

Group No 13

Name:

Dhruv Patel(202301095)

Prasham Shah(202301252)

Manthan Gajera(202301488)

DDL Script

```
-- Create Users table
CREATE TABLE Users (
    UserID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Email VARCHAR(100) UNIQUE NOT NULL,
    Password VARCHAR(255) NOT NULL,
    PhoneNumber VARCHAR(20),
    Address TEXT,
    UserType VARCHAR(20) NOT NULL CHECK (UserType IN ('Customer',
'RestaurantOwner', 'DeliveryAgent'))
);

-- Create Customer table
CREATE TABLE Customer (
    CustomerID INT PRIMARY KEY,
    UserID INT UNIQUE NOT NULL,
    RewardCoins INT DEFAULT 0,
    PaymentPreferences VARCHAR(100),
    FOREIGN KEY (UserID) REFERENCES Users(UserID) ON DELETE CASCADE
);

-- Create RestaurantOwner table
CREATE TABLE RestaurantOwner (
    RestaurantOwnerID INT PRIMARY KEY,
    UserID INT UNIQUE NOT NULL,
    LicenseNumber VARCHAR(50) NOT NULL,
    FOREIGN KEY (UserID) REFERENCES Users(UserID) ON DELETE CASCADE
);

-- Create DeliveryAgent table
CREATE TABLE DeliveryAgent (
    DeliveryAgentID INT PRIMARY KEY,
    UserID INT UNIQUE NOT NULL,
    Earnings DECIMAL(10, 2) DEFAULT 0.00,
    FOREIGN KEY (UserID) REFERENCES Users(UserID) ON DELETE CASCADE
);

-- Create Restaurant table
CREATE TABLE Restaurant (
    RestaurantID INT PRIMARY KEY,
```

```

    Name VARCHAR(100) NOT NULL,
    Address TEXT NOT NULL,
    ContactInfo VARCHAR(100),
    CuisineType VARCHAR(50),
    AverageRating DECIMAL(3, 2) DEFAULT 0.00,
    LicenseNumber VARCHAR(50) NOT NULL,
    RestaurantOwnerID INT NOT NULL,
    FOREIGN KEY (RestaurantOwnerID) REFERENCES
RestaurantOwner(RestaurantOwnerID) ON DELETE CASCADE
);

-- Create Menu table
CREATE TABLE Menu (
    MenuID INT PRIMARY KEY,
    RestaurantID INT NOT NULL,
    ItemName VARCHAR(100) NOT NULL,
    Price DECIMAL(10, 2) NOT NULL,
    Availability BOOLEAN DEFAULT TRUE,
    FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)
ON DELETE CASCADE
);

-- Create Orders table
CREATE TABLE Orders (
    OrderID INT PRIMARY KEY,
    CustomerID INT NOT NULL,
    RestaurantID INT NOT NULL,
    DeliveryAgentID INT,
    OrderDate TIMESTAMP NOT NULL,
    DeliveryAddress TEXT NOT NULL,
    Status VARCHAR(20) NOT NULL CHECK (Status IN ('Placed',
'Preparing', 'Ready', 'On the way', 'Delivered', 'Cancelled')),
    TotalAmount DECIMAL(10, 2) NOT NULL,
    RewardCoinsEarned INT DEFAULT 0,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID) ON
DELETE CASCADE,
    FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)
ON DELETE CASCADE,
    FOREIGN KEY (DeliveryAgentID) REFERENCES
DeliveryAgent(DeliveryAgentID) ON DELETE SET NULL
);

-- Create OrderDetails table
CREATE TABLE OrderDetails (
    OrderDetailsID INT PRIMARY KEY,
    OrderID INT NOT NULL,
    MenuID INT NOT NULL,
    Quantity INT NOT NULL,
    Price DECIMAL(10, 2) NOT NULL,

```

```
        FOREIGN KEY (OrderID) REFERENCES Orders(OrderID) ON DELETE
CASCADE,
        FOREIGN KEY (MenuID) REFERENCES Menu(MenuID) ON DELETE CASCADE
);
```

-- Create Payment table

```
CREATE TABLE Payment (
    TransactionID INT PRIMARY KEY,
    OrderID INT NOT NULL,
    PaymentMethod VARCHAR(50) NOT NULL,
    TotalAmount DECIMAL(10, 2) NOT NULL,
    RewardCoinsUsed INT DEFAULT 0,
    AmountPaid DECIMAL(10, 2) NOT NULL,
    PlatformCommission DECIMAL(10, 2) NOT NULL,
    RestaurantEarnings DECIMAL(10, 2) NOT NULL,
    PaymentStatus VARCHAR(20) NOT NULL CHECK (PaymentStatus IN
('Pending', 'Completed', 'Failed', 'Refunded')),
    PaymentDate TIMESTAMP NOT NULL,
    FOREIGN KEY (OrderID) REFERENCES Orders(OrderID) ON DELETE
CASCADE
);
```

-- Create RewardCoinUsage table

```
CREATE TABLE RewardCoinUsage (
    RewardCoinUsageID INT PRIMARY KEY,
    TransactionID INT NOT NULL,
    CustomerID INT NOT NULL,
    CoinsUsed INT NOT NULL,
    CoinsValue DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (TransactionID) REFERENCES Payment(TransactionID) ON
DELETE CASCADE,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID) ON
DELETE CASCADE
);
```

-- Create PayoutHistory table

```
CREATE TABLE PayoutHistory (
    PayoutID INT PRIMARY KEY,
    DeliveryAgentID INT NOT NULL,
    OrderID INT NOT NULL,
    BaseFee DECIMAL(10, 2) NOT NULL,
    DistanceFee DECIMAL(10, 2) NOT NULL,
    TotalPayout DECIMAL(10, 2) NOT NULL,
    PaymentStatus VARCHAR(20) NOT NULL CHECK (PaymentStatus IN
('Pending', 'Completed', 'Failed')),
    PayoutDate TIMESTAMP NOT NULL,
    FOREIGN KEY (DeliveryAgentID) REFERENCES
DeliveryAgent(DeliveryAgentID) ON DELETE CASCADE,
    FOREIGN KEY (OrderID) REFERENCES Orders(OrderID) ON DELETE
```

```

CASCADE
);

-- Create FinancialReport table
CREATE TABLE FinancialReport (
    ReportID INT PRIMARY KEY,
    RestaurantID INT,
    DeliveryAgentID INT,
    CommissionEarned DECIMAL(10, 2) NOT NULL,
    PayoutAmount DECIMAL(10, 2) NOT NULL,
    ReportDate DATE NOT NULL,
    FOREIGN KEY (RestaurantID) REFERENCES Restaurant(RestaurantID)
ON DELETE SET NULL,
    FOREIGN KEY (DeliveryAgentID) REFERENCES
DeliveryAgent(DeliveryAgentID) ON DELETE SET NULL
);

-- Create Review table
CREATE TABLE Review (
    ReviewID INT PRIMARY KEY,
    OrderID INT NOT NULL,
    Rating DECIMAL(2, 1) NOT NULL CHECK (Rating BETWEEN 0 AND 5),
    Comment TEXT,
    FOREIGN KEY (OrderID) REFERENCES Orders(OrderID) ON DELETE
CASCADE
);

```