SOURCE CODE:

// Borrower.java

package Project;

import java.util.ArrayList;

import java.util.List;

public class Borrower { private String borrowerId; private String name; private String region; private double creditScore;

private List<Loan> activeLoans;

public Borrower(String borrowerId, String name, String region, double creditScore) { this.borrowerId = borrowerId; this.name = name; this.region = region; this.creditScore = creditScore;

this.activeLoans = new ArrayList<>();

}

public void addLoan(Loan loan) { activeLoans.add(loan);

}

public List<Loan> getActiveLoans() {

return activeLoans;

}

public String getName() {

return name;

}

public String getBorrowerId() {

return borrowerId;

}

public String getRegion() {

return region;

}

public double getCreditScore() {

return creditScore;

}

public void setCreditScore(double creditScore) {

this.creditScore = creditScore;

}

}

// Lender.java

package Project;

import java.util.ArrayList;

import java.util.List;

public class Lender { private String lenderId; private String name; private double walletBalance;

private List<Loan> portfolio;

public Lender(String lenderId, String name, double walletBalance) { this.lenderId = lenderId; this.name = name; this.walletBalance = walletBalance;

this.portfolio = new ArrayList<>();

}

public void fundLoan(Loan loan, double amount) {

if (walletBalance >= amount) { walletBalance -= amount; loan.fund(amount);

portfolio.add(loan);

}

}

public double getWalletBalance() { return walletBalance; } public String getName() { return name; }

}

//Loan.java

package Project;

public abstract class Loan { protected String loanId; protected double principal; protected double interestRate; protected int tenure; // in months protected double fundedAmount; protected String status; // e.g., Pending, Funded, Repaying, Paid

public Loan(String loanId, double principal, double interestRate, int tenure) { this.loanId = loanId; this.principal = principal; this.interestRate = interestRate; this.tenure = tenure; this.fundedAmount = 0;

this.status = "Pending";

}

public void fund(double amount) { fundedAmount += amount; if (fundedAmount >= principal) {

status = "Funded";

}

}

public abstract double loanSchedule();

public void repay(double amount) { fundedAmount -= amount; if (fundedAmount <= 0) { status = "Paid";

}

}

public String getStatus() { return status; }

}

// EducationLoan.java

package Project;

public class EducationLoan extends Loan { public EducationLoan(String loanId, double principal, double interestRate, int tenure) { super(loanId, principal, interestRate, tenure);

}

@Override public double loanSchedule() { // Specific logic for education loans

return (principal \* (1 + (interestRate/100))) / tenure;

}

}

// BusinessLoan.java

package Project;

public class BusinessLoan extends Loan { public BusinessLoan(String loanId, double principal, double interestRate, int tenure) { super(loanId, principal, interestRate, tenure);

}

@Override public double loanSchedule() { // Specific logic for business loans

return (principal \* (1 + (interestRate/100))) / tenure;

}

}

// LendingService.java

package Project;

import java.util.ArrayList;

import java.util.List;

public class LendingService { private List<Lender> lenders; private List<Borrower> borrowers; private List<Loan> loans;

public LendingService() { lenders = new ArrayList<>(); borrowers = new ArrayList<>();

loans = new ArrayList<>();

}

public void registerLender(Lender lender) {

lenders.add(lender);

}

public void registerBorrower(Borrower borrower) { borrowers.add(borrower);

}

public void listLoan(Loan loan) {

loans.add(loan);

}

public void fundLoan(Lender lender, Loan loan, double amount) { lender.fundLoan(loan, amount);

}

public void disburseFunds(Borrower borrower, Loan loan) { borrower.addLoan(loan);

}

public void repayLoan(Loan loan, double amount) { loan.repay(amount);

}

public void showLoanSchedules() { for (Loan loan : loans) {

System.out.println("Loan ID: " + loan.loanId + ", Monthly Payment: " + loan.loanSchedule());

}

}

}

// LendingAppMain.java

package Project;

public class LendingAppMain {

public static void main(String[] args) {

System.out.println("=== Micro-Lending Console Application ===\n");

LendingService service = new LendingService();

// Register Lender and Borrower

Lender lender = new Lender("L001", "Kavin prasanth", 50000);

Borrower borrower = new Borrower("B001", "Kishore", "RegionX", 750); service.registerLender(lender);

service.registerBorrower(borrower);

System.out.println("Lender: " + lender.getName() + ", Wallet: ₹" + lender.getWalletBalance()); System.out.println("Borrower: " + borrower.getName() + " from " + borrower.getRegion());

// Create Loans

EducationLoan eduLoan = new EducationLoan("EL001", 20000, 5.0, 24); BusinessLoan busLoan = new BusinessLoan("BL001", 50000, 7.0, 36); service.listLoan(eduLoan);

service.listLoan(busLoan);

// Fund Loans service.fundLoan(lender, eduLoan, 20000); service.fundLoan(lender, busLoan, 50000); service.disburseFunds(borrower, eduLoan); service.disburseFunds(borrower, busLoan);

System.out.println("\n--- Loan Schedules ---"); service.showLoanSchedules();

// Simulate Repayments service.repayLoan(eduLoan, 1000);

service.repayLoan(busLoan, 5000);

System.out.println("\n--- Updated Loan Statuses ---");

System.out.println("Education Loan EL001 Status: " + eduLoan.getStatus()); System.out.println("Business Loan BL001 Status: " + busLoan.getStatus());

System.out.println("\nRemaining Lender Balance: ₹" + lender.getWalletBalance()); System.out.println("\n=== Process Complete ===");

}

}