📝 Loan Management System - Project Documentation

📘 Problem Statement

Create SQL Schema from the `Customer` and `Loan` class, use the class attributes for table column names.

---

1. Define a `Customer` class with the following confidential attributes:

a. Customer ID
b. Name
c. Email Address
d. Phone Number
e. Address
f. Credit Score

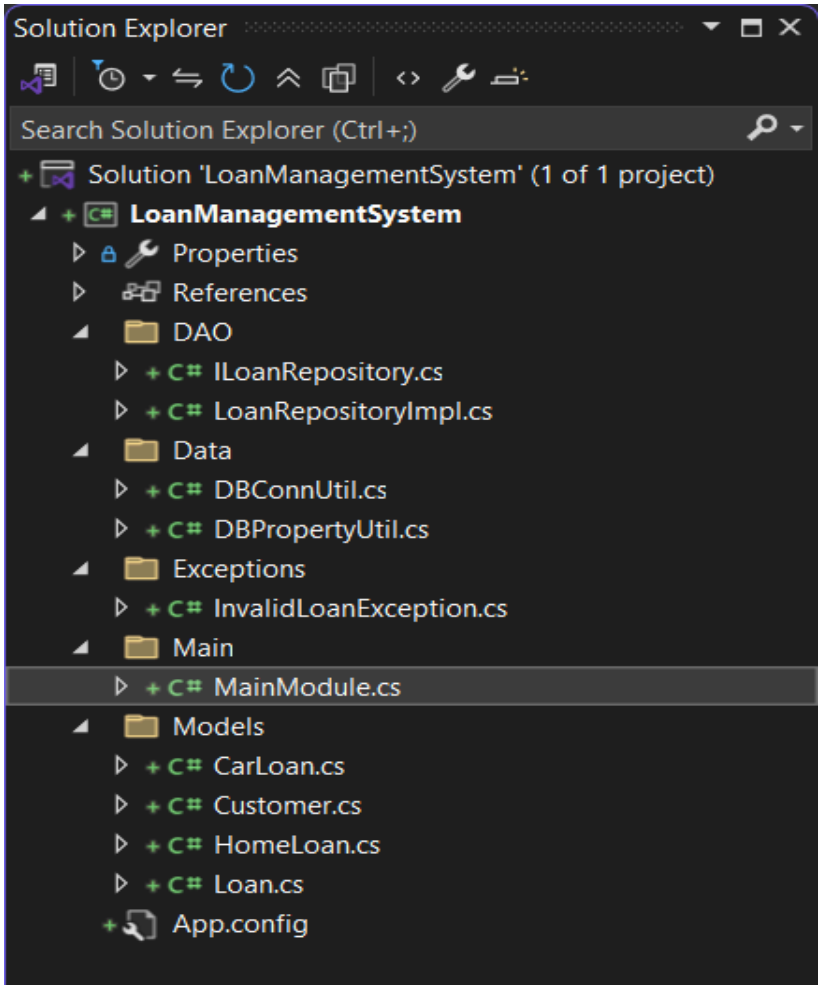
2. Define a base class `Loan` with the following attributes:

a. Loan ID
b. Customer (reference of customer class)
c. Principal Amount
d. Interest Rate
e. Loan Term (Loan Tenure in months)
f. Loan Type (`CarLoan`, `HomeLoan`)
g. Loan Status (`Pending`, `Approved`)

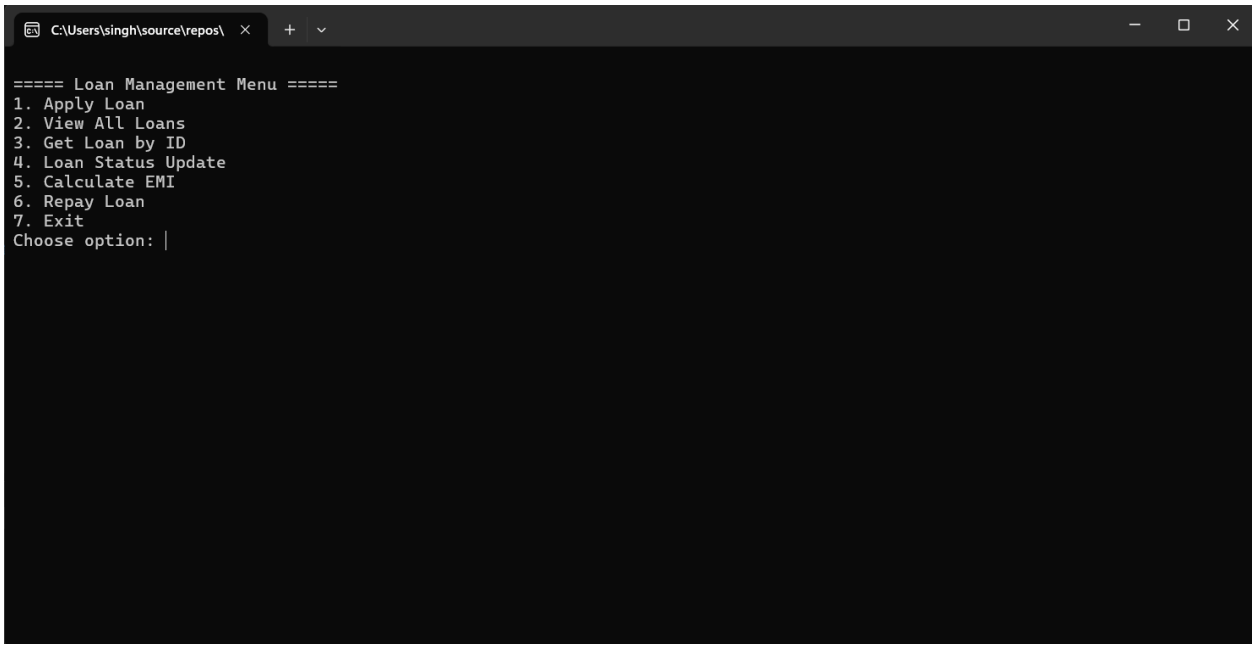
```sql
-- Create Database
CREATE DATABASE LoanDB;
GO

-- Use the database
USE LoanDB;
GO

-- Create Customer Table
CREATE TABLE Customer (
    CustomerId INT PRIMARY KEY,
    Name VARCHAR(100),
    EmailAddress VARCHAR(100),
    PhoneNumber VARCHAR(15),
    Address VARCHAR(200),
    CreditScore INT
);
GO

-- Create Base Loan Table
CREATE TABLE Loan (
    LoanId INT PRIMARY KEY,
    CustomerId INT NOT NULL,
    PrincipalAmount DECIMAL(18,2),
    InterestRate DECIMAL(5,2),
    LoanTerm INT, -- in months
    LoanType VARCHAR(20), -- 'CarLoan' or 'HomeLoan'
    LoanStatus VARCHAR(20), -- 'Pending', 'Approved', 'Rejected'
    FOREIGN KEY (CustomerId) REFERENCES Customer(CustomerId)
);
GO

-- Create CarLoan Details Table (inherits LoanId)
CREATE TABLE CarLoanDetails (
    LoanId INT PRIMARY KEY,
    CarModel VARCHAR(100),
    CarValue DECIMAL(18,2),
    FOREIGN KEY (LoanId) REFERENCES Loan(LoanId) ON DELETE CASCADE
);
GO
```

```sql
-- Create HomeLoan Details Table (inherits LoanId)
CREATE TABLE HomeLoanDetails (
    LoanId INT PRIMARY KEY,
    PropertyAddress VARCHAR(200),
    PropertyValue DECIMAL(18,2),
    FOREIGN KEY (LoanId) REFERENCES Loan(LoanId) ON DELETE CASCADE
);
GO
```

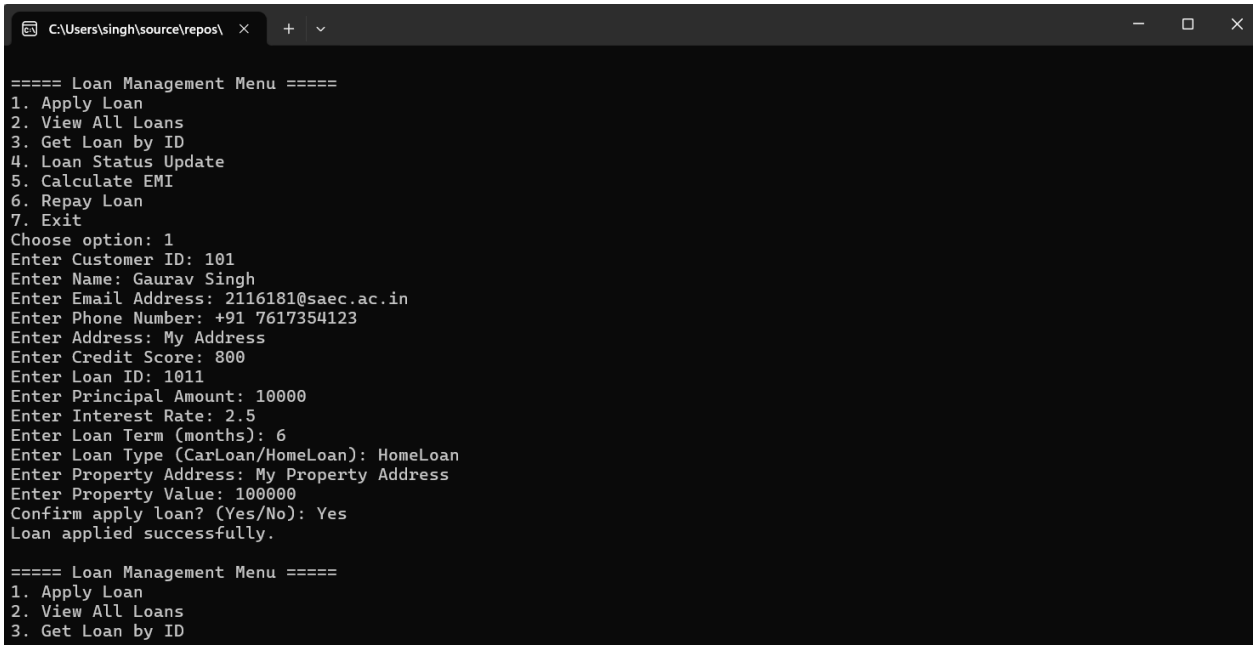
🗂️ Project Structure:



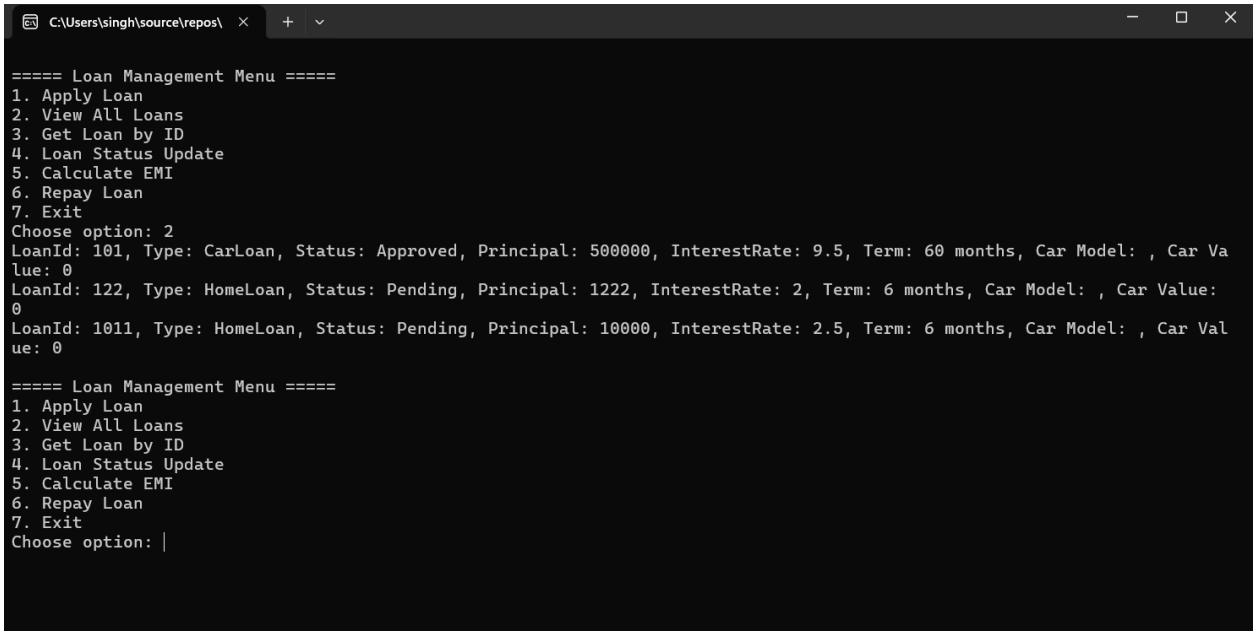
1. ✅ Main menu displayed in the console



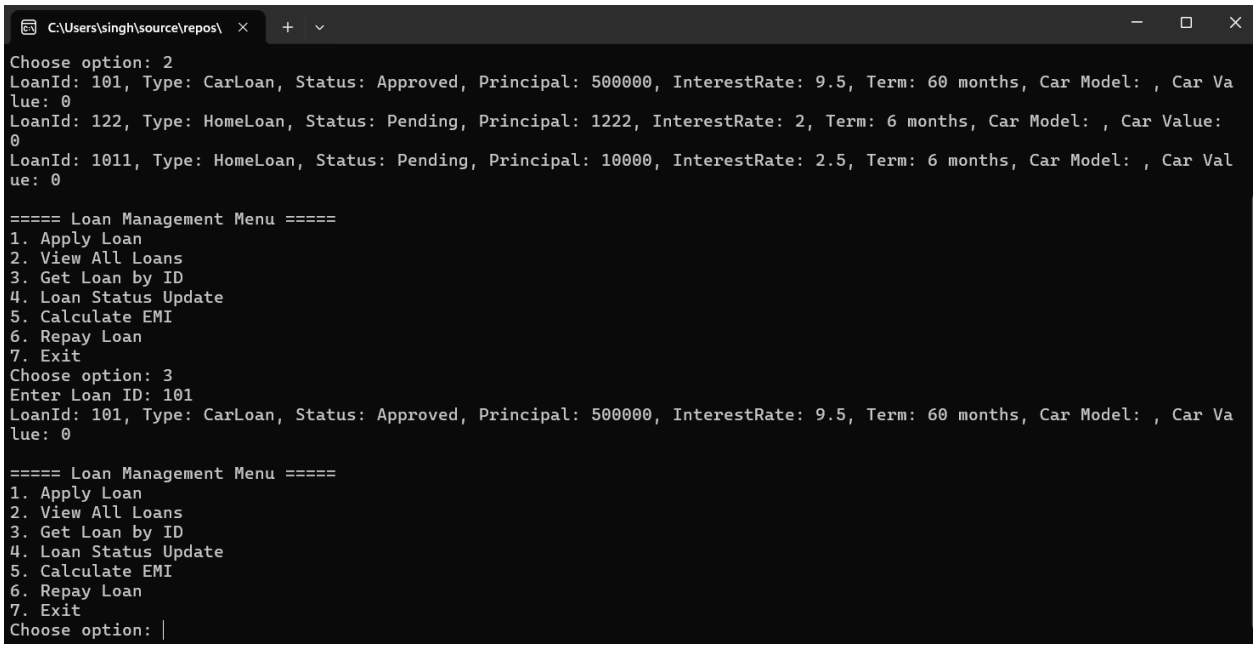
2. 📄 Successful loan application confirmation



3. 📋 View All Loans



4. 🔎 Get Loan by ID



5. 📊 Loan Status Update

```
===== Loan Management Menu =====
1. Apply Loan
2. View All Loans
3. Get Loan by ID
4. Loan Status Update
5. Calculate EMI
6. Repay Loan
7. Exit
Choose option: 4
Enter Loan ID to check status: 101
Loan status updated to: Approved
```

6. 💰 Calculate EMI

```
===== Loan Management Menu =====
1. Apply Loan
2. View All Loans
3. Get Loan by ID
4. Loan Status Update
5. Calculate EMI
6. Repay Loan
7. Exit
Choose option: 5
Enter Loan ID to calculate EMI: 101
EMI: 10500.93
```

7. 💳 Repay Loan

```
===== Loan Management Menu =====
1. Apply Loan
2. View All Loans
3. Get Loan by ID
4. Loan Status Update
5. Calculate EMI
6. Repay Loan
7. Exit
Choose option: 6
Enter Loan ID: 101
Enter amount to repay: 12000
Payment accepted. You can pay 1 EMI(s) of amount 10500.93 each.
```

🔗 Project Source Code

The complete source code for this project, including all implementation files, database utilities, and configuration, is available on GitHub.

GitHub Repository:
https://github.com/gauravsinghoffl/loan-management-system

Please refer to the repository for the full coding implementation, including project structure, business logic, and database interaction.