CSCI 4710-5710-01: DATABASES Course

- FINAL ASSIGNMENT -

MEMBER PARTICIPATION OF THE PROJECT:

Banner ID	Member Name
001368402	Xuan Mai Tran
001313112	Gayatri Jawharkar
001313628	Sarvesh Sonawane

PART 3: Implement the database and populate the tables with data.

```
-- CREATE DATABASE ------------------------------------------------------------
CREATE DATABASE Bus_System;
USE Bus_System;



-- ---------------------------------------------------------------------------
-- CREATE TABLES --------------------------------------------------------------

-- Customers table
CREATE TABLE CUSTOMER (
   Customer_Id INT PRIMARY KEY,
   SSN VARCHAR(20) NOT NULL,
   Customer_Name VARCHAR(100) NOT NULL,
   Address VARCHAR(200),
   Phone_Number VARCHAR(20),
   Email VARCHAR(100)
);

-- Buses table
```

```
CREATE TABLE  BUS (
    Bus_Id INT PRIMARY KEY,
    Bus_Plate VARCHAR(20) NOT NULL,
    License_Plate VARCHAR(20) NOT NULL,
    Expiry_Date DATE NOT NULL,
    Capacity INT NOT NULL
);

-- Drivers table
CREATE TABLE DRIVER (
    Driver_Id INT PRIMARY KEY,
    SSN VARCHAR(20) NOT NULL,
    Driver_Name VARCHAR(100) NOT NULL,
    Address VARCHAR(200),
    Phone_Number VARCHAR(20),
    Driver_License VARCHAR(20),
    License_Number VARCHAR(20),
    Expiry_Date DATE
);

-- Stops table
CREATE TABLE  STOP (
    Stop_Id INT PRIMARY KEY,
    Stop_Name VARCHAR(100) NOT NULL,
    Stop_Address VARCHAR(200) NOT NULL,
    GPS_Location VARCHAR(100) NOT NULL
);

-- Stations table
CREATE TABLE STATION (
    Station_Id INT PRIMARY KEY,
    Station_Name VARCHAR(100) NOT NULL,
    Address VARCHAR(200),
    Phone_Number VARCHAR(20)
    );

-- Routes table
CREATE TABLE ROUTE (
    Route_Id INT PRIMARY KEY,
    Departure_Location VARCHAR(100) NOT NULL,
    Arrival_Location VARCHAR(100) NOT NULL,
    Start_Date DATE,
    End_Date DATE,
    Start_Time TIME,
    End_Time TIME,
    Customer_Id INT,
    Driver_Id INT,
    Bus_Id INT,
    Station_Id INT,
    Stop_Id INT,
            FOREIGN KEY (Customer_Id) REFERENCES CUSTOMER(Customer_Id),
            FOREIGN KEY (Driver_Id) REFERENCES DRIVER(Driver_Id),
            FOREIGN KEY (Bus_Id) REFERENCES BUS(Bus_Id),
            FOREIGN KEY (Station_Id) REFERENCES STATION(Station_Id),
    FOREIGN KEY (Stop_Id) REFERENCES STOP(Stop_Id)
);


-- Payment table - Supertype class
CREATE TABLE PAYMENT (
    Payment_Id INT PRIMARY KEY,
    Payment_Type ENUM('S', 'B', 'T') NOT NULL
);

-- Monthly subscriptions table - Subtype of Payment
CREATE TABLE MonthlySubscription (
    Payment_Id INT PRIMARY KEY,
    Start_Date DATE NOT NULL,
    Expiry_Date DATE NOT NULL,
    FOREIGN KEY (Payment_Id) REFERENCES PAYMENT(Payment_Id)
);

-- Balance table - Subtype of Payment
CREATE TABLE Balance (
    Payment_Id INT PRIMARY KEY,
    Balance_Amount DECIMAL(10, 2) NOT NULL,
    Min_Balance_Required DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (Payment_Id) REFERENCES PAYMENT(Payment_Id)
);

-- Ticket table - Subtype of Payment
CREATE TABLE Ticket (
    Payment_Id INT PRIMARY KEY,
    Ticket_Type VARCHAR(255) NOT NULL,
    Ticket_Price DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (Payment_Id) REFERENCES PAYMENT(Payment_Id)
);
```

```
-- Ticket boxes table
CREATE TABLE  TICKET_BOX (
    Ticket_Box_Id INT PRIMARY KEY,
            Location VARCHAR(200),
    Station_Id INT,
            FOREIGN KEY (Station_Id) REFERENCES STATION(Station_Id)
);

-- Purchases table
CREATE TABLE PURCHASE (
    Purchase_Id INT PRIMARY KEY,
    Purchase_Amount DECIMAL(10, 2) NOT NULL,
    Purchase_Date datetime,
    Customer_Id INT,
    Payment_Id INT,
    Station_Id INT,
    Ticket_Box_Id INT,
    Route_Id INT,
    FOREIGN KEY (Customer_Id) REFERENCES CUSTOMER(Customer_Id),
    FOREIGN KEY (Payment_Id) REFERENCES PAYMENT(Payment_Id),
    FOREIGN KEY (Station_Id) REFERENCES STATION(Station_Id),
    FOREIGN KEY (Ticket_Box_Id) REFERENCES TICKET_BOX(Ticket_Box_Id),
            FOREIGN KEY (Route_Id) REFERENCES ROUTE(Route_Id)
);



-- -----------------------------------------------------------------------------------------
-- INSERT SAMPLE DATA --------------------------------------------------------------------

-- Insert sample data into CUSTOMER table
INSERT INTO CUSTOMER (Customer_Id, SSN, Customer_Name, Address, Phone_Number,
Email)
VALUES
    (1, '123456789', 'John Doe', '123 Main St, New York, NY 10001', '1234567890',
'john.doe@email.com'),
    (2, '987654321', 'Jane Smith', '456 Oak Ave, Los Angeles, CA 90001', '9876543210',
'jane.smith@email.com'),
    (3, '456789012', 'Michael Johnson', '789 Elm St, Chicago, IL 60601', '3125551234',
'michael.johnson@email.com'),
    (4, '789012345', 'Emily Davis', '321 Pine St, Seattle, WA 98101', '2065559876',
'emily.davis@email.com'),
    (5, '111222333', 'Alice Johnson', '456 Elm St, New York, NY 10001', '212-555-1111',
'alice.johnson@email.com'),
    (6, '222333444', 'Bob Smith', '789 Oak Ave, Los Angeles, CA 90001', '213-555-2222',
'bob.smith@email.com'),
    (7, '333444555', 'Eva Martinez', '321 Pine St, Seattle, WA 98101', '206-555-3333',
'eva.martinez@email.com');


-- Insert sample data into BUS table
INSERT INTO BUS (Bus_Id, Bus_Plate, License_Plate, Expiry_Date, Capacity)
VALUES
    (1, 'ABC123', 'LP456789', '2023-05-31', 50),
    (2, 'DEF456', 'LP012345', '2024-06-30', 40),
    (3, 'GHI789', 'LP678901', '2022-12-31', 60),
    (4, 'JKL012', 'LP890123', '2023-07-31', 55),
    (5, 'MNO345', 'LP234567', '2024-08-31', 45),
    (6, 'PQR678', 'LP456789', '2022-10-31', 50);


-- Insert sample data into DRIVER table
INSERT INTO DRIVER (Driver_Id, SSN, Driver_Name, Address, Phone_Number,
Driver_License, License_Number, Expiry_Date)
VALUES
    (1, '123456789', 'Michael Johnson', '456 Elm St, New York, NY 10001', '212-555-6789',
'DL123', 'LN456789', '2025-12-31'),
    (2, '987654321', 'Emily Davis', '789 Oak Ave, Los Angeles, CA 90001', '213-555-2345',
'DL456', 'LN987654', '2024-04-30'),
    (3, '456789012', 'Robert Wilson', '321 Pine St, Chicago, IL 60601', '312-555-7890', 'DL789',
'LN012345', '2026-03-15'),
    (4, '444555666', 'Daniel Brown', '123 1st St, Portland, OR 97201', '503-555-7890', 'DL999',
'LN111222', '2025-06-30'),
    (5, '555666777', 'Grace Lee', '456 2nd Ave, Seattle, WA 98101', '206-555-3456', 'DL888',
'LN333444', '2024-09-15'),
    (6, '666777888', 'Oliver Taylor', '789 3rd St, Chicago, IL 60601', '312-555-2345', 'DL777',
'LN555666', '2023-12-31');


-- Insert sample data into STATION table
INSERT INTO STATION (Station_Id, Station_Name, Address, Phone_Number)
VALUES
    (1, 'Grand Central Station', '89 E 42nd St, New York, NY 10017', '212-555-1234'),
    (2, 'Union Station', '800 N Alameda St, Los Angeles, CA 90012', '213-555-5678'),
    (3, 'Ogilvie Transportation Center', '500 W Madison St, Chicago, IL 60661', '312-555-9012'),
    (4, 'Magnolia Station', '1600 N Magnolia St, Los Angeles, CA 91002', '213-578-7890'),
    (5, 'Oxford Center', '300 W Oxford St, Chicago, IL 61066', '312-789-1245'),
    (6, 'Detroit Center', '1002 W Luther King St, Detroit, MI 78009', '512-980-0909');


-- Insert sample data into STOP table
INSERT INTO STOP (Stop_Id ,Stop_Name, Stop_Address, GPS_Location)
VALUES
    (1,'Main Street Stop', '123 Main St, New York, NY 10001', '40.7128, -74.0060'),
    (2,'Central Park Stop', '100 Central Park West, New York, NY 10024', '40.7812, -73.9665'),
    (3,'Hollywood Stop', '6925 Hollywood Blvd, Los Angeles, CA 90028', '34.1017, -118.3406'),
    (4,'Union Station Stop', '800 N Alameda St, Los Angeles, CA 90012', '34.0565, -118.2380'),
    (5,'Times Square Stop', 'Times Square, New York, NY 10036', '40.7580, -73.9855'),
    (6,'Chinatown Stop', 'Chinatown, Los Angeles, CA 90012', '34.0639, -118.2386'),
    (7,'Golden Gate Bridge Stop', 'Golden Gate Bridge, San Francisco, CA 94129', '37.8199, -122.4783');


-- Insert sample data into ROUTE table
INSERT INTO ROUTE (Route_Id, Departure_Location, Arrival_Location, Start_Date, End_Date, Start_Time, End_Time, Customer_Id, Driver_Id, Bus_Id, Station_Id, Stop_Id)
VALUES
    (1, 'New York', 'Boston', '2023-06-01', '2023-12-31', '08:00:00', '16:00:00', 2, 3, 1, 6, 4),
    (2, 'Los Angeles', 'San Francisco', '2023-06-01', '2023-12-31', '09:00:00', '15:00:00', 1, 2, 3, 5, 7),
    (3, 'Chicago', 'Detroit', '2023-06-01', '2023-12-31', '07:00:00', '14:00:00', 5, 6, 2, 1, 3),
    (4, 'New York', 'Philadelphia', '2023-06-01', '2023-12-31', '10:00:00', '14:00:00', 4, 5, 6, 2, 1),
    (5, 'Los Angeles', 'Las Vegas', '2023-06-01', '2023-12-31', '11:00:00', '15:00:00', 7, 4, 5, 3, 2),
    (6, 'San Francisco', 'Portland', '2023-06-01', '2023-12-31', '09:00:00', '18:00:00', 6, 1, 4, 4, 5);


-- Insert into Payment table
INSERT INTO Payment (Payment_Id, Payment_Type)
VALUES (1, 'S'),
       (2, 'B'),
       (3, 'T'),
       (4, 'S'),
       (5, 'B'),
       (6, 'T');

-- Insert into MonthlySubscription table
INSERT INTO MonthlySubscription (Payment_Id, Start_Date, Expiry_Date)
VALUES (1, '2024-05-05', '2024-06-05'),
       (4, '2024-05-08', '2024-06-08');
```

```
-- Insert into Balance table
INSERT INTO Balance (Payment_Id, Balance_Amount, Min_Balance_Required)
VALUES (2, 20.00, 5.00),
       (5, 25.00, 10.00);

-- Insert into Ticket table
INSERT INTO Ticket (Payment_Id, Ticket_Type, Ticket_Price)
VALUES (3, 'state', 20.00),
       (6, 'interstate', 45.00);


-- Insert sample data into TICKET_BOX table
INSERT INTO TICKET_BOX (Ticket_Box_Id, Location, Station_Id)
VALUES
           (1, 'Food Center', 3),
    (2, 'Front Door', 2),
    (3, 'Back Door', 4),
    (4, 'Bar Center', 5),
    (5, 'Vendor Booth', 1),
    (6, 'Food Booth', 6),
    (7, 'Front Door', 1);


-- Insert sample data into PURCHASE table
INSERT INTO PURCHASE (Purchase_Id, Purchase_Amount, Purchase_Date, Customer_Id,
Payment_Id, Station_Id, Ticket_Box_Id, Route_Id)
VALUES
    (1, 50.00, '2024-01-05', 1, 2, 3, 4, 5),
    (2, 25.00, '2024-02-15', 6, 5, 4, 3, 2),
    (3, 10.00, '2024-03-10', 5, 4, 6, 2, 1),
    (4, 12.00, '2024-04-02', 4, 3, 2, 1, 3),
    (5, 20.00, '2024-01-12', 3, 1, 1, 5, 6),
    (6, 30.00, '2024-03-20', 2, 6, 5, 6, 4),
    (7, 40.00, '2024-04-08', 1, 6, 5, 2, 3);
```

PART 4: **Write queries to accomplish the following:**

```
-- --------------------------------------------------------------------------------
-- QUERIES ------------------------------------------------------------------------


-- All Tables Execution
SELECT * FROM CUSTOMER;
SELECT * FROM Balance;
SELECT * FROM MonthlySubscription;
SELECT * FROM PURCHASE;
SELECT * FROM Ticket;
SELECT * FROM PAYMENT;
SELECT * FROM STOP;
SELECT * FROM ROUTE;
SELECT * FROM STATION;
SELECT * FROM DRIVER;
SELECT * FROM BUS;
```

1)-- **View all routes in the city and what buses go on these routes**

```
SELECT
    ROUTE.Route_Id,
    ROUTE.Departure_Location,
    ROUTE.Arrival_Location,
    BUS.Bus_Id,
    BUS.Bus_Plate
FROM
    ROUTE
JOIN
    BUS ON BUS.Bus_Id = ROUTE.Bus_Id;
```

2)-- **Search by bus stop to see which bus goes to that stop and display the bus number** and the full route

```
SELECT
    STOP.Stop_Name,
    BUS.Bus_Id,
    BUS.Bus_Plate,
    ROUTE.Departure_Location,
    ROUTE.Arrival_Location
FROM
    STOP
JOIN
```

```
    ROUTE ON STOP.Stop_Id = ROUTE.Stop_Id
JOIN
    BUS ON BUS.Bus_Id = ROUTE.Bus_Id;
```

3)-- **Display driver's information and the assigned bus**

```
SELECT
    DRIVER.Driver_Id,
    DRIVER.Driver_Name,
    DRIVER.Phone_Number,
    BUS.Bus_Id,
    BUS.Bus_Plate
FROM
    ROUTE
JOIN
    BUS ON BUS.Bus_Id = ROUTE.Bus_Id
JOIN
    DRIVER ON DRIVER.Driver_Id = ROUTE.Driver_Id;
```

4)-- **Search for expired license plates for the buses OR expired driver's license and display the bus information and the driver's information**

```
SELECT
    BUS.Bus_Id,
    BUS.Bus_Plate,
    BUS.License_Plate,
          BUS.Expiry_Date,
    BUS.Capacity,
    DRIVER.Driver_Id,
    DRIVER.Driver_Name,
    DRIVER.Phone_Number,
    DRIVER.Driver_License,
    DRIVER.License_Number,
    DRIVER.Expiry_Date
FROM
    ROUTE
JOIN
```

```
    BUS ON BUS.Bus_Id = ROUTE.Bus_Id
JOIN
    DRIVER ON DRIVER.Driver_Id = ROUTE.Driver_Id
WHERE
    BUS.Expiry_Date < CURDATE() OR DRIVER.Expiry_Date < CURDATE();
```

5)-- **Search for expired license plates for the buses AND expired driver's license and display the bus information and the driver's information**

```
SELECT
    BUS.Bus_Id,
    BUS.Bus_Plate,
    BUS.License_Plate,
        BUS.Expiry_Date,
    BUS.Capacity,
    DRIVER.Driver_Id,
    DRIVER.Driver_Name,
    DRIVER.Phone_Number,
    DRIVER.Driver_License,
    DRIVER.License_Number,
    DRIVER.Expiry_Date
FROM
    ROUTE
JOIN
    BUS ON BUS.Bus_Id = ROUTE.Bus_Id
JOIN
    DRIVER ON DRIVER.Driver_Id = ROUTE.Driver_Id
WHERE
    BUS.Expiry_Date < CURDATE() AND DRIVER.Expiry_Date < CURDATE();
```

5)-- **Search by station and see all buses stopping at that station along with the routes they will go on**

```
SELECT
    STATION.Station_Name,
    STATION.Address AS Station_Address,
    BUS.Bus_Id,
    BUS.Bus_Plate,
    ROUTE.Route_Id,
    ROUTE.Departure_Location,
    ROUTE.Arrival_Location
FROM
    ROUTE
JOIN
    STATION ON STATION.Station_Id = ROUTE.Station_Id
JOIN
    BUS ON BUS.Bus_Id = ROUTE.Bus_Id;
```

6)-- **View all buses and all routes for the buses**

```
SELECT
   BUS.Bus_Id,
   BUS.Bus_Plate,
   ROUTE.Route_Id,
   ROUTE.Departure_Location,
   ROUTE.Arrival_Location
FROM
   ROUTE
JOIN
        BUS ON BUS.Bus_Id = ROUTE.Bus_Id;
```

7)-- **Search by customers and show their subscription information**

```
SELECT
   CUSTOMER.Customer_Id,
   CUSTOMER.Customer_Name,
   MonthlySubscription.Payment_Id AS Subscription_ID,
   MonthlySubscription.Start_Date,
   MonthlySubscription.Expiry_Date
FROM
   PURCHASE
JOIN
   CUSTOMER ON CUSTOMER.Customer_Id = PURCHASE.Customer_Id
JOIN
        PAYMENT ON PAYMENT.Payment_Id = PURCHASE.Payment_Id
```

```
JOIN
	MonthlySubscription ON MonthlySubscription.Payment_Id = PAYMENT.Payment_Id
WHERE
	PAYMENT.Payment_Type = "S";
```