Food Delivery Platform Database SQL Statement

`

-- Table for Individual User

CREATE TABLE IndividualUser (

UserID INTEGER PRIMARY KEY,

Name VARCHAR(100) NOT NULL,

Email VARCHAR(100) UNIQUE NOT NULL,

Password VARCHAR(100) NOT NULL,

Phone VARCHAR(15),

UserType TEXT CHECK(UserType IN ('Customer', 'Delivery Person')) NOT NULL,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

UpdatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Table for Address

CREATE TABLE Address (

AddressID INTEGER PRIMARY KEY,

UserID INTEGER,

Street VARCHAR(100),

City VARCHAR(50),

State VARCHAR(50),

ZipCode VARCHAR(10),

Latitude DECIMAL(10, 8),

Longitude DECIMAL(11, 8),

FOREIGN KEY (UserID) REFERENCES IndividualUser(UserID)

);

-- Table for Restaurant

CREATE TABLE Restaurant (

RestaurantID INTEGER PRIMARY KEY,

Name VARCHAR(100) NOT NULL,

Description TEXT,

ContactNumber VARCHAR(15),

Email VARCHAR(100) UNIQUE,

LocationID INTEGER,

Rating DECIMAL(2,1),

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

UpdatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (LocationID) REFERENCES Address(AddressID)

);

-- Table for Menu

CREATE TABLE Menu (

MenuID INTEGER PRIMARY KEY,

RestaurantID INTEGER,

Name VARCHAR(100),

Description TEXT,

FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)

);

-- Table for MenuItem

CREATE TABLE MenuItem (

MenuItemID INTEGER PRIMARY KEY,

MenuID INTEGER,

Name VARCHAR(100) NOT NULL,

Description TEXT,

Price DECIMAL(10, 2),

AvailabilityStatus BOOLEAN DEFAULT TRUE,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

UpdatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (MenuID) REFERENCES Menu(MenuID)

);

-- Table for Order

CREATE TABLE `Order` (

OrderID INTEGER PRIMARY KEY,

UserID INTEGER,

RestaurantID INTEGER,

OrderStatus TEXT CHECK(OrderStatus IN ('Placed', 'In-Transit', 'Delivered', 'Canceled')) NOT NULL,

OrderDate TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

DeliveryDate TIMESTAMP,

TotalAmount DECIMAL(10, 2),

PaymentStatus TEXT CHECK(PaymentStatus IN ('Paid', 'Pending')) NOT NULL,

RatingID INTEGER,

FOREIGN KEY (UserID) REFERENCES IndividualUser(UserID),

FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID),

FOREIGN KEY (RatingID) REFERENCES Rating(RatingID)

);

-- Table for OrderItem

CREATE TABLE OrderItem (

OrderItemID INTEGER PRIMARY KEY,

OrderID INTEGER,

MenuItemID INTEGER,

Quantity INTEGER NOT NULL,

ItemPrice DECIMAL(10, 2),

TotalPrice DECIMAL(10, 2),

FOREIGN KEY (OrderID) REFERENCES `Order`(OrderID),

FOREIGN KEY (MenuItemID) REFERENCES MenuItem(MenuItemID)

);

-- Table for Delivery

CREATE TABLE Delivery (

DeliveryID INTEGER PRIMARY KEY,

OrderID INTEGER,

DeliveryPersonID INTEGER,

EstimatedDeliveryTime TIMESTAMP,

ActualDeliveryTime TIMESTAMP,

DeliveryStatus TEXT CHECK(DeliveryStatus IN ('Pending', 'Out for Delivery', 'Delivered')) NOT NULL,

FOREIGN KEY (OrderID) REFERENCES `Order`(OrderID),

FOREIGN KEY (DeliveryPersonID) REFERENCES IndividualUser(UserID)

);

-- Table for DeliveryPerson (as a subset of Individual User)

CREATE TABLE DeliveryPerson (

DeliveryPersonID INTEGER PRIMARY KEY,

DeliveryStatus TEXT CHECK(DeliveryStatus IN ('Available', 'Busy')) DEFAULT 'Available',

RealTimeLocation VARCHAR(255),

FOREIGN KEY (DeliveryPersonID) REFERENCES IndividualUser(UserID)

);

-- Table for Payment

CREATE TABLE Payment (

PaymentID INTEGER PRIMARY KEY,

OrderID INTEGER,

OriginalAmount DECIMAL(10, 2),

PromotionID INTEGER,

FinalAmount DECIMAL(10, 2),

PaymentDate TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

PaymentMethod TEXT CHECK(PaymentMethod IN ('Credit Card', 'PayPal')) NOT NULL,

PaymentStatus TEXT CHECK(PaymentStatus IN ('Paid', 'Pending')) NOT NULL,

FOREIGN KEY (OrderID) REFERENCES `Order`(OrderID),

FOREIGN KEY (PromotionID) REFERENCES Promotion(PromotionID)

);

-- Table for Rating

CREATE TABLE Rating (

RatingID INTEGER PRIMARY KEY,

UserID INTEGER,

RestaurantID INTEGER,

OrderID INTEGER,

RatingScore INTEGER CHECK (RatingScore BETWEEN 1 AND 5),

Comment TEXT,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (UserID) REFERENCES IndividualUser(UserID),

FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID),

FOREIGN KEY (OrderID) REFERENCES `Order`(OrderID)

);

-- Table for Promotion

CREATE TABLE Promotion (

PromotionID INTEGER PRIMARY KEY,

RestaurantID INTEGER,

Title VARCHAR(100),

Description TEXT,

DiscountPercentage DECIMAL(5, 2),

StartDate DATE,

EndDate DATE,

FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)

);

-- Table for CustomerFeedback

CREATE TABLE CustomerFeedback (

FeedbackID INTEGER PRIMARY KEY,

UserID INTEGER,

FeedbackType TEXT CHECK(FeedbackType IN ('Delivery', 'Order Quality', 'App Performance')) NOT NULL,

Comment TEXT,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (UserID) REFERENCES IndividualUser(UserID)

);

-- Table for Favorites

CREATE TABLE Favorites (

FavoriteID INTEGER PRIMARY KEY,

UserID INTEGER,

RestaurantID INTEGER,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (UserID) REFERENCES IndividualUser(UserID),

FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)

);

-- Table for Notification

CREATE TABLE Notification (

NotificationID INTEGER PRIMARY KEY,

UserID INTEGER,

Message TEXT,

Status TEXT CHECK(Status IN ('Read', 'Unread')) DEFAULT 'Unread',

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (UserID) REFERENCES IndividualUser(UserID)

);

-- Table for Discount

CREATE TABLE Discount (

DiscountID INTEGER PRIMARY KEY,

Code VARCHAR(50) UNIQUE,

Description TEXT,

DiscountAmount DECIMAL(10, 2),

ExpirationDate DATE,

MinOrderAmount DECIMAL(10, 2),

MaxUsage INTEGER

);

-- Table for OrderDiscount

CREATE TABLE OrderDiscount (

OrderDiscountID INTEGER PRIMARY KEY,

OrderID INTEGER,

DiscountID INTEGER,

DiscountAmount DECIMAL(10, 2),

FOREIGN KEY (OrderID) REFERENCES