<Project Car Rental System>

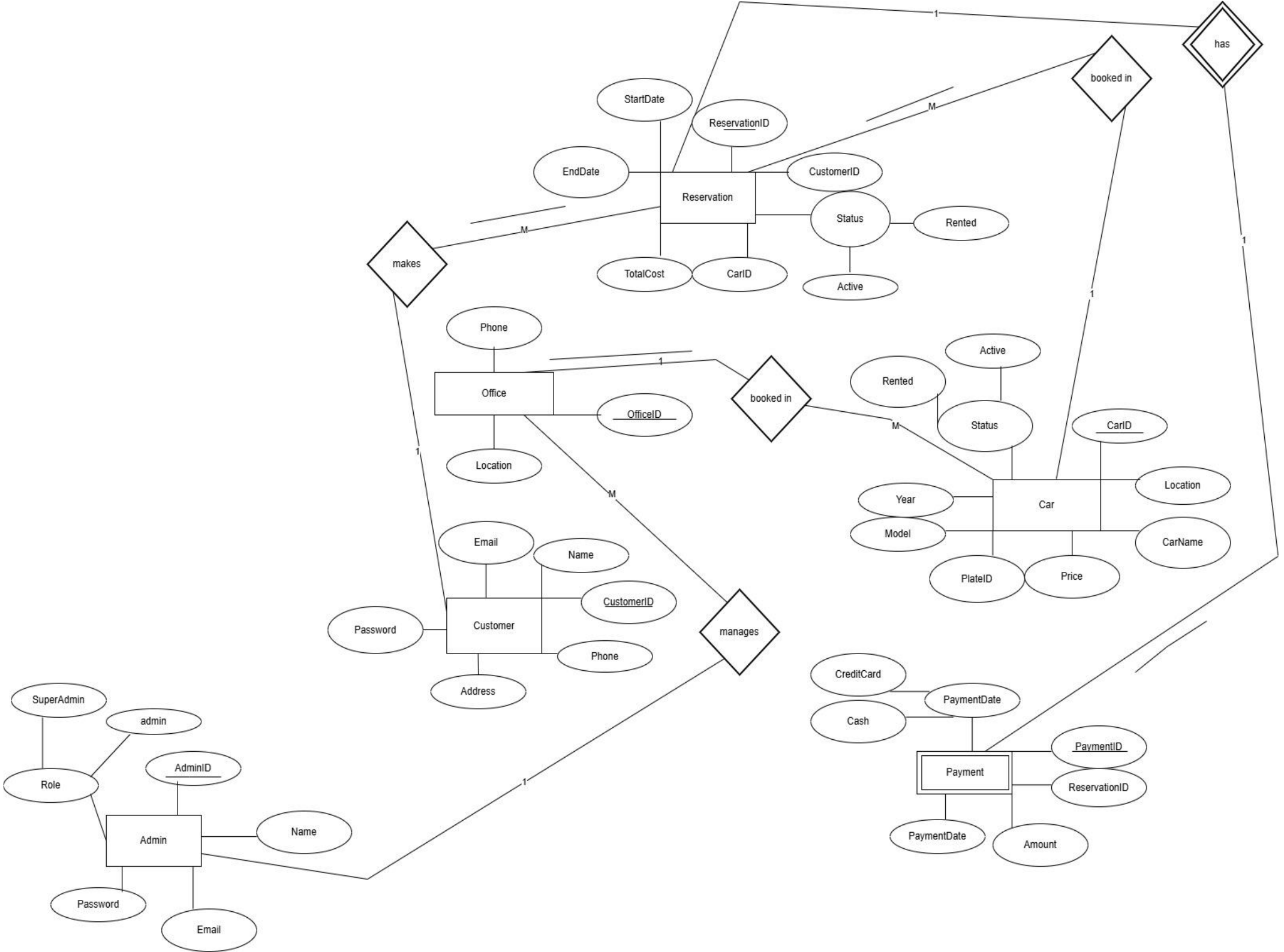
OUR TEAM:

Abdelrahman Mohamed Abdo 2305618

Engy Ahmed Ali 2305450

Asmaa Al_Shahat 2305130

Rawda Mostafa 230514


1
has
booked in
M
StartDate
ReservationID
EndDate
CustomerID
Reservation
Status
Rented
M
makes
TotalCost
CarID
Active
1
Phone
Office
1
booked in
Active
Rented
OfficeID
Status
CarID
1
M
Location
Location
Year
Car
M
Email
Name
Model
CarName
CustomerID
PlateID
Price
Password
Customer
manages
Phone
Address
CreditCard
PaymentDate
Cash
SuperAdmin
admin
PaymentID
AdminID
Payment
Role
ReservationID
1
Admin
Name
PaymentDate
Amount
Password
Email

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:

```
* CREATE DATABASE rental3;

CREATE TABLE Car (
    OfficeID INT PRIMARY KEY AUTO_INCREMENT,
    Location VARCHAR(255) NOT NULL,
    Phone VARCHAR(15) NOT NULL
) ENGINE=InnoDB;
```

Explanation:

TABLE Office → include PRIMARY KEY → OfficeID

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

```
CREATE TABLE car (
    CarID INT PRIMARY KEY AUTO_INCREMENT,
    CarName VARCHAR(50) NOT NULL,
    Model VARCHAR(50) NOT NULL,
    Year INT NOT NULL,
    PlateID VARCHAR(20) UNIQUE NOT NULL,
    Status ENUM('active', 'out of service', 'rented') NOT NULL,
    Location VARCHAR(50) NOT NULL,
    Price INT NOT NULL,
    FOREIGN KEY (Location) REFERENCES Office(Location) ON DELETE CASCADE
) ENGINE=InnoDB
```

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.

```
CREATE TABLE Customer
    CustomerID INT PRIMARY KEY AUTO_INCREMENT,
    Name VARCHAR (255) NOT NULL,
    Email VARCHAR(255) UNIQUE NOT NULL,
    Phone VARCHAR(15) NOT NULL,
    Address TEXT,
    password VARCHAR(255) NOT NULL
)ENGINE=InnoDB;
CREATE INDEX Indexemail ON Customer(email)
```

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.

```
>>Table Reservation:
CREATE TABLE Reservation (
    ReservationID INT PRIMARY KEY AUTO_INCREMENT,
    CustomerID INT NOT NULL,
    CarID INT NOT NULL,
    StartDate DATE NOT NULL,
    EndDate DATE NOT NULL,
    TotalCost DECIMAL(10, 2) NOT NULL,
    Status ENUM('Reserved', 'Picked Up', 'Returned') NOT NULL,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID) ON DELETE CASCADE,
    FOREIGN KEY (CarID) REFERENCES Car(CarID) ON DELETE CASCADE
) ENGINE=InnoDB
```

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.

```
>>Table Payment:
CREATE TABLE Payment (
    PaymentID INT PRIMARY KEY AUTO_INCREMENT,
    ReservationID INT NOT NULL,
    Amount DECIMAL(10, 2) NOT NULL,
    PaymentDate DATE NOT NULL,
    PaymentMethod ENUM('Credit Card', 'Cash', 'Other') NOT NULL,
   CardNumber VARCHAR(255) NULL,
   PayPalAccount VARCHAR(255) NULL,
   BankAccount VARCHAR(255) NULL
    FOREIGN KEY (ReservationID) REFERENCES Reservation(ReservationID) ON DELETE CASCADE
) ENGINE=InnoDB
```

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:
CREATE TABLE Admin (
AdminID INT PRIMARY KEY AUTO_INCREMENT,  AdminID    AUTO_INCREMENT
    Name VARCHAR(255) NOT NULL,
    Email VARCHAR(255) UNIQUE NOT NULL,
    Password VARCHAR(255) NOT NULL,
    Role ENUM('superadmin', 'admin') NOT NULL
) ENGINE=InnoDB;
CREATE INDEX IndexAdminEmail ON Admin(Email)
```

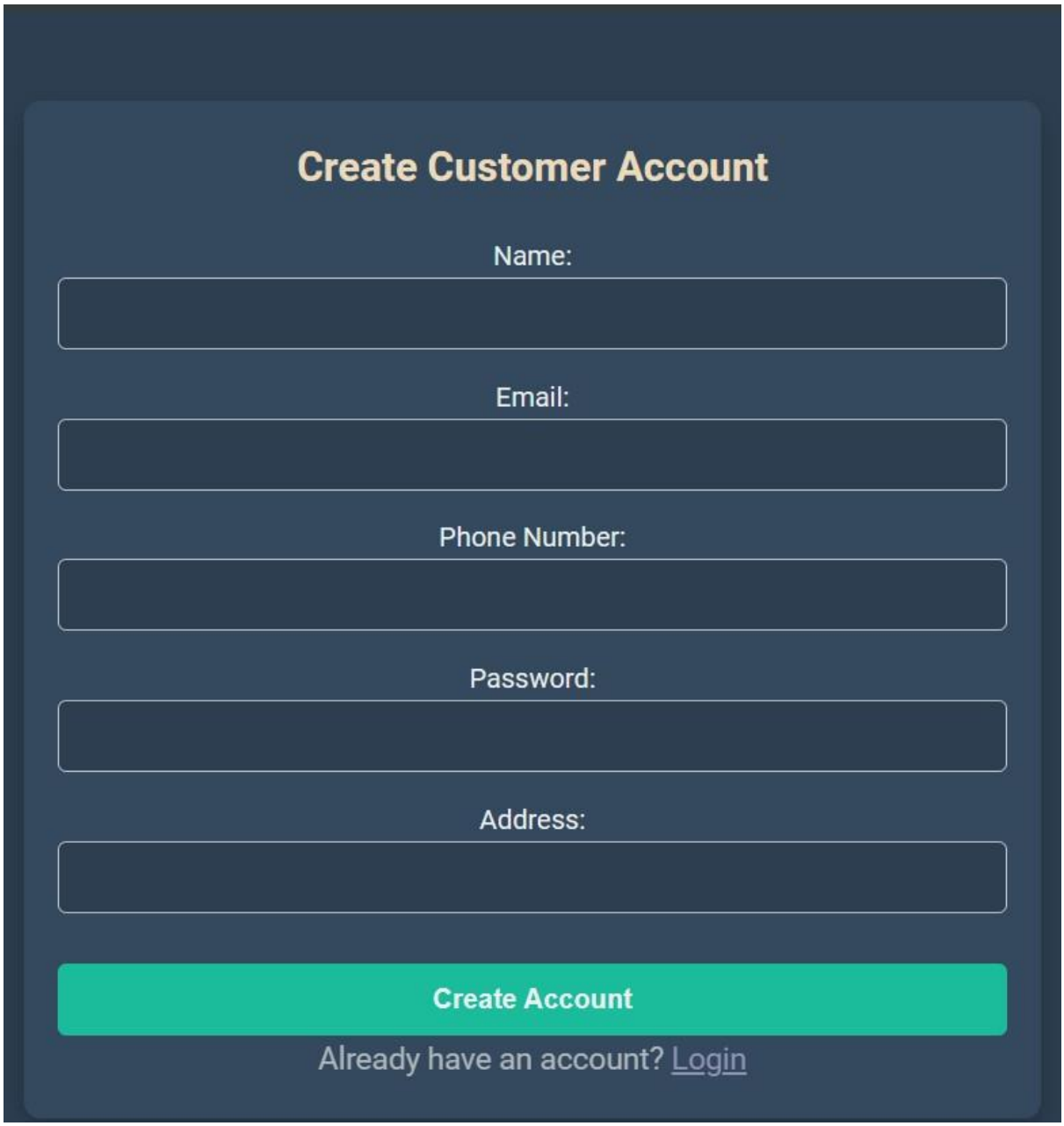
TABLE Admin → include PRIMARY KEY → AdminID

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.

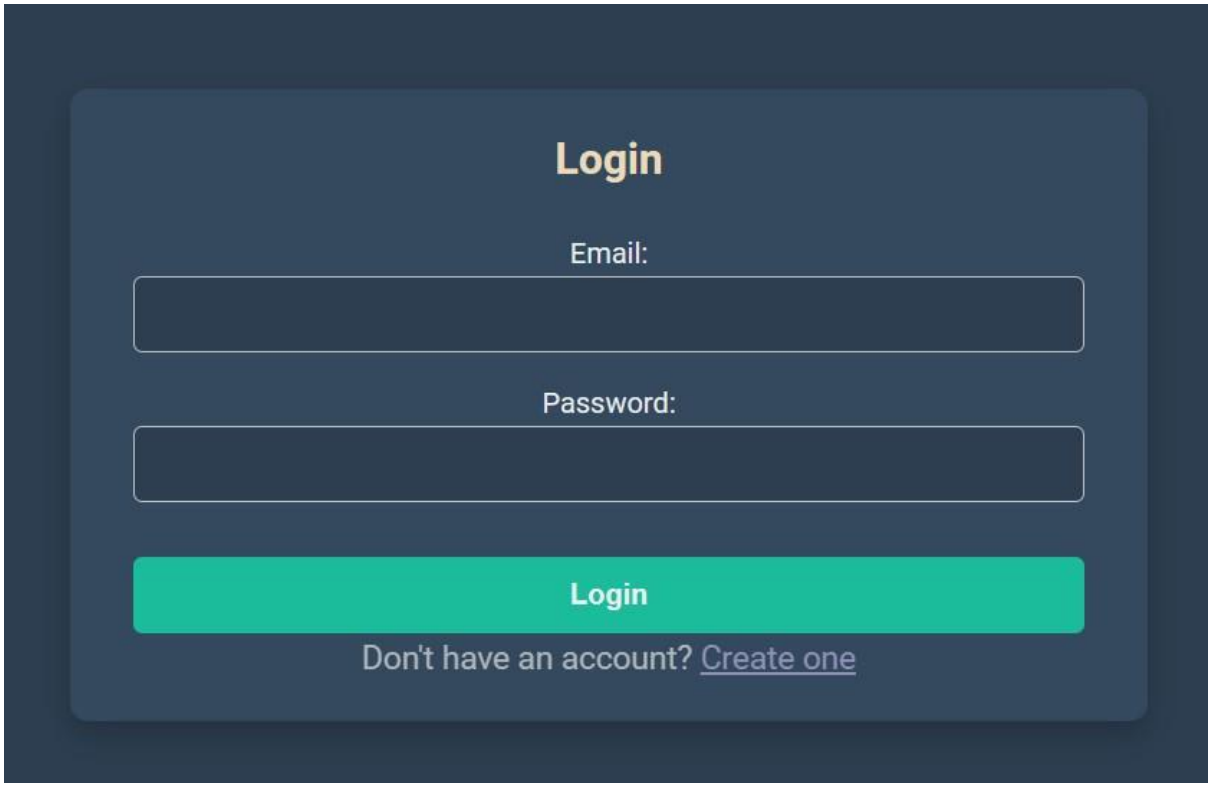
Code:

```
$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:

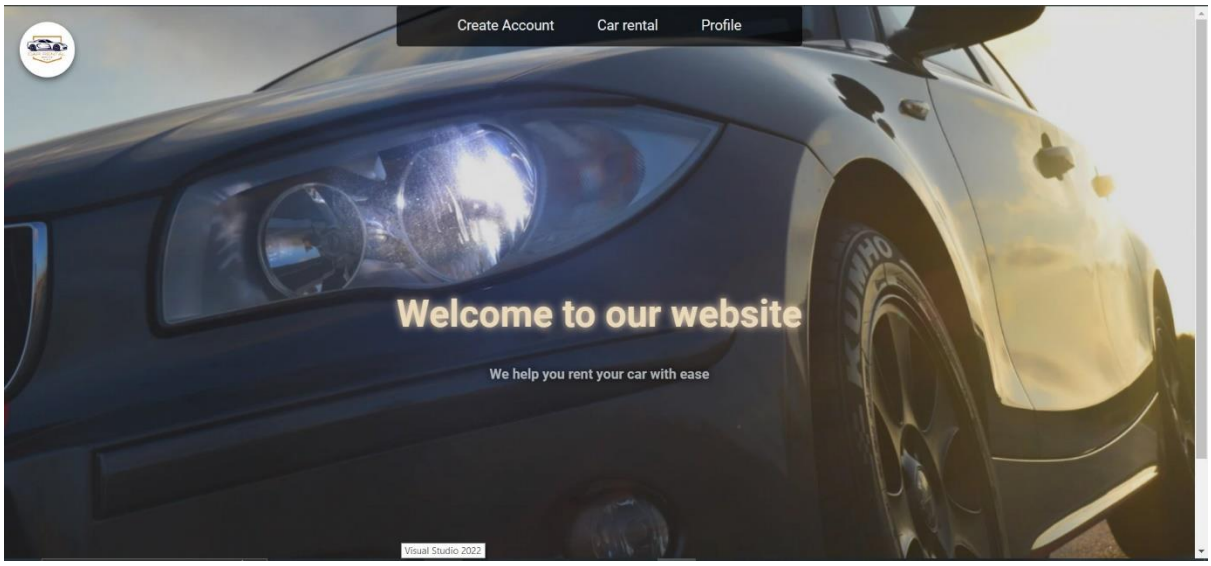
After customer create account nce clicked, you will be transferred to the login page.

"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!!!



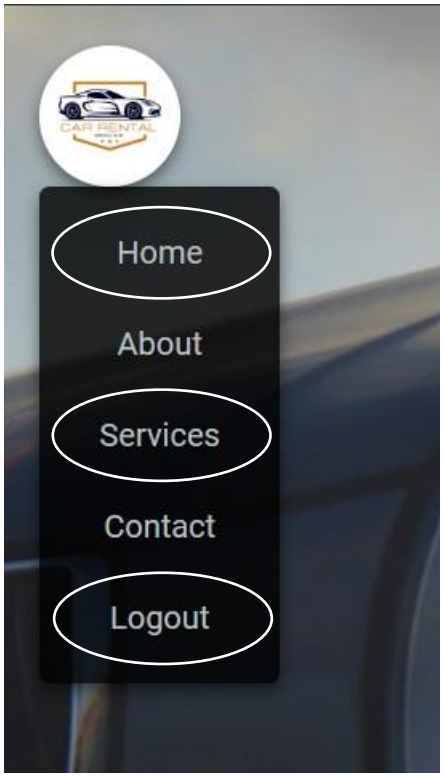
Home page:

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

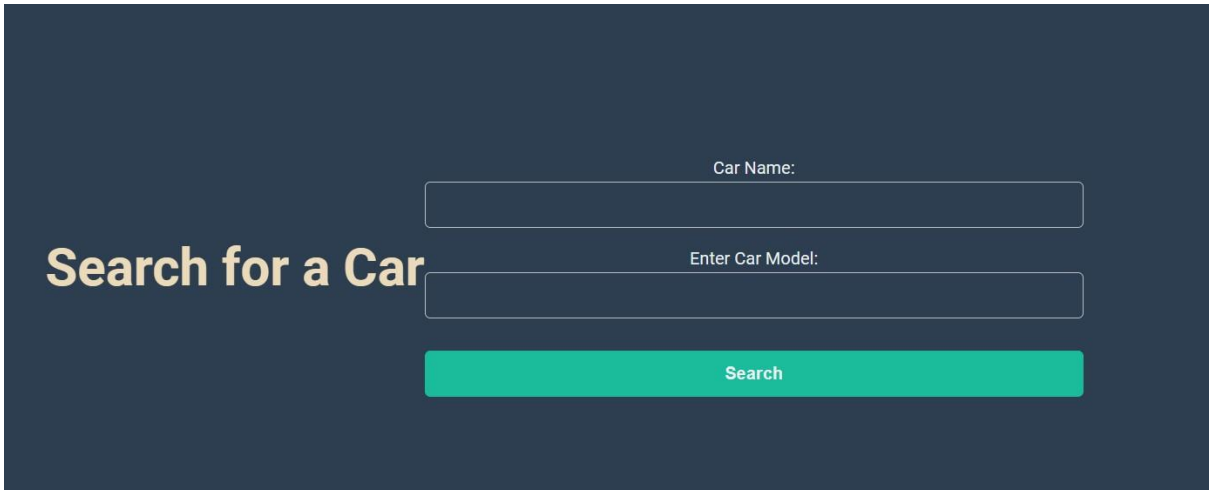
1- **The customer chooses one of the options in the box at the top (create an account, book a car, or profile**

or

2- **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 ىModel
   $sql = "SELECT * FROM Car
         WHERE LOWER(CarName) = LOWER('$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.

Searching for Car Name: bmw
Searching for Car Model: x6

Search Results:

Car Name	Model	Year	Plate ID	Price	Status
bmw	x6	2023	666	50	This car is currently rented.
bmw	x6	2024	2123	70	This car is currently rented.

Go To Home

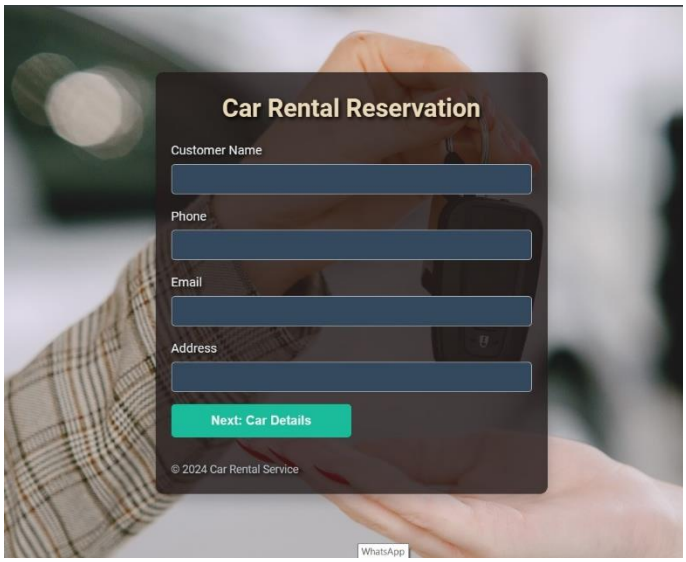
Searching for Car Name: kia
Searching for Car Model: k5gt

Search Results:

Car Name	Model	Year	Plate ID	Price	Status
kia	k5gt	2024	1111	120	The car is available for booking. Reserve Now

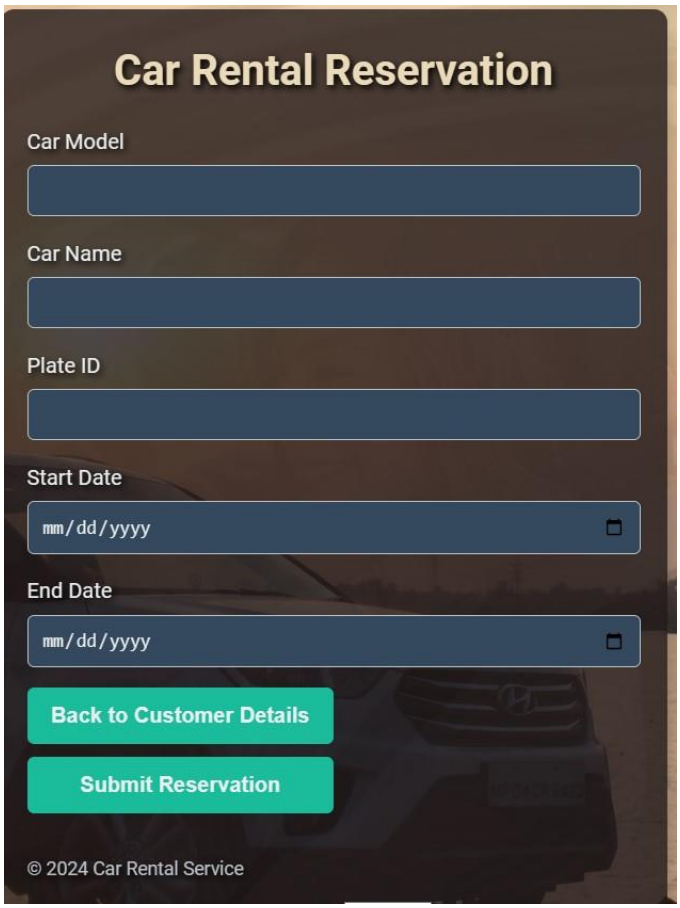
Go To Home

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.

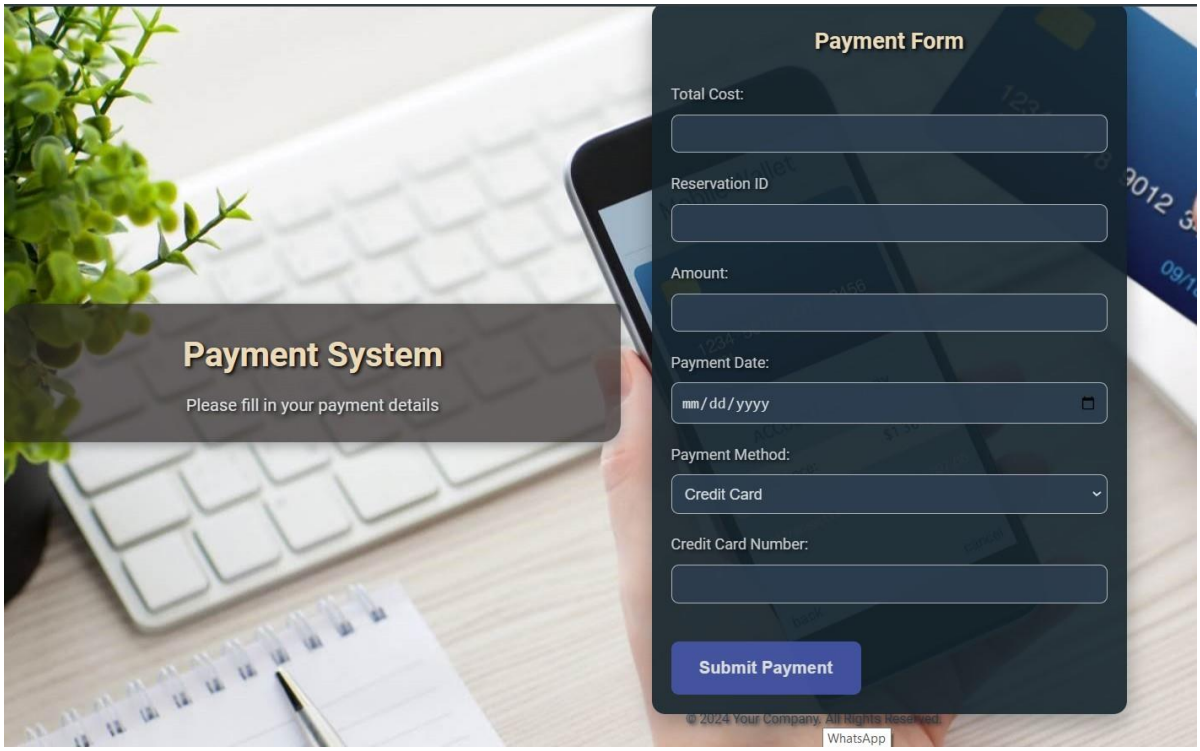
```
"SELECT CarID, Price FROM Car WHERE Model = ? AND CarName = ? AND PlateID = ? AND Status = 'active'";
```

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:

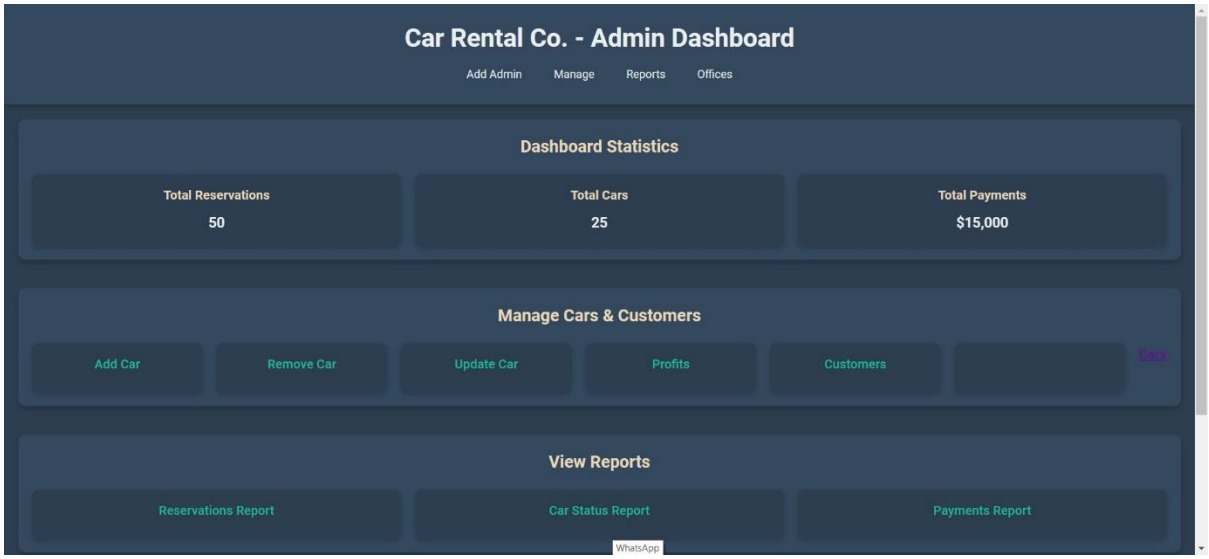
```
$query = "INSERT INTO payment (ReservationID, Amount, paymentDate, paymentMethod, CardNumber, PayPalAccount, BankAccount)  VALUES (?, ?, ?, ?, ?, ?, ?)";
```

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).

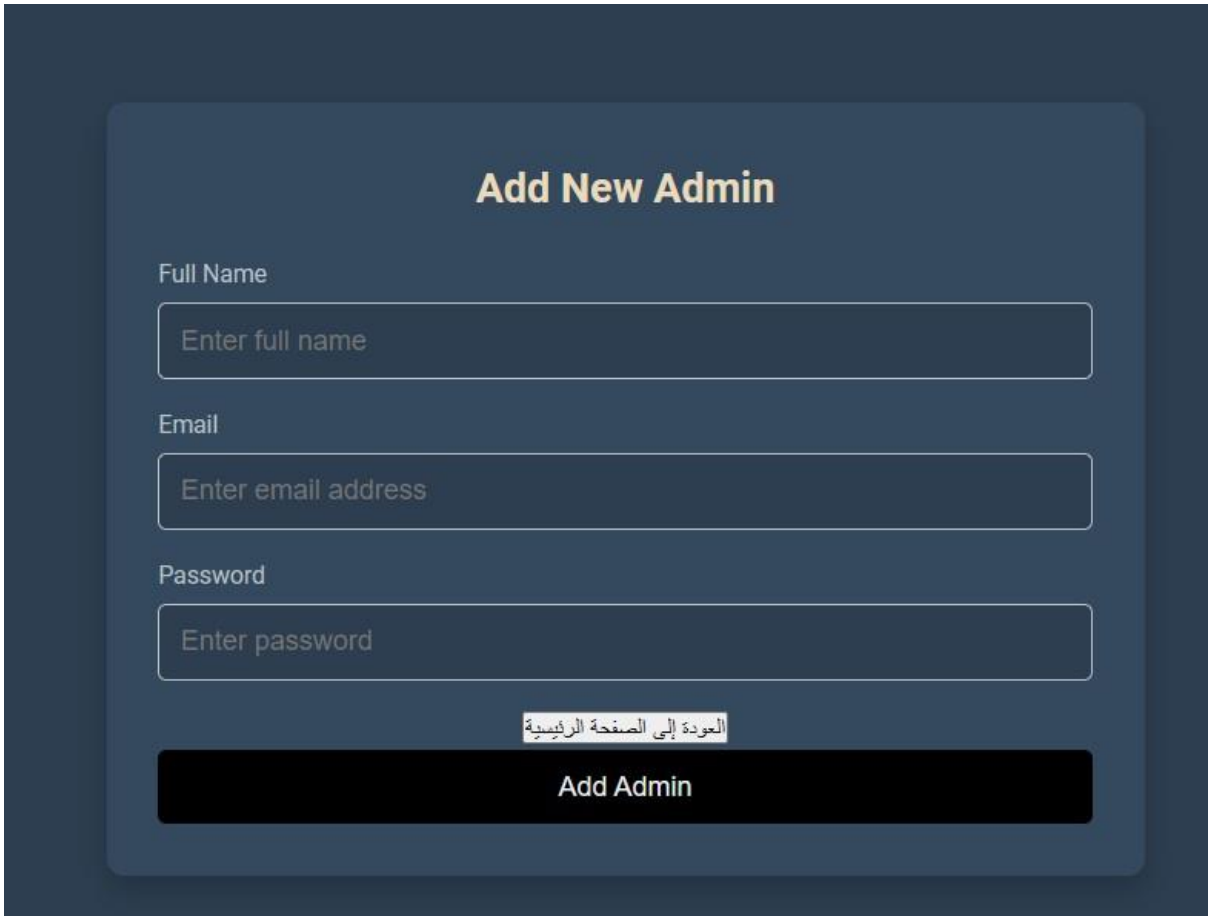
```
echo "Payment recorded successfully.";
    echo "<form action='index.html' method='POST'>
            <button type='submit'>Go To Home</button>
        </form>";
```

If this process is successfully this message will appear

Admin Dashbord:



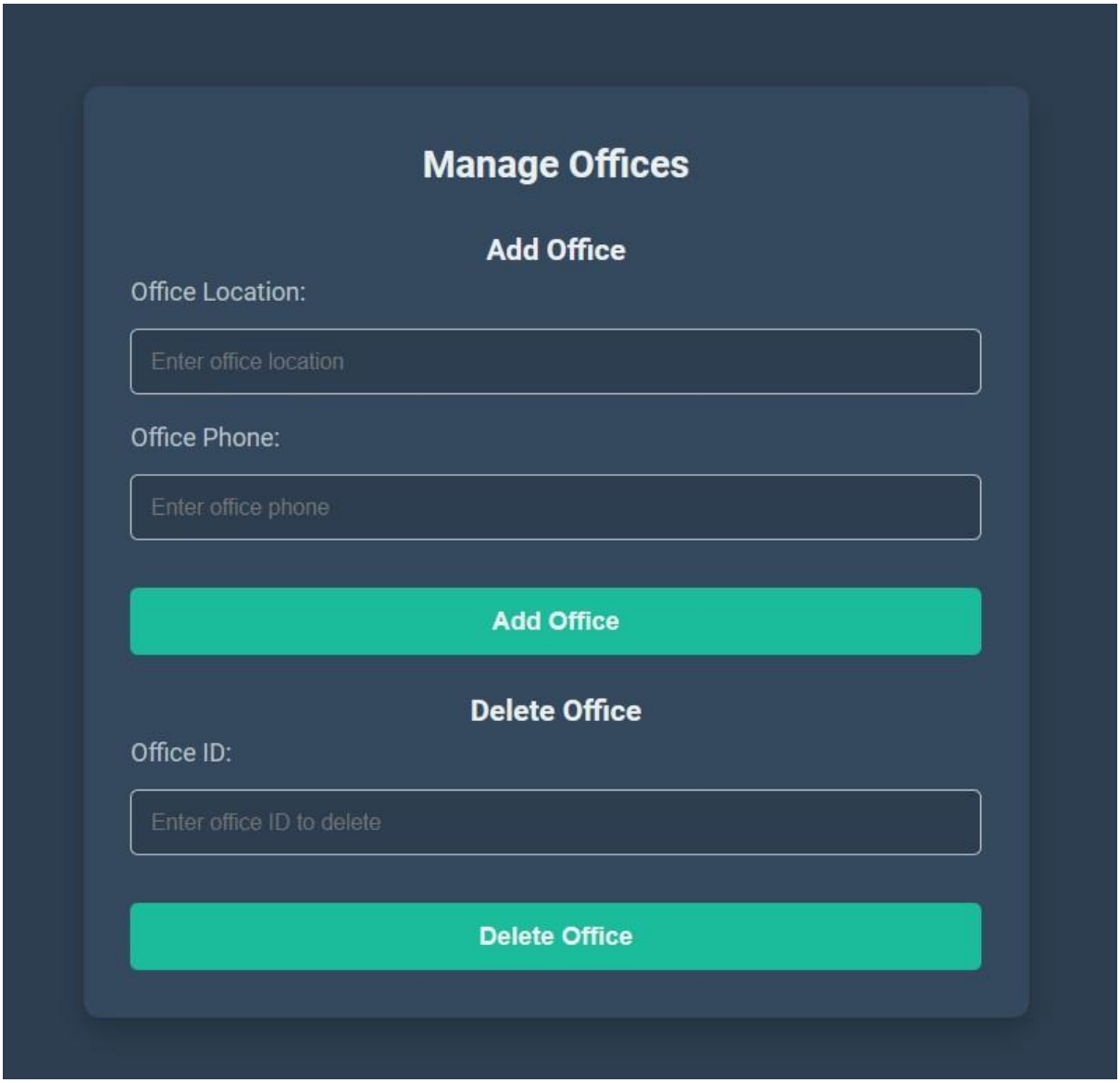
This is the admin page where you can control the site, such as adding or deleting a car, updating its status, displaying all cars or customers, deleting a customer, adding a new office, adding an admin, and another page through which you can display a report on the status of cars or reservations.



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

---

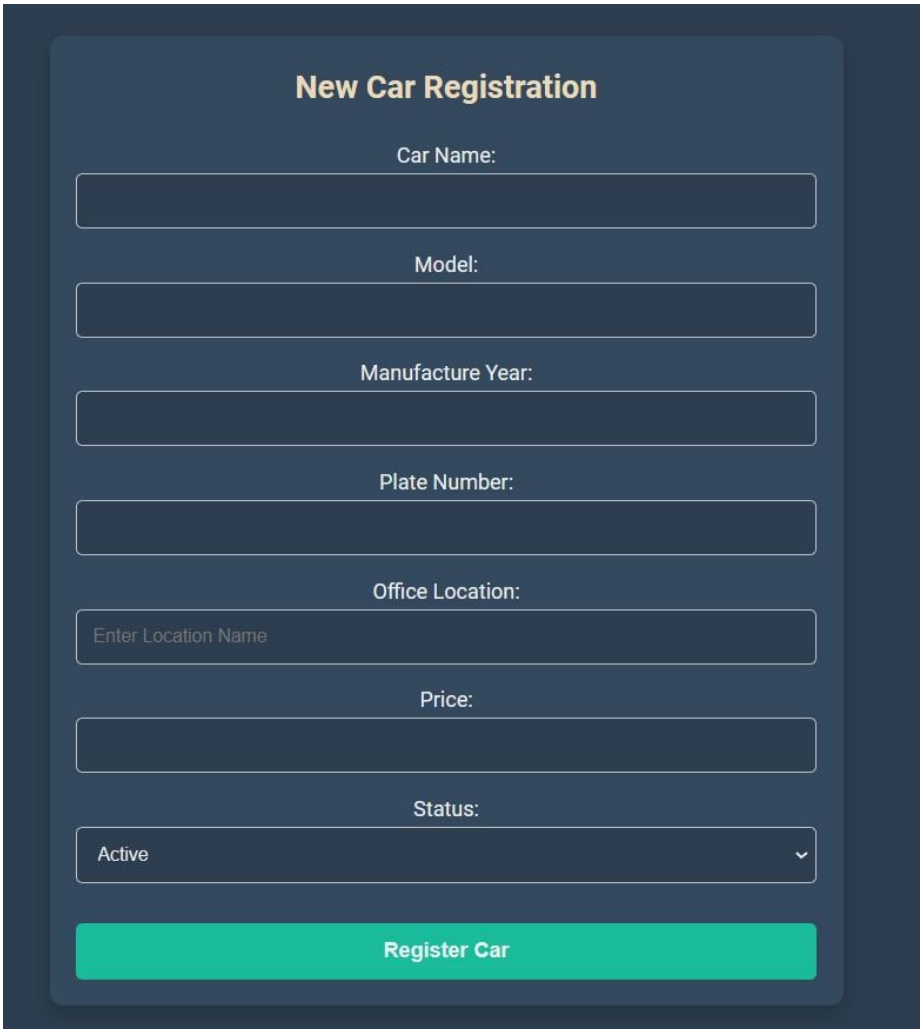
Form Add Car

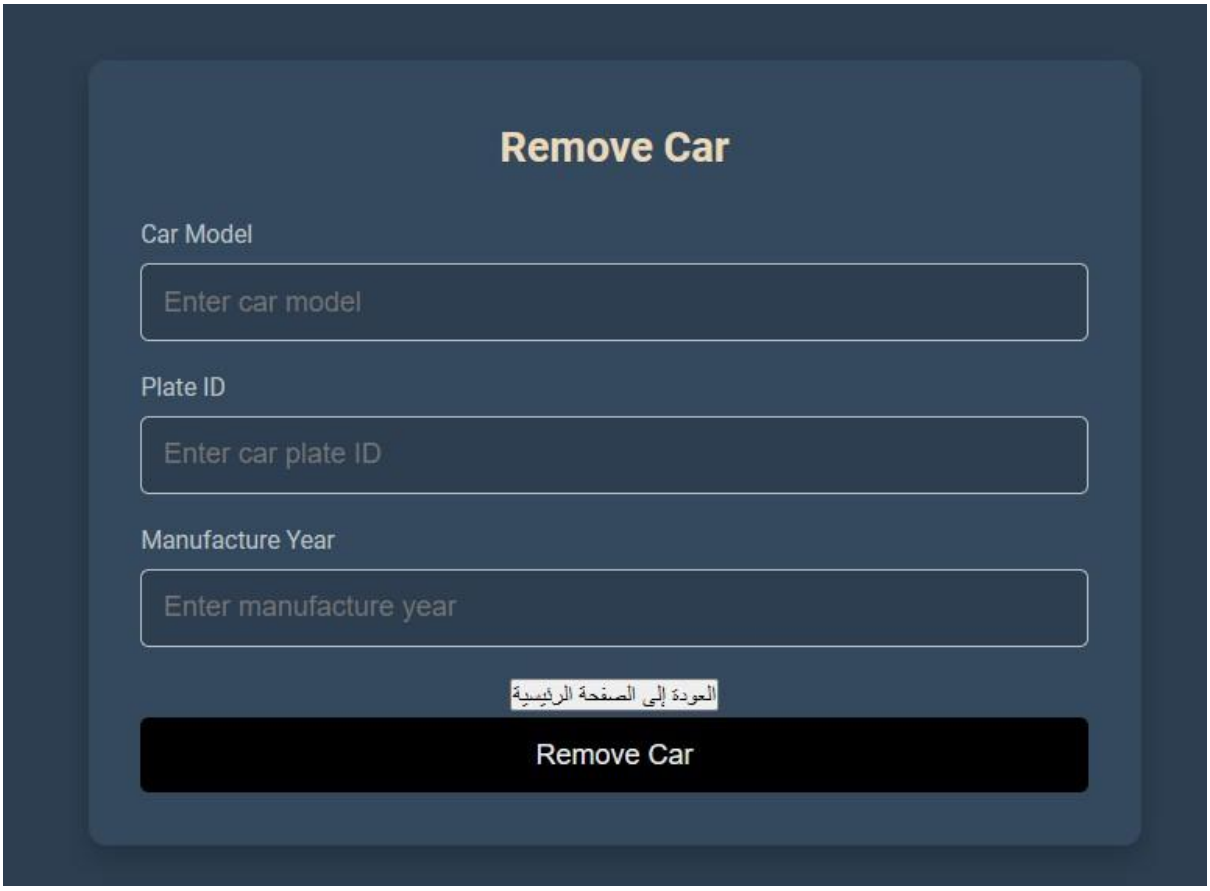
Here you can add car by filling this field.

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

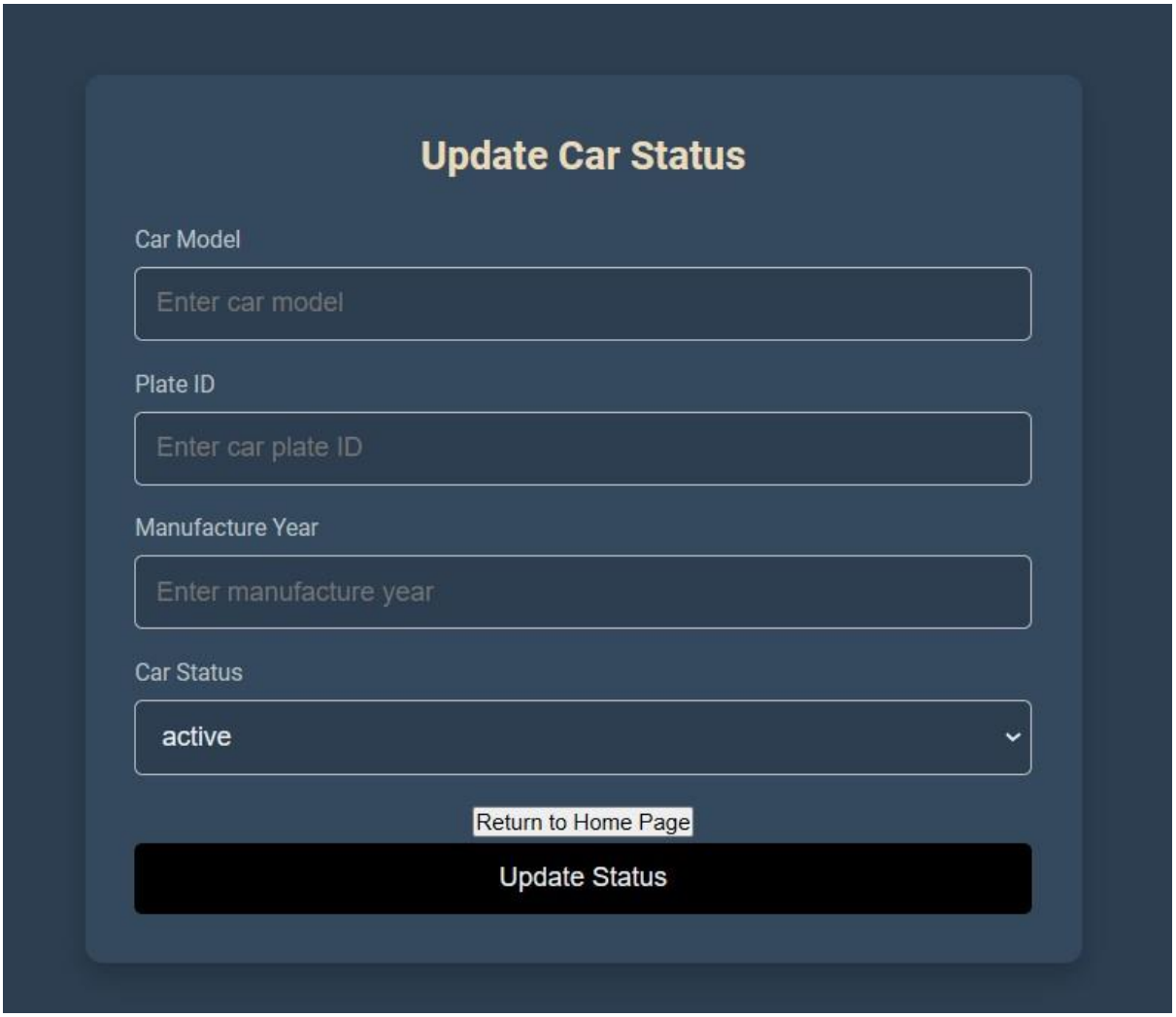
Code:

```
"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

Car Data

Car ID	Model	Car Name	Year	Plate ID	Location	Status
2	x3	bmw	2023	1236	USA	rented
3	x5	bmw	2020	222	USA	rented
4	diefinr 130	land rover	2024	909	USA	rented
5	fiat		128	2354	USA	rented
6	x6	bmw	2023	666	eygpt	rented
7	x6	bmw	2024	2123	eygpt	rented
8	s1	kai	2023	252	eygpt	rented
10	a5	audi	2023	878	eygpt	rented
11	k5gt	kia	2024	1111	USA	active

Customer Data

Customer ID	Name	Email	Phone	Address	Actions
1	abdo	abdo@gmail.com	01282444655	cairo	Delete
3	abdo	abdoy@gmail.com	0128244468	ab	Delete
4	abde	abde@gmail.com	01282444654	as	Delete
7	Abdelrahman	abdel@gmail.com	01282444653	Alex	Delete
10	Abdelrahman Mohamed	abdelr@gmail.com	01282444652	alex	Delete

	PaymentID	ReservationID	Amount	PaymentDate	PaymentMethod	CardNumber	PayPalAccount	BankAccount
Edit Copy Delete	4	10	71500.00	2024-12-29	Credit Card	528794436644		
Edit Copy Delete	5	13	36550.00	2024-12-30	Cash			
Edit Copy Delete	6	13	36550.00	2024-12-30	Cash			
Edit Copy Delete	7	13	36550.00	2024-12-30	Cash			

You can also edit most values by double-clicking directly on them.

	ReservationID	CustomerID	CarID	StartDate	EndDate	TotalCost	Status
Edit Copy Delete	8	1	6	2024-01-01	2025-01-01	36600.00	Reserved
Edit Copy Delete	10	7	8	2024-01-01	2026-01-01	73100.00	Reserved
Edit Copy Delete	11	7	3	2024-01-01	2025-01-01	36600.00	Reserved
Edit Copy Delete	12	7	7	2024-01-01	2025-01-01	25620.00	Reserved
Edit Copy Delete	13	7	10	2024-01-01	2026-01-01	36550.00	Reserved
Edit Copy Delete	14	7	4	2024-01-01	2026-01-02	73200.00	Reserved

<<TEST>>

Report Reservations:

Reservations from 2024-01-01 to 2025-01-01

Reservation ID	Start Date	End Date	Total Cost	Status	Customer Name	Customer Email	Customer Phone	Car Model	Car Year	Car Plate ID
8	2024-01-01	2025-01-01	36600.00	Reserved	abdo	abdo@gmail.com	01282444655	x6	2023	666
11	2024-01-01	2025-01-01	36600.00	Reserved	Abdelrahman	abdel@gmail.com	01282444653	x5	2020	222
12	2024-01-01	2025-01-01	25620.00	Reserved	Abdelrahman	abdel@gmail.com	01282444653	x6	2024	2123

Report Car Status:

Car Status for 2024-12-29

Car ID	Car Model	Car Year	Car Plate ID	Car Status
10	a5	2023	878	Reserved
4	diefinr 130	2024	909	Reserved
5	fiat	128	2354	Available
11	k5gt	2024	1111	Available
8	s1	2023	252	Reserved
2	x3	2023	1236	Available
3	x5	2020	222	Reserved
7	x6	2024	2123	Reserved
6	x6	2023	666	Reserved

Note in the table that cars during the specified period we see there are 6 reserved cars and there are 3 cars available

Car Status for 2026-12-29

Car ID	Car Model	Car Year	Car Plate ID	Car Status
10	a5	2023	878	Available
4	diefinr 130	2024	909	Available
5	fiat	128	2354	Available
11	k5gt	2024	1111	Available
8	s1	2023	252	Available
2	x3	2023	1236	Available
3	x5	2020	222	Available
7	x6	2024	2123	Available
6	x6	2023	666	Available

Note in the table that cars during the specified period we see that all recorded cars have become available

Report Daily Payment:

Date	Total Payments
2024-12-29	71500.00

This report displays the the toyal payments over a specified period

Report Customer:

Customer Name	Phone	Email	Car Model	Car Name	Plate ID	Start Date	End Date	Total Cost	Status
Abdelrahman	01282444653	abdel@gmail.com	s1	kai	252	2024-01-01	2026-01-01	73100.00	Reserved
Abdelrahman	01282444653	abdel@gmail.com	x5	bmw	222	2024-01-01	2025-01-01	36600.00	Reserved
Abdelrahman	01282444653	abdel@gmail.com	x6	bmw	2123	2024-01-01	2025-01-01	25620.00	Reserved
Abdelrahman	01282444653	abdel@gmail.com	a5	audi	878	2024-01-01	2026-01-01	36550.00	Reserved
Abdelrahman	01282444653	abdel@gmail.com	diefinr 130	land rover	909	2024-01-01	2026-01-02	73200.00	Reserved

This report displays a specific customer , showing all his data such as name , phon number, number of reservation and the cars he has reserved.