**<Project Car Rental System>**

****

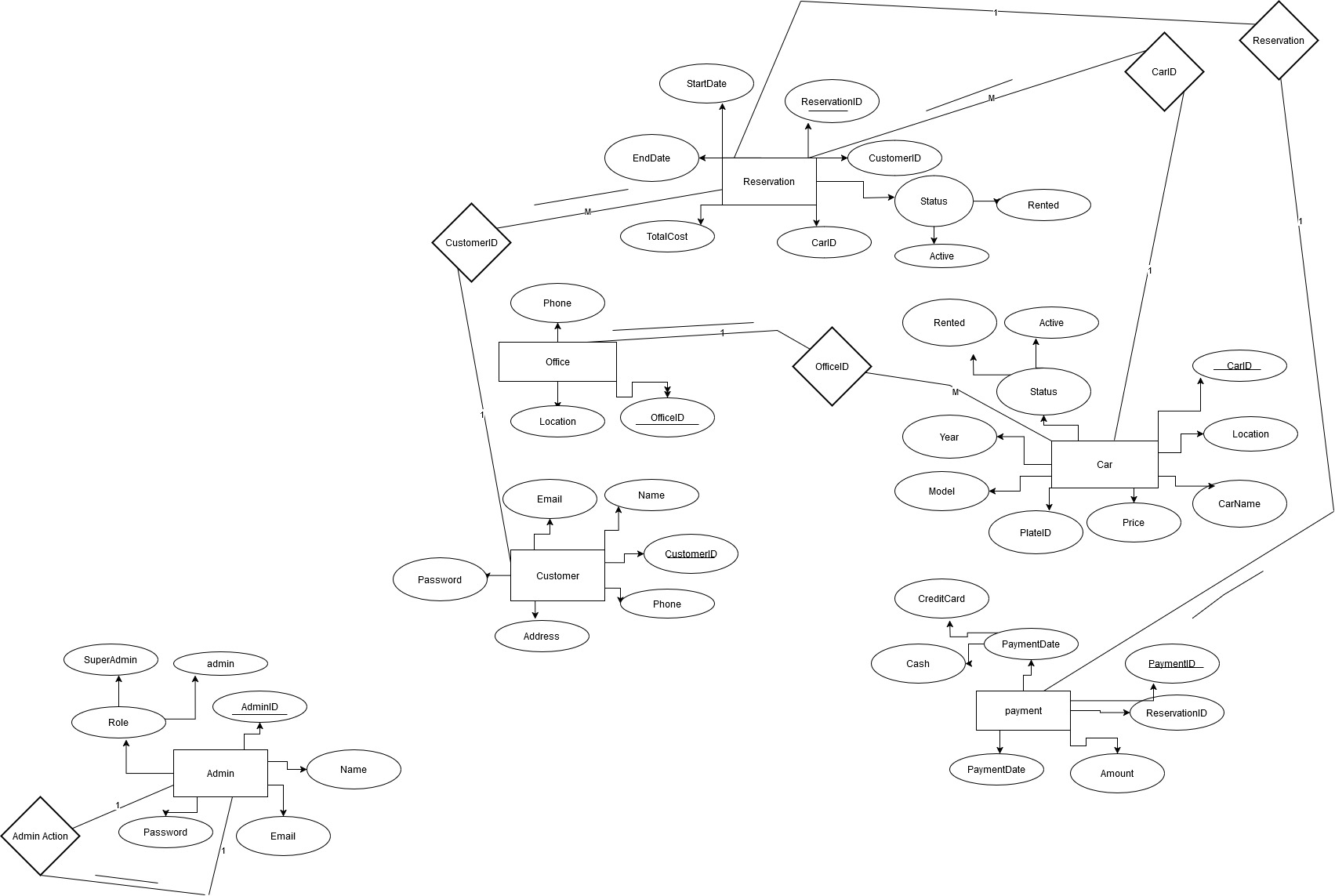
**OUR TEAM:**

**Abdelrahman Mohamed Abdo 2305618**

**Engy Ahmed Ali 2305450**

**Asmaa Al\_Shahat 2305130**

**Rawda Mostafa 230514**

**Project Introduction**

**This site provides the option to book a car in an easy and flexible way.**

**The most important thing that distinguishes our site:**

1. **Simple and easy user interface**

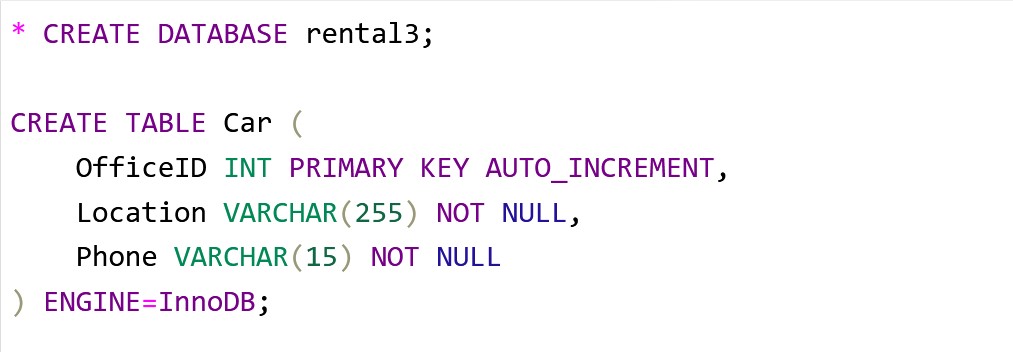
**2- You can book in an easy and comfortable way while you are in your place and from anywhere in the world in a few steps**

**3- The ability to search for the car**

**4- You have the ability to cancel the reservation at any time**

**5- You can extend the reservation period**

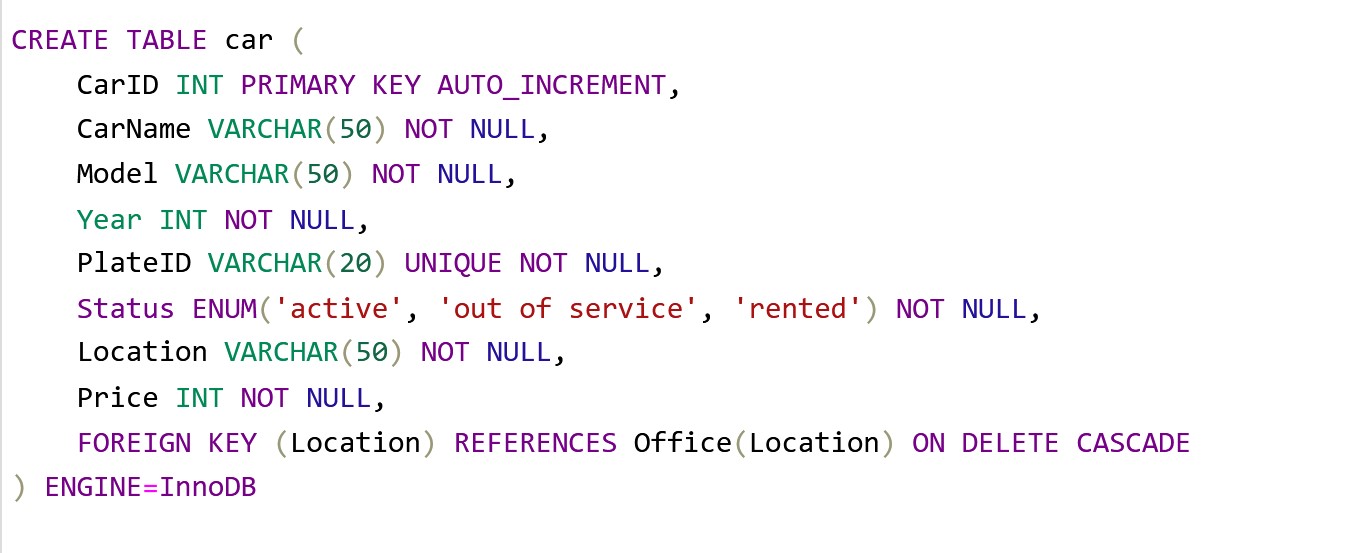
**🡪First Data Base:**

****

**Explanation:**

**TABLE Office 🡪 include PRIMARY KEY 🡪 OfficeID**

**This means that OfficeID is something special to the car and cannot be repeated.**

****

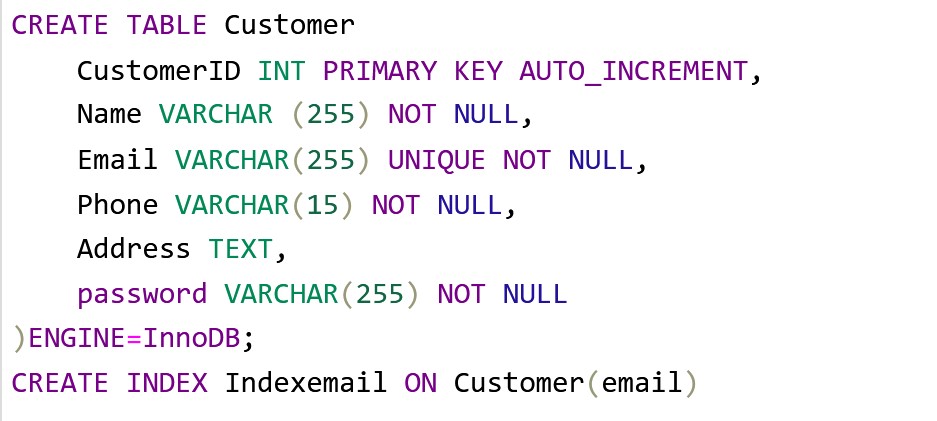
**Explanation:**

**TABLE car🡪 include PRIMARY KEY 🡪 CarID**

**This means that CarID is something special to the car and cannot be repeated.**

**FOREIGN KEY 🡪Location**

**Enforces a relationship between Location columns, and delete related records in the current table when an Office record is deleted.**

****

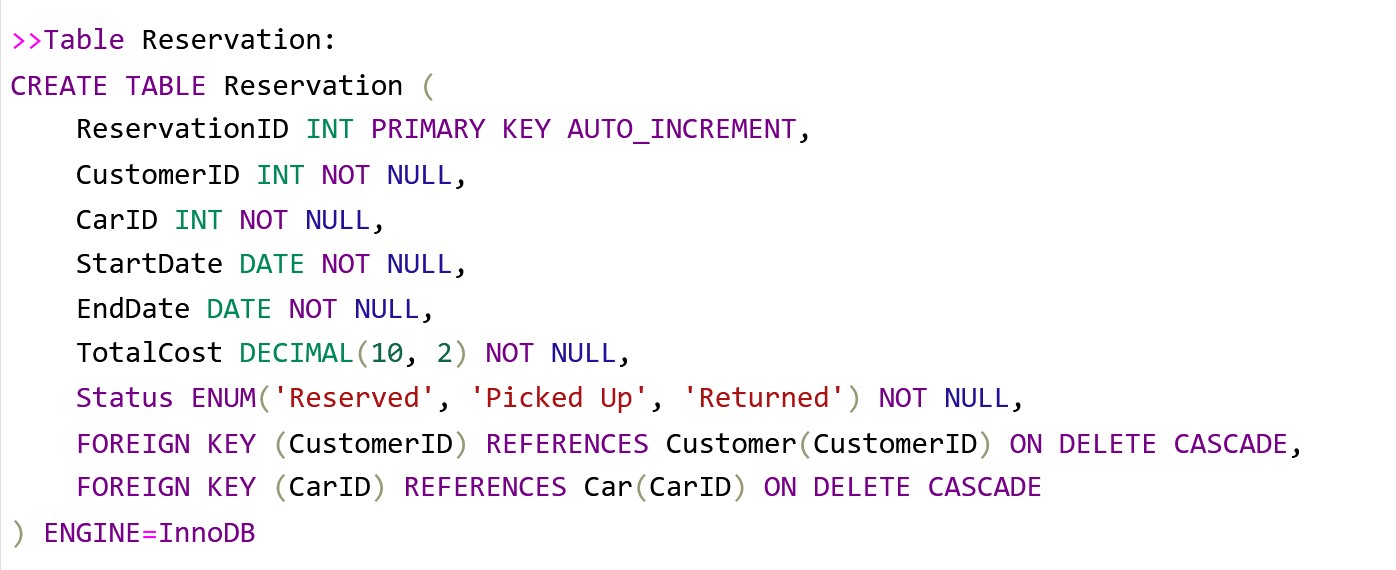
**Explanation:**

**TABLE Customer🡪 include PRIMARY KEY 🡪 Customer ID**

**This means that Customer ID is something special to the car and cannot be repeated.**

**UNIQUE 🡪Email**

**is a constraint in a database that ensures the values in a column or set of columns are unique and do not repeat within the table. It allows storing distinct values in the specified column, but NULL values can be allowed if the column supports it.**

****

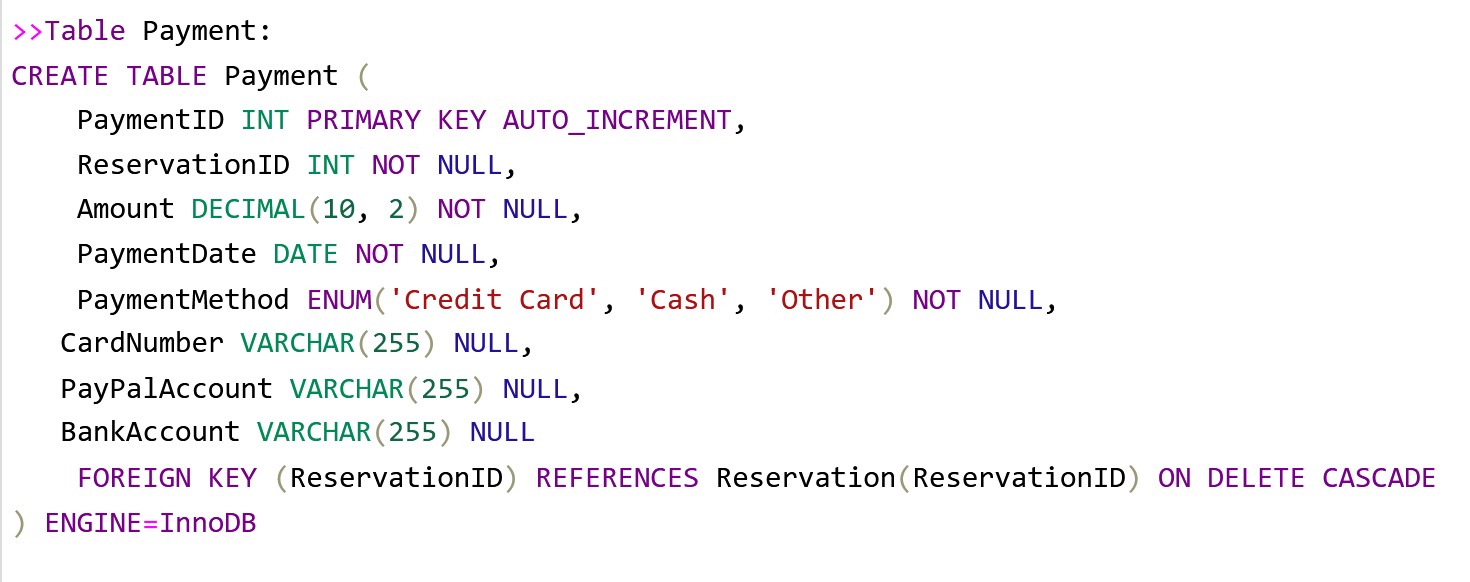
**Explanation:**

**TABLE Reservations 🡪 include PRIMARY KEY 🡪 ReservationID**

**This means that ReservationID is something special to the car and cannot be repeated.**

**FOREIGN KEY 🡪 CustomerID & CarID**

**The foreign key enforces a relationship between CustomerID & CarID columns, and deletes related records in the current table when an car & Customer record is deleted.**

****

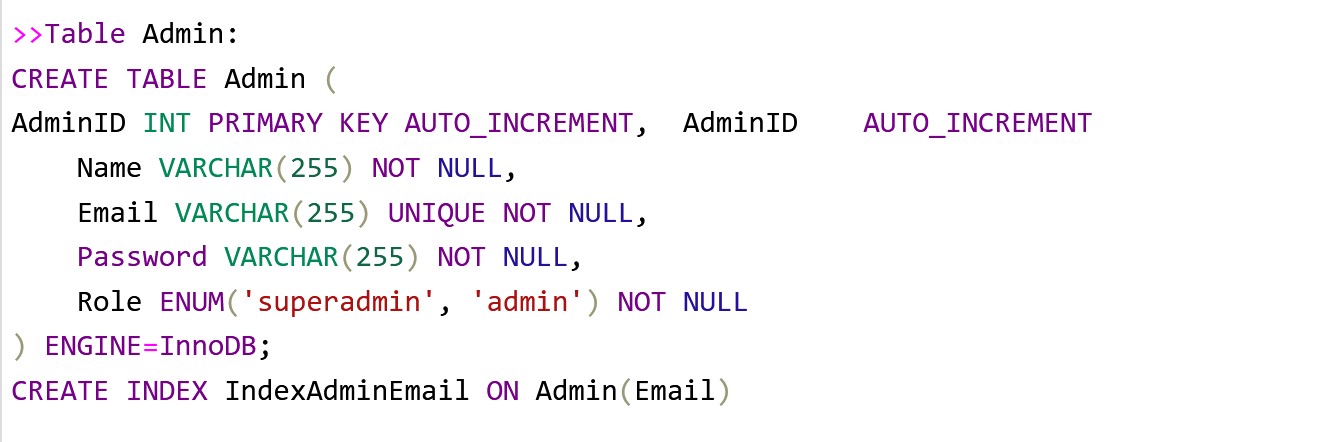
**Explanation:**

**TABLE Payment 🡪 include PRIMARY KEY 🡪 PaymentID**

**This means that PaymentID is something special to the car and cannot be repeated.**

**FOREIGN KEY 🡪 ReservationID**

**The foreign key enforces a relationship between ReservationID columns, and deletes related records in the current table when an car & Customer record is deleted.**

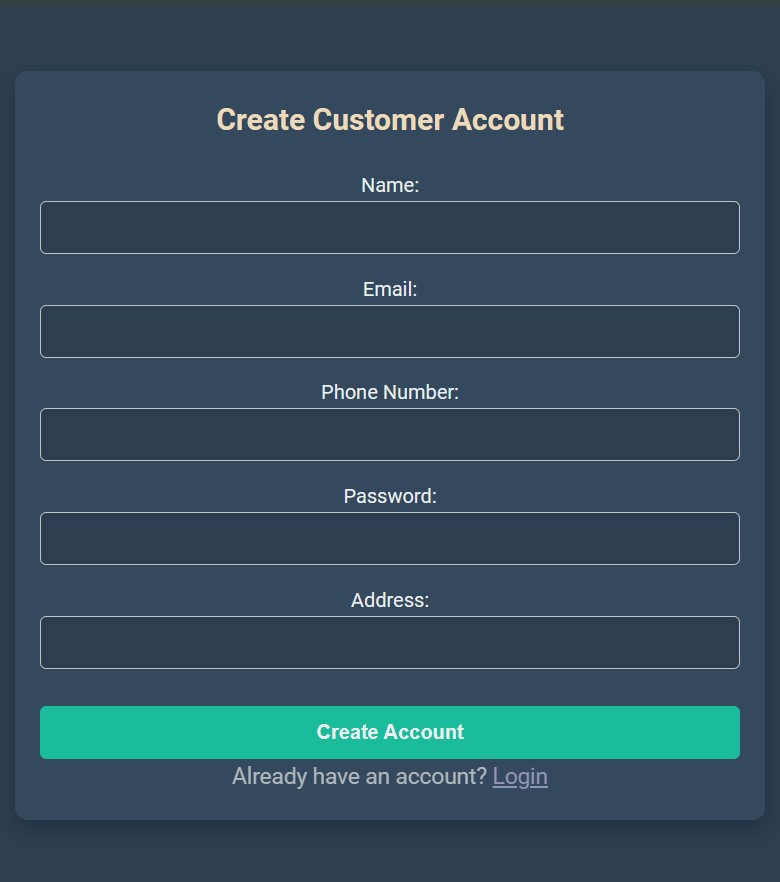
****

**TABLE Admin 🡪 include PRIMARY KEY 🡪 AdminID**

**This means that Admin is something special to the car and cannot be repeated.**

**UNIQUE 🡪Email**

**is a constraint in a database that ensures the values in a column or set of columns are unique and do not repeat within the table. It allows storing distinct values in the specified column, but NULL values can be allowed if the column supports it.**

**🡪Second Main Page:**

**Customer will create account or if he Already have account he will login.**

**After customer** **create account and click create account he will transfer to login page to login at web.**

**Code:**

**$name = $\_POST['customer-name'];**

**$email = $\_POST['customer-email'];**

**$phone = $\_POST['customer-phone'];**

**$password = $\_POST['customer-password'];**

**$address = $\_POST['customer-address'];**

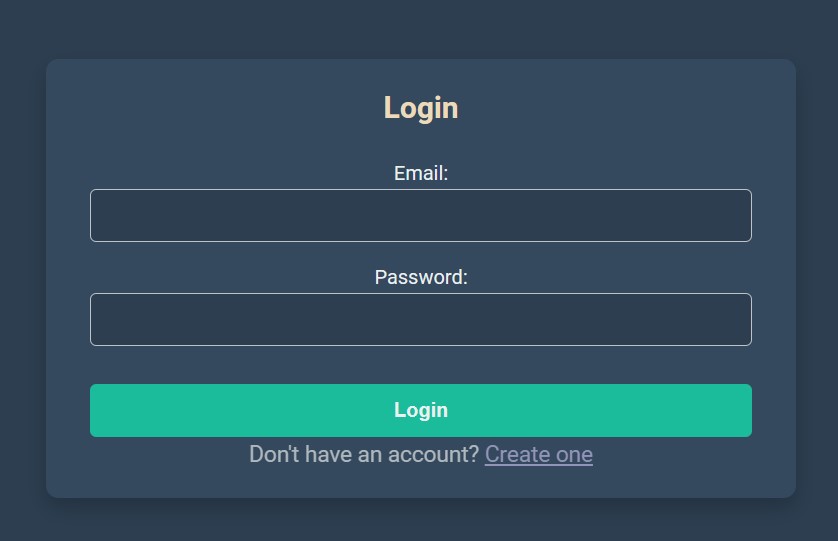
**These commends take values ​​from the booking form in the HTML page.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**$sql = "****INSERT INTO Customer (Name, Email, Phone, Password, Address)**

**VALUES ('$name', '$email', '$phone', '$hashedPassword', '$address')";**

**This commend will INSERT values ​​in DATABASE Specifically table customer.**

****

**Login page:**

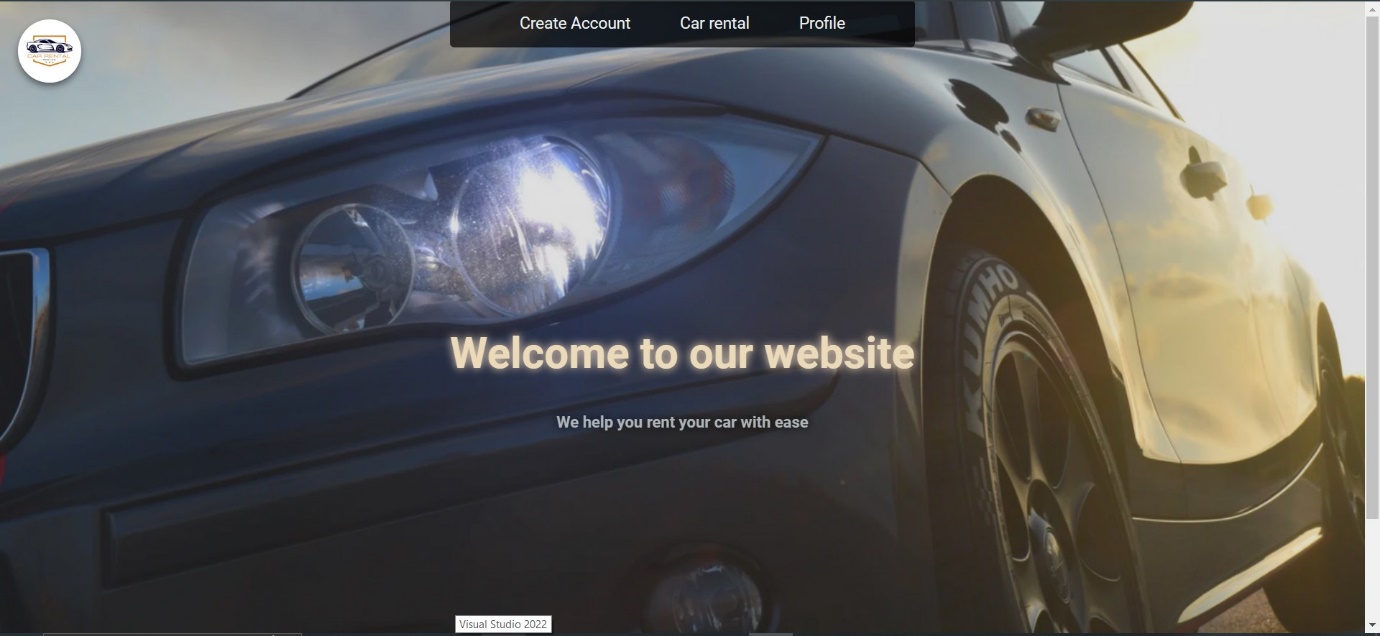
**After customer create account nce clicked, you will be transferred to the login page.**

**"SELECT \* FROM users WHERE username = ?"**

**This command is one of the query commands**

**And it calls the customer's data to make sure that he is actually registered. If he is not registered, the message "No user found with that username" appears.**

**If the registration process is successful, you will be directed to the home page!!!**

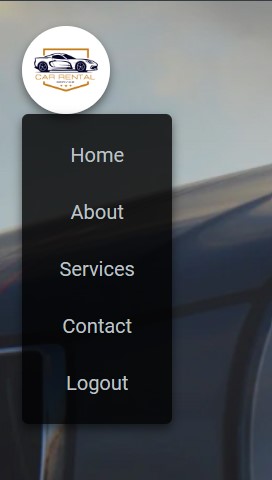
****

**Home page:**

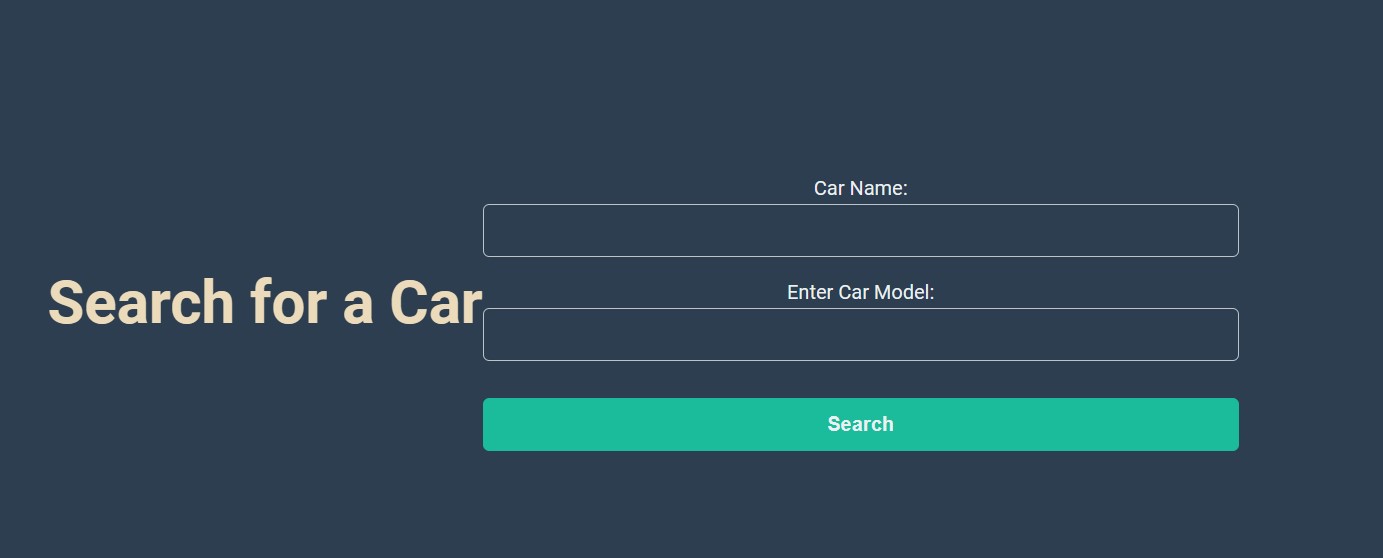
**This interface s** **This is a simple and easy-to-use website interface.**

1. **The customer chooses one of the options in the box at the top (create an account, book a car, or profile**

**or**

1. **He can stand on the logo and a menu will appear containing (Home Page, About Us, Contact Us, Our Services and Log Out)**

**Form Search:**



**When the customer starts to book the car, we will first go to the car search page to make sure it is available.**

**The booking page displays all cars that match the same data entered by the customer.**

**1- If it is available and available, the customer can book it.**

**2- If it is not available, a message will appear stating No car found with the specified name and model.**

**3- If it is available but not available, it will appear stating This car is currently rented OR**

**This car is out of service**

**Code:**

**if (isset($\_POST['search'])) {**

**$CarName = trim($\_POST['CarName']);**

**$CarModel = trim($\_POST['CarModel']);**

**echo "Searching for Car Name: " . $CarName . "<br>";**

**echo "Searching for Car Model: " . $CarModel . "<br>";**

**CarName وModel**

**$sql = "SELECT \* FROM Car**

**WHERE LOWER(CarName) = LOWER('$CarName')**

**AND LOWER(Model) = LOWER('$CarModel')";**

**$result = $conn->query($sql);**

**if (!$result) {**

**die("Error in query: " . $conn->error);**

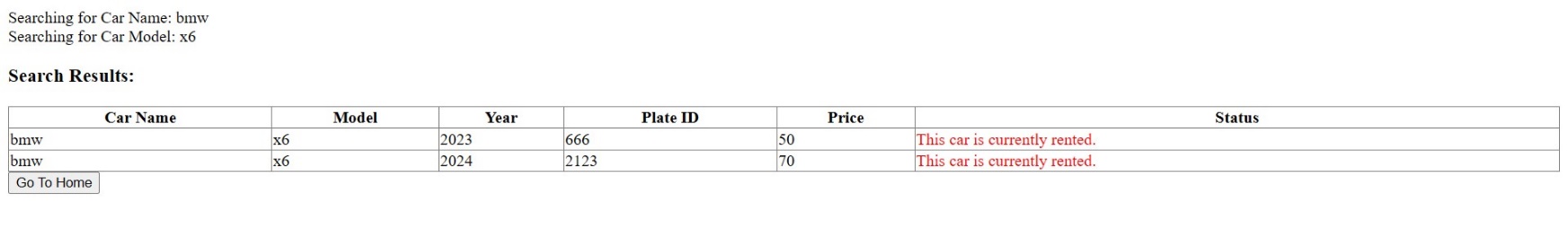
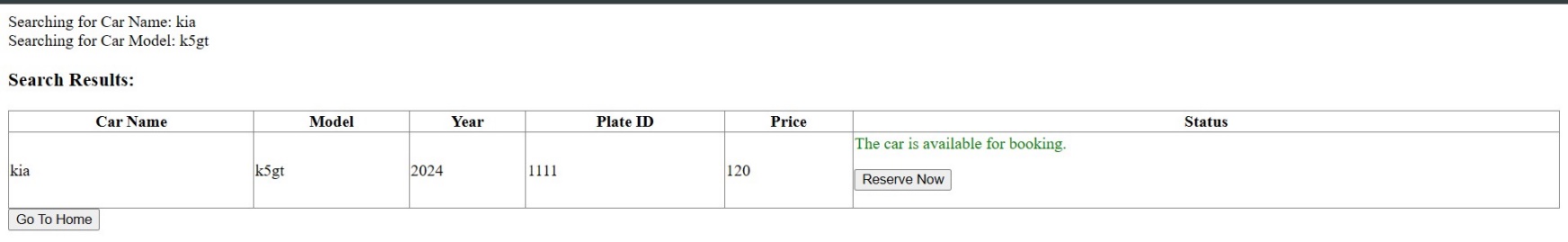
**}**

**In these commands**

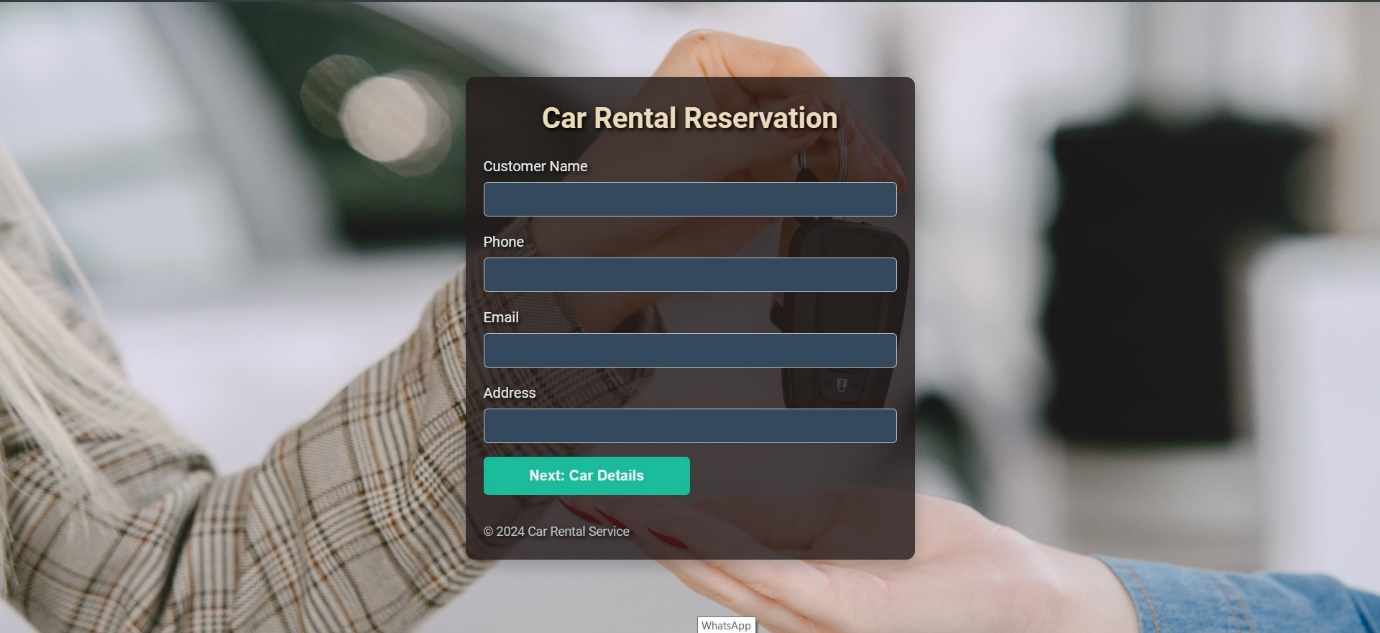
**1- The PHP code receives the car data from the search page, and here comes the role of the query code because it searches the database and determines the car schedule**

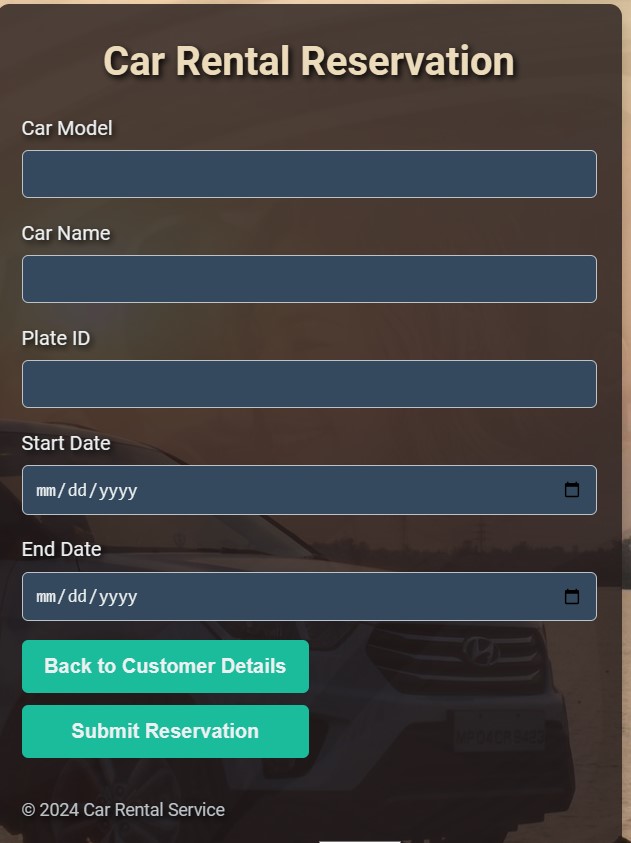
**\* If the car is found, it displays its full data and explains the most important thing, which is whether it is available for reservation or not**

**$sql = "SELECT \* FROM Car WHERE LOWER(CarName) = LOWER('$CarName') AND LOWER(Model) = LOWER('$CarModel')";**

**This code calls the car from the database, provided that it matches the same data of the car that the customer entered.**

**Form Reserve:**

****

**\*The customer will reserve car By filling out the reservation form that requests his and the car’s data**

**\*After the customer fills in his data, he enters the car data and then clicks to record the due amount.**

**Code:**

**"SELECT CustomerID FROM Customer WHERE Name = ? AND Phone = ? AND Email = ?";**

**This commend check if the Customer are actually exists**

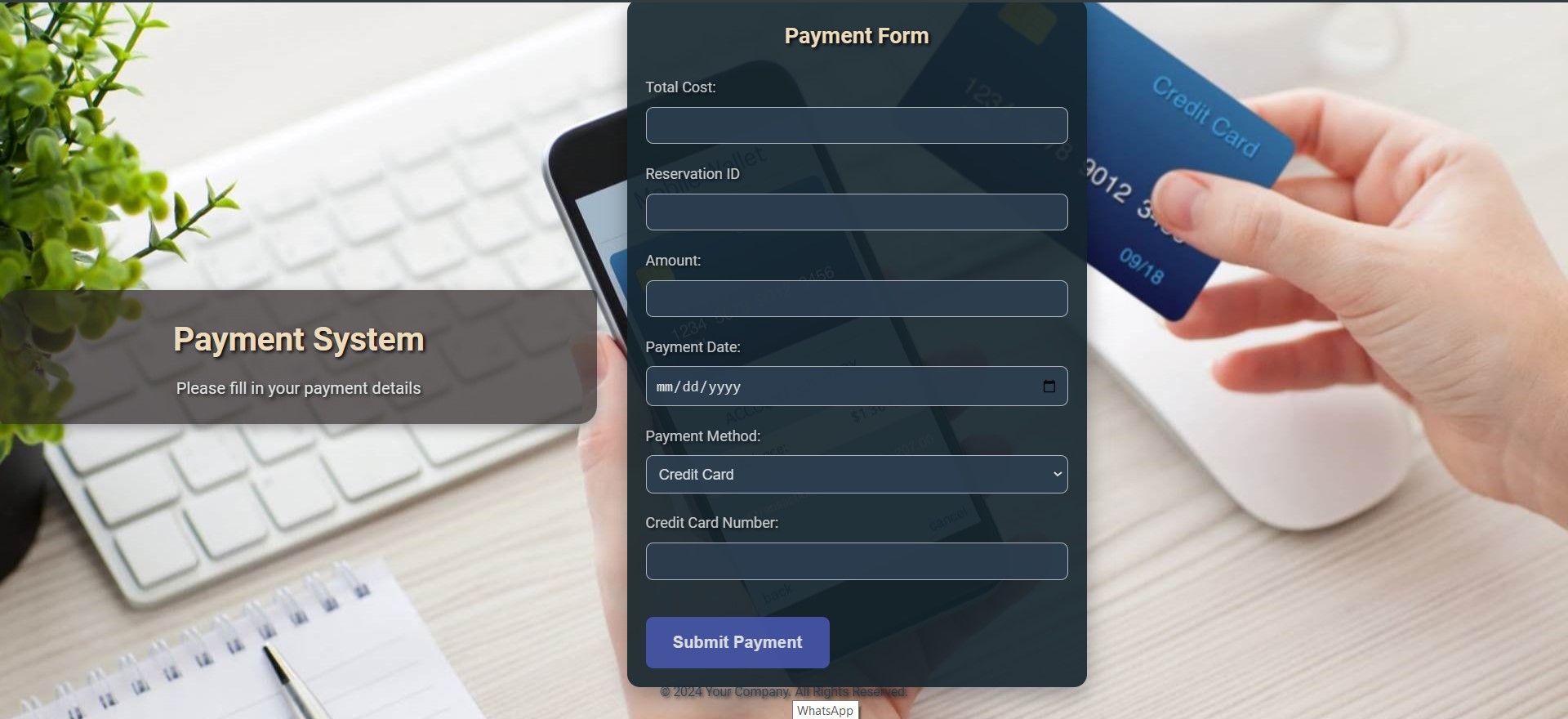
**"?" 🡪 It means that these variables receive non-fixed values, such as the name being different every time, and the same applies to email and phone.**

**"SELECT CarID, Price FROM Car WHERE Model = ? AND CarName = ? AND PlateID = ? AND Status = 'active'";**

**This commend check if the Car is actually exists**

**"?" 🡪 It means that these variables receive non-fixed values, such as the name being different every time, and the same applies to Model and carName…etc**

**Payment Form**

****

**This form asks the customer to enter payment data such as price, reservation number, reservation date and payment type.**

**Code:**

**$query = "INSERT INTO payment (ReservationID, Amount, paymentDate, paymentMethod, CardNumber, PayPalAccount, BankAccount) VALUES (?, ?, ?, ?, ?, ?, ?)";**

**In this query🡪 It enters the payment data into the payment table via Insert. If there is an error, the payment process will not be completed. Common errors include (the column names do not match those in the PHP code or the field names on the page do not match those in PHP).**

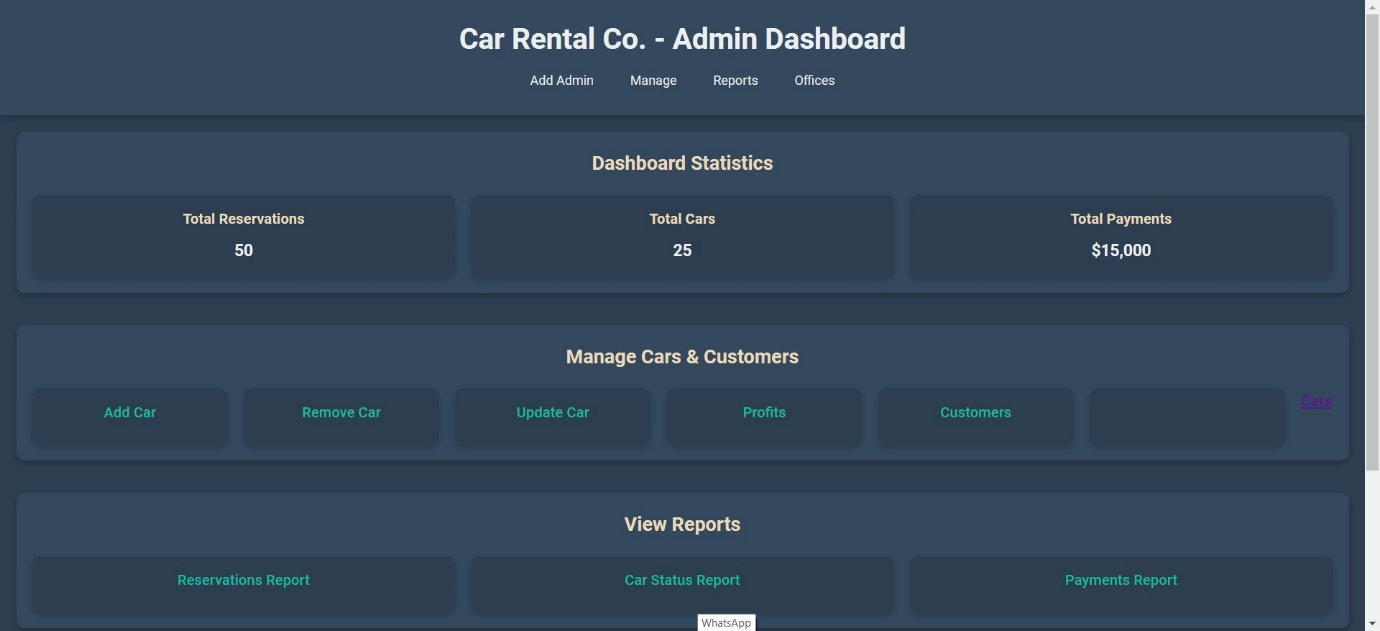
**echo "Payment recorded successfully.";**

**echo "<form action='index.html' method='POST'>**

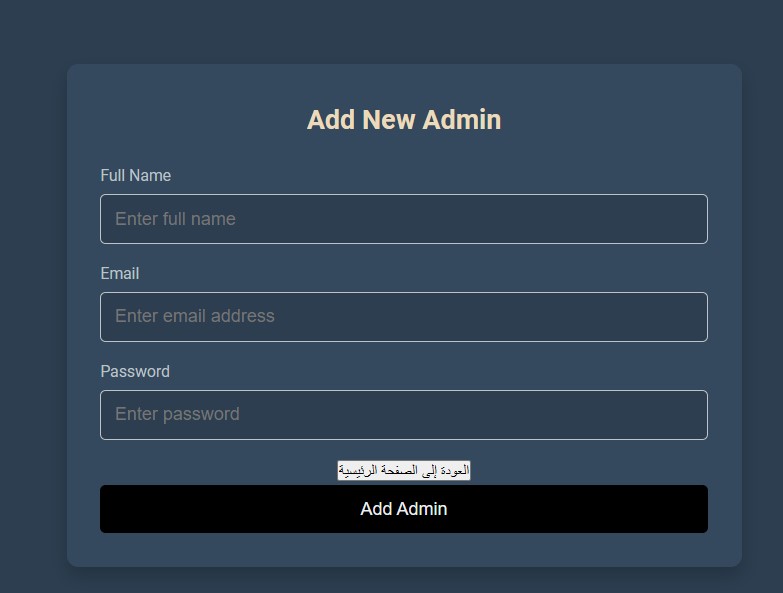
**<button type='submit'>Go To Home</button>**

**</form>";**

**If this process is successfully this message will appear**

**Admin Dashbord:**

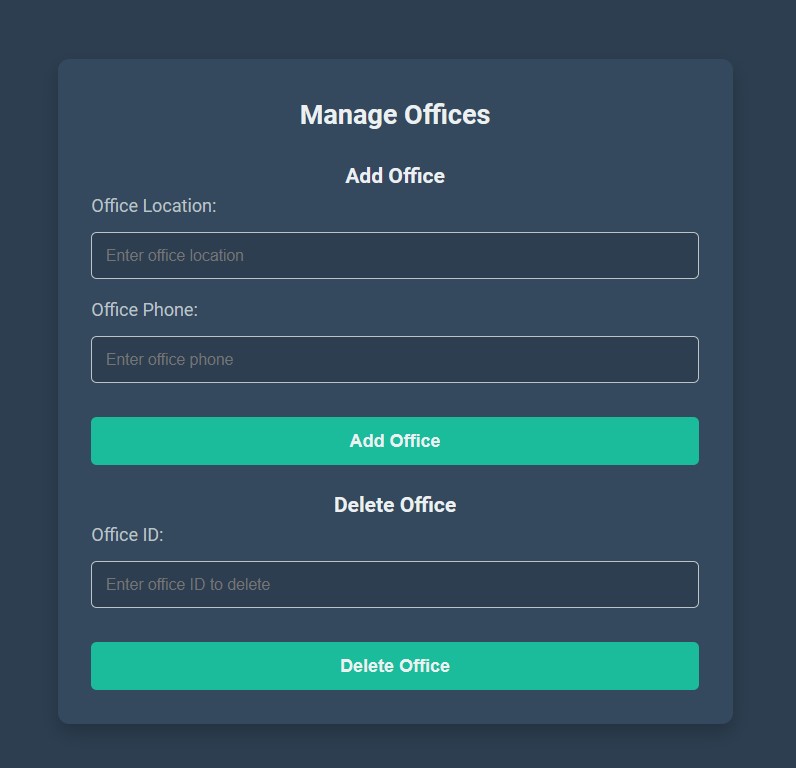
**This is the admin page where you can control the site, such as adding or deleting a car, updating its status, displaying all cars or customers, deleting a customer, adding a new office, adding an admin, and another page through which you can display a report on the status of cars or reservations.**

****

**Here you can add new admin**

**("INSERT INTO admin (name, email, password) VALUES (?, ?, ?)");**

**In this query🡪 It enters the a new admin data into the Admin table via Insert.**

****

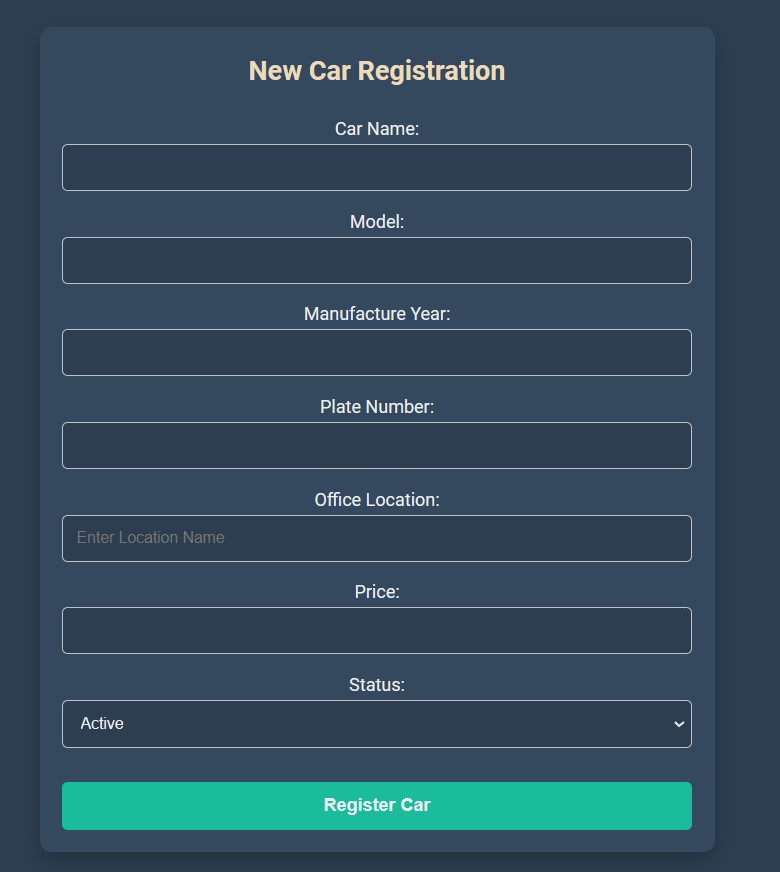
**Here you can Add office and delete**

**INSERT INTO Office (Location, Phone) VALUES (?, ?)";**

**In this query🡪 It enters the a new office data into the office table via Insert.**

**DELETE FROM Office WHERE OfficeID = ?";**

**In this query🡪 It delet the a office data into the office table via delete**

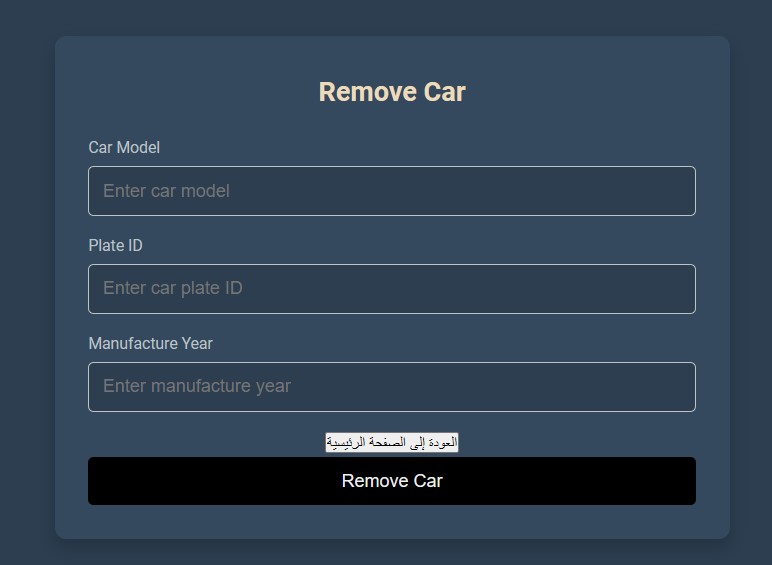
**Form Add Car**

**Here you can add car by filling this field.**

**Code:**

**INSERT INTO car (CarName, Model, Year, PlateID, Status, Location, price) VALUES (?, ?, ?, ?, ?, ?, ?)";**

**In this query🡪 It enters the a new car data into the car table via Insert.**

****

**Form Remove Car**

**Here you can remove car by filling details of car**

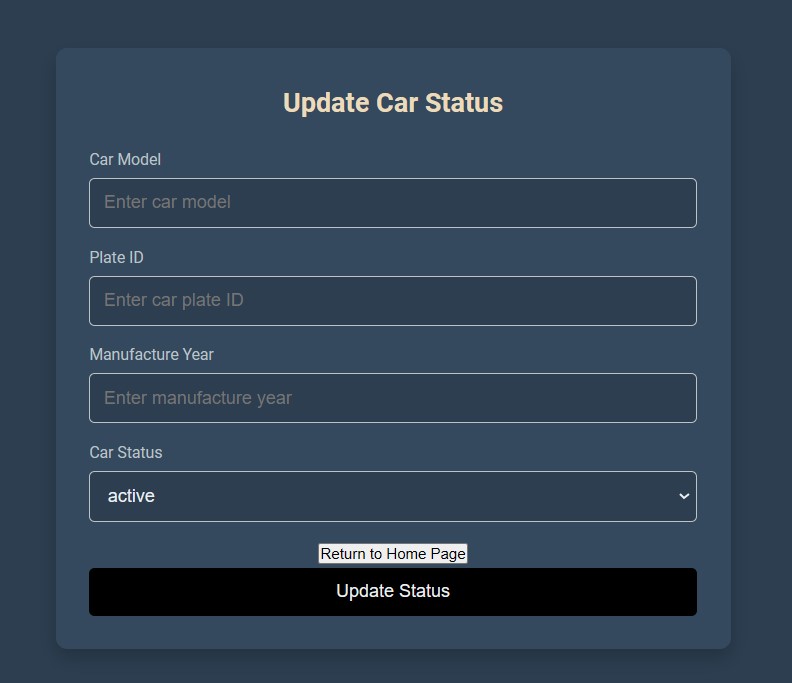
**Code:**

**"SELECT \* FROM Car WHERE Model = ? AND PlateID = ? AND Year = ?";**

**In this query🡪 It check if car is actually exists if it exists**

**"DELETE FROM Car WHERE Model = ? AND PlateID = ? AND Year = ?";**

**In this query🡪 It remove car if car is actually exists**

****

**Form Update Car**

**Here you can Update car by filling details of car**

**Code:**

**"SELECT \* FROM Car WHERE Model = ? AND PlateID = ? AND Year = ?";**

**In this query🡪 It check if car is actually exists if it exists**

**"UPDATE FROM Car WHERE Model = ? AND PlateID = ? AND Year = ?";**

**In this query🡪 It update car if car is actually exists**

****

**Report Reservations:**

**Here you can view report of any date by filling details of date**

**Code:**

**SELECT**

**Reservation.ReservationID,**

**Reservation.StartDate,**

**Reservation.EndDate,**

**Reservation.TotalCost,**

**Reservation.Status,**

**Customer.Name AS CustomerName,**

**Customer.Email AS CustomerEmail,**

**Customer.Phone AS CustomerPhone,**

**Car.Model AS CarModel,**

**Car.Year AS CarYear,**

**Car.PlateID AS CarPlateID**

**FROM Reservation**

**INNER JOIN Customer ON Reservation.CustomerID = Customer.CustomerID**

**INNER JOIN Car ON Reservation.CarID = Car.CarID**

**WHERE Reservation.StartDate >= ? AND Reservation.EndDate <= ?**

**ORDER BY Reservation.StartDate ASC;**

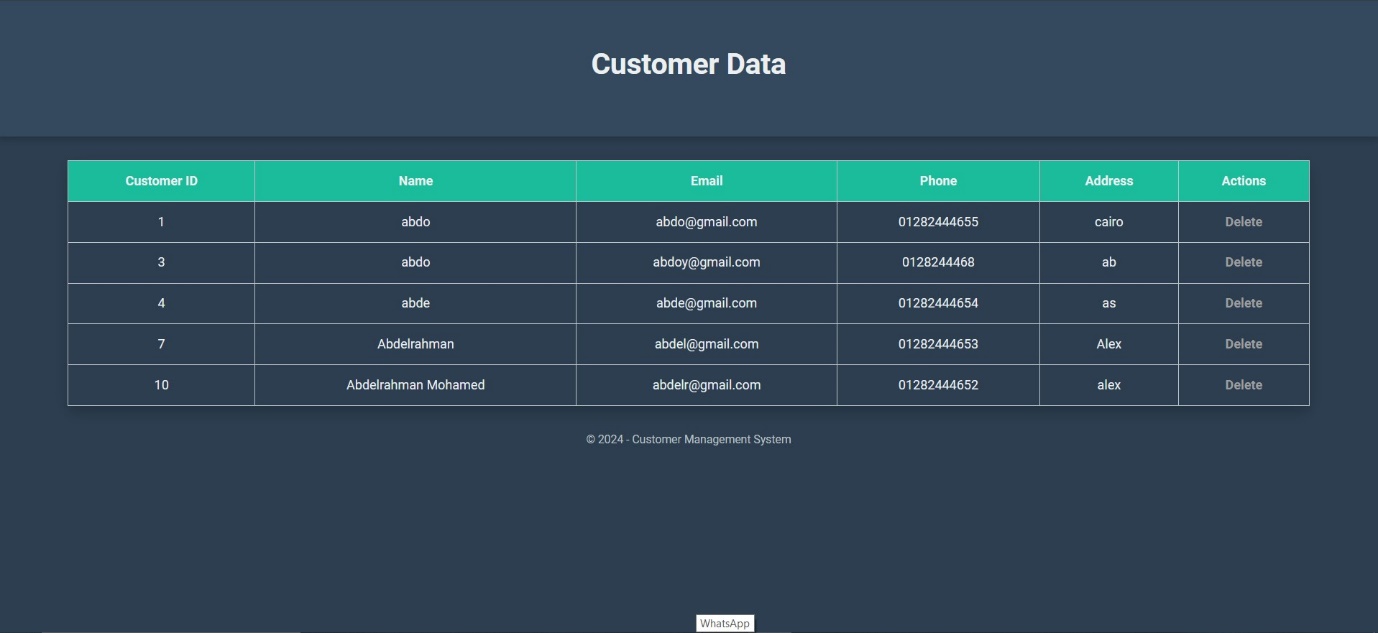
**";**

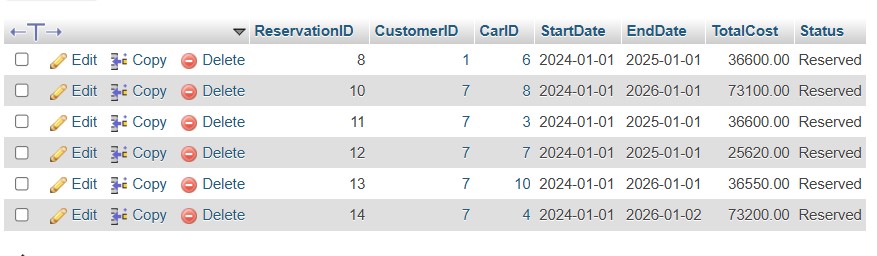
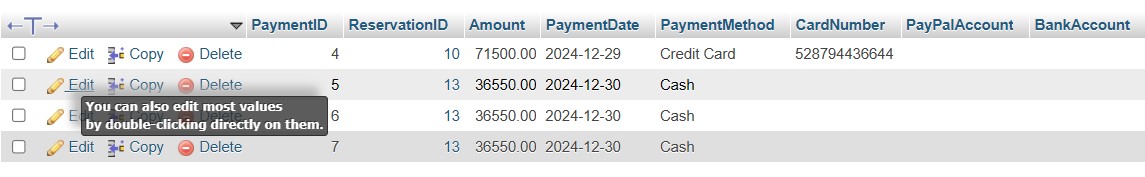
****

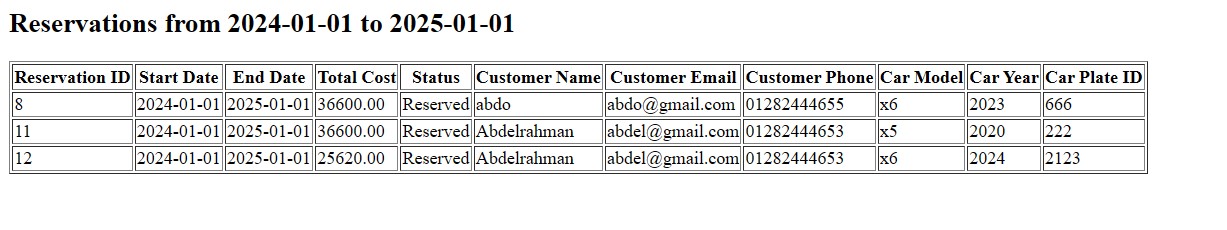
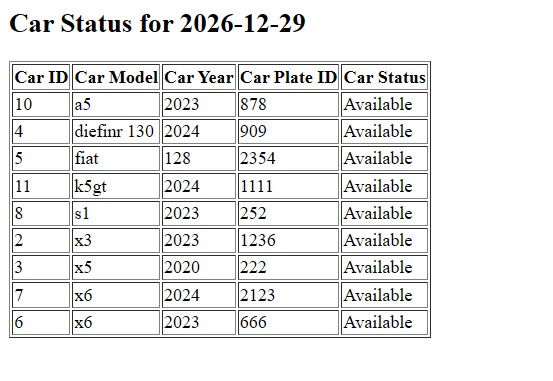
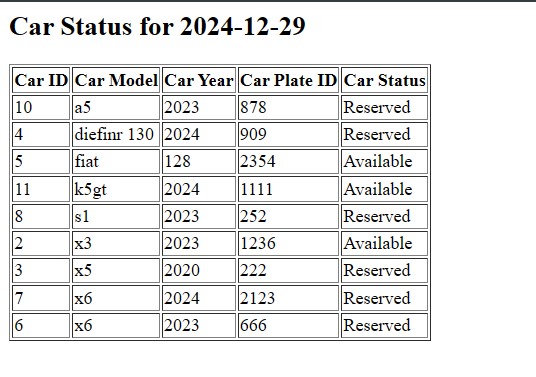
**Report cars status on specific day**

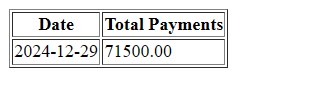
**Here display all cars status on specific day**







**<<TEST>>**

****