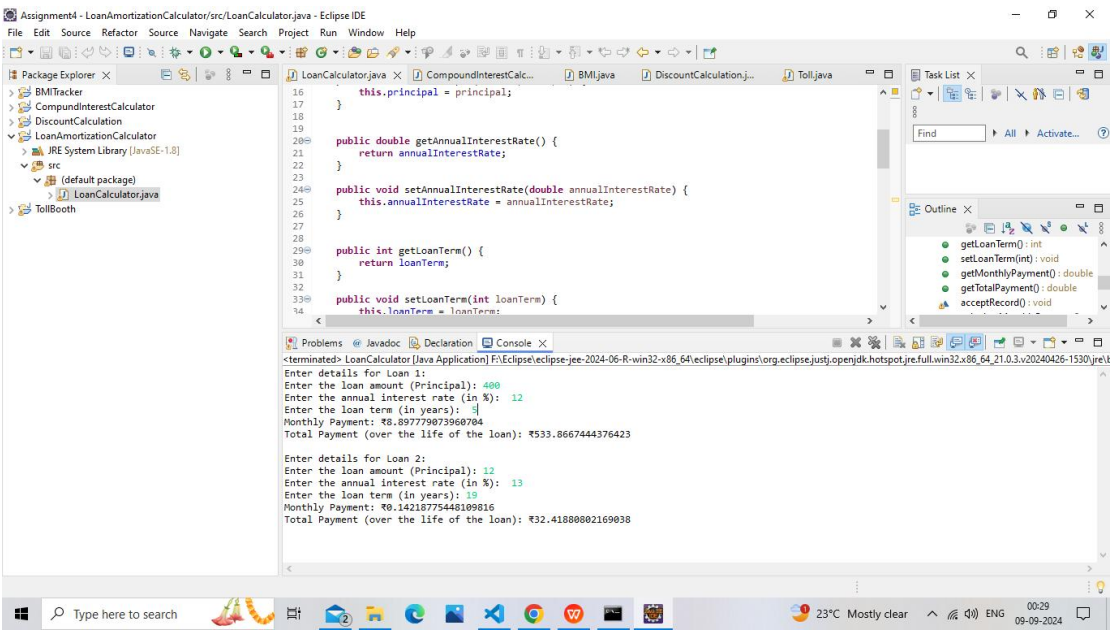
Question1

```
import java.util.Scanner;

class LoanAmortization {
private double principal;
private double annualInterestRate;
private int loanTerm;
private double monthlyPayment;
private double totalPayment;

public double getPrincipal() {
return principal;
}

public void setPrincipal(double principal) {
this.principal = principal;
}

public double getAnnualInterestRate() {
return annualInterestRate;
}

public void setAnnualInterestRate(double annualInterestRate) {
this.annualInterestRate = annualInterestRate;
}

public int getLoanTerm() {
return loanTerm;
}

public void setLoanTerm(int loanTerm) {
this.loanTerm = loanTerm;
}

public double getMonthlyPayment() {
return monthlyPayment;
}

public double getTotalPayment() {
return totalPayment;
}

void acceptRecord() {
Scanner sc = new Scanner(System.in);

System.out.print("Enter the loan amount (Principal): ");
setPrincipal(sc.nextDouble());

System.out.print("Enter the annual interest rate (in %): ");
setAnnualInterestRate(sc.nextDouble());

System.out.print("Enter the loan term (in years): ");
setLoanTerm(sc.nextInt());
}

void calculateMonthlyPayment() {
```

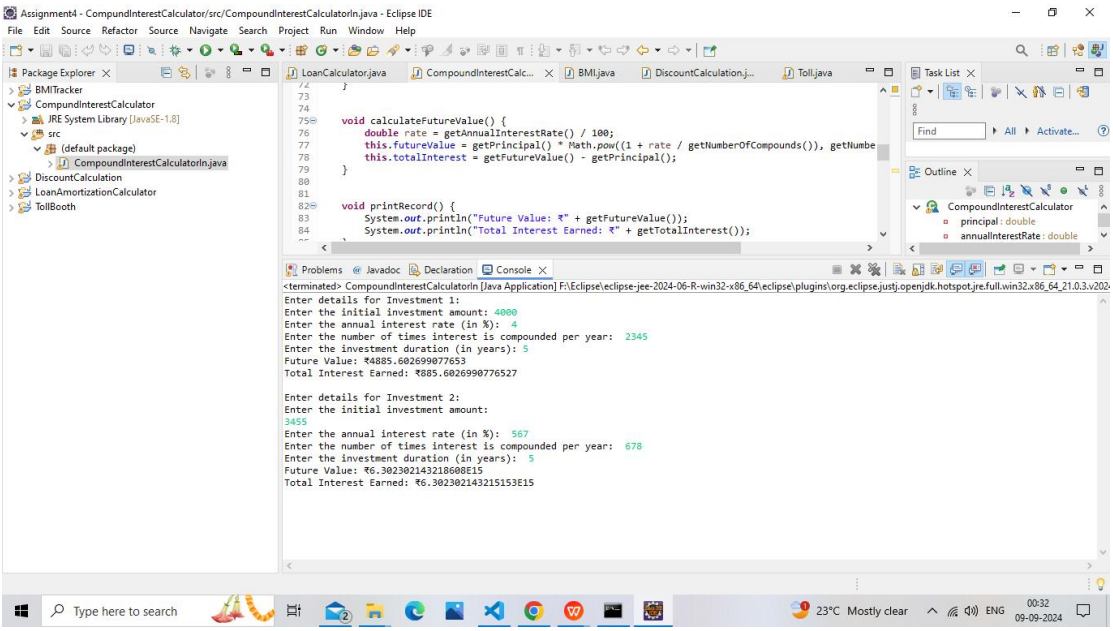
```
double monthlyInterestRate = getAnnualInterestRate() / 12 / 100;
int numberOfMonths = getLoanTerm() * 12;

this.monthlyPayment = getPrincipal() * (monthlyInterestRate * Math.pow((1 +
monthlyInterestRate), numberOfMonths)) /
(Math.pow((1 + monthlyInterestRate), numberOfMonths) - 1);

this.totalPayment = this.monthlyPayment * numberOfMonths;
}

void printRecord() {
System.out.println("Monthly Payment: ₹" + getMonthlyPayment());
System.out.println("Total Payment (over the life of the loan): ₹" +
getTotalPayment());
}
}

public class LoanCalculator {
public static void main(String[] args) {
LoanAmortization loan1 = new LoanAmortization();
LoanAmortization loan2 = new LoanAmortization();

System.out.println("Enter details for Loan 1:");
loan1.acceptRecord();
loan1.calculateMonthlyPayment();
loan1.printRecord();

System.out.println("\nEnter details for Loan 2:");
loan2.acceptRecord();
loan2.calculateMonthlyPayment();
loan2.printRecord();
}
}
```



Question2

```java
import java.util.Scanner;

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

public double getPrincipal() {
return principal;
}

public void setPrincipal(double principal) {
this.principal = principal;
}

public double getAnnualInterestRate() {
return annualInterestRate;
}

public void setAnnualInterestRate(double annualInterestRate) {
this.annualInterestRate = annualInterestRate;
}

public int getNumberOfCompounds() {
return numberOfCompounds;
}

public void setNumberOfCompounds(int numberOfCompounds) {
this.numberOfCompounds = numberOfCompounds;
}

public int getYears() {
return years;
}

public void setYears(int years) {
this.years = years;
}

public double getFutureValue() {
return futureValue;
}

public double getTotalInterest() {
return totalInterest;
}

void acceptRecord() {
Scanner sc = new Scanner(System.in);

System.out.print("Enter the initial investment amount: ");
setPrincipal(sc.nextDouble());

System.out.print("Enter the annual interest rate (in %): ");
```

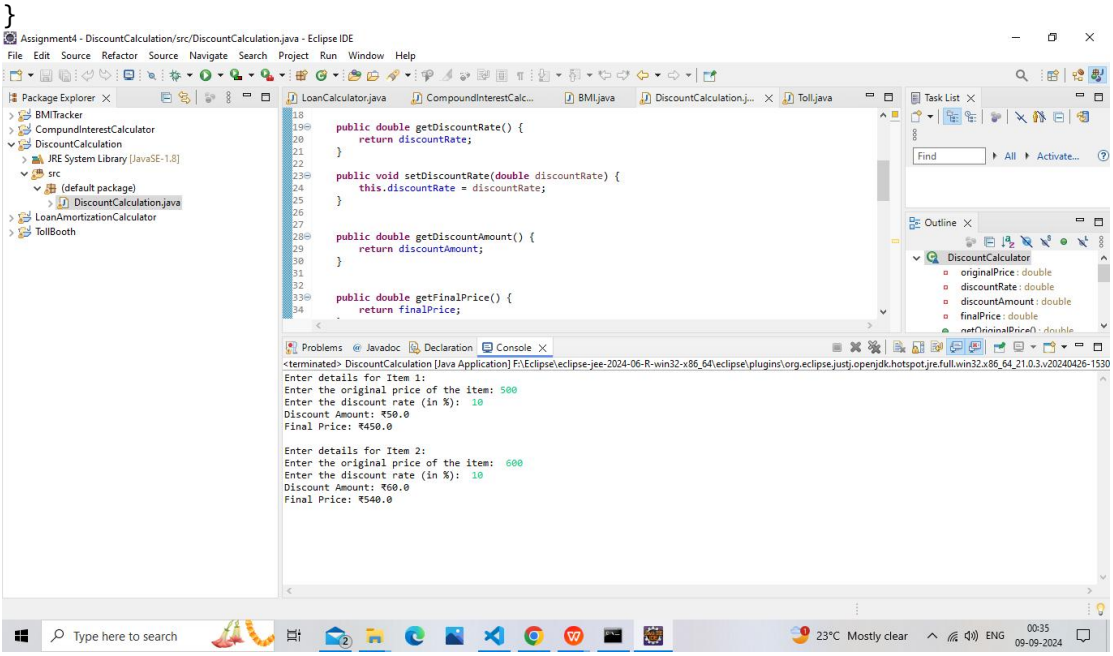
```
setAnnualInterestRate(sc.nextDouble());

System.out.print("Enter the number of times interest is compounded per year:
");
setNumberOfCompounds(sc.nextInt());

System.out.print("Enter the investment duration (in years): ");
setYears(sc.nextInt());
}

void calculateFutureValue() {
double rate = getAnnualInterestRate() / 100;
this.futureValue = getPrincipal() * Math.pow((1 + rate /
getNumberOfCompounds()), getNumberOfCompounds() * getYears());
this.totalInterest = getFutureValue() - getPrincipal();
}

void printRecord() {
System.out.println("Future Value: ₹" + getFutureValue());
System.out.println("Total Interest Earned: ₹" + getTotalInterest());
}
}

public class CompoundInterestCalculatorIn {
public static void main(String[] args) {
CompoundInterestCalculator investment1 = new CompoundInterestCalculator();
CompoundInterestCalculator investment2 = new CompoundInterestCalculator();

System.out.println("Enter details for Investment 1:");
investment1.acceptRecord();
investment1.calculateFutureValue();
investment1.printRecord();

System.out.println("\nEnter details for Investment 2:");
investment2.acceptRecord();
investment2.calculateFutureValue();
investment2.printRecord();
}
}
```



Question3

```
import java.util.Scanner;

class DiscountCalculator {
private double originalPrice;
private double discountRate;
private double discountAmount;
private double finalPrice;

public double getOriginalPrice() {
return originalPrice;
}

public void setOriginalPrice(double originalPrice) {
this.originalPrice = originalPrice;
}

public double getDiscountRate() {
return discountRate;
}

public void setDiscountRate(double discountRate) {
this.discountRate = discountRate;
}

public double getDiscountAmount() {
return discountAmount;
}

public double getFinalPrice() {
return finalPrice;
}

void calculateDiscount() {
this.discountAmount = getOriginalPrice() * (getDiscountRate() / 100);
this.finalPrice = getOriginalPrice() - getDiscountAmount();
```

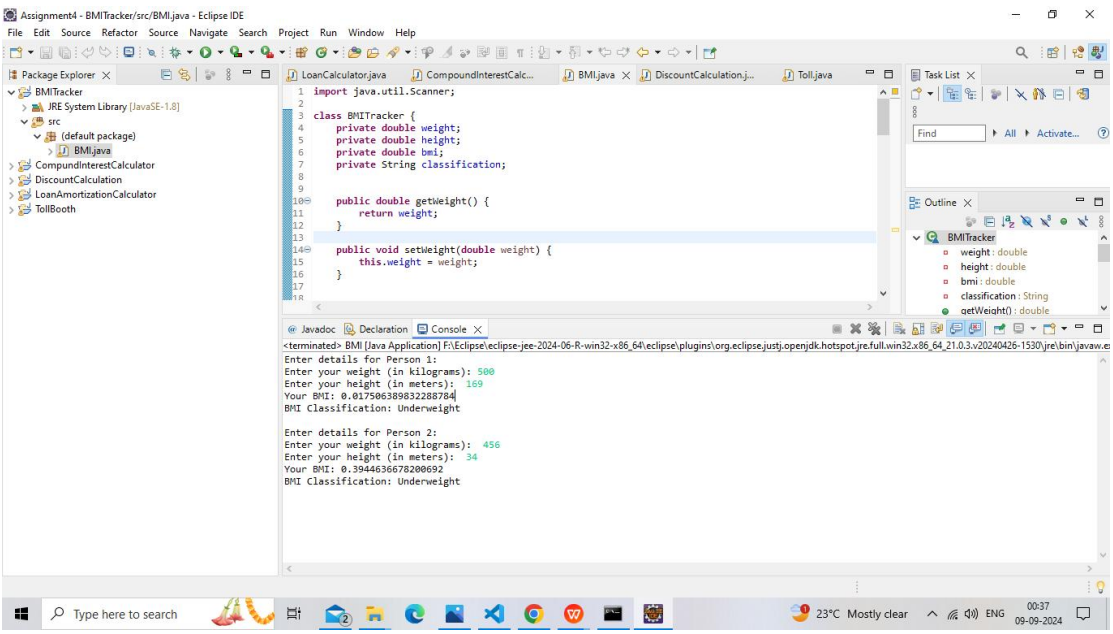
```
}

void printRecord() {
System.out.println("Discount Amount: ₹" + getDiscountAmount());
System.out.println("Final Price: ₹" + getFinalPrice());
}
}

public class DiscountCalculation {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);

DiscountCalculator item1 = new DiscountCalculator();
DiscountCalculator item2 = new DiscountCalculator();

System.out.println("Enter details for Item 1:");
System.out.print("Enter the original price of the item: ");
item1.setOriginalPrice(sc.nextDouble());

System.out.print("Enter the discount rate (in %): ");
item1.setDiscountRate(sc.nextDouble());

item1.calculateDiscount();
item1.printRecord();

System.out.println("\nEnter details for Item 2:");
System.out.print("Enter the original price of the item: ");
item2.setOriginalPrice(sc.nextDouble());

System.out.print("Enter the discount rate (in %): ");
item2.setDiscountRate(sc.nextDouble());

item2.calculateDiscount();
item2.printRecord();
}
}
```



Question4

```
import java.util.Scanner;

class BMITracker {
private double weight;
private double height;
private double bmi;
private String classification;

public double getWeight() {
return weight;
}

public void setWeight(double weight) {
this.weight = weight;
}

public double getHeight() {
return height;
}

public void setHeight(double height) {
this.height = height;
}

public double getBMI() {
return bmi;
}

public String getClassification() {
return classification;
}

void calculateBMI() {
this.bmi = getWeight() / (getHeight() * getHeight());
}

void classifyBMI() {
if (getBMI() < 18.5) {
this.classification = "Underweight";
} else if (getBMI() >= 18.5 && getBMI() < 24.9) {
this.classification = "Normal weight";
} else if (getBMI() >= 25 && getBMI() < 29.9) {
this.classification = "Overweight";
} else {
this.classification = "Obese";
}
}

void printRecord() {
System.out.println("Your BMI: " + getBMI());
System.out.println("BMI Classification: " + getClassification());
}
}

public class BMI {
```

```
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);

BMITracker person1 = new BMITracker();
BMITracker person2 = new BMITracker();

System.out.println("Enter details for Person 1:");
System.out.print("Enter your weight (in kilograms): ");
person1.setWeight(sc.nextDouble());
System.out.print("Enter your height (in meters): ");
person1.setHeight(sc.nextDouble());

person1.calculateBMI();
person1.classifyBMI();
person1.printRecord();

System.out.println("\nEnter details for Person 2:");
System.out.print("Enter your weight (in kilograms): ");
person2.setWeight(sc.nextDouble());
System.out.print("Enter your height (in meters): ");
person2.setHeight(sc.nextDouble());

person2.calculateBMI();
person2.classifyBMI();
person2.printRecord();
}
}
```



Question5

```
import java.util.Scanner;

class TollBoothRevenueManager {
private double carRate;
private double truckRate;
private double motorcycleRate;
```

```
private int numberOfCars;
private int numberOfTrucks;
private int numberOfMotorcycles;
private double totalRevenue;
private int totalVehicles;

public double getCarRate() {
return carRate;
}

public void setCarRate(double carRate) {
this.carRate = carRate;
}

public double getTruckRate() {
return truckRate;
}

public void setTruckRate(double truckRate) {
this.truckRate = truckRate;
}


public double getMotorcycleRate() {
return motorcycleRate;
}

public void setMotorcycleRate(double motorcycleRate) {
this.motorcycleRate = motorcycleRate;
}

public int getNumberOfCars() {
return numberOfCars;
}

public void setNumberOfCars(int numberOfCars) {
this.numberOfCars = numberOfCars;
}

public int getNumberOfTrucks() {
return numberOfTrucks;
}

public void setNumberOfTrucks(int numberOfTrucks) {
this.numberOfTrucks = numberOfTrucks;
}

public int getNumberOfMotorcycles() {
return numberOfMotorcycles;
}

public void setNumberOfMotorcycles(int numberOfMotorcycles) {
this.numberOfMotorcycles = numberOfMotorcycles;
}

public double getTotalRevenue() {
return totalRevenue;
```

```
}

public int getTotalVehicles() {
return totalVehicles;
}

void acceptRecord() {
Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of Cars passing through: ");
setNumberOfCars(sc.nextInt());

System.out.print("Enter the number of Trucks passing through: ");
setNumberOfTrucks(sc.nextInt());

System.out.print("Enter the number of Motorcycles passing through: ");
setNumberOfMotorcycles(sc.nextInt());
}

void setTollRates() {
Scanner sc = new Scanner(System.in);

System.out.print("Enter the toll rate for Cars (₹): ");
setCarRate(sc.nextDouble());

System.out.print("Enter the toll rate for Trucks (₹): ");
setTruckRate(sc.nextDouble());

System.out.print("Enter the toll rate for Motorcycles (₹): ");
setMotorcycleRate(sc.nextDouble());
}

void calculateRevenue() {
this.totalRevenue = (getNumberOfCars() * getCarRate()) +
(getNumberOfTrucks() * getTruckRate()) +
(getNumberOfMotorcycles() * getMotorcycleRate());

this.totalVehicles = getNumberOfCars() + getNumberOfTrucks() +
getNumberOfMotorcycles();
}

void printRecord() {
System.out.println("Total Number of Vehicles: " + getTotalVehicles());
System.out.println("Total Revenue Collected: ₹" + getTotalRevenue());
}
}

public class Toll {
public static void main(String[] args) {
TollBoothRevenueManager tollBooth = new TollBoothRevenueManager();

tollBooth.setTollRates();

tollBooth.acceptRecord();

tollBooth.calculateRevenue();
```

```
tollBooth.printRecord();
}
}
```