**package** com.cdac;

**import** java.util.Scanner;

**class** Loan {

**private** **int** principle\_amount;

**private** **float** annual\_interest\_rate;

**private** **int** loan\_term;

**double** monthly\_Interest\_Rate;

**double** no\_Of\_Months;

**double** monthly\_Payment;

**public** **void** acceptRecord() {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the principle amount (loan amount):");

principle\_amount = sc.nextInt();

System.***out***.println("Enter the annual interset rate:");

annual\_interest\_rate = sc.nextFloat();

System.***out***.println("Enter the loan term in years :");

loan\_term = sc.nextInt();

sc.close();

}

**public** **void** calculateMonthlyPayment() {

monthly\_Interest\_Rate = annual\_interest\_rate / 12 / 100;

no\_Of\_Months = loan\_term \* 12;

monthly\_Payment = principle\_amount \* (monthly\_Interest\_Rate \* Math.*pow*(1 + monthly\_Interest\_Rate, no\_Of\_Months)

/ (Math.*pow*(1 + monthly\_Interest\_Rate, no\_Of\_Months) - 1));

}

**public** **void** printRecord() {

// double monthly\_payment\_value=calculateMonthlyPayment();

System.***out***.println("monthly payment is:" + **this**.monthly\_Payment);

System.***out***.println("total amount paid is:" + **this**.monthly\_Payment \* 12 \* loan\_term);

}

}

**public** **class** LoanAmortizationCalculator {

**public** **static** **void** main(String[] args) {

Loan loan = **new** Loan();

loan.acceptRecord();

loan.calculateMonthlyPayment();

loan.printRecord();

}

}

OUTPUT:

Enter the principle amount (loan amount):

10000

Enter the annual interset rate:

12

Enter the loan term in years :

5

monthly payment is:222.44447549360956

total amount paid is:13346.668529616572

**package** com.cdac;

**import** java.util.Scanner;

/\*1. Accept the initial investment amount, a

\* nnual 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:

♣ futureValue = principal \* (1 + annualInterestRate

/ numberOfCompounds)^(numberOfCompounds \* years)

o Total Interest Earned: totalInterest = futureValue - 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.\*/

**class** CompoundInterest {

**private** **int** initial\_investment\_amount;

**private** **float** annual\_interest\_rate;

**private** **float** no\_of\_times\_interest\_componded;

**private** **int** investment\_duration;

**private** **double** futureValue;

**private** **double** Total\_Interest\_Earned;

**public** **void** acceptRecord()

{

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter the initial amount:");

**this**.initial\_investment\_amount=sc.nextInt();

System.***out***.println("Enter the annual rate:");

**this**.annual\_interest\_rate=sc.nextFloat();

System.***out***.println("Enter the no of times interest compounded:");

**this**.no\_of\_times\_interest\_componded=sc.nextFloat();

System.***out***.println("Enter the invetment duration:");

**this**.investment\_duration=sc.nextInt();

sc.close();

}

**public** **void** calculateFutureValue() {

**this**.futureValue = initial\_investment\_amount \* Math.*pow*(

1 + annual\_interest\_rate / no\_of\_times\_interest\_componded,

no\_of\_times\_interest\_componded \* investment\_duration);

**this**.Total\_Interest\_Earned= futureValue - initial\_investment\_amount;

}

**public** **void** printRecord() {

System.***out***.println("future value of the investment:" + **this**.futureValue);

System.***out***.println("total interest earned is:" + **this**.Total\_Interest\_Earned);

}

}

**public** **class** CompoundInterestCalculator {

**public** **static** **void** main(String[] args) {

CompoundInterest ci=**new** CompoundInterest();

ci.acceptRecord();

ci.calculateFutureValue();

ci.printRecord();

}

}

OUTPUT:

**package** com.cdac;

**import** java.util.Scanner;

/\* Create a system to calculate and classify Body Mass Index (BMI). The system should:

1. Accept weight (in kilograms) and height (in meters) from the user.

2. Calculate the BMI using the formula:

o BMI Calculation: BMI = weight / (height \* height)

3. Classify the BMI into one of the following categories:

o Underweight: BMI < 18.5

o Normal weight: 18.5 ≤ BMI < 24.9

o Overweight: 25 ≤ BMI < 29.9

o Obese: BMI ≥ 30

4. Display the BMI value and its classification.

Define class BMITracker with methods acceptRecord, calculateBMI, classifyBMI & printRecord and test the functionality in main method.

\*/

**class** BMITracker {

**private** **float** weight;

**private** **float** height;

**private** **float** calculated\_Bmi;

**void** acceptRecord() {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the weight in kg:");

weight = sc.nextFloat();

System.***out***.println("Enter the height in meters:");

height = sc.nextFloat();

sc.close();

}

**void** calculateBMI() {

calculated\_Bmi = weight / (height \* height);

}

**void** classify() {

**if** (calculated\_Bmi < 18.5) {

System.***out***.println("you are underweight");

} **else** **if** (calculated\_Bmi >= 18.5 && calculated\_Bmi < 24.9) {

System.***out***.println("you are is normal");

} **else** **if** (calculated\_Bmi >= 25 && calculated\_Bmi < 29.9) {

System.***out***.println("you are overweight");

} **else**

System.***out***.println("you are obese");

}

**void** printRecord() {

System.***out***.println(" bmi is :" + calculated\_Bmi);

}

}

**public** **class** Test {

**public** **static** **void** main(String[] args) {

BMITracker bmi = **new** BMITracker();

bmi.acceptRecord();

bmi.calculateBMI();

bmi.printRecord();

bmi.classify();

}

}

**package** com.cdac;

**import** java.util.Scanner;

/\*Design a system to calculate the final price of an item after applying a discount.

\* The system should:

1. Accept the original price of an item and the discount percentage from the user.

2. Calculate the discount amount and the final price using the following formulas:

o Discount Amount Calculation: discountAmount = originalPrice \* (discountRate / 100)

o Final Price Calculation: finalPrice = originalPrice - discountAmount

3. Display the discount amount and the final price of the item, in Indian Rupees (₹).

Define class DiscountCalculator with methods acceptRecord, calculateDiscount &

printRecord and test the functionality in main method.\*/

**class** DiscountCalculator{

**private** **float** original\_price;

**private** **float** discount\_rate;

**private** **double** final\_price;

**private** **double** discount\_amount;

**public** **void** acceptRecord() {

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter the original price:");

**this**.original\_price=sc.nextFloat();

System.***out***.println("Enter the discount rate in(%)");

**this**.discount\_rate=sc.nextFloat();

sc.close();

}

**void** calculateDiscount() {

**this**.discount\_amount = original\_price \* (discount\_rate / 100);

**this**.final\_price=original\_price-discount\_amount;

}

**void** printRecord() {

System.***out***.println("Discount Amount is:" + discount\_amount);

System.***out***.println("final price is :" +final\_price);

}

}

**public** **class** Test {

**public** **static** **void** main(String[] args) {

DiscountCalculator dc=**new** DiscountCalculator();

dc.acceptRecord();

dc.calculateDiscount();

dc.printRecord();

}

}

**package** com.cdac;

**import** java.util.Scanner;

/\*5. Toll Booth Revenue Management

Develop a system to simulate a toll booth for collecting revenue. The system should:

1. Allow the user to set toll rates for different vehicle types: Car, Truck, and Motorcycle.

2. Accept the number of vehicles of each type passing through the toll booth.

3. Calculate the total revenue based on the toll rates and number of vehicles.

4. Display the total number of vehicles and the total revenue collected, in Indian Rupees (₹).

• Toll Rate Examples:

o Car: ₹50.00

o Truck: ₹100.00

o Motorcycle: ₹30.00

Define class TollBoothRevenueManager with methods acceptRecord, setTollRates, calculateRevenue

& printRecord and test the functionality in main method.\*/

**class** TollBoothManager{

**private** **int** no\_of\_car;

**private** **int** no\_of\_truck;

**private** **int** no\_of\_motorcycle;

**private** **float** car\_toll;

**private** **float** truck\_toll;

**private** **float** motorcycle\_toll;

**private** **float** total\_revenue;

**void** acceptRecord() {

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter the no of car :");

no\_of\_car=sc.nextInt();

System.***out***.println("Enter the no of truck:");

no\_of\_truck=sc.nextInt();

System.***out***.println("Enter the no of motorcycle:");

no\_of\_motorcycle=sc.nextInt();

}

**void** setToll() {

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("enter the toll for car:");

**this**.car\_toll=sc.nextFloat();

System.***out***.println("enter the toll for truck:");

**this**.truck\_toll=sc.nextFloat();

System.***out***.println("enter the toll for motorcycle:");

**this**.motorcycle\_toll=sc.nextFloat();

}

**void** calculateRevenue() {

**this**.total\_revenue=(car\_toll\*no\_of\_car)+(truck\_toll \* no\_of\_truck)+(motorcycle\_toll\*no\_of\_motorcycle);

System.***out***.println(**this**.total\_revenue);

}

**void** printRecord() {

System.***out***.println("total no of vehicles:" + (no\_of\_car+no\_of\_motorcycle+no\_of\_truck));

System.***out***.println("total revenue is:" + **this**.total\_revenue);

}

}

**public** **class** Test {

**public** **static** **void** main(String[] args) {

TollBoothManager tm=**new** TollBoothManager();

tm.acceptRecord();

tm.setToll();

tm.calculateRevenue();

tm.printRecord();

}

}