**Loan Application Management**

**Problem Statement**: XYZ Finance Company requires an application where customer can apply for loan. The Application’s first screen will be as follows:

XYZ Finance Company welcomes you

1. Register Customer
2. Exit

When customer selects first option application displays the following menu:

1. Register Customer

**Register Customer: The** menu option then application asks to enter the Customer details as:

**Enter Customer Name**

John

**Enter Address**

Bangalore

**Enter Email**

John@capgemini.com

**Customer ID should be generated by Math.random () method.**

After adding the customer info into customerEntry HashMap application should display success message as follows.

Customer information saved successfully.

Your Customer Id is <customer id>

After registering the Customer successfully below message should be populated

Do you wish to apply for Loan? (Yes/No)

If user enters yes then read the below details:

**Enter the loan amount**

400000

**Enter the loan duration**

5 years

**Display Details of CustomerEntry Map if user enters no**

After accepting amount and duration, it will calculate the EMI as follows.

Fix Interest rate is 9.5% for all loans.

**EMI per month = P \* r \*(1 + r) n/ ((1 + r) n - 1)**

Where P is loan amount and n is number of months.

After calculating the EMI it will display the message as follows:

For loan amount <loan amount> and <no of years> Years duration.

You EMI per month will be <calculated EMI>

Do you want to apply for loan now? (Yes/No)

If user enters yes then loan record should be added in loanEntry HashMap where loan ID will be key and Loan object will be key and it will display the success message as follows:

Your Loan request is generated.

Your Loan ID is <loan ID>

***Display Details of customerEntry and loanEntry Map after this.***

* Write test cases for all methods of DAO class for valid and invalid input values.

**Classes to be created:**

**com.capgemini.xyz.ui**

public class ExecuterMain {

public static void main (String[] args) {

// User Interface which display the menu and accept the Input from user

// Create object for service and execute the respective methods

// Menu for loan and exit

}

**com.capgemini. xyz.bean**

public class Customer {

private long custId;

private String custName;

private String address;

private long mobile;

private String email;

//Getter and Setter methods for all above data members

}

public class Loan{

private long loanID;

private double loanAmount;

private long custId;

private int duration;

//Getter and Setter methods for all above data members

}

**com.capgemini. xyz.service** // Service utility classes and Interfaces

public interface ILoanService {

public loan applyLoan (Loan loan);

public Customer validateCustomer(Customer customer);

public long insertCust(Customer cust);

public double calculateEMI(double amount,int duration);

}

public class LoanService implements ILoanService {

public long applyLoan (Loan loan){

//pass loan to dao and invoke applyLoan from dao }

Customer validateCustomer(Customer customer){

//use regex to validate name and address

}

public long insertCust(Customer cust){

//pass customer object to dao to invoke insertCustomer method from dao

}

public double calculateEMI(double amount, int duration){

//pass the amount and duration and calculate the EMI

}}

**com.capgemini. xyz.dao** // Class to perform Data Accessing logic

public interface ILoanDao {

private Map<Integer,Customer> customerEntry;

private Map<Integer,Loan> loanEntry;

public long applyLoan (Loan loan);

public long insertCust(Customer cust);

}

public class LoanDao implements ILoanDao {

public long applyLoan (Loan loan){ //insertion of loan info in database }

long insertCust(Customer cust){

//insert customer info in the database

}}

Add appropriate user defined exception classes and any other supporting classes required.

Also, if extra methods are added to the given classes, please justify in comments what is the purpose of those methods