**Car Rental System**

**Task 1: Design the Database Schema**

-- Creating Customers Table

CREATE TABLE Customers (

CUSTOMER\_ID NUMBER PRIMARY KEY,

FIRST\_NAME VARCHAR2(50),

LAST\_NAME VARCHAR2(50),

EMAIL VARCHAR2(100),

PHONE\_NUMBER VARCHAR2(15),

ADDRESS VARCHAR2(255)

);

-- Creating Cars Table

CREATE TABLE Cars (

CAR\_ID NUMBER PRIMARY KEY,

MAKE VARCHAR2(50),

MODEL VARCHAR2(50),

YEAR NUMBER,

RATE\_PER\_DAY NUMBER,

AVAILABLE NUMBER -- 1 for available, 0 for not available

);

-- Creating Rentals Table

CREATE TABLE Rentals (

RENTAL\_ID NUMBER PRIMARY KEY,

CUSTOMER\_ID NUMBER,

CAR\_ID NUMBER,

START\_DATE DATE,

END\_DATE DATE,

STATUS VARCHAR2(50),

FOREIGN KEY (CUSTOMER\_ID) REFERENCES Customers(CUSTOMER\_ID),

FOREIGN KEY (CAR\_ID) REFERENCES Cars(CAR\_ID)

);

**Task 2: Populate the Database with Sample Data**

--Insert Sample Data into Customers Table

INSERT INTO Customers (CUSTOMER\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, ADDRESS)

VALUES (1, 'John', 'Doe', 'john.doe@example.com', '1234567890', '123 Elm Street');

INSERT INTO Customers (CUSTOMER\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, ADDRESS)

VALUES (2, 'Jane', 'Smith', 'jane.smith@example.com', '0987654321', '456 Oak Avenue');

--Insert Sample Data into Cars Table

INSERT INTO Cars (CAR\_ID, MAKE, MODEL, YEAR, RATE\_PER\_DAY, AVAILABLE)

VALUES (1, 'Toyota', 'Camry', 2020, 50, 1);

INSERT INTO Cars (CAR\_ID, MAKE, MODEL, YEAR, RATE\_PER\_DAY, AVAILABLE)

VALUES (2, 'Honda', 'Civic', 2019, 45, 1);

--Insert Sample Data into Rentals Table

INSERT INTO Rentals (RENTAL\_ID, CUSTOMER\_ID, CAR\_ID, START\_DATE, END\_DATE, STATUS)

VALUES (1, 1, 1, TO\_DATE('2024-09-01', 'YYYY-MM-DD'), TO\_DATE('2024-09-05', 'YYYY-MM-DD'), 'Completed');

INSERT INTO Rentals (RENTAL\_ID, CUSTOMER\_ID, CAR\_ID, START\_DATE, END\_DATE, STATUS)

VALUES (2, 2, 2, TO\_DATE('2024-09-03', 'YYYY-MM-DD'), TO\_DATE('2024-09-07', 'YYYY-MM-DD'), 'Ongoing');

**Task 3: Develop PL/SQL Procedures**

--Procedure for Rental Management

CREATE OR REPLACE PROCEDURE ManageRental(

p\_rental\_id IN NUMBER,

p\_customer\_id IN NUMBER,

p\_car\_id IN NUMBER,

p\_start\_date IN DATE,

p\_end\_date IN DATE,

p\_status IN VARCHAR2,

p\_action IN VARCHAR2

) AS

BEGIN

IF p\_action = 'INSERT' THEN

INSERT INTO Rentals (RENTAL\_ID, CUSTOMER\_ID, CAR\_ID, START\_DATE, END\_DATE, STATUS)

VALUES (p\_rental\_id, p\_customer\_id, p\_car\_id, p\_start\_date, p\_end\_date, p\_status);

UPDATE Cars SET AVAILABLE = 0 WHERE CAR\_ID = p\_car\_id;

ELSIF p\_action = 'UPDATE' THEN

UPDATE Rentals SET CUSTOMER\_ID = p\_customer\_id, CAR\_ID = p\_car\_id,

START\_DATE = p\_start\_date, END\_DATE = p\_end\_date, STATUS = p\_status

WHERE RENTAL\_ID = p\_rental\_id;

ELSIF p\_action = 'DELETE' THEN

DELETE FROM Rentals WHERE RENTAL\_ID = p\_rental\_id;

UPDATE Cars SET AVAILABLE = 1 WHERE CAR\_ID = p\_car\_id;

END IF;

END;

/

--Procedure for Customer Registration

CREATE OR REPLACE PROCEDURE RegisterCustomer(

p\_customer\_id IN NUMBER,

p\_first\_name IN VARCHAR2,

p\_last\_name IN VARCHAR2,

p\_email IN VARCHAR2,

p\_phone\_number IN VARCHAR2,

p\_address IN VARCHAR2

) AS

BEGIN

INSERT INTO Customers (CUSTOMER\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, ADDRESS)

VALUES (p\_customer\_id, p\_first\_name, p\_last\_name, p\_email, p\_phone\_number, p\_address);

END;

/

--Procedure for Car Availability Tracking

CREATE OR REPLACE PROCEDURE TrackCarAvailability(

p\_car\_id IN NUMBER,

p\_available IN NUMBER

) AS

BEGIN

UPDATE Cars SET AVAILABLE = p\_available WHERE CAR\_ID = p\_car\_id;

END;

/

--Registering new Customer

BEGIN

RegisterCustomer(

p\_customer\_id => 3,

p\_first\_name => 'Alice',

p\_last\_name => 'Johnson',

p\_email => 'alice.johnson@example.com',

p\_phone\_number => '9876543210',

p\_address => '789 Pine Street'

);

END;

/

-- Verify by querying the Customers table

SELECT \* FROM Customers WHERE CUSTOMER\_ID = 3;

--Insert a New Rental:

BEGIN

ManageRental(

p\_rental\_id => 3,

p\_customer\_id => 3,

p\_car\_id => 1,

p\_start\_date => TO\_DATE('2024-09-10', 'YYYY-MM-DD'),

p\_end\_date => TO\_DATE('2024-09-15', 'YYYY-MM-DD'),

p\_status => 'Scheduled',

p\_action => 'INSERT'

);

END;

/

-- Verify by querying the Rentals table

SELECT \* FROM Rentals WHERE RENTAL\_ID = 3;

-- Check if car availability is updated

SELECT AVAILABLE FROM Cars WHERE CAR\_ID = 1;

--Update an Existing Rental:

BEGIN

ManageRental(

p\_rental\_id => 3,

p\_customer\_id => 3,

p\_car\_id => 1,

p\_start\_date => TO\_DATE('2024-09-12', 'YYYY-MM-DD'),

p\_end\_date => TO\_DATE('2024-09-17', 'YYYY-MM-DD'),

p\_status => 'Updated',

p\_action => 'UPDATE'

);

END;

/

-- Verify by querying the Rentals table

select \* from Rentals;

SELECT \* FROM Rentals WHERE RENTAL\_ID = 3;

--Delete a Rental:

BEGIN

ManageRental(

p\_rental\_id => 3,

p\_customer\_id => 3,

p\_car\_id => 1,

p\_start\_date => NULL,

p\_end\_date => NULL,

p\_status => NULL,

p\_action => 'DELETE'

);

END;

/

-- Verify by querying the Rentals table

SELECT \* FROM Rentals WHERE RENTAL\_ID = 3;

-- Check if car availability is updated back to available

SELECT AVAILABLE FROM Cars WHERE CAR\_ID = 1;

--Mark a Car as Not Available:

BEGIN

TrackCarAvailability(

p\_car\_id => 2,

p\_available => 0

);

END;

/

-- Verify by querying the Cars table

SELECT \* FROM Cars WHERE CAR\_ID = 2;

--Mark a Car as Available:

BEGIN

TrackCarAvailability(

p\_car\_id => 2,

p\_available => 1

);

END;

/

select car\_id, AVAILABLE from Cars;

-- Verify by querying the Cars table

SELECT \* FROM Cars WHERE CAR\_ID = 2;