

OOPJ

Assignment-3

Note:

- The assignment is designed to practice class, fields, and methods only.
- Create a separate project for each question.
- Do not use getter/setter methods or constructors for these assignments.
- Define two classes: one class to implement the logic and another class to test it.

1. Loan Amortization Calculator

Implement a system to calculate and display the monthly payments for a mortgage loan. The system should:

1. Accept the principal amount (loan amount), annual interest rate, and loan term (in years) from the user.
2. Calculate the monthly payment using the standard mortgage formula:

◦ Monthly Payment Calculation:

$$\$ \text{monthlyPayment} = \text{principal} * (\text{monthlyInterestRate} * (1 + \text{monthlyInterestRate})^{\text{numberOfMonths}}) / ((1 + \text{monthlyInterestRate})^{\text{numberOfMonths}} - 1)$$

$\$$ Where $\text{monthlyInterestRate} = \text{annualInterestRate} / 12 / 100$ and $\text{numberOfMonths} = \text{loanTerm} * 12$

$\$$ Note: Here ^ means power and to find it you can use Math.pow() method

3. Display the monthly payment and the total amount paid over the life of the loan, in Indian Rupees (₹).

Define class LoanAmortizationCalculator with methods acceptRecord, calculateMonthlyPayment & printRecord and test the functionality in main method.

```
package com.practice.q1;  
import java.util.Scanner;
```

```
class LoanAmortizationCalculator {  
  
    private double principal;  
    private double annualInterestRate;  
    private double loanTerm;  
    private double monthlyInterestRate;  
    private double numberOfMonths;  
    private double monthlyPayment;  
    private double totalAmount;
```

```

    public void acceptRecord() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Principal: ");
        this.principal = sc.nextDouble();
        System.out.print("Enter Interest Rate: ");
        this.annualInterestRate = sc.nextDouble();
        System.out.print("Enter Loan Term: ");
        this.loanTerm = sc.nextDouble();
    }

    public double calculateMonthlyPayment() {
        monthlyInterestRate = annualInterestRate / 12 / 100;
        numberOfMonths = loanTerm * 12;
        monthlyPayment = principal * (monthlyInterestRate * Math.pow((1 + monthlyInterestRate),
(numberOfMonths))) / (Math.pow((1 + monthlyInterestRate), (numberOfMonths) - 1);
        return monthlyPayment;
    }

    public double calculateTotalAmount() {
        totalAmount = calculateMonthlyPayment() * loanTerm * 12;
        return totalAmount;
    }

    public void printRecord() {
        System.out.printf("Monthly Payment: %.2f%n", monthlyPayment);
        System.out.printf("Total Amount Paid: %.2f%n", totalAmount);
    }
}

public class Program {
    public static void main(String[] args) {

        LoanAmortizationCalculator lac = new LoanAmortizationCalculator();

        lac.acceptRecord();
        lac.calculateMonthlyPayment();
        lac.calculateTotalAmount();
        lac.printRecord();
    }
}

```

Output

```

Enter Principal: 1000000
Enter Interest Rate: 10
Enter Loan Term: 10
Monthly Payment: 13215.07
Total Amount Paid: 1585808.84

```

2. Compound Interest Calculator for Investment

Develop a system to compute the future value of an investment with compound interest. The system should:

1. Accept the initial investment amount, annual interest rate, number of times the interest is compounded per year, and investment duration (in years) from the user.
2. Calculate the future value of the investment using the formula:
 - o **Future Value Calculation:**

$$\text{futureValue} = \text{principal} * (1 + \text{annualInterestRate} / \text{numberOfCompounds}) ^ (\text{numberOfCompounds} * \text{years})$$
 - o **Total Interest Earned:** $\text{totalInterest} = \text{futureValue} - \text{principal}$
3. Display the future value and the total interest earned, in Indian Rupees (₹).

Define class CompoundInterestCalculator with methods acceptRecord , calculateFutureValue, printRecord and test the functionality in main method.

```
package com.practice.q2;
import java.util.Scanner;
```

```
class CompoundInterestCalculator {
    private double principal;
    private double annualInterestRate;
    private double numberOfCompounds;
    private double years;
    private double futureValue;
    private double totalInterest;

    public void acceptRecord() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Principal: ");
        this.principal = sc.nextDouble();
        System.out.print("Enter Interest Rate: ");
        this.annualInterestRate = sc.nextDouble();
        System.out.print("Enter number of compounds: ");
        this.numberOfCompounds = sc.nextDouble();
        System.out.print("Enter duration: ");
        this.years = sc.nextDouble();
    }

    public double calculateFutureValue() {
        futureValue = principal * Math.pow((1 + annualInterestRate / numberOfCompounds / 100),
        (numberOfCompounds * years));
        return futureValue;
    }

    public double calculateTotalInterest() {
        totalInterest = futureValue - principal;
        return totalInterest;
    }

    public void printRecord() {
        System.out.printf("Future Value: %.2f%n", futureValue);
        System.out.printf("Total Interest: %.2f%n", totalInterest);
    }
}

public class Program {
```

```
public static void main(String[] args) {  
    CompoundInterestCalculator cic = new CompoundInterestCalculator();  
  
    cic.acceptRecord();  
    cic.calculateFutureValue();  
    cic.calculateTotalInterest();  
    cic.printRecord();  
}  
}
```

Output

```
Enter Principal: 100000  
Enter Interest Rate: 8  
Enter number of compounds: 2  
Enter duration: 3  
Future Value: 126531.90  
Total Interest: 26531.90
```