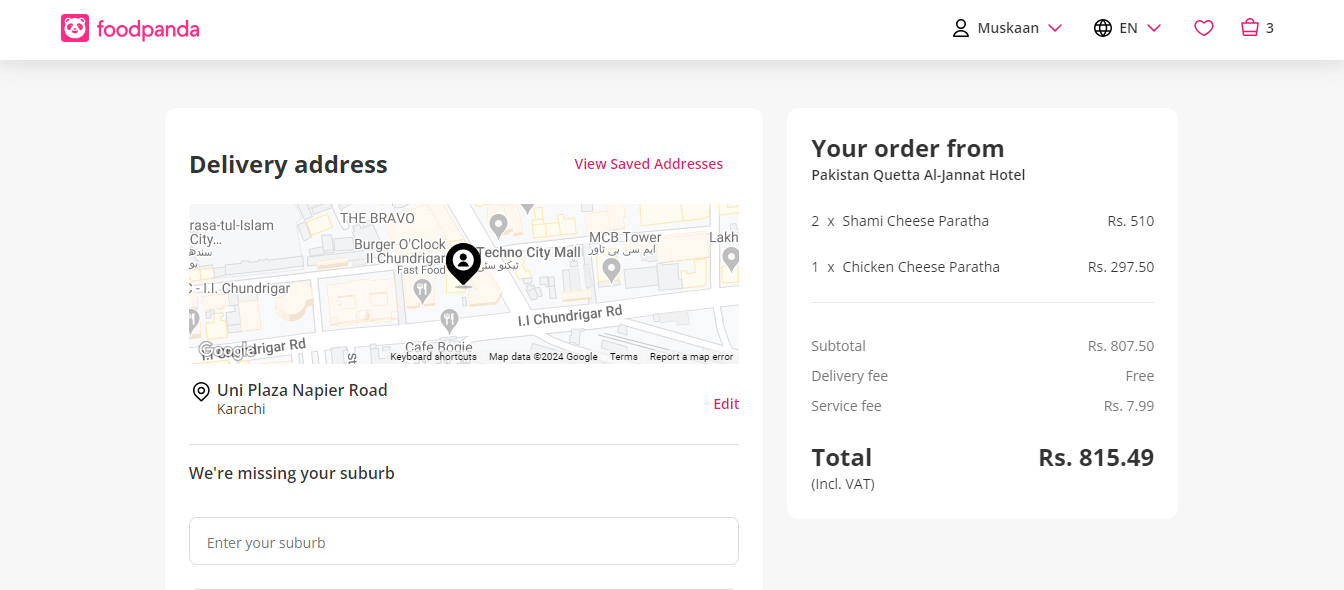


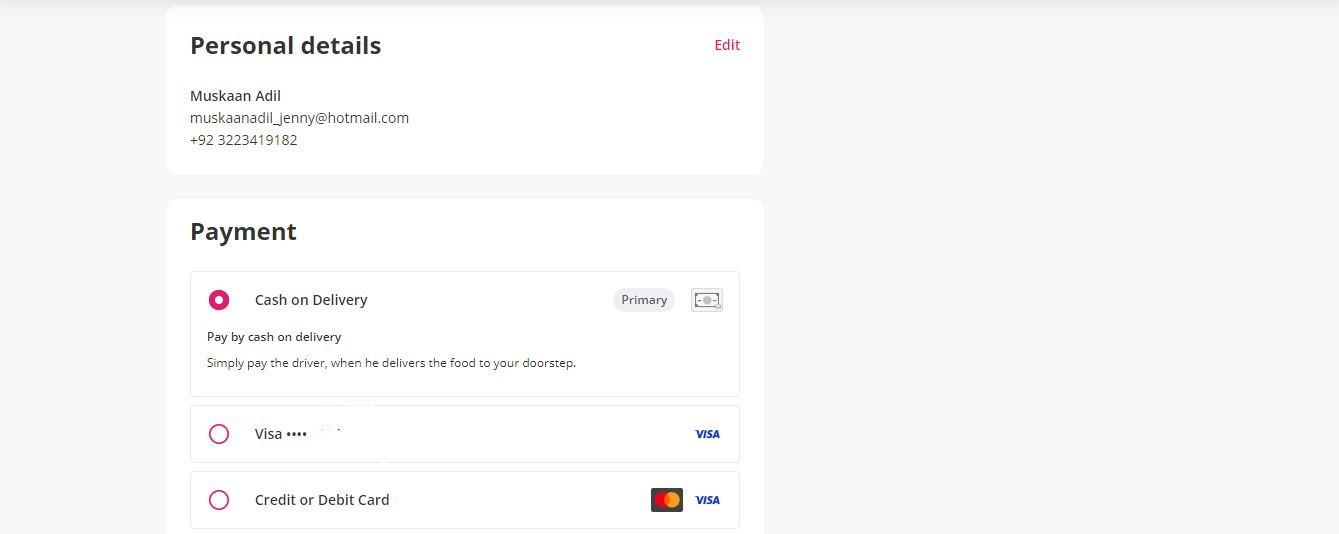


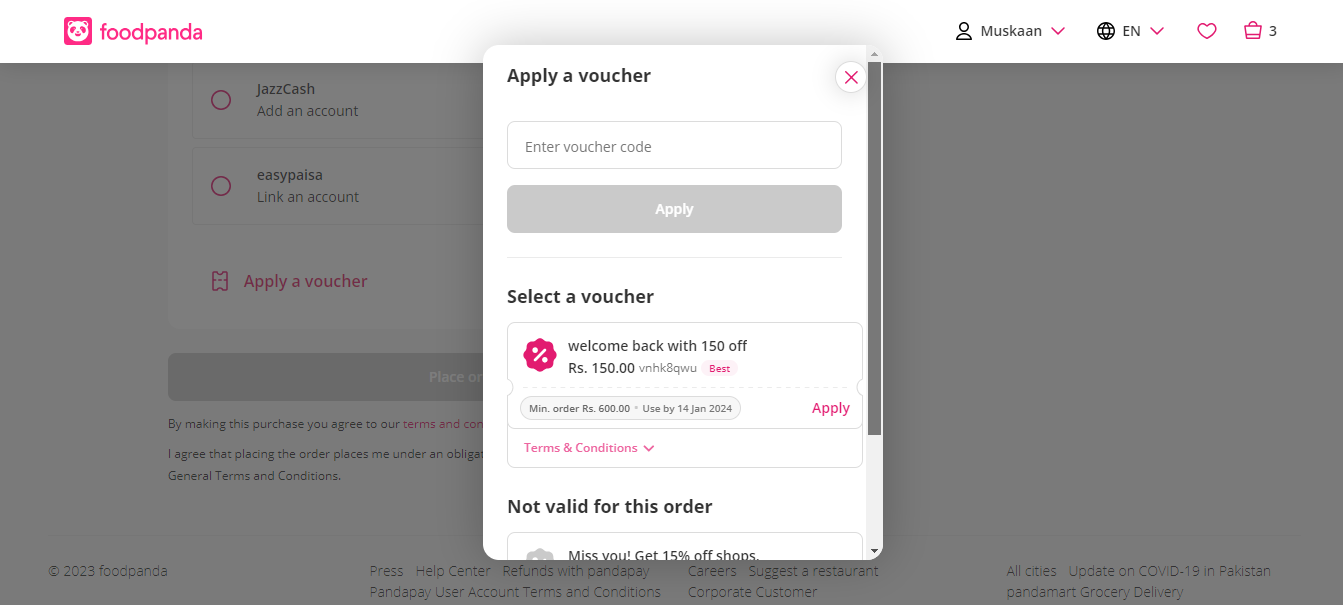


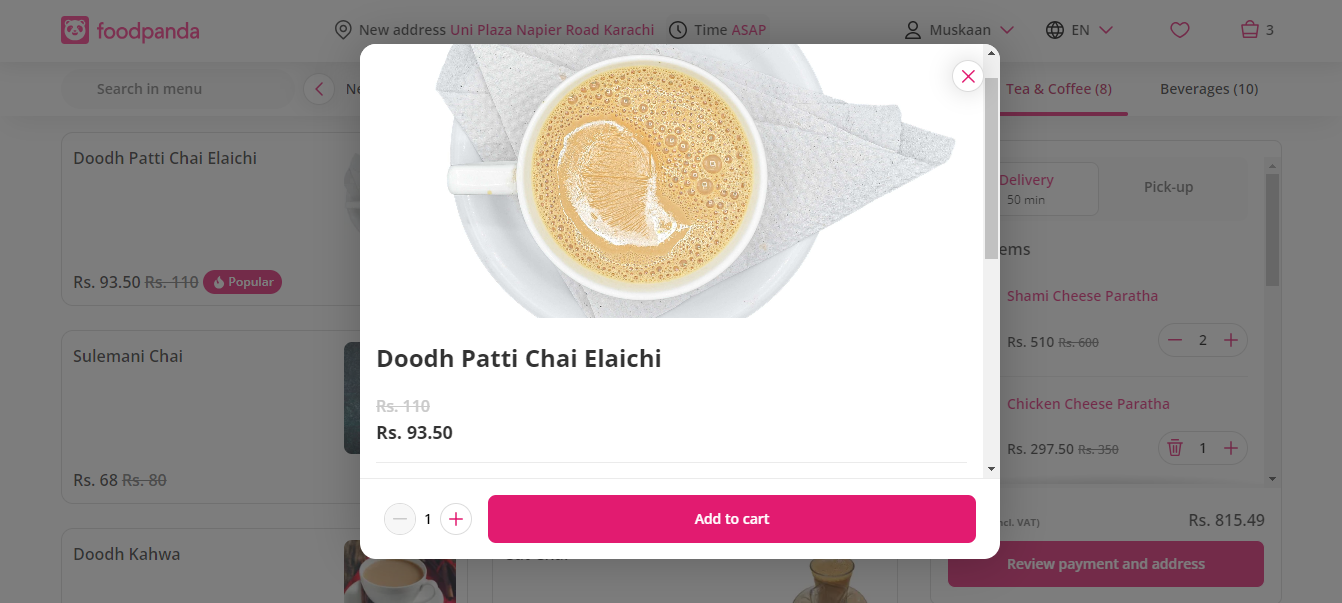


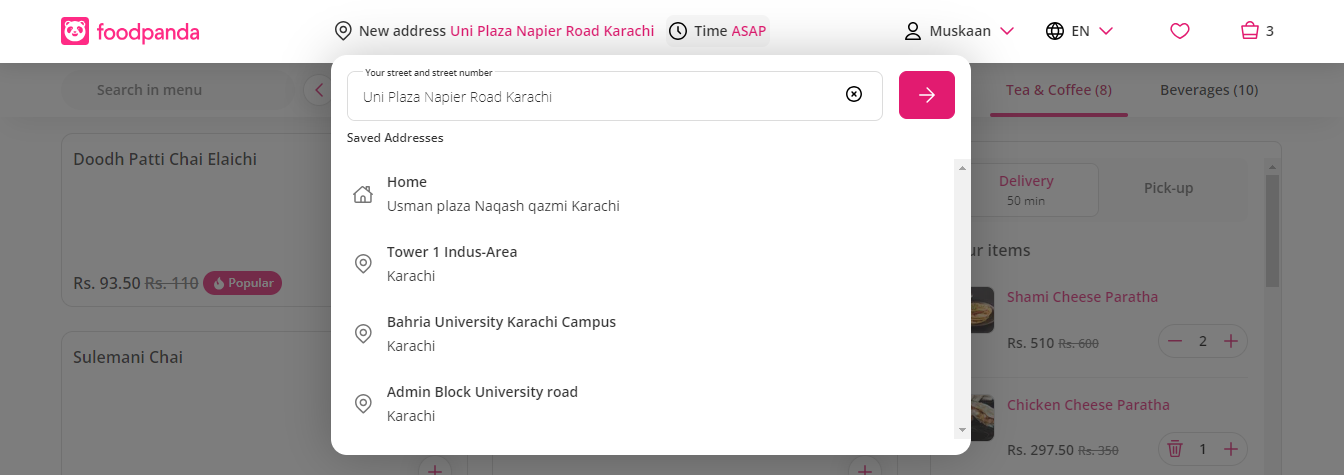


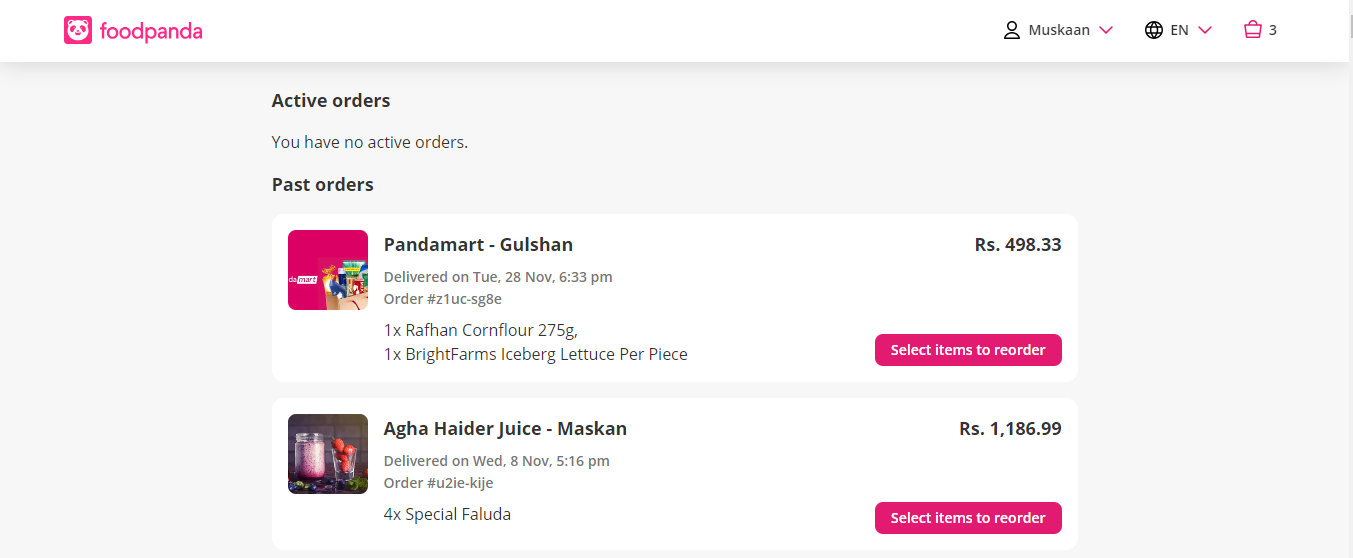


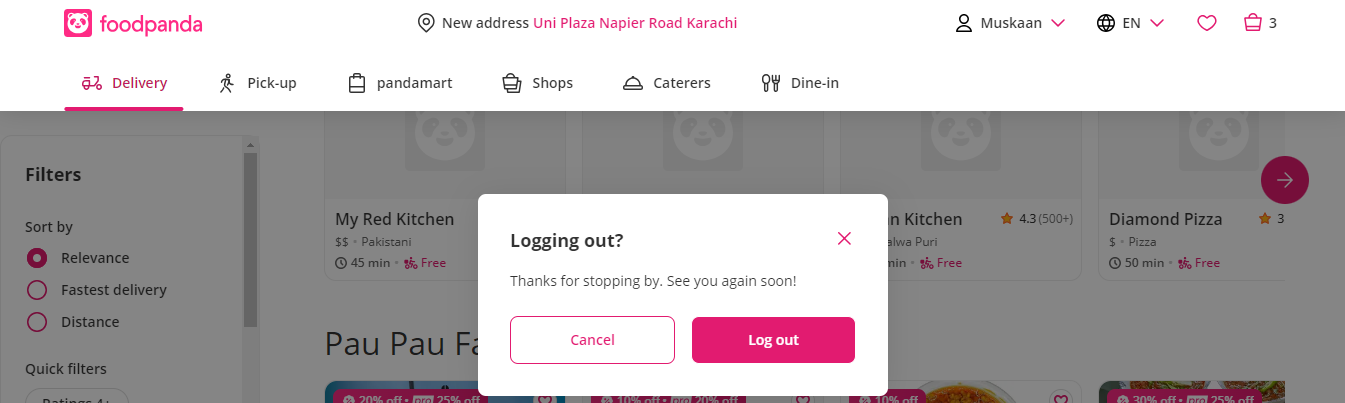












### Scope of the project

* **Customer Management:** Manage customer profiles and order history.
* **Restaurant Information:** Store and update details about partnered restaurants, including menu items, prices, and categories.
* **Order Processing:** Facilitate order placement, payment, and order status.
* **Delivery Logistics:** Implement features for assigning delivery personnel.
* **User Interfaces:** Develop user interfaces for customers and delivery staff, each tailored to their specific needs.

## Methodology

### Business Rules to be applied in the DBMS

* **Category Table:**

A category can be associated with zero or many restaurants.

A restaurant is only associated with one category.

* **Restaurant Table:**

A restaurant can have one or many menu items.

Each menu item belongs to one and only one restaurant.

A restaurant can have zero or many reviews and ratings.

Each review and rating is associated with one restaurant.

A restaurant may take one or many orders.

An order can be taken for one restaurant.

* **MenuItem Table:**

Each order item is associated with one menu item.

* **Order Table:**

An order can have one or many order items.

Each order item is associated with one order.

* **OrderItem Table:**

Each order item must be associated with one order.

Each order item is associated with one menu item.

* **User Table:**

A user can place one or many orders.

Each order is associated with one user.

A user can have one reviews and ratings for one restaurant.

A user can have one or many addresses.

Each address is associated with one or many users.

* **Employee Table:**

An employee can be associated with one vehicle.

An employee delivers many orders.

One order is delivered by one employee.

* **Payment Table:**

A payment must be associated with one order.

Each order can have one payment.

* **ReviewRating Table:**

Each review and rating must be associated with one order.

Each review and rating is associated with one user.

Each review and rating is associated with one restaurant.

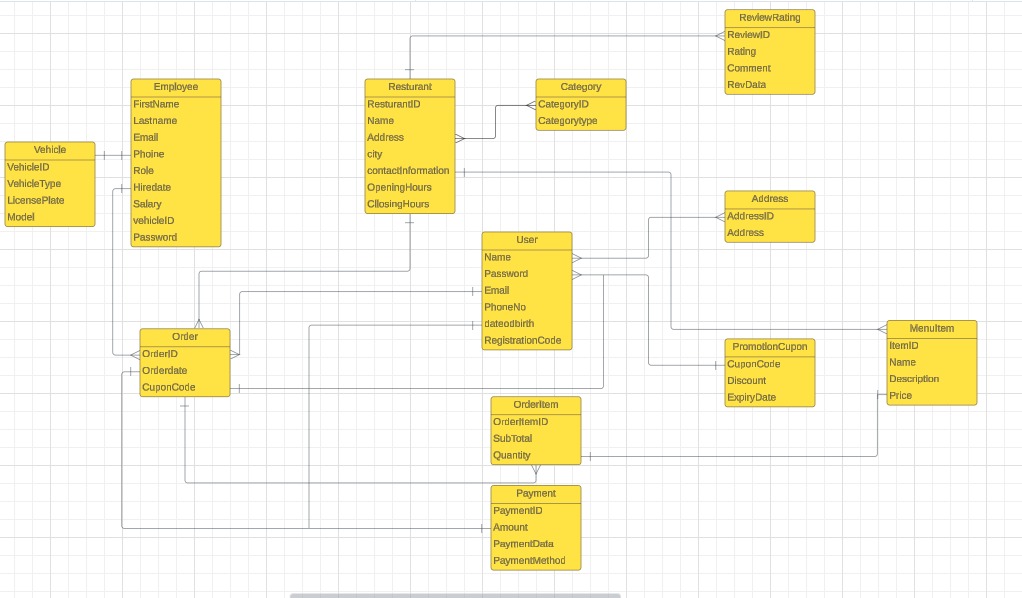
* **PromotionCoupon Table:**

A promotion coupon is associated with many users.

Each user can have many promotion coupons.

One or many users may use one promotion coupon against one order.

### Entity Relationship Diagram



## Implementation

### Conceptual to Logical Mapping

* **Employee Table:** Email -> FirstName, LastName, Phone, Role, HireDate, Salary, VehicleID, Password

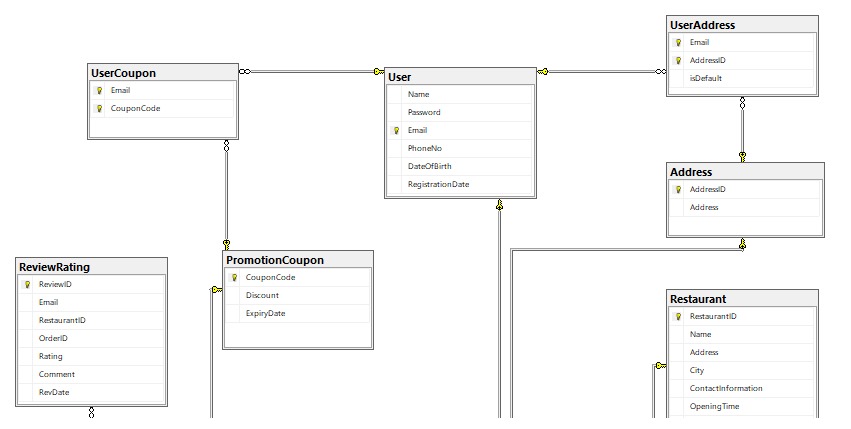
VehicleID -> VehicleType, LicensePlate, Model

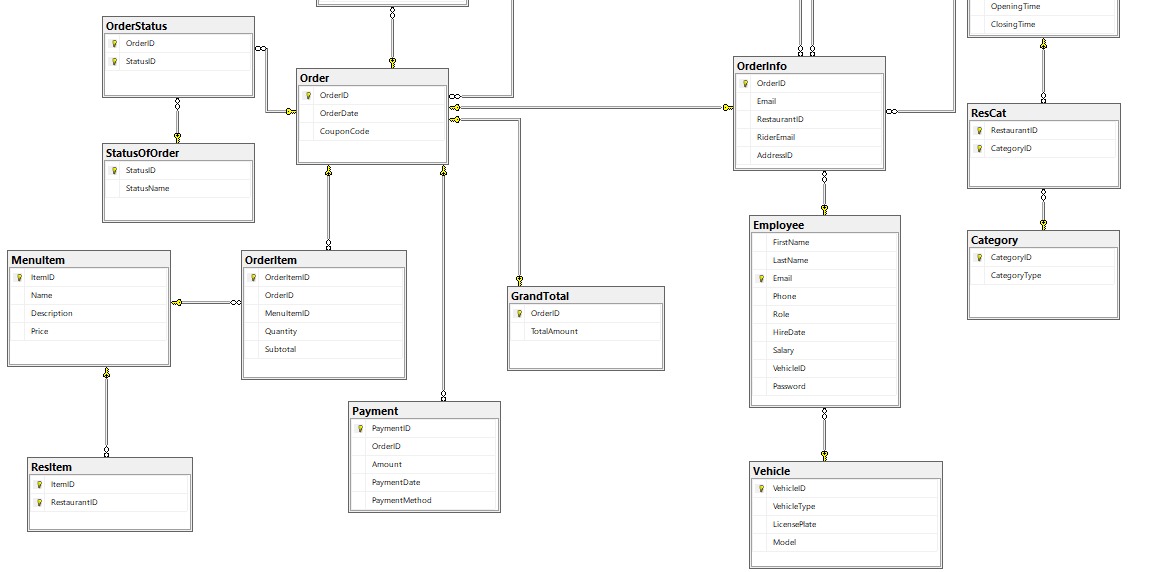
* **GrandTotal Table:** OrderID -> TotalAmount
* **Order Table:** OrderID -> OrderDate, CouponCode
* **OrderInfo Table:** OrderID -> Email, RestaurantID, RiderEmail, AddressID

Email -> Name, Password, PhoneNo, DateOfBirth, RegistrationDate

* **OrderStatus Table:** OrderID, StatusID -> NULL
* **StatusOfOrder Table:** StatusID -> StatusName
* **Payment Table:** PaymentID -> OrderID, Amount, PaymentDate, PaymentMethod
* **PromotionCoupon Table:** CouponCode -> Discount, ExpiryDate
* **Restaurant Table:** RestaurantID -> Name, Address, City, ContactInformation, OpeningTime, ClosingTime
* **Vehicle Table:** VehicleID -> VehicleType, LicensePlate, Model
* **User Table:** Email -> Name, Password, PhoneNo, DateOfBirth, RegistrationDate
* **Address Table:** AddressID -> Address
* **UserAddress Table:** Email, AddressID -> isDefault
* **MenuItem Table:** ItemID -> Name, Description, Price
* **OrderItem Table:** OrderItemID -> OrderID, MenuItemID, Quantity, Subtotal
* **ReviewRating Table:** ReviewID -> Email, RestaurantID, OrderID, Rating, Comment, RevDate
* **ResCat Table:** RestaurantID, CategoryID -> NULL
* **ResItem Table:** ItemID, RestaurantID -> NULL
* **UserCoupon:** Email, CouponCode -> NULL

### Normalized Tables up to BCNF (SQL Server Schema Diagram)





### Code snippets

#### DDL

ALTER TABLE [dbo].[Employee] WITH CHECK ADD CONSTRAINT [fk\_vID] FOREIGN KEY([VehicleID])

REFERENCES [dbo].[Vehicle] ([VehicleID])

GO

ALTER TABLE [dbo].[Employee] CHECK CONSTRAINT [fk\_vID]

GO

ALTER TABLE [dbo].[GrandTotal] WITH CHECK ADD CONSTRAINT [fk\_gtOID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[GrandTotal] CHECK CONSTRAINT [fk\_gtOID]

GO

ALTER TABLE [dbo].[Order] WITH CHECK ADD CONSTRAINT [fk\_couponCode] FOREIGN KEY([CouponCode])

REFERENCES [dbo].[PromotionCoupon] ([CouponCode])

GO

ALTER TABLE [dbo].[Order] CHECK CONSTRAINT [fk\_couponCode]

GO

ALTER TABLE [dbo].[OrderInfo] WITH CHECK ADD CONSTRAINT [fk\_oiAID] FOREIGN KEY([AddressID])

REFERENCES [dbo].[Address] ([AddressID])

GO

ALTER TABLE [dbo].[OrderInfo] CHECK CONSTRAINT [fk\_oiAID]

GO

ALTER TABLE [dbo].[OrderInfo] WITH CHECK ADD CONSTRAINT [fk\_oURID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[OrderInfo] CHECK CONSTRAINT [fk\_oURID]

GO

ALTER TABLE [dbo].[OrderInfo] WITH CHECK ADD CONSTRAINT [fk\_oURRID] FOREIGN KEY([RestaurantID])

REFERENCES [dbo].[Restaurant] ([RestaurantID])

GO

ALTER TABLE [dbo].[OrderInfo] CHECK CONSTRAINT [fk\_oURRID]

GO

ALTER TABLE [dbo].[OrderInfo] WITH CHECK ADD CONSTRAINT [fk\_oURUE] FOREIGN KEY([Email])

REFERENCES [dbo].[User] ([Email])

GO

ALTER TABLE [dbo].[OrderInfo] CHECK CONSTRAINT [fk\_oURUE]

GO

ALTER TABLE [dbo].[OrderInfo] WITH CHECK ADD CONSTRAINT [fk\_remail] FOREIGN KEY([RiderEmail])

REFERENCES [dbo].[Employee] ([Email])

GO

ALTER TABLE [dbo].[OrderInfo] CHECK CONSTRAINT [fk\_remail]

GO

ALTER TABLE [dbo].[OrderItem] WITH CHECK ADD CONSTRAINT [fk\_oIIID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[OrderItem] CHECK CONSTRAINT [fk\_oIIID]

GO

ALTER TABLE [dbo].[OrderItem] WITH CHECK ADD CONSTRAINT [fk\_oIMTID] FOREIGN KEY([MenuItemID])

REFERENCES [dbo].[MenuItem] ([ItemID])

GO

ALTER TABLE [dbo].[OrderItem] CHECK CONSTRAINT [fk\_oIMTID]

GO

ALTER TABLE [dbo].[OrderStatus] WITH CHECK ADD CONSTRAINT [fk\_osOID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[OrderStatus] CHECK CONSTRAINT [fk\_osOID]

GO

ALTER TABLE [dbo].[OrderStatus] WITH CHECK ADD CONSTRAINT [fk\_osSOS] FOREIGN KEY([StatusID])

REFERENCES [dbo].[StatusOfOrder] ([StatusID])

GO

ALTER TABLE [dbo].[OrderStatus] CHECK CONSTRAINT [fk\_osSOS]

GO

ALTER TABLE [dbo].[Payment] WITH CHECK ADD CONSTRAINT [fk\_pOID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[Payment] CHECK CONSTRAINT [fk\_pOID]

GO

ALTER TABLE [dbo].[ResCat] WITH CHECK ADD CONSTRAINT [fk\_catID] FOREIGN KEY([CategoryID])

REFERENCES [dbo].[Category] ([CategoryID])

GO

ALTER TABLE [dbo].[ResCat] CHECK CONSTRAINT [fk\_catID]

GO

ALTER TABLE [dbo].[ResCat] WITH CHECK ADD CONSTRAINT [fk\_res] FOREIGN KEY([RestaurantID])

REFERENCES [dbo].[Restaurant] ([RestaurantID])

GO

ALTER TABLE [dbo].[ResCat] CHECK CONSTRAINT [fk\_res]

GO

ALTER TABLE [dbo].[ResItem] WITH CHECK ADD CONSTRAINT [fk\_itemID] FOREIGN KEY([ItemID])

REFERENCES [dbo].[MenuItem] ([ItemID])

GO

ALTER TABLE [dbo].[ResItem] CHECK CONSTRAINT [fk\_itemID]

CREATE TABLE [dbo].[Employee](

[FirstName] [varchar](25) NOT NULL,

[LastName] [varchar](25) NOT NULL,

[Email] [varchar](25) NOT NULL,

[Phone] [varchar](25) NOT NULL,

[Role] [varchar](25) NOT NULL,

[HireDate] [date] NOT NULL,

[Salary] [decimal](10, 2) NOT NULL,

[VehicleID] [int] NULL,

[Password] [varchar](25) NULL,

CONSTRAINT [pk\_eID] PRIMARY KEY CLUSTERED

(

[Email] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[GrandTotal](

[OrderID] [int] NOT NULL,

[TotalAmount] [decimal](10, 2) NOT NULL,

CONSTRAINT [pk\_gtOID] PRIMARY KEY CLUSTERED

(

[OrderID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[Order](

[OrderID] [int] IDENTITY(1,1) NOT NULL,

[OrderDate] [date] NOT NULL,

[CouponCode] [varchar](25) NULL,

CONSTRAINT [pk\_orderID] PRIMARY KEY CLUSTERED

(

[OrderID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[OrderInfo](

[OrderID] [int] NOT NULL,

[Email] [varchar](25) NOT NULL,

[RestaurantID] [int] NOT NULL,

[RiderEmail] [varchar](25) NOT NULL,

[AddressID] [int] NULL,

CONSTRAINT [pk\_oURID] PRIMARY KEY CLUSTERED

(

[OrderID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[OrderStatus](

[OrderID] [int] NOT NULL,

[StatusID] [int] NOT NULL,

CONSTRAINT [pk\_osID] PRIMARY KEY CLUSTERED

(

[OrderID] ASC,

[StatusID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[StatusOfOrder](

[StatusID] [int] IDENTITY(1,1) NOT NULL,

[StatusName] [varchar](25) NOT NULL,

CONSTRAINT [pk\_sID] PRIMARY KEY CLUSTERED

(

[StatusID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[User](

[Name] [varchar](20) NOT NULL,

[Password] [varchar](20) NOT NULL,

[Email] [varchar](25) NOT NULL,

[PhoneNo] [varchar](20) NOT NULL,

[DateOfBirth] [date] NOT NULL,

[RegistrationDate] [date] NOT NULL,

CONSTRAINT [pk\_uE] PRIMARY KEY CLUSTERED

(

[Email] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[Address](

[AddressID] [int] IDENTITY(1,1) NOT NULL,

[Address] [varchar](50) NOT NULL,

CONSTRAINT [pk\_aID] PRIMARY KEY CLUSTERED

(

[AddressID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[UserAddress](

[Email] [varchar](25) NOT NULL,

[AddressID] [int] NOT NULL,

[isDefault] [bit] NOT NULL,

PRIMARY KEY CLUSTERED

(

[Email] ASC,

[AddressID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[MenuItem](

[ItemID] [int] IDENTITY(1,1) NOT NULL,

[Name] [varchar](25) NOT NULL,

[Description] [varchar](50) NULL,

[Price] [decimal](10, 2) NOT NULL,

CONSTRAINT [pk\_mID] PRIMARY KEY CLUSTERED

(

[ItemID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[OrderItem](

[OrderItemID] [int] IDENTITY(1,1) NOT NULL,

[OrderID] [int] NOT NULL,

[MenuItemID] [int] NOT NULL,

[Quantity] [int] NOT NULL,

[Subtotal] [decimal](10, 2) NOT NULL,

CONSTRAINT [pk\_oIID] PRIMARY KEY CLUSTERED

(

[OrderItemID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[Payment](

[PaymentID] [int] IDENTITY(5023,1) NOT NULL,

[OrderID] [int] NOT NULL,

[Amount] [decimal](10, 2) NOT NULL,

[PaymentDate] [date] NOT NULL,

[PaymentMethod] [varchar](25) NOT NULL,

CONSTRAINT [pk\_pID] PRIMARY KEY CLUSTERED

(

[PaymentID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[PromotionCoupon](

[CouponCode] [varchar](25) NOT NULL,

[Discount] [decimal](5, 2) NOT NULL,

[ExpiryDate] [date] NOT NULL,

CONSTRAINT [pk\_pCID] PRIMARY KEY CLUSTERED

(

[CouponCode] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[Restaurant](

[RestaurantID] [int] IDENTITY(1,1) NOT NULL,

[Name] [varchar](25) NOT NULL,

[Address] [varchar](50) NOT NULL,

[City] [varchar](25) NOT NULL,

[ContactInformation] [varchar](25) NOT NULL,

[OpeningTime] [time](7) NOT NULL,

[ClosingTime] [time](7) NOT NULL,

CONSTRAINT [pk\_resID] PRIMARY KEY CLUSTERED

(

[RestaurantID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[Vehicle](

[VehicleID] [int] IDENTITY(1,1) NOT NULL,

[VehicleType] [varchar](20) NOT NULL,

[LicensePlate] [varchar](7) NOT NULL,

[Model] [varchar](4) NOT NULL,

CONSTRAINT [pk\_vID] PRIMARY KEY CLUSTERED

(

[VehicleID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[UserCoupon](

[Email] [varchar](25) NOT NULL,

[CouponCode] [varchar](25) NOT NULL,

CONSTRAINT [pk\_emailcoupon] PRIMARY KEY CLUSTERED

(

[Email] ASC,

[CouponCode] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[ResItem](

[ItemID] [int] NOT NULL,

[RestaurantID] [int] NOT NULL,

CONSTRAINT [pk\_itResID] PRIMARY KEY CLUSTERED

(

[ItemID] ASC,

[RestaurantID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[ReviewRating](

[ReviewID] [int] IDENTITY(200,1) NOT NULL,

[Email] [varchar](25) NOT NULL,

[RestaurantID] [int] NOT NULL,

[OrderID] [int] NOT NULL,

[Rating] [int] NOT NULL,

[Comment] [varchar](75) NULL,

[RevDate] [date] NOT NULL,

CONSTRAINT [pk\_revID] PRIMARY KEY CLUSTERED

(

[ReviewID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

CREATE TABLE [dbo].[ResCat](

[RestaurantID] [int] NOT NULL,

[CategoryID] [int] NOT NULL,

CONSTRAINT [pk\_rescat] PRIMARY KEY CLUSTERED

(

[RestaurantID] ASC,

[CategoryID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[ReviewRating] WITH CHECK ADD CONSTRAINT [fk\_rrOID] FOREIGN KEY([OrderID])

REFERENCES [dbo].[Order] ([OrderID])

GO

ALTER TABLE [dbo].[ReviewRating] CHECK CONSTRAINT [fk\_rrOID]

GO

ALTER TABLE [dbo].[UserAddress] WITH CHECK ADD FOREIGN KEY([AddressID])

REFERENCES [dbo].[Address] ([AddressID])

GO

ALTER TABLE [dbo].[UserAddress] WITH CHECK ADD FOREIGN KEY([AddressID])

REFERENCES [dbo].[Address] ([AddressID])

GO

ALTER TABLE [dbo].[UserAddress] WITH CHECK ADD FOREIGN KEY([Email])

REFERENCES [dbo].[User] ([Email])

GO

ALTER TABLE [dbo].[UserAddress] WITH CHECK ADD FOREIGN KEY([Email])

REFERENCES [dbo].[User] ([Email])

GO

ALTER TABLE [dbo].[UserCoupon] WITH CHECK ADD CONSTRAINT [fk\_coupcode] FOREIGN KEY([CouponCode])

REFERENCES [dbo].[PromotionCoupon] ([CouponCode])

GO

ALTER TABLE [dbo].[UserCoupon] CHECK CONSTRAINT [fk\_coupcode]

GO

ALTER TABLE [dbo].[UserCoupon] WITH CHECK ADD CONSTRAINT [fk\_useremail] FOREIGN KEY([Email])

REFERENCES [dbo].[User] ([Email])

GO

ALTER TABLE [dbo].[UserCoupon] CHECK CONSTRAINT [fk\_useremail]

GO

#### DML

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (1, N'Street 7 Block H North Nazimabad')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (19, N'House 16 DHA Phase 2')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (20, N'Street14 Clifton Block 5')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (21, N'Gulshan-e-Iqbal Block 13, House 202 ')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (22, N'PECHS Block 6 Bungalow 105')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (23, N'Street 10 North Nazimabad Block L')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (24, N'Bahadurabad Street 14 ,House 11')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (25, N'House 205 Block D Korangi Industrial Area')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (26, N'Askari 5 House 62 Malir Cantt')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (27, N'House 25 Lines Area Saddar')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (28, N'Test Area')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (29, N'Jamshaid Road, Karachi')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (30, N'Gorahpur Road')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (31, N'Test Area')

INSERT [dbo].[Address] ([AddressID], [Address]) VALUES (32, N'Askari 4, Karachi')

INSERT [dbo].[Category] ([CategoryID], [CategoryType]) VALUES (1, N'Fastfood')

INSERT [dbo].[Category] ([CategoryID], [CategoryType]) VALUES (2, N'Bistro')

INSERT [dbo].[Category] ([CategoryID], [CategoryType]) VALUES (3, N'Fried Rice')

INSERT [dbo].[Category] ([CategoryID], [CategoryType]) VALUES (4, N'Ice Cream Parlour')

INSERT [dbo].[Category] ([CategoryID], [CategoryType]) VALUES (5, N'Desi')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Ameer', N'Hamza', N'ah1@gmail.com', N'3013300309', N'Rider', CAST(N'2021-05-24' AS Date), CAST(19000.00 AS Decimal(10, 2)), 1, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Affan', N'Shakir', N'as1@gmail.com', N'3460007772', N'Rider', CAST(N'2021-01-20' AS Date), CAST(19000.00 AS Decimal(10, 2)), 2, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Ameen', N'Shehbaz', N'as2@gmail.com', N'03002223334', N'Manager', CAST(N'2020-06-30' AS Date), CAST(25000.00 AS Decimal(10, 2)), NULL, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Haris', N'Bashir', N'hb@gmail.com', N'3221445599', N'Rider', CAST(N'2021-10-12' AS Date), CAST(19000.00 AS Decimal(10, 2)), 3, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Hamza', N'Siddiqui', N'hs@gmail.com', N'3212226007', N'Rider', CAST(N'2021-11-08' AS Date), CAST(19000.00 AS Decimal(10, 2)), 4, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Shayan', N'Jawed', N'sj@gmail.com', N'3214929190', N'Rider', CAST(N'2020-08-15' AS Date), CAST(19000.00 AS Decimal(10, 2)), 5, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Shahreyar', N'Khan', N'sk@gmail.com', N'3317070900', N'Rider', CAST(N'2022-02-05' AS Date), CAST(19000.00 AS Decimal(10, 2)), 6, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Usama', N'Khalid', N'uk@gmail.com', N'3214433225', N'Rider', CAST(N'2021-07-25' AS Date), CAST(19000.00 AS Decimal(10, 2)), 7, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Zeyan', N'Ahmed', N'za1@gmail.com', N'3318678773', N'Rider', CAST(N'2020-08-25' AS Date), CAST(19000.00 AS Decimal(10, 2)), 8, N'abc')

INSERT [dbo].[Employee] ([FirstName], [LastName], [Email], [Phone], [Role], [HireDate], [Salary], [VehicleID], [Password]) VALUES (N'Zahid', N'Khan', N'zk@fd.com', N'03004005006', N'Rider', CAST(N'2020-06-30' AS Date), CAST(19000.00 AS Decimal(10, 2)), 9, N'abc')

INSERT [dbo].[GrandTotal] ([OrderID], [TotalAmount]) VALUES (1, CAST(1128.00 AS Decimal(10, 2)))

INSERT [dbo].[GrandTotal] ([OrderID], [TotalAmount]) VALUES (2, CAST(398.23 AS Decimal(10, 2)))

INSERT [dbo].[GrandTotal] ([OrderID], [TotalAmount]) VALUES (17, CAST(380.00 AS Decimal(10, 2)))

INSERT [dbo].[GrandTotal] ([OrderID], [TotalAmount]) VALUES (20, CAST(1130.00 AS Decimal(10, 2)))

INSERT [dbo].[GrandTotal] ([OrderID], [TotalAmount]) VALUES (21, CAST(450.00 AS Decimal(10, 2)))

INSERT [dbo].[GrandTotal] ([OrderID], [TotalAmount]) VALUES (22, CAST(450.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (4, N'Belgian Chocolate Shake', N'Creamy Chocolate Shake', CAST(664.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (5, N'Choco Sprinkle', N'Chocolate Donut', CAST(204.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (6, N'The Majesty Burger', N'Double Crispy Chicken Patty', CAST(629.10 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (7, N'Pau Pau Deal', N'Chick N Crunch & Crispy Chicken Piece', CAST(499.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (65, N'Chicken Honey Mustard', N'Chicken Honey Mustard Served In Butter Croissant', CAST(788.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (66, N'BreakFast Croissant', N'Dill Cream Scrambled Eggs Rocket Cheese', CAST(788.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (67, N'Spicy Pakistani', N'Chicken Masala Onions Cheese In Butter Croissant', CAST(788.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (68, N'Cookies', N'Its A Chewy Cookies With Dark Chocolate Chunks', CAST(398.23 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (69, N'Brownies', N'Dark Chocolate Fudge Brownies Made With Chocolate', CAST(398.23 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (70, N'Vanilla Swirl', N'Vanilla flavoured icecream', CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (71, N'Chocolate Swirl', N'Chocolate flavoured ice cream', CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (72, N'Vanilla Brownie Swirl', N'Vanilla ice cream topped with brownie crumb', CAST(180.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (73, N'Vanilla Nuts Sundae', N'Vanilla sundae topped with nuts', CAST(320.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (74, N'Vanilla Stawberry Sundae', N'Vanilla sundae topped with strawberry syrup', CAST(360.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (75, N'Chicken Biryani', N'fluffy basmati rice layered over pieces of chicken', CAST(180.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (76, N'Cheesecake', N'Rich And Creamy, With A Perfect Tangy Finish', CAST(490.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (77, N'Super Cornetto Cup', N'Cup Filled Vanilla Base Topped With Waffle Chunks', CAST(220.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (78, N'Chicken Tikka Pizza', N'Pizza topped with tikka flavoured chickenn,cheese', CAST(400.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (79, N'Chicken Sub And Drink', N'Chicken sub with choice of drink', CAST(599.00 AS Decimal(10, 2)))

INSERT [dbo].[MenuItem] ([ItemID], [Name], [Description], [Price]) VALUES (80, N'Chicken Fajita', N'Tender chicken pieces tossed in fajita seasoning,', CAST(784.00 AS Decimal(10, 2)))

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (1, CAST(N'2023-12-25' AS Date), N'ASKARI30')

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (2, CAST(N'2023-12-29' AS Date), N'NEWYEAR24')

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (3, CAST(N'2023-12-29' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (4, CAST(N'2023-12-29' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (5, CAST(N'2023-12-29' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (6, CAST(N'2023-12-29' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (9, CAST(N'2024-01-01' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (17, CAST(N'2024-01-07' AS Date), N'ASKARI30')

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (18, CAST(N'2024-01-07' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (19, CAST(N'2024-01-07' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (20, CAST(N'2024-01-07' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (21, CAST(N'2024-01-07' AS Date), NULL)

INSERT [dbo].[Order] ([OrderID], [OrderDate], [CouponCode]) VALUES (22, CAST(N'2024-01-07' AS Date), NULL)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (1, N'za@gmail.com', 3, N'zk@fd.com', 1)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (2, N'as@gmail.com', 84, N'hb@gmail.com', 22)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (3, N'as@gmail.com', 85, N'hb@gmail.com', 22)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (4, N'mah@gmail.com', 101, N'sk@gmail.com', 26)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (5, N'ma@gmail.com', 102, N'uk@gmail.com', 25)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (6, N'et@gmail.com', 103, N'hb@gmail.com', 20)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (17, N'as@gmail.com', 84, N'ah1@gmail.com', 30)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (20, N'as@gmail.com', 84, N'ah1@gmail.com', 30)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (21, N'as@gmail.com', 84, N'ah1@gmail.com', 30)

INSERT [dbo].[OrderInfo] ([OrderID], [Email], [RestaurantID], [RiderEmail], [AddressID]) VALUES (22, N'as@gmail.com', 84, N'ah1@gmail.com', 30)

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (1, 1, 6, 1, CAST(629.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (2, 1, 7, 1, CAST(499.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (3, 2, 68, 1, CAST(398.23 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (4, 3, 7, 2, CAST(998.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (5, 3, 70, 2, CAST(260.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (18, 17, 73, 1, CAST(320.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (19, 17, 74, 1, CAST(360.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (20, 17, 70, 1, CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (21, 17, 70, 1, CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (22, 20, 70, 1, CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (23, 20, 73, 2, CAST(640.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (24, 20, 74, 1, CAST(360.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (25, 21, 70, 1, CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (26, 21, 73, 1, CAST(320.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (27, 22, 70, 1, CAST(130.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderItem] ([OrderItemID], [OrderID], [MenuItemID], [Quantity], [Subtotal]) VALUES (28, 22, 73, 1, CAST(320.00 AS Decimal(10, 2)))

INSERT [dbo].[OrderStatus] ([OrderID], [StatusID]) VALUES (1, 2)

INSERT [dbo].[OrderStatus] ([OrderID], [StatusID]) VALUES (2, 1)

INSERT [dbo].[OrderStatus] ([OrderID], [StatusID]) VALUES (3, 2)

INSERT [dbo].[OrderStatus] ([OrderID], [StatusID]) VALUES (22, 1)

INSERT [dbo].[Payment] ([PaymentID], [OrderID], [Amount], [PaymentDate], [PaymentMethod]) VALUES (5023, 1, CAST(1128.00 AS Decimal(10, 2)), CAST(N'2023-12-25' AS Date), N'Cash')

INSERT [dbo].[Payment] ([PaymentID], [OrderID], [Amount], [PaymentDate], [PaymentMethod]) VALUES (5024, 21, CAST(450.00 AS Decimal(10, 2)), CAST(N'2024-01-07' AS Date), N'Cash')

INSERT [dbo].[Payment] ([PaymentID], [OrderID], [Amount], [PaymentDate], [PaymentMethod]) VALUES (5025, 22, CAST(450.00 AS Decimal(10, 2)), CAST(N'2024-01-07' AS Date), N'Cash')

INSERT [dbo].[PromotionCoupon] ([CouponCode], [Discount], [ExpiryDate]) VALUES (N'ASKARI30', CAST(300.00 AS Decimal(5, 2)), CAST(N'2024-01-15' AS Date))

INSERT [dbo].[PromotionCoupon] ([CouponCode], [Discount], [ExpiryDate]) VALUES (N'NEWYEAR24', CAST(240.00 AS Decimal(5, 2)), CAST(N'2024-01-01' AS Date))

INSERT [dbo].[PromotionCoupon] ([CouponCode], [Discount], [ExpiryDate]) VALUES (N'PEHLAORDER', CAST(250.00 AS Decimal(5, 2)), CAST(N'2024-12-12' AS Date))

INSERT [dbo].[PromotionCoupon] ([CouponCode], [Discount], [ExpiryDate]) VALUES (N'PROMO10', CAST(150.00 AS Decimal(5, 2)), CAST(N'2024-01-10' AS Date))

INSERT [dbo].[PromotionCoupon] ([CouponCode], [Discount], [ExpiryDate]) VALUES (N'WINTER10', CAST(100.00 AS Decimal(5, 2)), CAST(N'2024-02-02' AS Date))

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (3, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (4, 2)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (5, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (82, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (83, 2)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (84, 4)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (85, 4)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (86, 3)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (87, 3)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (88, 4)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (100, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (101, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (102, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (103, 1)

INSERT [dbo].[ResCat] ([RestaurantID], [CategoryID]) VALUES (104, 1)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (4, 3)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (5, 3)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (6, 1)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (7, 1)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (65, 4)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (66, 4)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (67, 83)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (68, 83)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (69, 4)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (70, 84)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (71, 85)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (72, 85)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (73, 84)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (74, 84)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (75, 86)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (76, 88)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (77, 100)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (78, 101)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (79, 102)

INSERT [dbo].[ResItem] ([ItemID], [RestaurantID]) VALUES (80, 103)

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (3, N'Kababjees Fried Chicken', N'Ground Floor Komal Heaven Apartment Jauhar', N'Karachi', N'kababjees@gmail.com', CAST(N'11:00:00' AS Time), CAST(N'00:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (4, N'OD', N'5 Star Chowrangi, North Nazimabad', N'Karachi', N'od.customercare@gmail.com', CAST(N'10:00:00' AS Time), CAST(N'00:30:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (5, N'Kababjees Fried Chicken', N'Shop No.1, Saima Paari Tower, Hyderi', N'Karachi', N'kababjees@gmail.com', CAST(N'11:00:00' AS Time), CAST(N'00:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (82, N'Kababjees Fried Chicken', N'8 F Mohammad Ali Society Muhammad Ali Chs (Machs)', N'Karachi', N'kababjees@gmail.com', CAST(N'11:00:00' AS Time), CAST(N'00:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (83, N'OD', N'Shop 15 plot SB 3 KDA scheme 1', N'Karachi', N'od.customercare@gmail.com', CAST(N'09:00:00' AS Time), CAST(N'20:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (84, N'Soft Swirl', N' Plot No Al Mustafa Terrace Shop 7 Jamshed Rd', N'Karachi', N'softswirl@gmail.com', CAST(N'20:00:00' AS Time), CAST(N'03:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (85, N'Soft Swirl', N'Khaliq-uz-Zaman Rd Block 8 Clifton', N'Karachi', N'softswirl@gmail.com', CAST(N'20:00:00' AS Time), CAST(N'03:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (86, N'AR REHMAN Biryani', N'Abdullah Haroon Rd Saddar Saddar Town', N'Karachi', N'rehmanbiryani@gmail.com', CAST(N'10:00:00' AS Time), CAST(N'01:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (87, N'AR REHMAN Biryani', N'Clayton Rd Soldier Bazaar Garden East', N'Karachi', N'rehmanbiryani@gmail.com', CAST(N'10:00:00' AS Time), CAST(N'01:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (88, N'Sweet Creme', N'86 Alamgir Rd CP & Berar Society CP & Berar CHS', N'Karachi', N'sweetcreme@gmail.com', CAST(N'17:00:00' AS Time), CAST(N'03:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (100, N'Pizza Hut', N' Plot # 20, Time Square Plaza, I-8 Markaz', N'Islamabad', N'pizzahut@gmail.com', CAST(N'11:00:00' AS Time), CAST(N'22:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (101, N'Pizza Hut', N'The Centaurus Mall, Jinnah Avenue, F 8/4 F-8', N'Islamabad', N'pizzahut@gmail.com', CAST(N'11:00:00' AS Time), CAST(N'22:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (102, N'Pizza Hut', N' Near Arena Cinema, Phase IV, Bahria Town Phase 4', N'Islamabad', N'pizzahut@gmail.com', CAST(N'17:00:00' AS Time), CAST(N'23:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (103, N'SUBWAY', N'Abdul Haque Rd, Block G1 Phase 1 Johar Town', N'Lahore', N'subway@gmail.com', CAST(N'17:00:00' AS Time), CAST(N'23:00:00' AS Time))

INSERT [dbo].[Restaurant] ([RestaurantID], [Name], [Address], [City], [ContactInformation], [OpeningTime], [ClosingTime]) VALUES (104, N'SUBWAY', N'2, Chenab Block Allama Iqbal Town', N'Lahore', N'subway@gmail.com', CAST(N'12:00:00' AS Time), CAST(N'19:00:00' AS Time))

INSERT [dbo].[ReviewRating] ([ReviewID], [Email], [RestaurantID], [OrderID], [Rating], [Comment], [RevDate]) VALUES (201, N'za@gmail.com', 3, 1, 5, N'Good taste and timely delivery', CAST(N'2023-12-30' AS Date))

INSERT [dbo].[ReviewRating] ([ReviewID], [Email], [RestaurantID], [OrderID], [Rating], [Comment], [RevDate]) VALUES (210, N'as@gmail.com', 84, 2, 1, N'Poor', CAST(N'2024-01-04' AS Date))

INSERT [dbo].[StatusOfOrder] ([StatusID], [StatusName]) VALUES (1, N'Pending')

INSERT [dbo].[StatusOfOrder] ([StatusID], [StatusName]) VALUES (2, N'Delivered')

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'ajay khan', N'fbad', N'a@gmail.com', N'03000000', CAST(N'2000-03-15' AS Date), CAST(N'2024-01-03' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Ahsan Ansari', N'ahsan1', N'ansari@gmail.com', N'0300400500', CAST(N'2000-09-05' AS Date), CAST(N'2024-01-08' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Arham Raza', N'ar2223', N'ar@gmail.com', N'3008562400', CAST(N'1980-07-12' AS Date), CAST(N'2023-11-22' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Adeel Sharif', N'as1213', N'as@gmail.com', N'3410001234', CAST(N'1992-08-05' AS Date), CAST(N'2023-12-29' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Dua Adeel', N'da1415', N'da@gmail.com', N'3054009488', CAST(N'1985-04-22' AS Date), CAST(N'2023-12-30' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Emad Tariq', N'et789', N'et@gmail.com', N'3234500087', CAST(N'1995-05-20' AS Date), CAST(N'2023-12-27' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Isam Kamleen', N'ik1617', N'ik@gmail.com', N'3002226007', CAST(N'1990-09-15' AS Date), CAST(N'2023-12-31' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Feroze', N'fk', N'Khan', N'030000000', CAST(N'2000-02-01' AS Date), CAST(N'2024-01-03' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Muskaan Adil', N'ma1819', N'ma@gmail.com', N'3005393531', CAST(N'1983-06-28' AS Date), CAST(N'2023-12-24' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Mahira', N'mah2021', N'mah@gmail.com', N'3134404356', CAST(N'1998-02-18' AS Date), CAST(N'2023-12-23' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Sofia Haider', N'sh1011', N'sh@gmail.com', N'3215356143', CAST(N'1988-11-10' AS Date), CAST(N'2023-12-28' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'test', N'test', N'test@gmail.com', N'505', CAST(N'2020-05-02' AS Date), CAST(N'2024-01-03' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Zaman Afridi', N'za123', N'za@gmail.com', N'0311200000', CAST(N'2000-02-01' AS Date), CAST(N'2023-12-25' AS Date))

INSERT [dbo].[User] ([Name], [Password], [Email], [PhoneNo], [DateOfBirth], [RegistrationDate]) VALUES (N'Zobia Gul', N'zg456', N'zg@gmail.com', N'3332189931', CAST(N'2000-01-15' AS Date), CAST(N'2023-12-26' AS Date))

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'a@gmail.com', 30, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'ansari@gmail.com', 32, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'ar@gmail.com', 27, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'as@gmail.com', 1, 0)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'as@gmail.com', 22, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'da@gmail.com', 23, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'et@gmail.com', 20, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'ik@gmail.com', 24, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'Khan', 29, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'ma@gmail.com', 25, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'mah@gmail.com', 26, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'sh@gmail.com', 21, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'test@gmail.com', 28, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'test@gmail.com', 31, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'za@gmail.com', 1, 1)

INSERT [dbo].[UserAddress] ([Email], [AddressID], [isDefault]) VALUES (N'zg@gmail.com', 19, 1)

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (1, N'Bike', N'CDK-555', N'2010')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (2, N'Bike', N'ABC-222', N'2011')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (3, N'Bike', N'AAA-132', N'2013')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (4, N'Bike', N'ACK-444', N'2014')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (5, N'Bike', N'KSM-567', N'2015')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (6, N'Bike', N'MIK-890', N'2016')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (7, N'Bike', N'JKL-134', N'2017')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (8, N'Bike', N'MNO-567', N'2018')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (9, N'Bike', N'PQR-111', N'2019')

INSERT [dbo].[Vehicle] ([VehicleID], [VehicleType], [LicensePlate], [Model]) VALUES (10, N'Bike', N'STU-88', N'2020')

#### Views

create view getRestaurants

as

Select top(3) Name, Address

from Restaurant

create view getCategory

as

Select Top(3) CategoryType from Category

create view TopPicks

as

Select Top(3) r.Name, r.Address, avg(Rating)'Rating' from Restaurant r, ReviewRating rr

where r.RestaurantID=rr.RestaurantID

group by rr.RestaurantID,r.Name, r.Address

order by avg(Rating)

create view RidersOnDelivery

as

SELECT oi.RiderEmail

FROM Employee e

LEFT JOIN OrderInfo oi ON e.Email = oi.RiderEmail JOIN OrderStatus os ON oi.OrderID = os.OrderID

LEFT JOIN StatusOfOrder so ON so.StatusID = os.StatusID

WHERE (so.StatusName IS NULL OR so.StatusName = 'Pending')

AND e.Role = 'Rider'

create view RidersAvailable

as

select distinct e.Email as 'RiderEmail'

from Employee e, RidersOnDelivery r

where e.Role='Rider'

AND NOT e.Email=r.RiderEmail

create view getAvailableCities

as

Select distinct City from Restaurant R

create view viewUserReviews

as

select rr.RevDate, u.Name'CustomerName', r.Name'RestaurantName', rr.Comment,rr.Rating

from ReviewRating rr, Restaurant r, [User] u

where u.email=rr.email

AND r.RestaurantID=rr.RestaurantID

#### Stored Procedures

create procedure addUserCoupon

@Email varchar(25), @CouponCode varchar(25)

as

begin

insert into UserCoupon(Email,CouponCode) values (@Email,@CouponCode)

end

create procedure addUserAddress

@Email varchar(25),

@Address varchar(50)

as

begin

declare @AddressID int;

if NOT exists(Select \* from Address where Address=@Address)

begin

Insert into Address(Address)values(@Address)

select Top 1 @AddressID=AddressID from [Address]

order by AddressID desc

end

if exists(Select \* from Address where Address=@Address)

begin

select @AddressID=AddressID from Address

where Address=@Address

end

Insert into UserAddress(Email,AddressID,isDefault) values (@Email,@AddressID,0)

end

create procedure CustomerSignUp

@name varchar(25),

@email varchar(25),

@password varchar(25),

@phoneno varchar(25),

@address varchar(50),

@city varchar(25),

@dob date

as

begin

insert into Address(Address) values (@address)

select @address=AddressID from Address where Address=@address

insert into [user](Name, Email, Password, PhoneNo, DateOfBirth, RegistrationDate) values (@name, @email, @password, @phoneno, @dob, GETDATE())

insert into UserAddress(email, AddressID, isDefault) values (@email, @address, 1)

end

create procedure submitReview

@Email varchar(25),

@ResID int,

@OrderID int,

@Rating int,

@Comment varchar(75)

as

begin

Insert into ReviewRating(Email,RestaurantID,OrderID,Rating,Comment,RevDate) values (@Email,@ResID,@OrderID,@Rating,@Comment,getdate())

end

create procedure placeOrder

@Code varchar(25)

as

begin

if (@Code='N/A')

begin

insert into [Order](OrderDate) values (getdate())

end

else

begin

insert into [Order](OrderDate,CouponCode) values (getdate(),@Code)

end

end

create procedure placeItemOrder

@MName varchar(25),

@Price decimal(10,2),

@Qty int,

@Subtotal decimal(10,2)

as

begin

declare @OrderID int, @MenuID int;

SELECT TOP 1 @OrderID=OrderID

FROM [Order]

ORDER BY OrderID DESC;

select @MenuID=ItemID

from MenuItem

where Name=@MName AND Price=@Price

insert into OrderItem(OrderID,MenuItemID,Quantity,Subtotal) values (@OrderID,@MenuID,@Qty,@Subtotal)

end

create procedure confirmCartOrder

@Address varchar(50),

@RName varchar(50),

@RAddress varchar(50),

@UEmail varchar(25),

@PM varchar(25)

as

begin

declare @RiderEmail varchar(25), @OrderID int, @AddressID int, @ResID int, @GTotal decimal(10,2);

SELECT TOP 1 @OrderID=OrderID

FROM [Order]

ORDER BY OrderID DESC;

select TOP 1 @RiderEmail=RiderEmail from RidersAvailable

select @AddressID=AddressID from Address

where Address=@Address

Select @ResID=RestaurantID from Restaurant

where Name=@RName AND Address=@RAddress

insert into OrderInfo(OrderID,Email,RiderEmail,RestaurantID,AddressID)

values (@OrderID,@UEmail,@RiderEmail,@ResID,@AddressID)

select @GTotal=dbo.calGrandTotal(@OrderID)

insert into GrandTotal(OrderID,TotalAmount) values (@OrderID,@GTotal)

insert into Payment(OrderID,Amount,PaymentDate,PaymentMethod) values (@OrderID,@GTotal,getdate(),@PM)

insert into OrderStatus(OrderID,StatusID) values (@OrderID,1)

end

create procedure riderDeliverOrder

@OrderID int, @msg varchar(30) OUTPUT

as

begin

if exists (Select StatusID from OrderStatus where StatusID=1 AND OrderID=@OrderID)

begin

Update OrderStatus set StatusID=2

where OrderID=@OrderID

set @msg='Updated!'

end

else

set @msg='Order is already delivered!'

end

CREATE PROCEDURE InsertUser

@Name VARCHAR(20),

@Password VARCHAR(20),

@Email VARCHAR(25),

@PhoneNo VARCHAR(20),

@DateOfBirth DATE,

@RegistrationDate DATE

AS

BEGIN

INSERT INTO [User] (Name, Password, Email, PhoneNo, DateOfBirth, RegistrationDate)

VALUES (@Name, @Password, @Email, @PhoneNo, @DateOfBirth, @RegistrationDate)

END

CREATE PROCEDURE InsertAddress

@Address VARCHAR(50)

AS

BEGIN

INSERT INTO Address (Address)

VALUES (@Address)

END

CREATE PROCEDURE InsertUserAddress

@Email VARCHAR(25),

@AddressID INT,

@isDefault BIT

AS

BEGIN

INSERT INTO UserAddress (Email, AddressID, isDefault)

VALUES (@Email, @AddressID, @isDefault)

END

CREATE PROCEDURE InsertEmployee

@FirstName VARCHAR(25),

@LastName VARCHAR(25),

@Email VARCHAR(25),

@Phone VARCHAR(25),

@Role VARCHAR(25),

@HireDate DATE,

@Salary DECIMAL(10, 2),

@VehicleID INT,

@Password VARCHAR(25)

AS

BEGIN

INSERT INTO Employee (FirstName, LastName, Email, Phone, Role, HireDate, Salary, VehicleID, Password)

VALUES (@FirstName, @LastName, @Email, @Phone, @Role, @HireDate, @Salary, @VehicleID, @Password)

END

CREATE PROCEDURE InsertGrandTotal

@OrderID INT,

@TotalAmount DECIMAL(10, 2)

AS

BEGIN

INSERT INTO GrandTotal (OrderID, TotalAmount)

VALUES (@OrderID, @TotalAmount)

END

CREATE PROCEDURE InsertMenuItem

@Name VARCHAR(25),

@Description VARCHAR(50),

@Price DECIMAL(10, 2)

AS

BEGIN

INSERT INTO MenuItem (Name, Description, Price)

VALUES (@Name, @Description, @Price)

END

CREATE PROCEDURE InsertOrder

@OrderDate DATE,

@CouponCode VARCHAR(25)

AS

BEGIN

INSERT INTO [Order] (OrderDate, CouponCode)

VALUES (@OrderDate, @CouponCode)

END

CREATE PROCEDURE InsertOrderInfo

@OrderID INT,

@Email VARCHAR(25),

@RestaurantID INT,

@RiderEmail VARCHAR(25),

@AddressID INT

AS

BEGIN

INSERT INTO OrderInfo (OrderID, Email, RestaurantID, RiderEmail, AddressID)

VALUES (@OrderID, @Email, @RestaurantID, @RiderEmail, @AddressID)

END

CREATE PROCEDURE InsertOrderItem

@OrderID INT,

@MenuItemID INT,

@Quantity INT,

@Subtotal DECIMAL(10, 2)

AS

BEGIN

INSERT INTO OrderItem (OrderID, MenuItemID, Quantity, Subtotal)

VALUES (@OrderID, @MenuItemID, @Quantity, @Subtotal)

END

CREATE PROCEDURE InsertPayment

@OrderID INT,

@Amount DECIMAL(10, 2),

@PaymentDate DATE,

@PaymentMethod VARCHAR(25)

AS

BEGIN

INSERT INTO Payment (OrderID, Amount, PaymentDate, PaymentMethod)

VALUES (@OrderID, @Amount, @PaymentDate, @PaymentMethod)

END

CREATE PROCEDURE InsertPromotionCoupon

@CouponCode VARCHAR(25),

@Discount DECIMAL(5, 2),

@ExpiryDate DATE

AS

BEGIN

INSERT INTO PromotionCoupon (CouponCode, Discount, ExpiryDate)

VALUES (@CouponCode, @Discount, @ExpiryDate)

END

CREATE PROCEDURE InsertRestaurant

@Name VARCHAR(25),

@Address VARCHAR(50),

@City VARCHAR(25),

@ContactInformation VARCHAR(25),

@OpeningTime TIME,

@ClosingTime TIME

AS

BEGIN

INSERT INTO Restaurant (Name, Address, City, ContactInformation, OpeningTime, ClosingTime)

VALUES (@Name, @Address, @City, @ContactInformation, @OpeningTime, @ClosingTime)

END

CREATE PROCEDURE InsertVehicle

@VehicleType VARCHAR(20),

@LicensePlate VARCHAR(7),

@Model VARCHAR(4)

AS

BEGIN

INSERT INTO Vehicle (VehicleType, LicensePlate, Model)

VALUES (@VehicleType, @LicensePlate, @Model)

END

CREATE PROCEDURE DeleteUser

@Email VARCHAR(25)

AS

BEGIN

DELETE FROM [User]

WHERE Email = @Email

END

CREATE PROCEDURE DeleteAddress

@AddressID INT

AS

BEGIN

DELETE FROM Address

WHERE AddressID = @AddressID

END

CREATE PROCEDURE DeleteUserAddress

@Email VARCHAR(25),

@AddressID INT

AS

BEGIN

DELETE FROM UserAddress

WHERE Email = @Email AND AddressID = @AddressID

END

CREATE PROCEDURE DeleteEmployee

@Email VARCHAR(25)

AS

BEGIN

DELETE FROM Employee

WHERE Email = @Email

END

CREATE PROCEDURE DeleteGrandTotal

@OrderID INT

AS

BEGIN

DELETE FROM GrandTotal

WHERE OrderID = @OrderID

END

CREATE PROCEDURE DeleteMenuItem

@ItemID INT

AS

BEGIN

DELETE FROM MenuItem

WHERE ItemID = @ItemID

END

CREATE PROCEDURE DeleteOrder

@OrderID INT

AS

BEGIN

DELETE FROM [Order]

WHERE OrderID = @OrderID

END

CREATE PROCEDURE DeleteOrderInfo

@OrderID INT

AS

BEGIN

DELETE FROM OrderInfo

WHERE OrderID = @OrderID

END

CREATE PROCEDURE DeleteOrderItem

@OrderItemID INT

AS

BEGIN

DELETE FROM OrderItem

WHERE OrderItemID = @OrderItemID

END

CREATE PROCEDURE DeletePayment

@PaymentID INT

AS

BEGIN

DELETE FROM Payment

WHERE PaymentID = @PaymentID

END

CREATE PROCEDURE DeletePromotionCoupon

@CouponCode VARCHAR(25)

AS

BEGIN

DELETE FROM PromotionCoupon

WHERE CouponCode = @CouponCode

END

CREATE PROCEDURE DeleteRestaurant

@RestaurantID INT

AS

BEGIN

DELETE FROM Restaurant

WHERE RestaurantID = @RestaurantID

END

CREATE PROCEDURE DeleteVehicle

@VehicleID INT

AS

BEGIN

DELETE FROM Vehicle

WHERE VehicleID = @VehicleID

END

#### Stored Functions

create function nextRestaurants

(@RestaurantName varchar(50), @RAddr varchar(50))

returns table

as

return

(Select Top(3) Name, Address from Restaurant

where RestaurantID>(Select RestaurantID from Restaurant

where Name=@RestaurantName AND Address=@RAddr))

create function prevRestaurants

(@RestaurantName varchar(50), @RAddr varchar(50))

returns table

as

return

(Select Top(3) Name from Restaurant

where RestaurantID<(Select RestaurantID from Restaurant

where Name=@RestaurantName AND Address=@RAddr))

create function searchRestaurantbyCity

(@City varchar(25))

returns table

as

return

(Select Top(3) Name, Address from Restaurant

where City like @City)

create function nextSearchRestByCity -- PASS C4

(@City varchar(25), @RName varchar(50),@RAdd varchar(50))

returns table

as

return

(Select top(3) Name, Address from Restaurant

where RestaurantID>(Select RestaurantID from Restaurant where Name=@RName AND City like @City AND Address=@RAdd))

create function prevSearchRestByCity -- PASS C4

(@City varchar(25), @RName varchar(50),@RAdd varchar(50))

returns table

as

return

(Select top(3) Name, Address from Restaurant

where RestaurantID<(Select RestaurantID from Restaurant where Name=@RName AND City like @City AND Address=@RAdd))

create function nextCategory

(@CName varchar(25))

returns table

as

return

(Select Top(3) CategoryType from Category

where CategoryID>(Select CategoryID from Category where CategoryType=@CName))

create function prevCategory

(@CName varchar(25))

returns table

as

return

(Select Top(3) CategoryType from Category

where CategoryID<(Select CategoryID from Category where CategoryType=@CName))

create function getRestByCatCity

(@City varchar(25), @CName varchar(25))

returns table

as

return

(Select Top(3) Name, Address from Restaurant r, ResCat rc, Category C

where r.RestaurantID=rc.RestaurantID

AND rc.CategoryID=c.CategoryID

AND r.City=@City AND CategoryType=@CName)

create function getNextResByCatCity

(@City varchar(25),@CName varchar(25), @RName varchar(50), @RAdd varchar(50))

returns table

as

return

(Select Top(3) Name, Address from Restaurant r, ResCat rc, Category C

where r.RestaurantID=rc.RestaurantID

AND rc.CategoryID=c.CategoryID

AND r.RestaurantID>(Select RestaurantID from Restaurant

where Name=@RName AND Address=@RAdd)

AND r.City=@City AND CategoryType=@CName)

create function getPrevResByCatCity

(@City varchar(25),@CName varchar(25), @RName varchar(50), @RAdd varchar(50))

returns table

as

return

(Select Top(3) Name, Address from Restaurant r, ResCat rc, Category C

where r.RestaurantID=rc.RestaurantID

AND rc.CategoryID=c.CategoryID

AND r.RestaurantID<(Select RestaurantID from Restaurant

where Name=@RName AND Address=@RAdd)

AND r.City=@City AND CategoryType=@CName)

create function getUserCoupons

(@uEmail varchar(25))

returns table

as

return

(SELECT pc.\*

FROM PromotionCoupon pc

LEFT JOIN UserCoupon uc ON pc.CouponCode = uc.CouponCode

WHERE (uc.CouponCode IS NULL OR uc.Email <> @uEmail)

AND ExpiryDate>=getdate())

create function getUserDetail

(@Email varchar(25),@Password varchar(25))

returns table

as

return

(Select \* from [User]

where Email=@Email AND Password=@Password)

CREATE FUNCTION loginUser

(@Email VARCHAR(25), @Password VARCHAR(25))

RETURNS VARCHAR(10)

AS

BEGIN

DECLARE @userCount INT, @Result VARCHAR(10);

SELECT @userCount = COUNT(\*)

FROM [User]

WHERE Email = @Email AND Password = @Password;

-- Return 'Success' if a matching user is found, otherwise 'Fail'

IF @userCount > 0

SET @Result = 'Success';

ELSE

SET @Result = 'Fail';

RETURN @Result;

END;

create function getUserAddress

(@Email varchar(25))

returns varchar(50)

as

begin

declare @Address varchar(50)

select @Address=Address from Address a, UserAddress ua

where a.AddressID=ua.AddressID and ua.isDefault=1

return @Address

end

create function getUserAddressNotDefault

(@Email varchar(25))

returns table

as

return

(select Address from Address a, UserAddress ua

where a.AddressID=ua.AddressID and ua.isDefault=0 AND Email=@Email)

create function getRestaurantByName

(@RName varchar(25))

RETURNS TABLE

AS

RETURN

( SELECT TOP(3) Name, Address

FROM Restaurant

WHERE Name LIKE '%'+ @RName + '%')

create function nextRestaurantByName

(@RName varchar(50),@RsName varchar(50),@RAdd varchar(50) )

returns table

as

return

(Select Top(3) Name, Address from Restaurant

where Name LIKE '%'+ @RName + '%' and RestaurantID>(Select RestaurantID from Restaurant

where name=@RsName and Address=@RAdd))

create function prevRestaurantByName

(@RName varchar(50),@RsName varchar(50),@RAdd varchar(50) )

returns table

as

return

(Select Top(3) Name, Address from Restaurant

where Name LIKE '%'+ @RName + '%' and RestaurantID<(Select RestaurantID from Restaurant

where name=@RsName and Address=@RAdd))

create function RestaurantReview

(@RName varchar(50))

returns table

as

return

(Select rr.Rating, rr.Comment, rr.RevDate, u.Name

from ReviewRating rr, [User] u, Restaurant r

where u.Email=rr.Email

AND r.RestaurantID=rr.RestaurantID

AND [r.Name=@RName](mailto:r.Name=@RName))

create function getUserOrderInfo

(@OrderID int)

returns table

as

return

(select (e.FirstName+' '+e.LastName)'RiderName', v.LicensePlate, e.Phone, r.Name as 'RestaurantName',

r.Address'RestaurantAddress', o.OrderID, a.Address'DeliveryAddress', m.Name as 'Menu Item', m.Price'ItemPrice',pc.CouponCode,pc.Discount, oi.Quantity,oi.Subtotal'ItemSubTotal',

gt.TotalAmount-Discount'OrderSubTotal', p.PaymentMethod, (gt.TotalAmount)'Total'

from MenuItem m, OrderItem oi, OrderInfo oin, [Order] o,

Restaurant r, UserAddress ua, Address a, Employee e, Vehicle v, PromotionCoupon pc, GrandTotal gt, Payment p

where

r.RestaurantID=oin.RestaurantID

AND p.OrderID=1

AND ua.Email=oin.Email

AND m.ItemID=oi.MenuItemID

and e.Email=oin.RiderEmail

and e.VehicleID=v.VehicleID

and pc.CouponCode=o.CouponCode

and o.OrderID=@OrderID

and oi.OrderID=@OrderID

and gt.OrderID=@OrderID

AND oin.OrderID=@OrderID

and o.OrderDate<=getdate()

AND oin.AddressID=a.AddressID)

create function showUserOrders

(@Email varchar(25))

returns table

as

return

(select distinct o.OrderID, o.Orderdate, gt.TotalAmount

from [Order] o, GrandTotal gt, [User] u,

OrderInfo oi

where

oi.OrderID=o.OrderID

AND gt.OrderID=o.OrderID

AND oi.Email=@Email)

create function calGrandTotal

(@OrderID INT)

returns decimal(10,2)

as

begin

declare @GrandTotal decimal(10,2);

select @GrandTotal = (sum(oi.Subtotal) - isNULL(pc.Discount, 0))

from OrderItem oi

left join [Order] o on oi.OrderID = o.OrderID

left join PromotionCoupon pc on o.CouponCode = pc.CouponCode

where oi.OrderID = @OrderID

group by pc.Discount

RETURN @GrandTotal;

END

create function checkReviewEligibility

(@Email varchar(25),

@OrderID int)

returns varchar(15)

as

begin

declare @Eligibility varchar(15), @count int;

if exists(

select \*

from

ReviewRating rr, OrderStatus os

where rr.OrderID=@OrderID AND Email=@Email

AND os.OrderID=@OrderID)

begin

set @count=1

end

if not exists (select \*

from

ReviewRating rr, OrderStatus os

where rr.OrderID=@OrderID AND Email=@Email

AND os.OrderID=@OrderID) AND exists (Select \* from OrderStatus os, StatusOfOrder so where OrderID=@OrderID AND os.StatusID=so.StatusID and StatusName='Pending')

begin

set @count=1

end

if not exists (select \*

from

ReviewRating rr, OrderStatus os

where rr.OrderID=@OrderID AND Email=@Email

AND os.OrderID=@OrderID) and not exists (Select \* from OrderStatus os, StatusOfOrder so where OrderID=@OrderID AND os.StatusID=so.StatusID and StatusName='Pending')

begin

set @count=0

end

if @count > 0

set @Eligibility='Not Eligible'

else

set @Eligibility='Eligible'

return @Eligibility

end

create function getRestaurantID

(@RName varchar(50), @RAdd varchar(50))

returns int

as

begin

declare @RID int;

select @RID=RestaurantID from Restaurant

where Name=@RName AND Address=@RAdd

return @RID

end

create function getRestaurantMenu

(@ResName varchar(25),@RAddress varchar(50))

returns table

as

return

(select Top(3) m.Name, m.Description, m.Price

from MenuItem m, Restaurant r, ResItem ri

where ri.RestaurantID=r.RestaurantID

AND r.Name=@ResName

AND r.Address=@RAddress

AND m.ItemID=ri.ItemID)

create function getnextRestaurantMenu

(@MName varchar(50), @ResName varchar(50),@RAddress varchar(50))

returns table

as

return

(select Top(3) m.Name, m.Description, m.Price

from MenuItem m, Restaurant r, ResItem ri

where ri.RestaurantID=r.RestaurantID

AND r.Name=@ResName

AND r.Address=@RAddress

AND ri.ItemID=m.ItemID

AND m.ItemID>(Select ItemID from MenuItem where Name=@MName))

create function getprevRestaurantMenu

(@MName varchar(50), @ResName varchar(50),@RAddress varchar(50))

returns table

as

return

(select Top(3) m.Name, m.Description, m.Price

from MenuItem m, Restaurant r, ResItem ri

where ri.RestaurantID=r.RestaurantID

AND r.Name=@ResName

AND r.Address=@RAddress

AND ri.ItemID=m.ItemID

AND m.ItemID<(Select ItemID from MenuItem where Name=@MName))

create function nextRestaurantItem

(@MName varchar(25), @ResName varchar(25),@RAddress varchar(50))

returns table

as

return

(select Top(3) m.Name, m.Description, m.Price

from MenuItem m, Restaurant r, ResItem ri

where ri.RestaurantID=r.RestaurantID

AND r.[Name]=@ResName

AND m.ItemID=ri.ItemID

AND r.[Address]=@RAddress

AND m.ItemID>(Select ItemID from MenuItem where Name=@MName))

select \* from prevRestaurantItem('Spicy Pakistani','Kababjees Fried Chicken','Ground Floor Komal Heaven Apartment Jauhar')

create function prevRestaurantItem

(@MName varchar(25), @ResName varchar(25),@RAddress varchar(50))

returns table

as

return

(select Top(3) m.Name, m.Description, m.Price

from MenuItem m, Restaurant r, ResItem ri

where ri.RestaurantID=r.RestaurantID

AND r.[Name]=@ResName

AND m.ItemID=ri.ItemID

AND r.[Address]=@RAddress

AND m.ItemID<(Select ItemID from MenuItem where Name=@MName))

create function searchItem

(@ResName varchar(25), @RAddress varchar(50),@Desc varchar(25))

returns table

as

return

(Select distinct Top(3) m.Name, m.Description, m.Price

from MenuItem m, Restaurant r, ResItem ri

where ri.RestaurantID=r.RestaurantID

AND r.[Name]=@ResName

AND m.ItemID=ri.ItemID

AND r.[Address]=@RAddress

AND (m.Description like '%'+@Desc+'%'

OR m.Description like (@Desc+'%') OR m.Description like ('%'+@Desc)

OR m.Description like @Desc))

create function getRestaurantRating

(@ResName varchar(25), @RAddress varchar(50))

returns decimal(2,1)

as

begin

declare @Rating decimal(2,1)

Select @Rating=avg(rr.Rating) from Restaurant r, ReviewRating rr

where r.RestaurantID=(Select RestaurantID from Restaurant where Name=@ResName AND Address=@RAddress)

RETURN ISNULL(@Rating, 0.0); -- Return the computed average rating, or 0.0 if no rating found

End

create FUNCTION loginRider

(@Email VARCHAR(25), @Password VARCHAR(25))

RETURNS VARCHAR(10)

AS

BEGIN

DECLARE @userCount INT, @Result VARCHAR(10);

SELECT @userCount = COUNT(\*)

FROM Employee

WHERE Email = @Email AND Password = @Password AND Role='Rider';

-- Return 'Success' if a matching user is found, otherwise 'Fail'

IF @userCount > 0

SET @Result = 'Success';

ELSE

SET @Result = 'Fail';

RETURN @Result;

END;

create function getRiderDetail

(@Email varchar(25),@Password varchar(25))

returns table

as

return

(Select \* from Employee

where Email=@Email AND Password=@Password

AND Role='Rider')

create function riderOrders

(@RiderEmail varchar(25))

returns table

as

return

SELECT oi.OrderID, gt.TotalAmount, o.OrderDate, StatusName

FROM Employee e

JOIN OrderInfo oi ON e.Email = oi.RiderEmail JOIN OrderStatus os ON oi.OrderID = os.OrderID

JOIN StatusOfOrder so ON so.StatusID = os.StatusID JOIN GrandTotal gt on oi.OrderID=gt.OrderID

JOIN [Order] o on oi.OrderID=o.OrderID

WHERE e.Role = 'Rider'

AND RiderEmail=@RiderEmail

create function getRiderOrderInfo

(@OrderID int)

returns table

as

return

(select (u.Name)'CustomerName', u.PhoneNo,a.Address'DeliveryAddress', r.Name as 'RestaurantName',

r.Address'RestaurantAddress', o.OrderID, m.Name as 'Menu Item', m.Price'ItemPrice',pc.CouponCode,pc.Discount, oi.Quantity,oi.Subtotal'ItemSubTotal',

gt.TotalAmount-Discount'OrderSubTotal', p.PaymentMethod, (gt.TotalAmount)'Total'

from MenuItem m, OrderItem oi, OrderInfo oin, [Order] o,

Restaurant r, UserAddress ua, Address a, Employee e, Vehicle v, PromotionCoupon pc, GrandTotal gt, Payment p,

[User] u

where

r.RestaurantID=oin.RestaurantID

AND p.OrderID=@OrderID

AND ua.Email=oin.Email

AND m.ItemID=oi.MenuItemID

and e.Email=oin.RiderEmail

and e.VehicleID=v.VehicleID

and pc.CouponCode=o.CouponCode

and o.OrderID=@OrderID

and oi.OrderID=@OrderID

and gt.OrderID=@OrderID

AND oin.OrderID=@OrderID

AND oin.Email=u.Email

AND oin.AddressID=a.AddressID

and o.OrderDate<=getdate())

create function getLastOrder()

returns int

as

begin

declare @OrderID int;

select top(1) @OrderID=OrderID from [Order]

order by OrderID desc

return @OrderID

end

#### Triggers

create trigger trig\_pend\_order

on OrderInfo

for insert,update

as

begin

declare @count int;

select @count=count(Email) from OrderInfo oi, StatusOfOrder so, OrderStatus os

where oi.OrderID=os.OrderID AND so.StatusID=os.StatusID

AND StatusName='Pending'

group by Email

if (@count>1)

begin

Print 'Error! You cant place another order while already having one being delivered'

rollback

end

end

## Conclusion

### Evaluation of the project's success in meeting its objectives

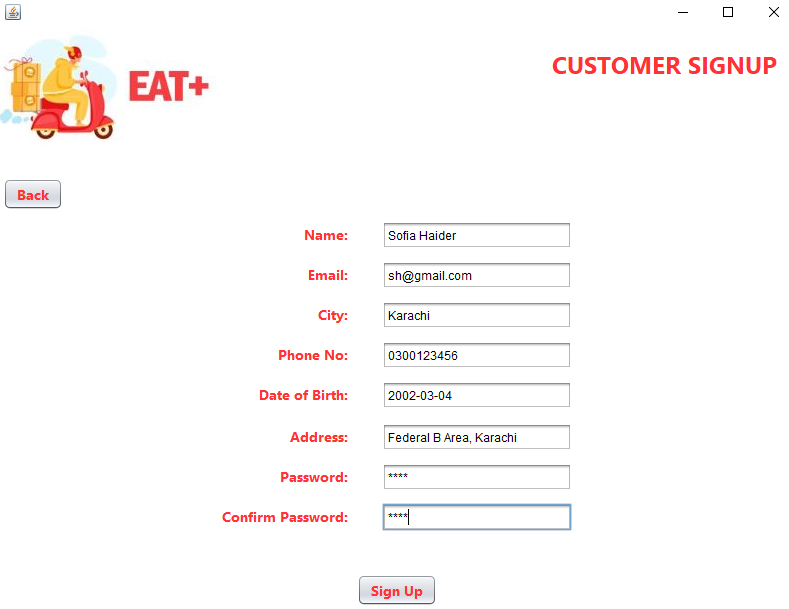
The success of the Food Delivery Database System can be evaluated based on the extent to which it achieves its stated objectives. Here's the success of our project:

* **Efficient Order Management:** The project has successfully achieved its objective of efficient order management. The database system consistently records order details accurately, facilitating smooth processing. Also, manages orders effectively while minimizing errors.
* **Menu and Restaurant Information:** The system has met the objective of providing a comprehensive repository for menu details and restaurant information. The database effectively stores and updates menu items, prices, categories, operating hours, and special coupons. Restaurants information can be managed easily, and the system allows for prompt updates as well.
* **User-Friendly Interface:** The project has successfully delivered on its objective of creating a user-friendly interface making the order placement process seamless and providing improved efficiency for delivery staff.
* **Scalability:** The system demonstrates successful scalability, meeting the goal of accommodating the growth of the food delivery service. It has effectively handled an increasing number of restaurants, menu items, and customer orders without compromising performance.

In summary, the Food Delivery Database System has achieved success in meeting its objectives. Efficient order management, comprehensive menu and restaurant information, a user-friendly interface, and scalability have been successfully implemented.

### Screenshots of major modules’ outputs





A screenshot of a login form

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A search box with red text

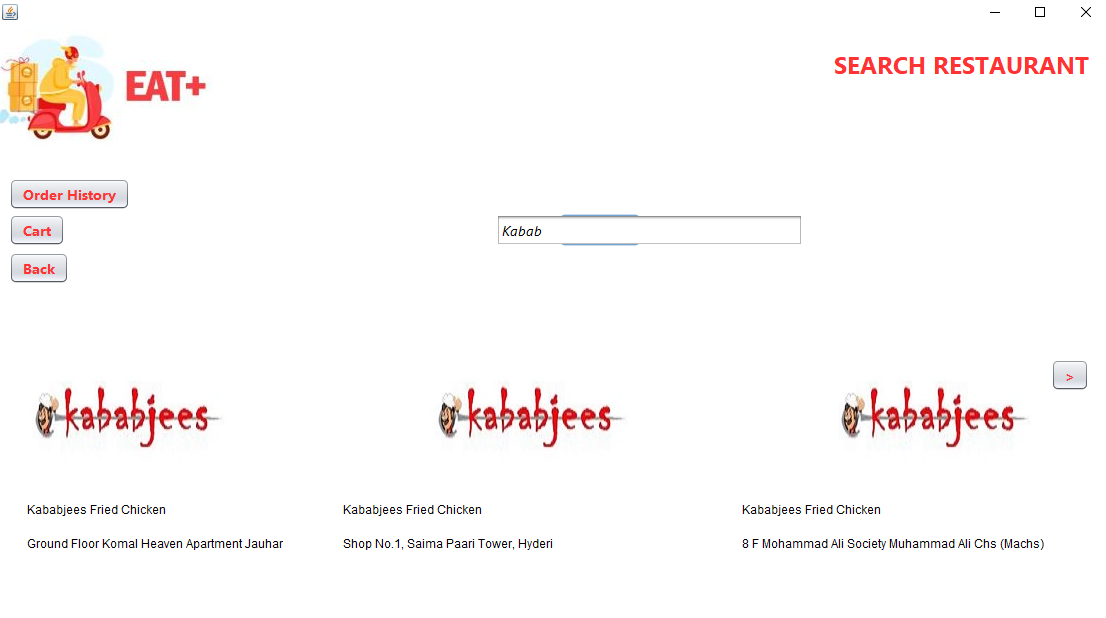
Description automatically generated

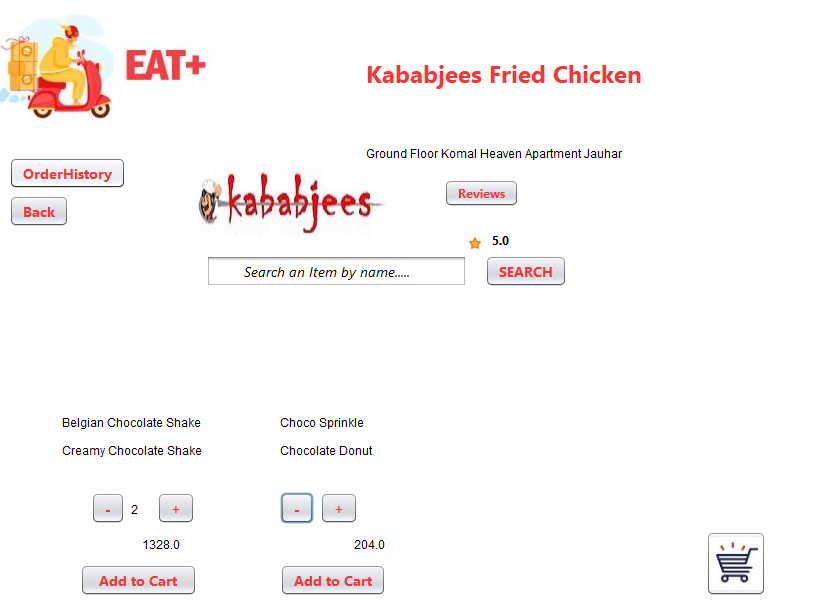
A screenshot of a menu

Description automatically generated

A screenshot of a search engine

Description automatically generated





A screenshot of a menu

Description automatically generated

A screenshot of a food order

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer message

Description automatically generatedA screenshot of a menu

Description automatically generatedA screenshot of a coupon

Description automatically generated

A screenshot of a food order

Description automatically generated

A screenshot of a computer

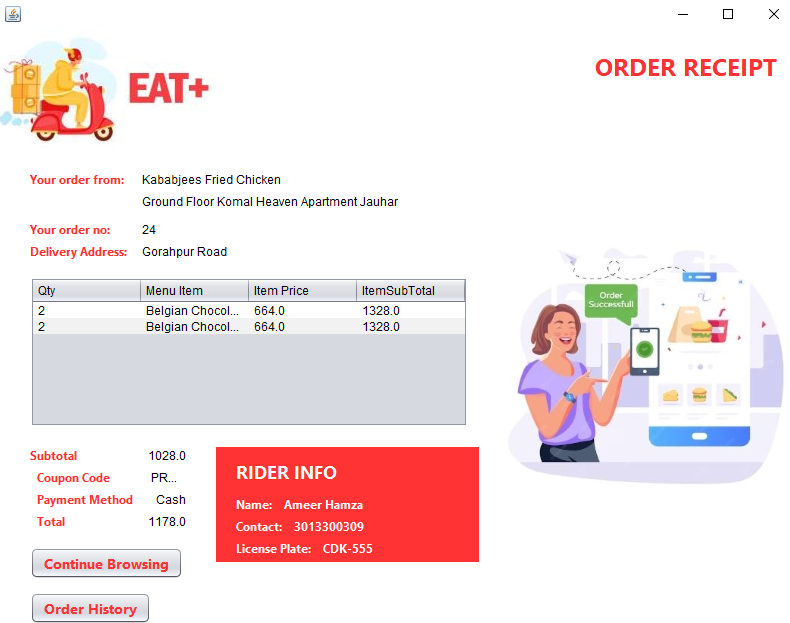
Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a login screen

Description automatically generated



A screenshot of a menu

Description automatically generated

A blue and black text

Description automatically generated

A table of email address

Description automatically generated

A table with numbers and letters

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

Description automatically generated