BSC. (HONS) IN ELECTRONICS AND TELECOMMUNICATIONS ENGINEERING

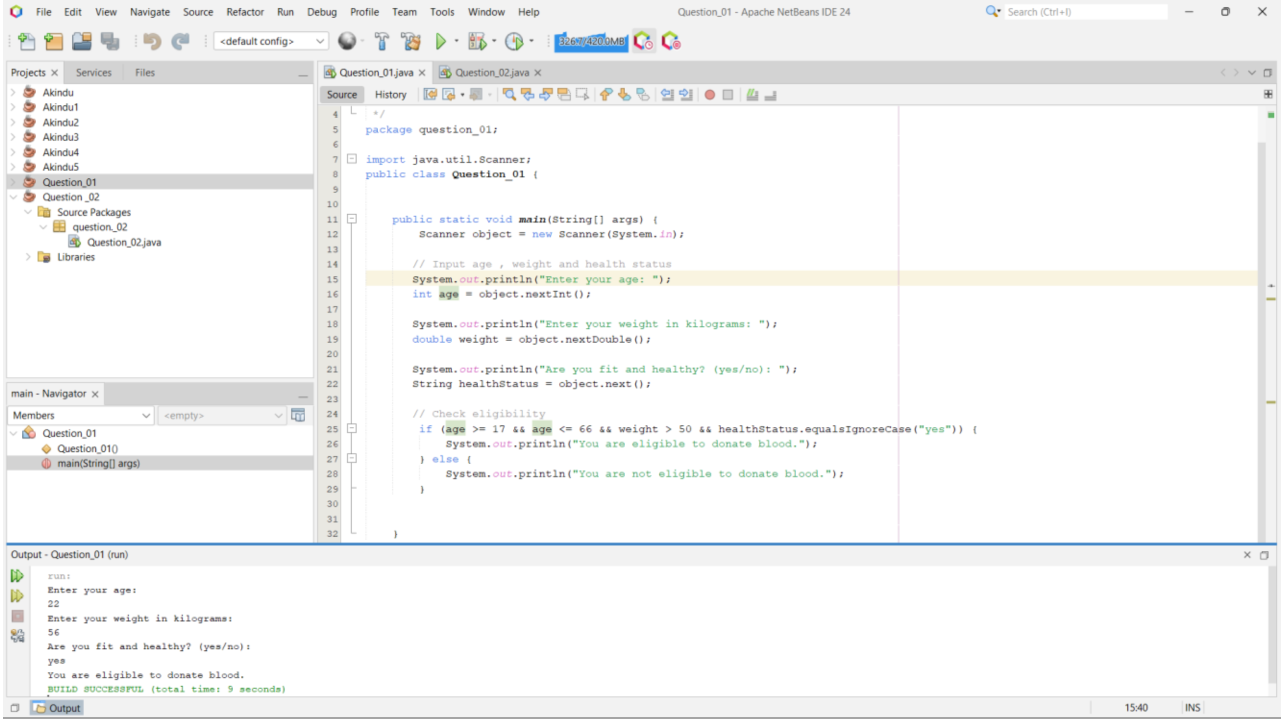
ECS2301 – Software Engineering and Project

LAB ASSIGNMENT NO. : 02

INDEX NUMBER : 23UG1- 0152_Akindu Randira

7th JANUARY 2025

1. You should be able to give blood if you are fit and healthy, weigh over 50kg and are between 17 and 66 years. Write a simple program to check the eligibility of blood donation.



```java
 */
package question_01;

import java.util.Scanner;
public class Question_01 {


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

        // Input age , weight and health status
        System.out.println("Enter your age: ");
        int age = object.nextInt();

        System.out.println("Enter your weight in kilograms: ");
        double weight = object.nextDouble();

        System.out.println("Are you fit and healthy? (yes/no): ");
        String healthStatus = object.next();

        // Check eligibility
         if (age >= 17 && age <= 66 && weight > 50 && healthStatus.equalsIgnoreCase("yes")) {
             System.out.println("You are eligible to donate blood.");
         } else {
             System.out.println("You are not eligible to donate blood.");
         }


    }
```

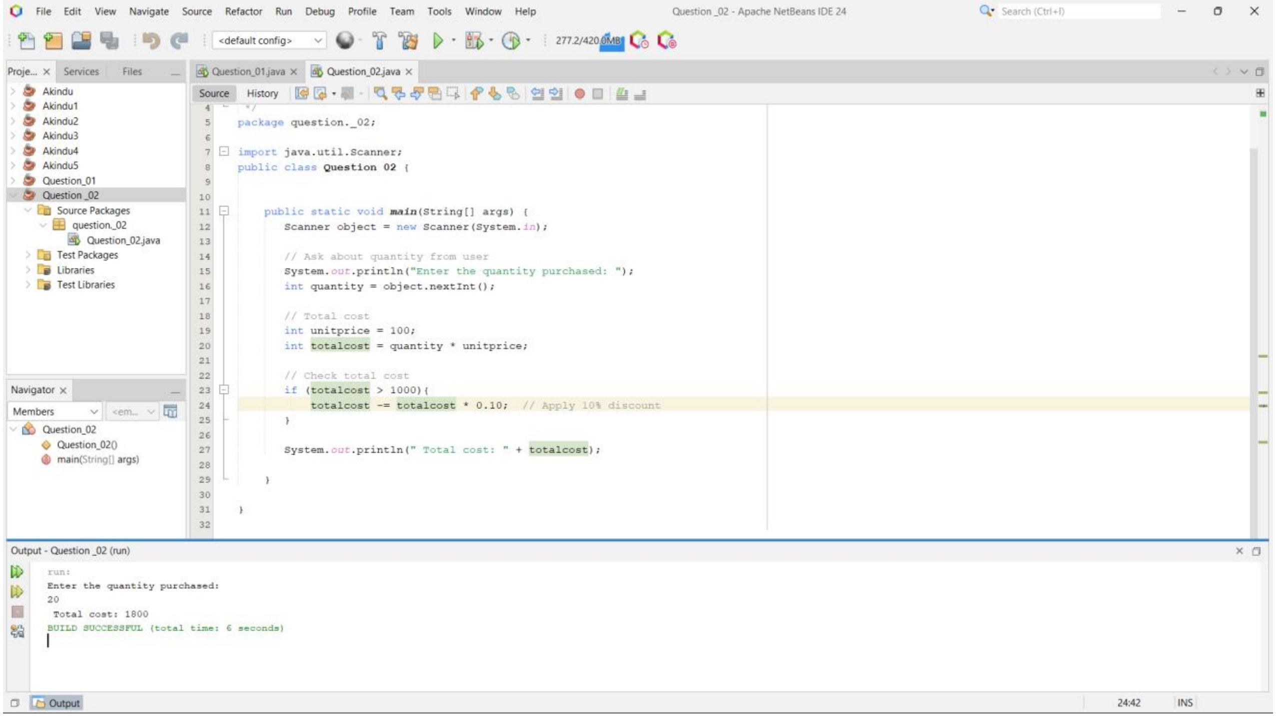
```
run:
Enter your age:
22
Enter your weight in kilograms:
56
Are you fit and healthy? (yes/no):
yes
You are eligible to donate blood.
BUILD SUCCESSFUL (total time: 9 seconds)
```

2. A shop will give a discount of 10% if the cost of the quantity purchased is more than 1000.

a. Ask user for quantity

b. Suppose, one unit will cost 100.

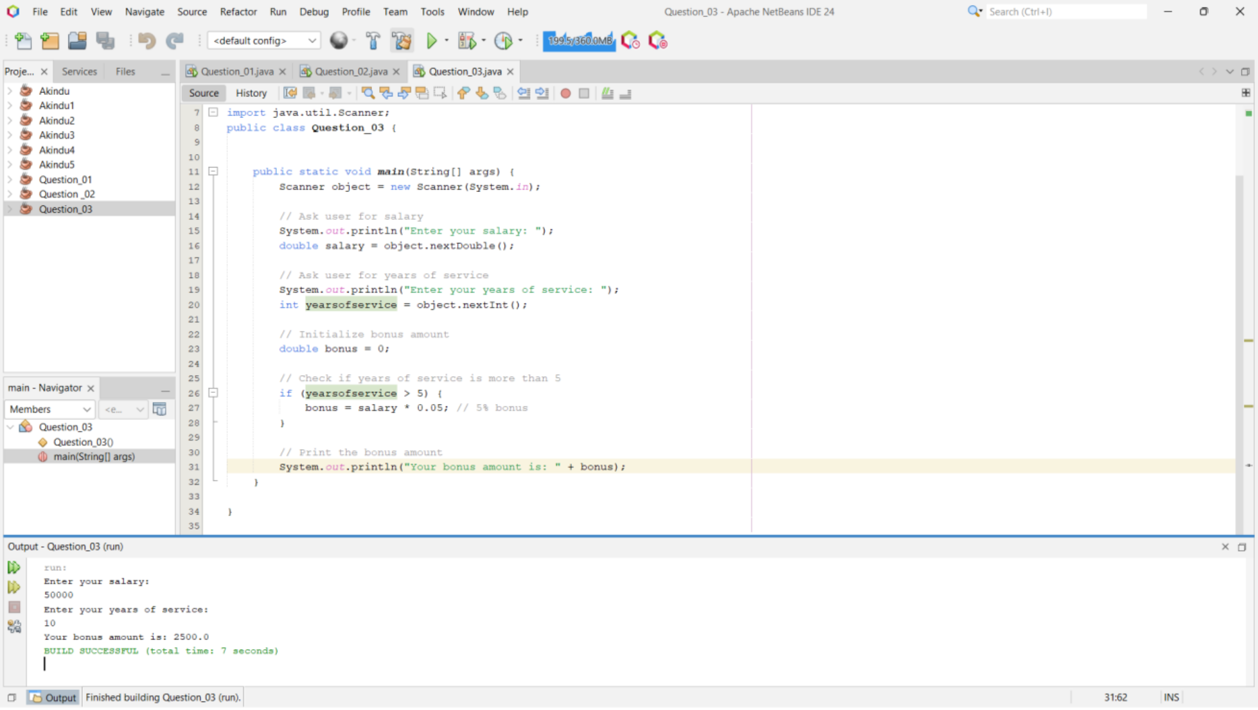
c. Judge and print total cost for the user.



```java
 */
package question._02;

import java.util.Scanner;
public class Question 02 {


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

       // Ask about quantity from user
       System.out.println("Enter the quantity purchased: ");
       int quantity = object.nextInt();

       // Total cost
       int unitprice = 100;
       int totalcost = quantity * unitprice;

       // Check total cost
       if (totalcost > 1000){
           totalcost -= totalcost * 0.10;  // Apply 10% discount
       }

       System.out.println(" Total cost: " + totalcost);

    }

}
```

```
run:
Enter the quantity purchased:
20
 Total cost: 1800
BUILD SUCCESSFUL (total time: 6 seconds)
```

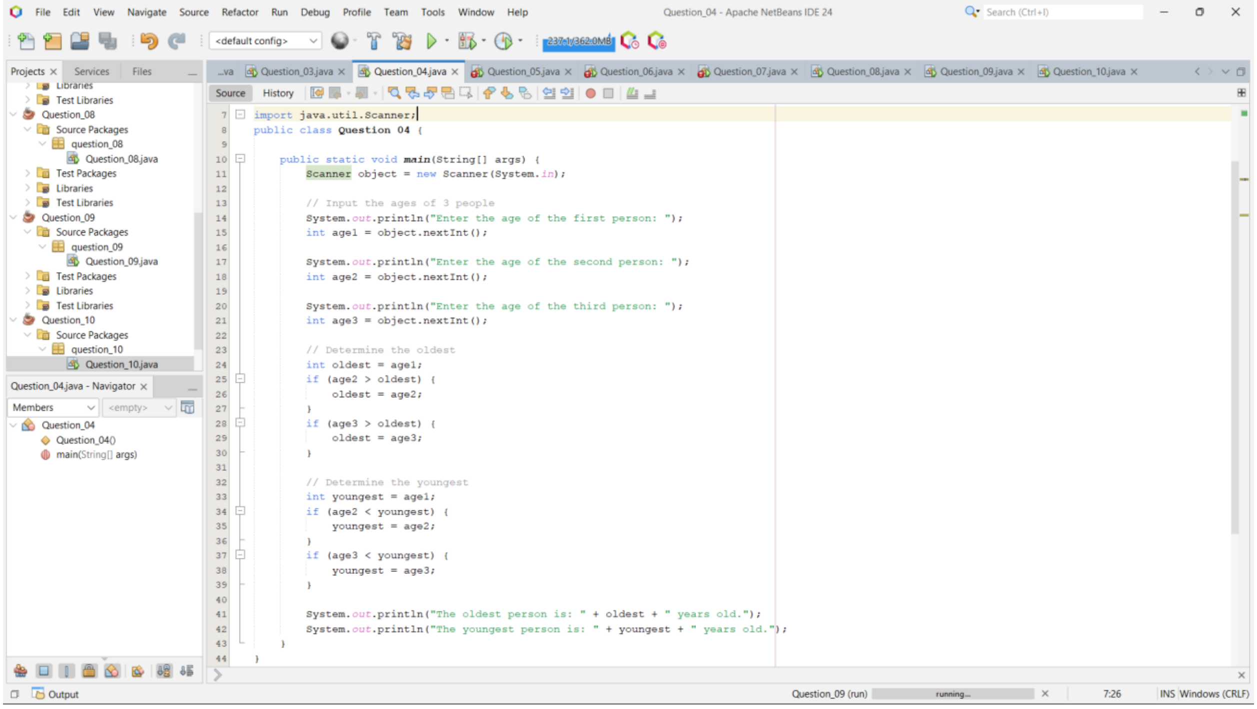
3. A company decided to give a bonus of 5% to an employee if his/her year of service is more than 5 years. Ask users for their salary and year of service and print the net bonus amount.



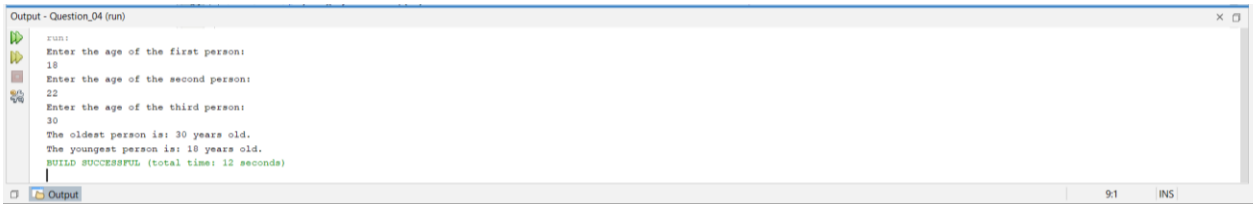
```java
import java.util.Scanner;
public class Question_03 {


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

        // Ask user for salary
        System.out.println("Enter your salary: ");
        double salary = object.nextDouble();

        // Ask user for years of service
        System.out.println("Enter your years of service: ");
        int yearsofservice = object.nextInt();

        // Initialize bonus amount
        double bonus = 0;

        // Check if years of service is more than 5
        if (yearsofservice > 5) {
            bonus = salary * 0.05; // 5% bonus
        }

        // Print the bonus amount
        System.out.println("Your bonus amount is: " + bonus);
    }

}
```

```
run:
Enter your salary:
50000
Enter your years of service:
10
Your bonus amount is: 2500.0
BUILD SUCCESSFUL (total time: 7 seconds)
```

4. Take input of the age of 3 people by user and determine oldest and youngest among them.

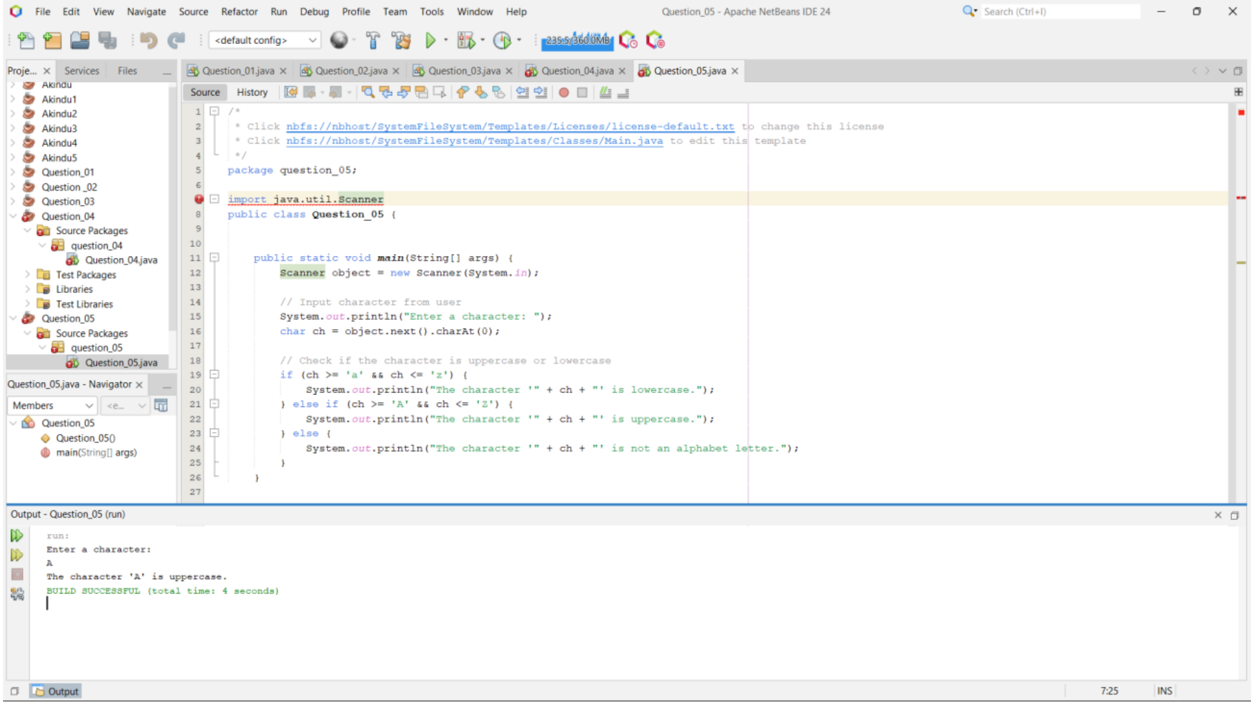


```java
import java.util.Scanner;
public class Question_04 {

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

        // Input the ages of 3 people
        System.out.println("Enter the age of the first person: ");
        int age1 = object.nextInt();

        System.out.println("Enter the age of the second person: ");
        int age2 = object.nextInt();

        System.out.println("Enter the age of the third person: ");
        int age3 = object.nextInt();

        // Determine the oldest
        int oldest = age1;
        if (age2 > oldest) {
            oldest = age2;
        }
        if (age3 > oldest) {
            oldest = age3;
        }

        // Determine the youngest
        int youngest = age1;
        if (age2 < youngest) {
            youngest = age2;
        }
        if (age3 < youngest) {
            youngest = age3;
        }

        System.out.println("The oldest person is: " + oldest + " years old.");
        System.out.println("The youngest person is: " + youngest + " years old.");
    }
}
```



```
run:
Enter the age of the first person:
18
Enter the age of the second person:
22
Enter the age of the third person:
30
The oldest person is: 30 years old.
The youngest person is: 18 years old.
BUILD SUCCESSFUL (total time: 12 seconds)
```

5. Write a program to check whether an entered character is lowercase (a to z) or uppercase (A to Z).


Question_05 - Apache NetBeans IDE 24
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Question_01.java
Question_02.java
Question_03.java
Question_04.java
Question_05.java
Source
History
Output - Question_05 (run)

```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
 */
package question_05;

import java.util.Scanner
public class Question_05 {


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

        // Input character from user
        System.out.println("Enter a character: ");
        char ch = object.next().charAt(0);

        // Check if the character is uppercase or lowercase
        if (ch >= 'a' && ch <= 'z') {
            System.out.println("The character '" + ch + "' is lowercase.");
        } else if (ch >= 'A' && ch <= 'Z') {
            System.out.println("The character '" + ch + "' is uppercase.");
        } else {
            System.out.println("The character '" + ch + "' is not an alphabet letter.");
        }
    }

```

```
run:
Enter a character:
A
The character 'A' is uppercase.
BUILD SUCCESSFUL (total time: 4 seconds)
```

6. The two roots of a quadratic equation ax 2 + bx + c = 0 can be obtained using the following formula: $r1 = \frac{-b+\sqrt{b^2-4ac}}{2a}$ and r2 = $\frac{-b-\sqrt{b^2-4ac}}{2a}$

$b^2 - 4ac$ is called the discriminant of the quadratic equation. If it is positive, the equation has 2 − $4ac$ two real roots. If it is zero, the equation has one root. If it is negative, the equation has no real roots. Write a program that prompts the user to enter values for a, b, and c and displays the result based on the discriminant. If the discriminant is positive, display two roots. If the discriminant is 0, display one root. Otherwise, display “The equation has no real roots”. Note that you can use Math.pow(x, 0.5) to compute 2x. Here are some sample runs.

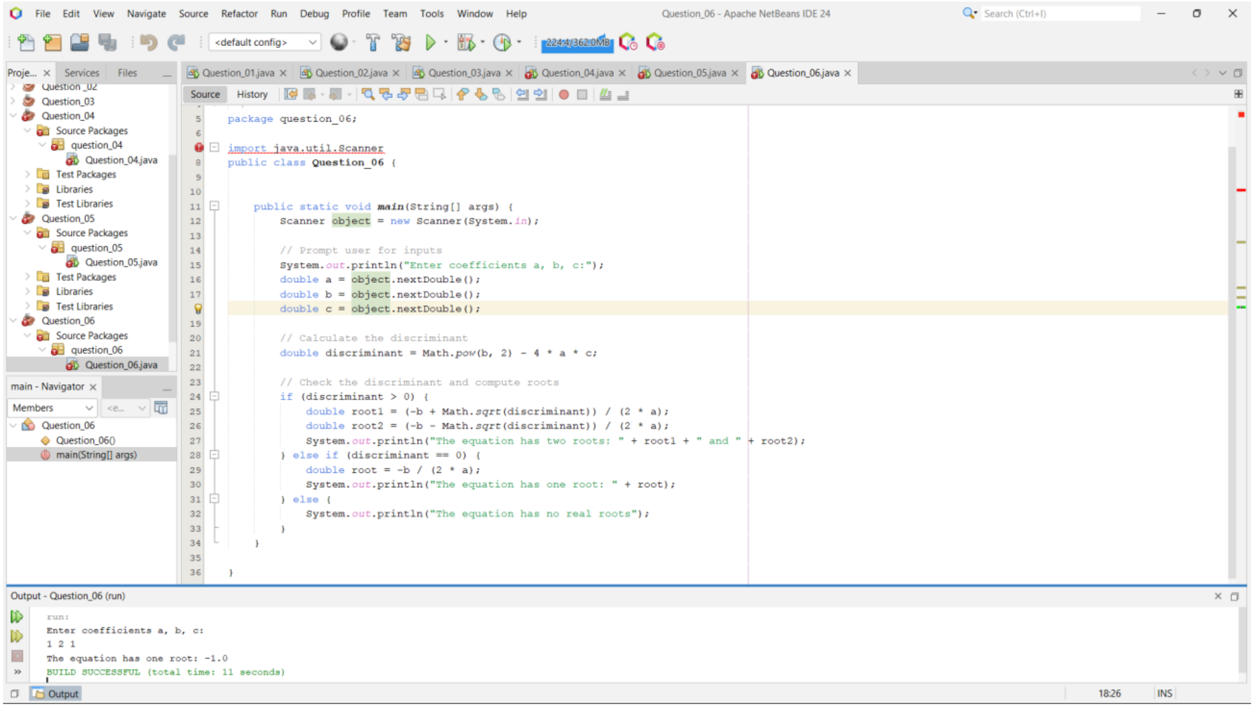
Enter a, b, c: 1.0 3 1

The equation has two roots -0.381966 and -2.61803

Enter a, b, c: 1 2.0 1

The equation has one root -1

Enter a, b, c: 1 2 3 The equation has no real roots



```java
package question_06;

import java.util.Scanner
public class Question_06 {


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

        // Prompt user for inputs
        System.out.println("Enter coefficients a, b, c:");
        double a = object.nextDouble();
        double b = object.nextDouble();
        double c = object.nextDouble();

        // Calculate the discriminant
        double discriminant = Math.pow(b, 2) - 4 * a * c;

        // Check the discriminant and compute roots
        if (discriminant > 0) {
            double root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
            double root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
            System.out.println("The equation has two roots: " + root1 + " and " + root2);
        } else if (discriminant == 0) {
            double root = -b / (2 * a);
            System.out.println("The equation has one root: " + root);
        } else {
            System.out.println("The equation has no real roots");
        }
    }

}
```

Output - Question_06 (run)

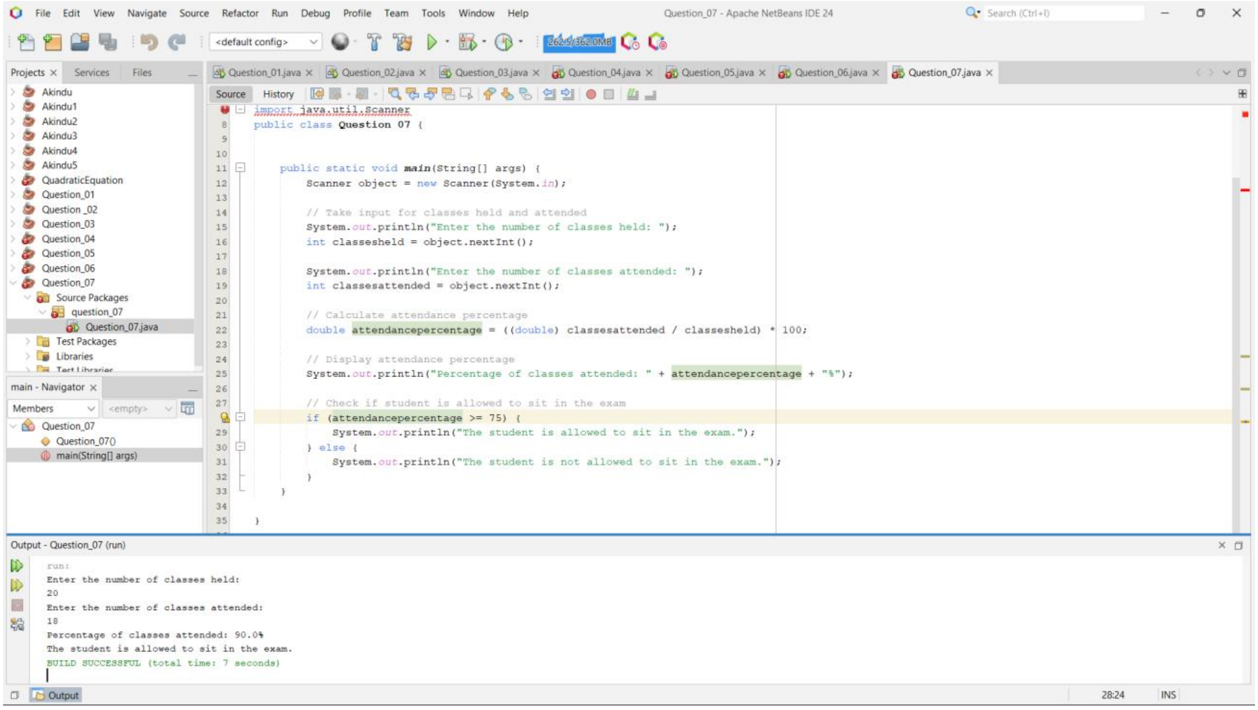
```
run:
Enter coefficients a, b, c:
1 2 1
The equation has one root: -1.0
BUILD SUCCESSFUL (total time: 11 seconds)
```

7. A student will not be allowed to sit an exam if his/her attendance is less than 75%. Take following input from user

a. Number of classes held

b. Number of classes attended. And print

c. percentage of class attended d. Is the student allowed to sit in the exam or not.

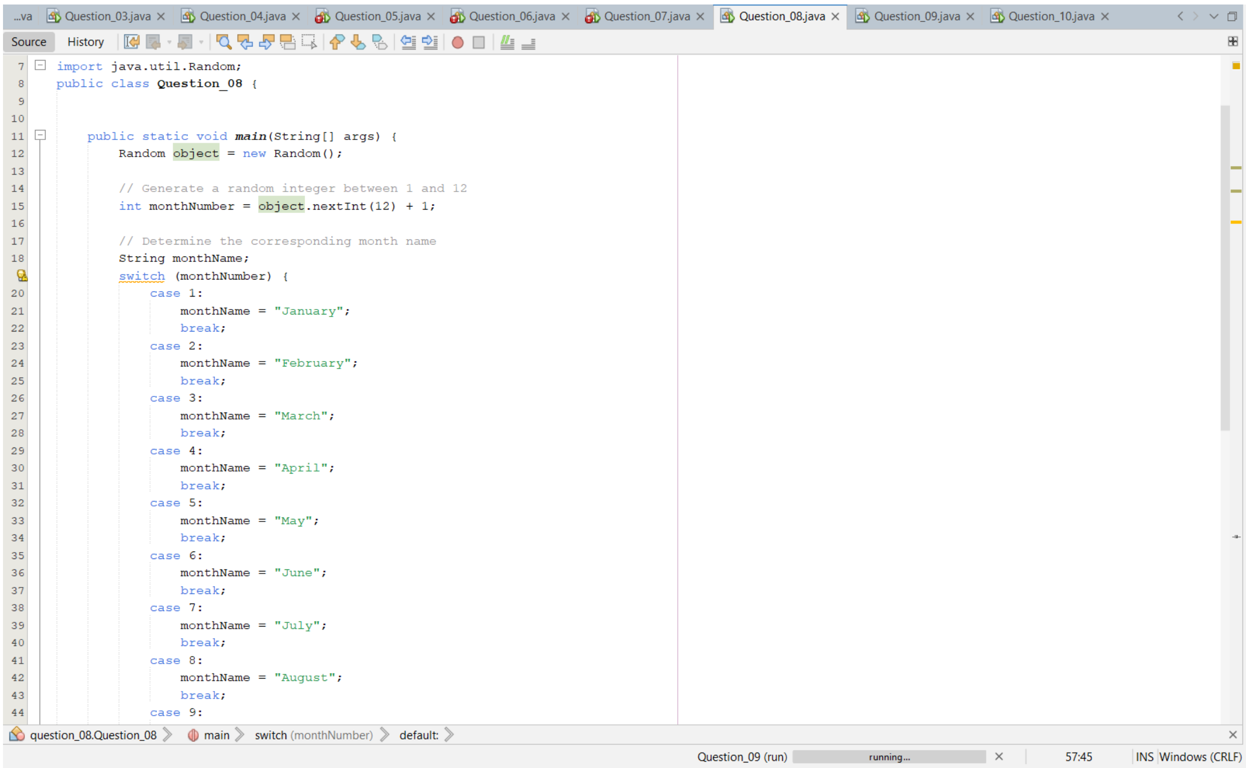


```java
import java.util.Scanner
public class Question_07 {


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

        // Take input for classes held and attended
        System.out.println("Enter the number of classes held: ");
        int classesheld = object.nextInt();

        System.out.println("Enter the number of classes attended: ");
        int classesattended = object.nextInt();

        // Calculate attendance percentage
        double attendancepercentage = ((double) classesattended / classesheld) * 100;

        // Display attendance percentage
        System.out.println("Percentage of classes attended: " + attendancepercentage + "%");

        // Check if student is allowed to sit in the exam
        if (attendancepercentage >= 75) {
            System.out.println("The student is allowed to sit in the exam.");
        } else {
            System.out.println("The student is not allowed to sit in the exam.");
        }
    }

}
```

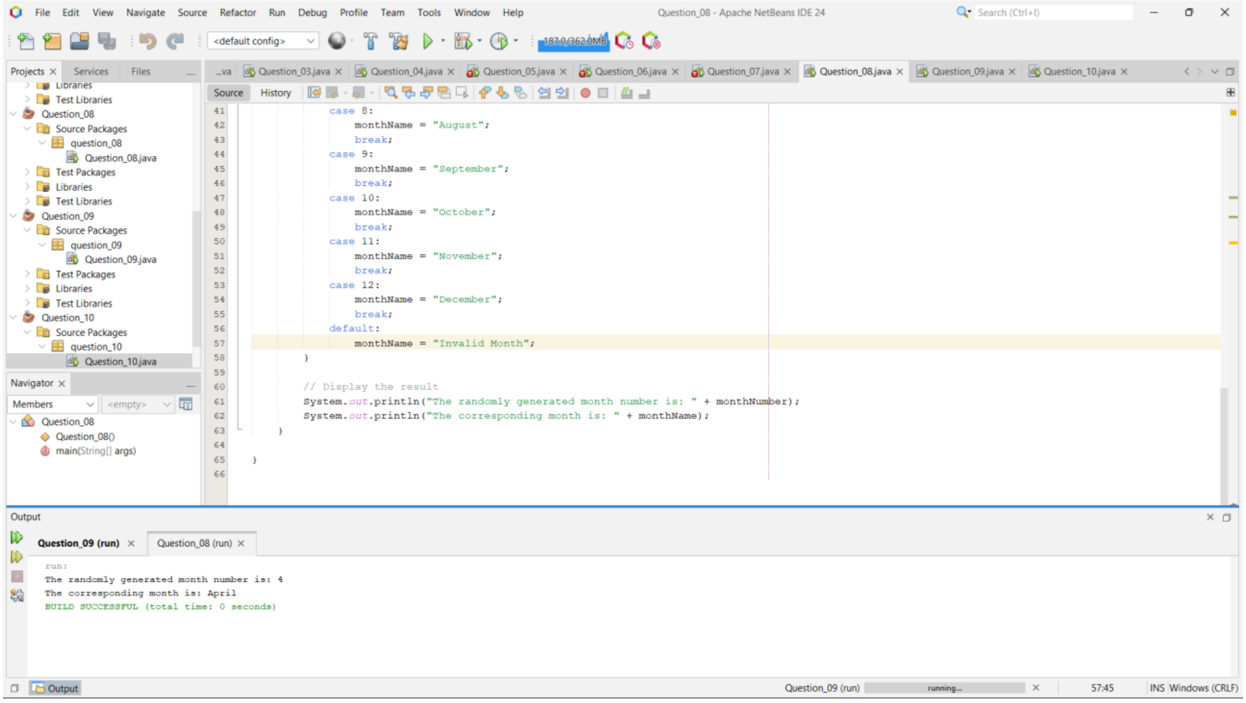
Output - Question_07 (run)

```
run:
Enter the number of classes held:
20
Enter the number of classes attended:
18
Percentage of classes attended: 90.0%
The student is allowed to sit in the exam.
BUILD SUCCESSFUL (total time: 7 seconds)
```

8. Write a program that randomly generates an integer between 1 and 12 and displays the English month name January, February, ..., December for the number 1, 2, ..., 12, accordingly.



```
import java.util.Random;
public class Question_08 {


    public static void main(String[] args) {
        Random object = new Random();

        // Generate a random integer between 1 and 12
        int monthNumber = object.nextInt(12) + 1;

        // Determine the corresponding month name
        String monthName;
        switch (monthNumber) {
            case 1:
                monthName = "January";
                break;
            case 2:
                monthName = "February";
                break;
            case 3:
                monthName = "March";
                break;
            case 4:
                monthName = "April";
                break;
            case 5:
                monthName = "May";
                break;
            case 6:
                monthName = "June";
                break;
            case 7:
                monthName = "July";
                break;
            case 8:
                monthName = "August";
                break;
            case 9:
```



```
            case 8:
                monthName = "August";
                break;
            case 9:
                monthName = "September";
                break;
            case 10:
                monthName = "October";
                break;
            case 11:
                monthName = "November";
                break;
            case 12:
                monthName = "December";
                break;
            default:
                monthName = "Invalid Month";
        }

        // Display the result
        System.out.println("The randomly generated month number is: " + monthNumber);
        System.out.println("The corresponding month is: " + monthName);
    }

}
```

```
run:
The randomly generated month number is: 4
The corresponding month is: April
BUILD SUCCESSFUL (total time: 0 seconds)
```

9. An ISBN-10 (International Standard Book Number) consists of 10 digits: d1d2d3d4d5d6d7d8d9d10 . The last digit, d10 , is a checksum, which is calculated from the other nine digits using the following formula:

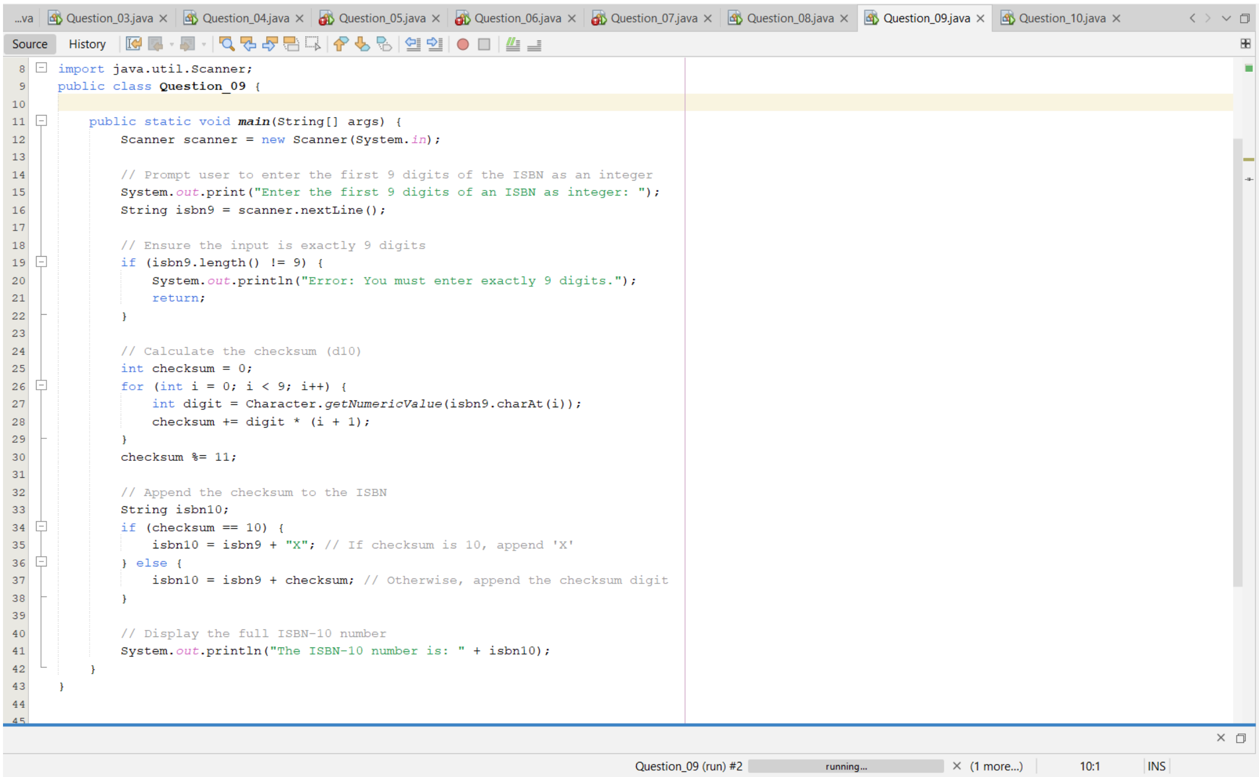
(d1 * 1 + d2 * 2 + d3 * 3 + d4 * 4 + d5 * 5 + d6 * 6 + d7 * 7 + d8 * 8 + d9 * 9) % 11

If the checksum is 10, the last digit is denoted as X according to the ISBN-10 convention.

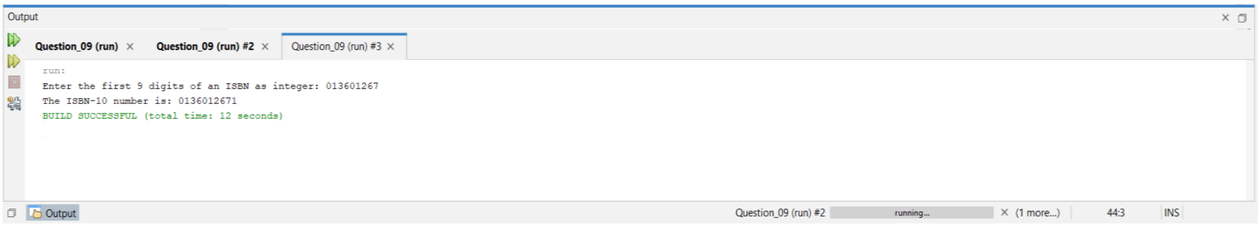
Write a program that prompts the user to enter the first 9 digits and displays the 10-digit ISBN (including leading zeros). Your program should read the input as an integer. Here are sample runs:

Enter the first 9 digits of an ISBN as integer: 013601267 The ISBN-10 number is 0136012671

Enter the first 9 digits of an ISBN as integer: 013031997 The ISBN-10 number is 013031997X



```
import java.util.Scanner;
public class Question_09 {

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

        // Prompt user to enter the first 9 digits of the ISBN as an integer
        System.out.print("Enter the first 9 digits of an ISBN as integer: ");
        String isbn9 = scanner.nextLine();

        // Ensure the input is exactly 9 digits
        if (isbn9.length() != 9) {
            System.out.println("Error: You must enter exactly 9 digits.");
            return;
        }

        // Calculate the checksum (d10)
        int checksum = 0;
        for (int i = 0; i < 9; i++) {
            int digit = Character.getNumericValue(isbn9.charAt(i));
            checksum += digit * (i + 1);
        }
        checksum %= 11;

        // Append the checksum to the ISBN
        String isbn10;
        if (checksum == 10) {
            isbn10 = isbn9 + "X"; // If checksum is 10, append 'X'
        } else {
            isbn10 = isbn9 + checksum; // Otherwise, append the checksum digit
        }

        // Display the full ISBN-10 number
        System.out.println("The ISBN-10 number is: " + isbn10);
    }
}
```



```
run:
Enter the first 9 digits of an ISBN as integer: 013601267
The ISBN-10 number is: 0136012671
BUILD SUCCESSFUL (total time: 12 seconds)
```

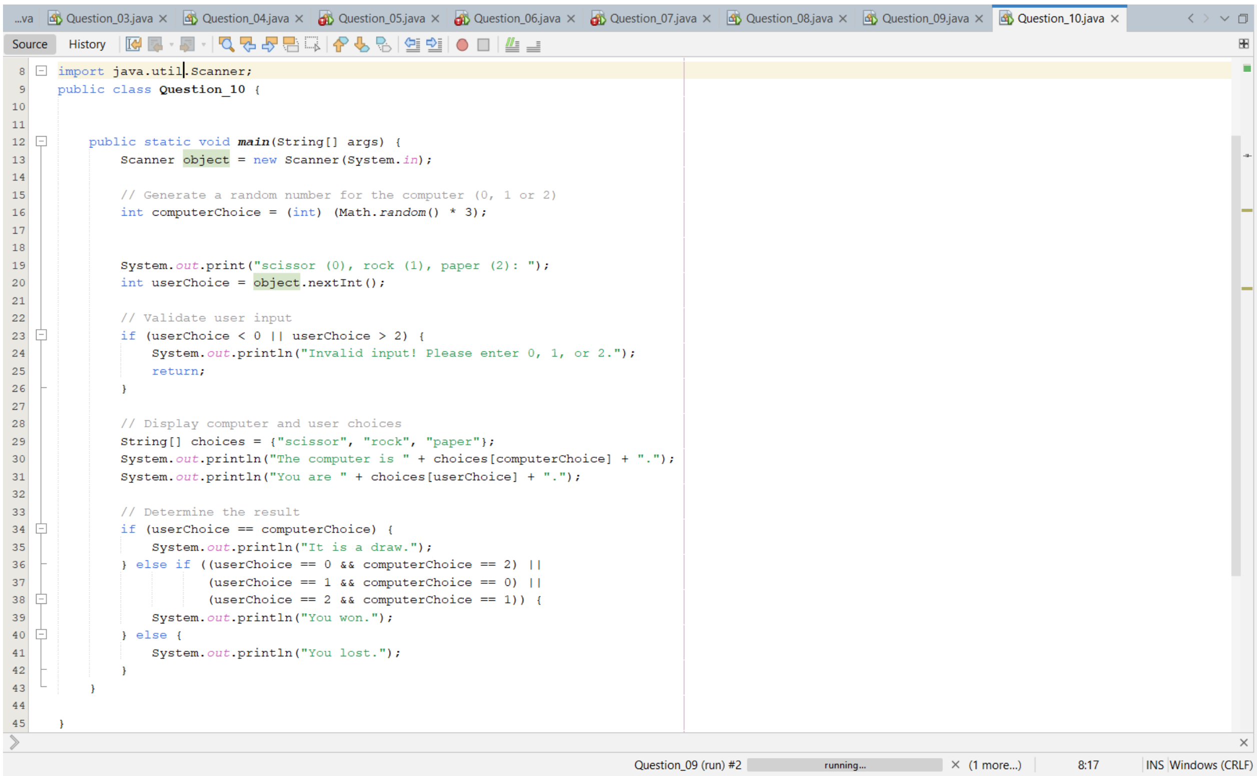
10. (Game: scissor, rock, paper) Write a program that plays the popular scissor-rockpaper game. (A scissor can cut a paper, a rock can knock a scissor, and a paper can wrap a rock.) The program randomly generates a number 0, 1, or 2 representing scissor, rock, and paper. The program prompts the user to enter a number 0, 1, or 2 and displays a message indicating whether the user or the computer wins, loses, or draws. Here are sample runs:

scissor (0), rock (1), paper (2): 1

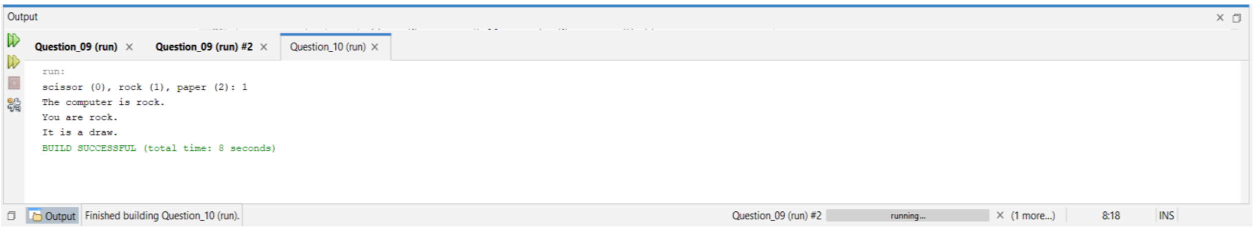
The computer is scissor. You are rock. You won

scissor (0), rock (1), paper (2): 2

The computer is paper. You are paper too. It is a draw



```java
import java.util.Scanner;
public class Question_10 {


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

        // Generate a random number for the computer (0, 1 or 2)
        int computerChoice = (int) (Math.random() * 3);


        System.out.print("scissor (0), rock (1), paper (2): ");
        int userChoice = object.nextInt();

        // Validate user input
        if (userChoice < 0 || userChoice > 2) {
            System.out.println("Invalid input! Please enter 0, 1, or 2.");
            return;
        }

        // Display computer and user choices
        String[] choices = {"scissor", "rock", "paper"};
        System.out.println("The computer is " + choices[computerChoice] + ".");
        System.out.println("You are " + choices[userChoice] + ".");

        // Determine the result
        if (userChoice == computerChoice) {
            System.out.println("It is a draw.");
        } else if ((userChoice == 0 && computerChoice == 2) ||
                   (userChoice == 1 && computerChoice == 0) ||
                   (userChoice == 2 && computerChoice == 1)) {
            System.out.println("You won.");
        } else {
            System.out.println("You lost.");
        }
    }

}
```



```
run:
scissor (0), rock (1), paper (2): 1
The computer is rock.
You are rock.
It is a draw.
BUILD SUCCESSFUL (total time: 8 seconds)
```