Laboratory Work

Subject: Java Technologies
Branch: B.Tech. (CE)
Semester: IV
Batch: A1

Student Roll No: CE001
Student Name: NISARG KALPESHBHAI AMLANI

Department of Computer Engineering,
Faculty of Technology,
Dharmsinh Desai University, Nadiad – 387001.
Gujarat, INDIA.

LAB 01

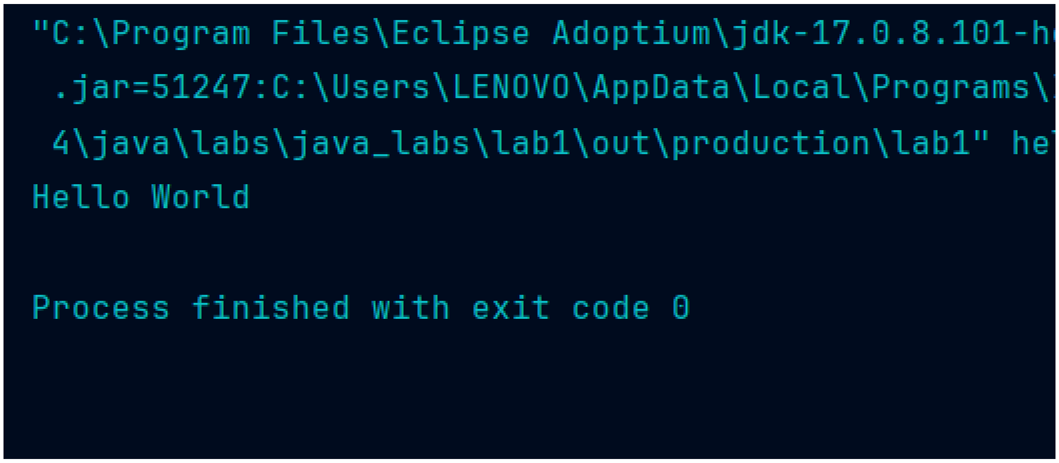
Question 1)

Write a Java program to display “Hello World”.

Solution)

```
class helloworld {

    public static void main(String[] args) {
        System.out.println("Hello World");
    }

}
```

Screenshots)



```
"C:\Program Files\Eclipse Adoptium\jdk-17.0.8.101-h
 .jar=51247:C:\Users\LENOVO\AppData\Local\Programs\
 4\java\labs\java_labs\lab1\out\production\lab1" he
Hello World


Process finished with exit code 0
```

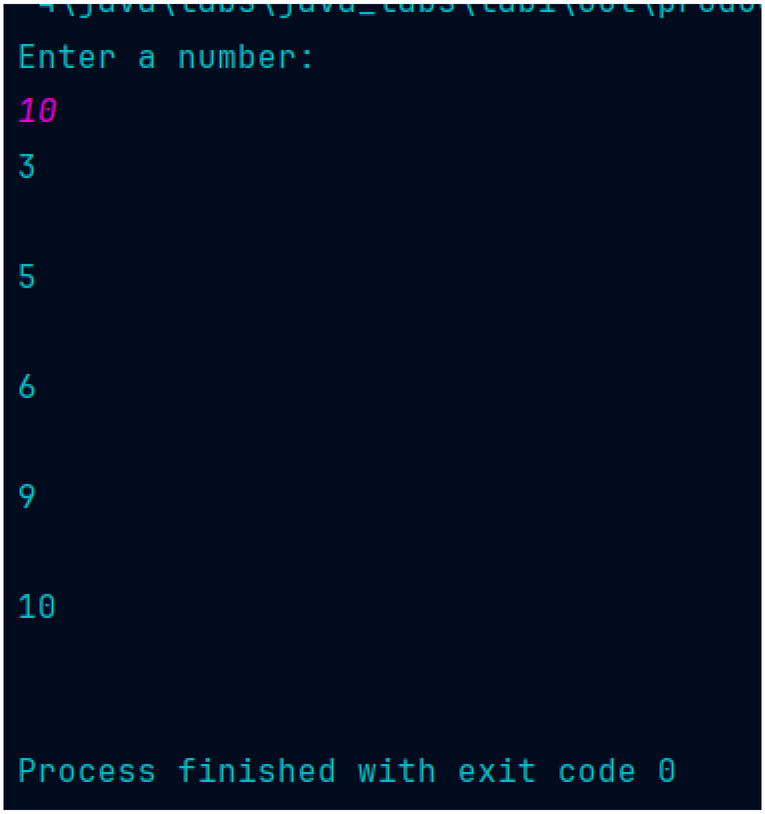
Question 2)

Write a Java program to print numbers between 1 to n which are divisible by 3, 5 and by both(3 and 5) by taking n as an input from the user.

Solution)

```
import java.util.Scanner;

class _input {

    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        int n = obj.nextInt();

        for (int i = 1; i <= n; i++) {
            if (i % 3 == 0 || i % 5 == 0) {
                System.out.println(i + "\n");
            }
        }
    }

}
```

Screenshots



```
4\java\labs\java_labs\lab1\out\produc
Enter a number:
10
3

5

6

9

10


Process finished with exit code 0
```

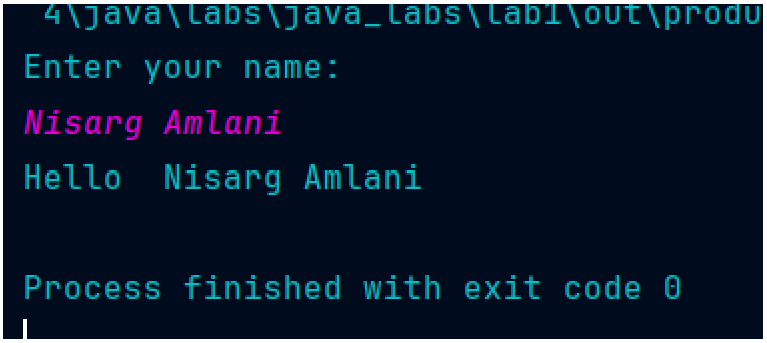
<span>Question 3)</span>

Write a class named Greeter that prompts the user for his or her name, and then prints a personalized greeting. As an example, if the user entered “Era”, the program should respond “Hello Era!”.

Solution)

```
import java.util.Scanner;

class Greeter {
    void greeting(String name) {
        System.out.println("Hello  " + name);
    }
}

class Greet {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        Greeter Obj = new Greeter();
        String name = obj.nextLine();
        Obj.greeting(name);
    }
}
```

Screenshot



```
 4\java\labs\java_labs\lab1\out\produ
Enter your name:
Nisarg Amlani
Hello  Nisarg Amlani

Process finished with exit code 0
```

Question 4)

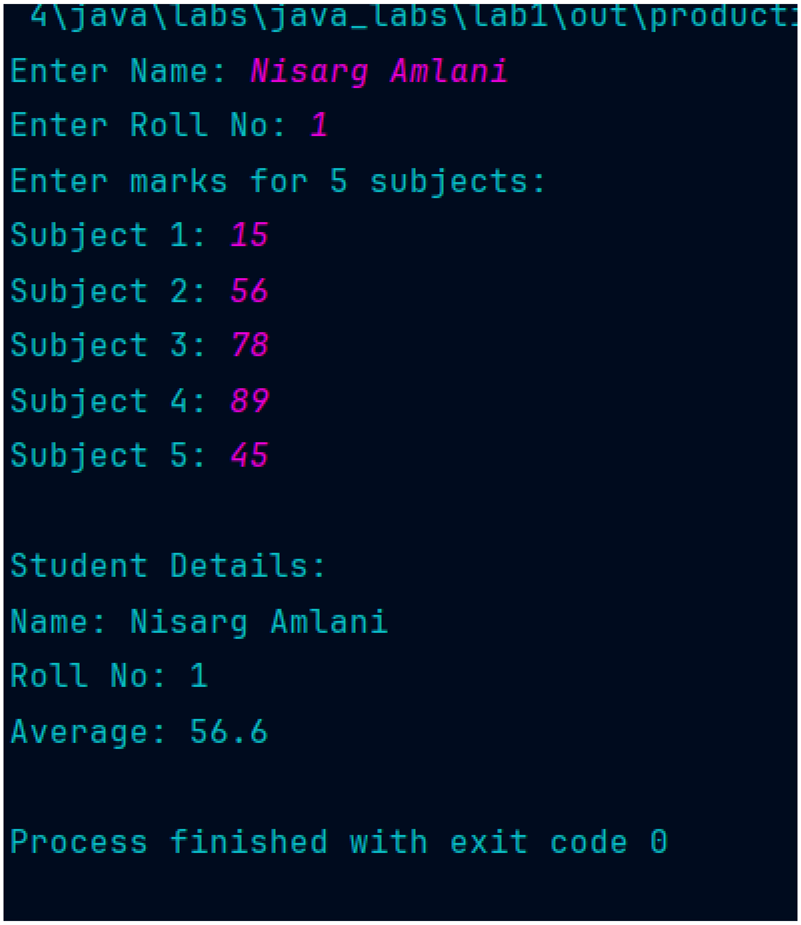
Write a Java program that takes Name, Roll No and marks of 5 subjects as input and
gives a formatted output as:
Name: ABCD
Roll No. : 1
Average: 84
Also display the grade (e.g. A, B, C…etc) using the average.

Solution

```
import java.util.Scanner;

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

```
    Scanner scanner = new Scanner(System.in);

    // Input
    System.out.print("Enter Name: ");
    String name = scanner.nextLine();

    System.out.print("Enter Roll No: ");
    int rollNo = scanner.nextInt();

    System.out.println("Enter marks for 5 subjects:");
    int[] marks = new int[5];
    int sum = 0;

    for (int i = 0; i < 5; i++) {
      System.out.print("Subject " + (i + 1) + ": ");
      marks[i] = scanner.nextInt();
      sum += marks[i];
    }

    // Calculate average
    double average = (double) sum / 5;

    // Output
    System.out.println("\nStudent Details:");
    System.out.println("Name: " + name);
    System.out.println("Roll No: " + rollNo);
    System.out.println("Average: " + average);

    // Display grade
    scanner.close();
  }
}
```

Screenshot



```
 4\java\labs\java_labs\lab1\out\producti
Enter Name: Nisarg Amlani
Enter Roll No: 1
Enter marks for 5 subjects:
Subject 1: 15
Subject 2: 56
Subject 3: 78
Subject 4: 89
Subject 5: 45

Student Details:
Name: Nisarg Amlani
Roll No: 1
Average: 56.6

Process finished with exit code 0
```

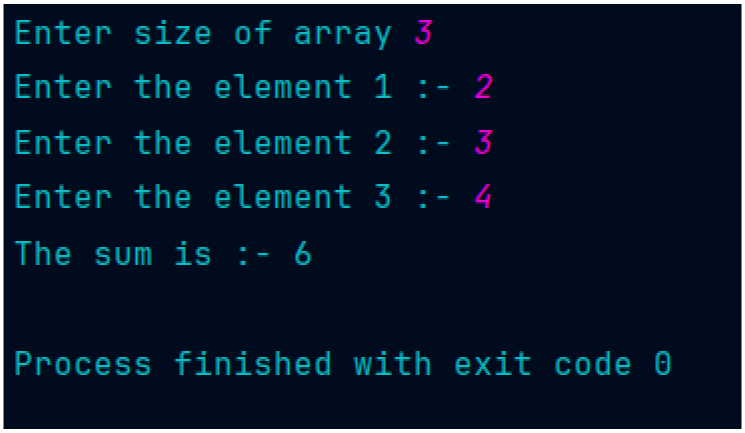
Question 5)

Calculate and return the sum of all the even numbers present in the numbers array passed to the method calculateSumOfEvenNumbers. Implement the logic inside calculateSumOfEvenNumbers() method.
Test the functionalities using the main() method of the Tester class.

Solution

```
import java.util.Scanner;

class Tester {
    void calculateSumOfEvenNumbers(int arr[]) {
        int n = arr.length;
        int sum = 0;
        for (int i = 0; i < n; i++) {
            if (arr[i] % 2 == 0) sum += arr[i];
        }
        System.out.println("The sum is :- " + sum);
    }
}

class res {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        Tester Obj = new Tester();
        System.out.print("Enter size of array ");
        int n = obj.nextInt();
        int arr[] = new int[n];
        for (int i = 0; i < n; i++) {
            System.out.print("Enter the element " + (i + 1) + " :- ");
            arr[i] = obj.nextInt();
        }
        Obj.calculateSumOfEvenNumbers(arr);
    }
}
```

Screenshots



```
Enter size of array 3
Enter the element 1 :- 2
Enter the element 2 :- 3
Enter the element 3 :- 4
The sum is :- 6

Process finished with exit code 0
```

Question 6)

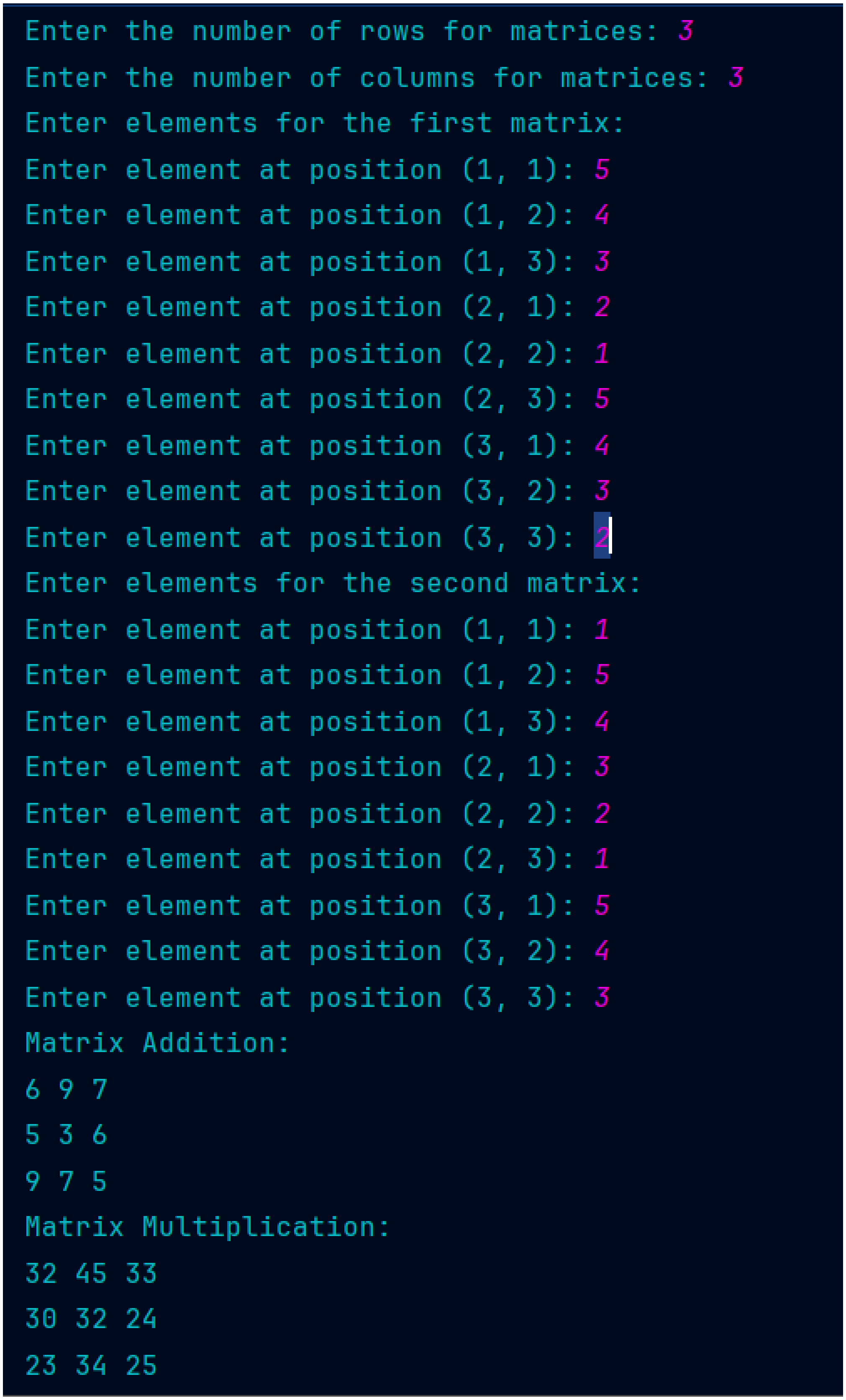
Write a program to perform matrix addition and matrix multiplication on two given matrices. Use for-each form of for loop to display the matrices.

Solution)

```
import java.util.Scanner;

class MatrixOperations {
    public static void displayMatrix(int[][] matrix) {
        for (int[] row : matrix) {
            for (int element : row) {
                System.out.print(element + " ");
            }
            System.out.println();
        }
    }

    public static int[][] addMatrices(int[][] matrix1, int[][] matrix2) {
        int rows = matrix1.length;
        int columns = matrix1[0].length;
        int[][] resultMatrix = new int[rows][columns];

        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                resultMatrix[i][j] = matrix1[i][j] + matrix2[i][j];
            }
        }

        return resultMatrix;
    }

    public static int[][] multiplyMatrices(int[][] matrix1, int[][]
matrix2) {
        int rowsA = matrix1.length;
        int columnsA = matrix1[0].length;
        int columnsB = matrix2[0].length;
        int[][] resultMatrix = new int[rowsA][columnsB];

        for (int i = 0; i < rowsA; i++) {
            for (int j = 0; j < columnsB; j++) {
                int elementSum = 0;
                for (int k = 0; k < columnsA; k++) {
                    elementSum += matrix1[i][k] *
matrix2[k][j];
                }
                resultMatrix[i][j] = elementSum;
            }
        }

        return resultMatrix;
    }

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

        System.out.print("Enter the number of rows for
matrices: ");
        int rows = scanner.nextInt();
        System.out.print("Enter the number of columns for
matrices: ");
        int columns = scanner.nextInt();

        // Input for the first matrix
        System.out.println("Enter elements for the first
matrix:");
        int[][] matrixA = new int[rows][columns];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                System.out.print("Enter element at position
(" + (i + 1) + ", " + (j + 1) + "): ");
                matrixA[i][j] = scanner.nextInt();
            }
        }

        // Input for the second matrix
        System.out.println("Enter elements for the second
matrix:");
        int[][] matrixB = new int[rows][columns];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                System.out.print("Enter element at position
(" + (i + 1) + ", " + (j + 1) + "): ");
```

```
                matrixB[i][j] = scanner.nextInt();
            }
        }

        // Matrix addition
        int[][] sumMatrix = addMatrices(matrixA, matrixB);
        System.out.println("Matrix Addition:");
        displayMatrix(sumMatrix);

        // Matrix multiplication
        int[][] productMatrix = multiplyMatrices(matrixA,
matrixB);
        System.out.println("Matrix Multiplication:");
        displayMatrix(productMatrix);

        scanner.close();
    }
}
```

Screenshots



```
Enter the number of rows for matrices: 3
Enter the number of columns for matrices: 3
Enter elements for the first matrix:
Enter element at position (1, 1): 5
Enter element at position (1, 2): 4
Enter element at position (1, 3): 3
Enter element at position (2, 1): 2
Enter element at position (2, 2): 1
Enter element at position (2, 3): 5
Enter element at position (3, 1): 4
Enter element at position (3, 2): 3
Enter element at position (3, 3): 2
Enter elements for the second matrix:
Enter element at position (1, 1): 1
Enter element at position (1, 2): 5
Enter element at position (1, 3): 4
Enter element at position (2, 1): 3
Enter element at position (2, 2): 2
Enter element at position (2, 3): 1
Enter element at position (3, 1): 5
Enter element at position (3, 2): 4
Enter element at position (3, 3): 3
Matrix Addition:
6 9 7
5 3 6
9 7 5
Matrix Multiplication:
32 45 33
30 32 24
23 34 25
```

LAB 02

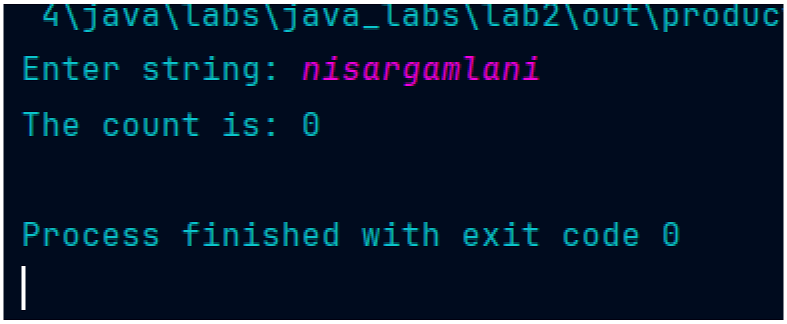
Question 1

Write a program that returns the number of times that the string “hi” appears anywhere in the given string.

Solution

```
import java.util.Scanner;

class Substr {

   public static void main(String[] args) {
      Scanner obj = new Scanner(System.in);
      System.out.print("Enter string: ");
      String str = obj.nextLine();

      int count = 0;
      int index = 0;

      while ((index = str.indexOf("hi", index)) != -1) {
         count++;
         index += 2;
      }

      System.out.println("The count is: " + count);
   }
}
```

Screenshot



```
 4\java\labs\java_labs\lab2\out\produc
Enter string: nisargamlani
The count is: 0

Process finished with exit code 0
```

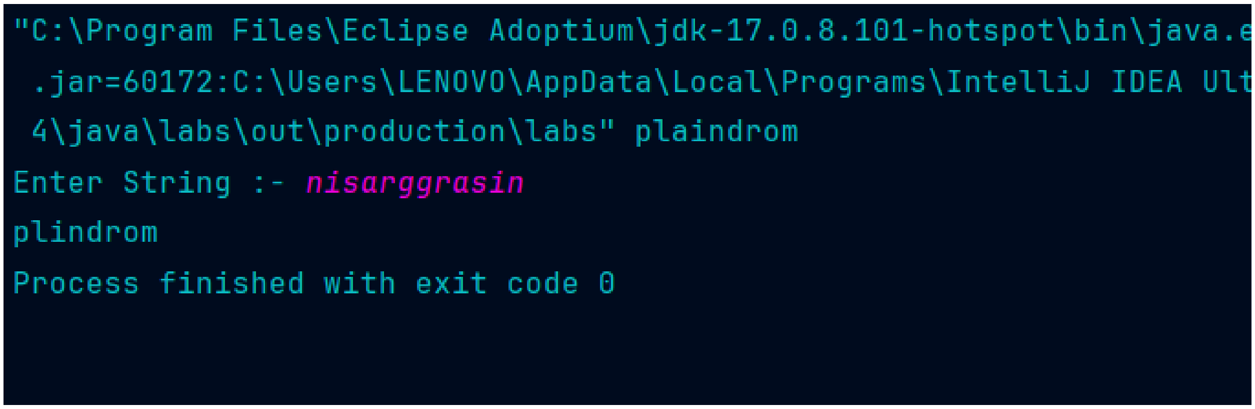
Question 2

Write a program which checks whether the input string is palindrome or not and then display an appropriate message [e.g. “Refer” is a palindrome string].

Solution

```
import java.util.Scanner;

class plaindrom{
   public static void main (String [] args)
   {
      Scanner obj = new Scanner(System.in);
      System.out.print("Enter String :- ");
      String  s =  obj.nextLine();
      StringBuilder s1 = new StringBuilder(s.toLowerCase());
      StringBuilder s2 = new StringBuilder(s.toLowerCase());
      s2.reverse();

      if((s1.toString()).equals(s2.toString()))
System.out.print("plindrom");
      else System.out.print(" not palindrom");
   }
}
```

Screenshot



```
"C:\Program Files\Eclipse Adoptium\jdk-17.0.8.101-hotspot\bin\java.e
 .jar=60172:C:\Users\LENOVO\AppData\Local\Programs\IntelliJ IDEA Ult
 4\java\labs\out\production\labs" plaindrom
Enter String :- nisarggrasin
plindrom
Process finished with exit code 0
```

Question 3

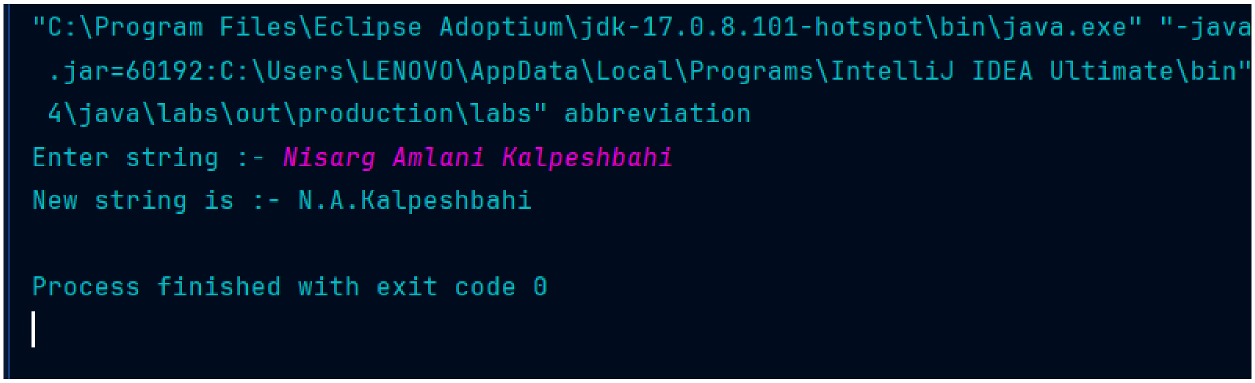
Write a program that takes your full name as input and displays the abbreviations of the first and middle names except the last name which is displayed as it is. For example, if your name is Robert Brett Roser, then the output should be R.B.Roser.

Solution

```
import java.util.*;

class abbreviation{
   public static void  main(String [] args)
   {
      Scanner obj = new Scanner(System.in);
      System.out.print("Enter string :- ");
      String S = obj.nextLine();

      int count = 1;
      String []Fullname = S.split(" ");
```

```
        StringBuffer n = new StringBuffer();
        for(String s : Fullname)
        {
            if(count%3 == 0)
            {
                n.append(s);
            }
            else{
                n.append(String.valueOf(s.charAt(0)) + "");
            }
            count++;
        }

        System.out.println("New string is :- " + n );
    }
}
```

Screenshot



```
"C:\Program Files\Eclipse Adoptium\jdk-17.0.8.101-hotspot\bin\java.exe" "-java
 .jar=60192:C:\Users\LENOVO\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin"
 4\java\labs\out\production\labs" abbreviation
Enter string :- Nisarg Amlani Kalpeshbahi
New string is :- N.A.Kalpeshbahi

Process finished with exit code 0
```

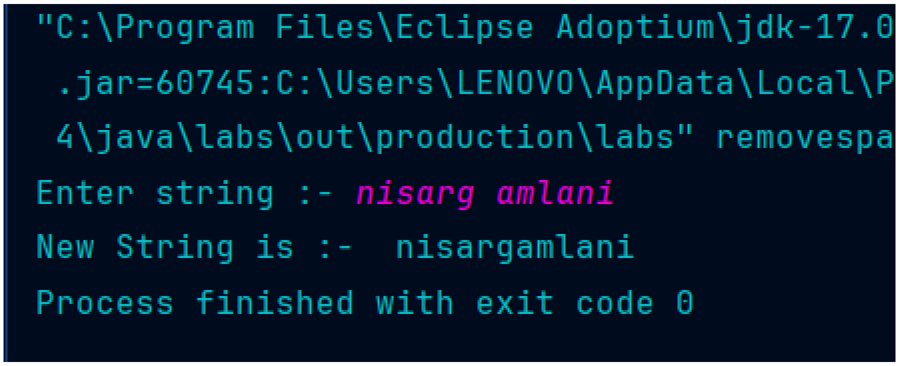
Question4

Write a method String removeWhiteSpaces(String str) method that removes all the white spaces from the string passed to the method and returns the modified string. Test the functionalities using the main() method of the Tester class.

Solution

```
import java.util.*;

class removespaces{

    static String remove_space(String s)
    {
        String []sarr = s.split(" ");
        StringBuffer sb = new StringBuffer();
        for( String str  : sarr )
        {
            sb.append(str);
        }

        return sb.toString();

    }
    public static void main (String[] s)
    {
        Scanner obj = new Scanner(System.in);
        System.out.print("Enter string :- ");
        String str = obj.nextLine();

        System.out.print("New String is :-  " + remove_space(str));

    }
}
```

Screenshot



```
"C:\Program Files\Eclipse Adoptium\jdk-17.0
 .jar=60745:C:\Users\LENOVO\AppData\Local\P
 4\java\labs\out\production\labs" removespa
Enter string :- nisarg amlani
New String is :-  nisargamlani
Process finished with exit code 0
```

Question 5

Write a class Student with member variables int roll_no, String name and an array to store marks of 5 subjects. Demonstrate constructor overloading and use this keyword. Write a findAverage() method that returns double value. Write a TestStudent class containing main() method to do the following:

a) Store the details of one student by creating one object of Student class and display them.

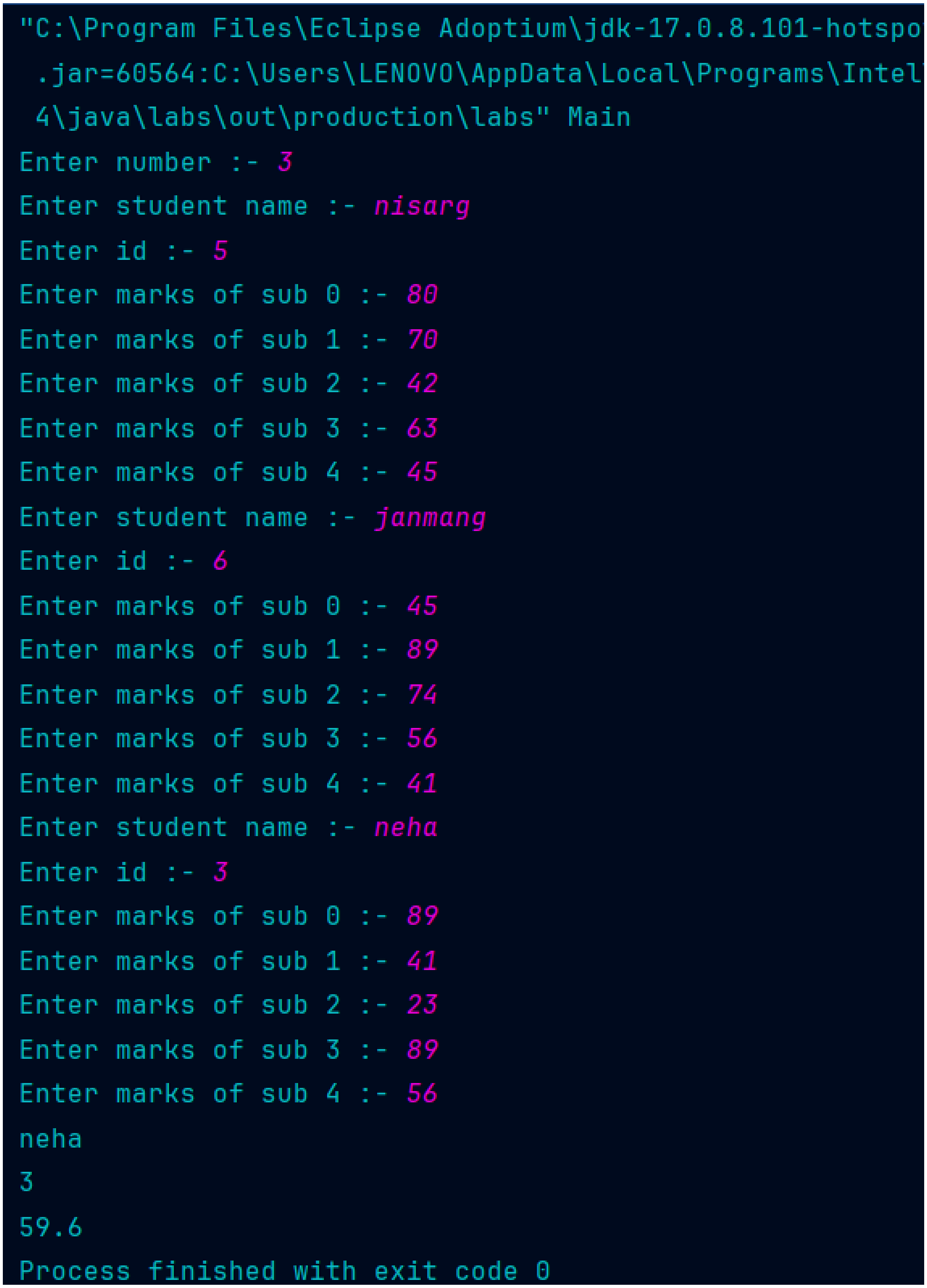
b) Store the details of 3 students by creating an array of objects of Student class and display the details of the student who has the highest average amongst the three students.

Solution

```
import java.util.*;

class Student{
    int roll;
    String name;
    float Average;
    int[] marks;
    Student(String name , int roll , int[] marks)
    {
        this.name = name;
        this.roll = roll;
        this.marks = marks;

        average_marks(marks);
    }

    void average_marks(int[] marks)
    {
        int n = marks.length;
        int sum = 0;

        for (int mark : marks) {
            sum += mark;
        }
        this.Average = sum*1.0f/n;
    }
}

class Main
{
    public  static void main(String[] args)
    {
        Scanner obj = new Scanner(System.in);
        System.out.print("Enter number :- ");
        int n = obj.nextInt();
        Student[] stud = new Student[n];
        int highest = 0;
        int index = 0;
        for (int i = 0;i<n;i++)
        {
            obj.nextLine();
            System.out.print("Enter student name :- ");
```

```
      String name = obj.nextLine();
      System.out.print("Enter id :- ");
      int roll = obj.nextInt();
      int[] marks = new int[5];

      for (int j = 0;j<5;j++)
      {
        System.out.print("Enter marks of sub " + j +  " :- ");
        marks[j] = obj.nextInt();
      }

      stud[i] = new Student(name,roll,marks);
      if(stud[i].Average > highest )
      {
        index = i;
      }
    }

    System.out.print(stud[index].name + "\n" + stud[index].roll + "\n"
+ stud[index].Average    );

  }

}
```

Screenshot



```
"C:\Program Files\Eclipse Adoptium\jdk-17.0.8.101-hotspo
 .jar=60564:C:\Users\LENOVO\AppData\Local\Programs\Intel
 4\java\labs\out\production\labs" Main
Enter number :- 3
Enter student name :- nisarg
Enter id :- 5
Enter marks of sub 0 :- 80
Enter marks of sub 1 :- 70
Enter marks of sub 2 :- 42
Enter marks of sub 3 :- 63
Enter marks of sub 4 :- 45
Enter student name :- janmang
Enter id :- 6
Enter marks of sub 0 :- 45
Enter marks of sub 1 :- 89
Enter marks of sub 2 :- 74
Enter marks of sub 3 :- 56
Enter marks of sub 4 :- 41
Enter student name :- neha
Enter id :- 3
Enter marks of sub 0 :- 89
Enter marks of sub 1 :- 41
Enter marks of sub 2 :- 23
Enter marks of sub 3 :- 89
Enter marks of sub 4 :- 56
neha
3
59.6
Process finished with exit code 0
```

LAB 03

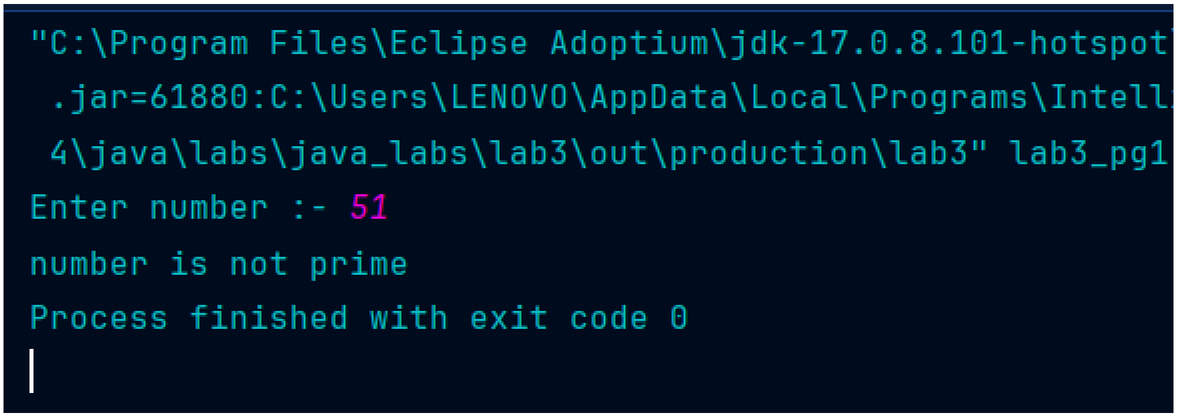
Question 1

Write a Java program that checks for prime number using the object oriented approach. [Hint: create a class NumberClass with a member value and method isPrimeNumber()]

Solution

```
import java.util.*;

class NumberClass {
    private int value ;
    private boolean[] isprime;

    NumberClass(int val)
    {
        this.value = val;
        this.isprime = new boolean[val + 1];
        for(int i = 0;i<=val;i++) isprime[i] = true;
    }

    boolean isPrimeNumber()
    {

        isprime[0] = isprime[1] = false;
        for(int i = 2;i<=value ; i++)
        {
            if(isprime[i])
            {
                for(int j = 2*i ; j<=value;j += i)
                {
                    isprime[j] = false;
                }
            }
        }

        return isprime[value];
    }
}

class lab3_pg1 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner obj = new Scanner(System.in);
        System.out.print("Enter number :- ");
        int n = obj.nextInt();
        NumberClass c1 = new NumberClass(n);

        if(c1.isPrimeNumber())  System.out.print("Number is
prime");
        else System.out.print("number is not prime ");

    }

}
```

Screenshot



```
"C:\Program Files\Eclipse Adoptium\jdk-17.0.8.101-hotspot
 .jar=61880:C:\Users\LENOVO\AppData\Local\Programs\Intelli
 4\java\labs\java_labs\lab3\out\production\lab3" lab3_pg1
Enter number :- 51
number is not prime
Process finished with exit code 0
```

Question 2

Create two classes:

class Person

Derive a class Student from class Person.

Person

- name : String
- age : int
+ Person()
+ Person(name : String, age : int)
+ getName() : String
+ getAge() : int
+ setName(name : String) : void
+ setAge(age : int) : void
+ toString() : String

Student

- rollno : int
- marks : double[]
+ Student()

+ Student(rollno : int)
+ Student(rollno : int, marks : double[])
+ Student(rollno : int, name : String, age : int, marks : double[])
+ getRollno() : int
+ getMarks() : double[]
+ setRollno(rollno: int) : void
+ setMarks(marks : double[]) : void
+ toString() : String
+ displayDetails() : void

Add the following to Student class:

● a static variable count(to count the number of objects)
● a static block to initialize count variable to zero
● a static method String getCount() that returns the number of student objects created
● Write a TestStudent class containing the main() method.
● Store the details of 3 students by creating an array of objects of Student class and display
the student who has highest average amongst the three students as follows using
displayDetails() method for that object:
e.g.
RollNo = 100
Name = ABC
Age = 20
Marks=78 86 88 67 92
● Create one more object of the Student class and then call the getCount() to display the
number of Student objects created.

Person class

```
public class Person {

    private String name;
    private int age;

    public Person() {

    }

    ;

    public Person(String name, int age) {
        super();
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }

    @Override
    public String toString() {
        return "Person [name=" + name + ", age=" + age + "]";
    }
}
```

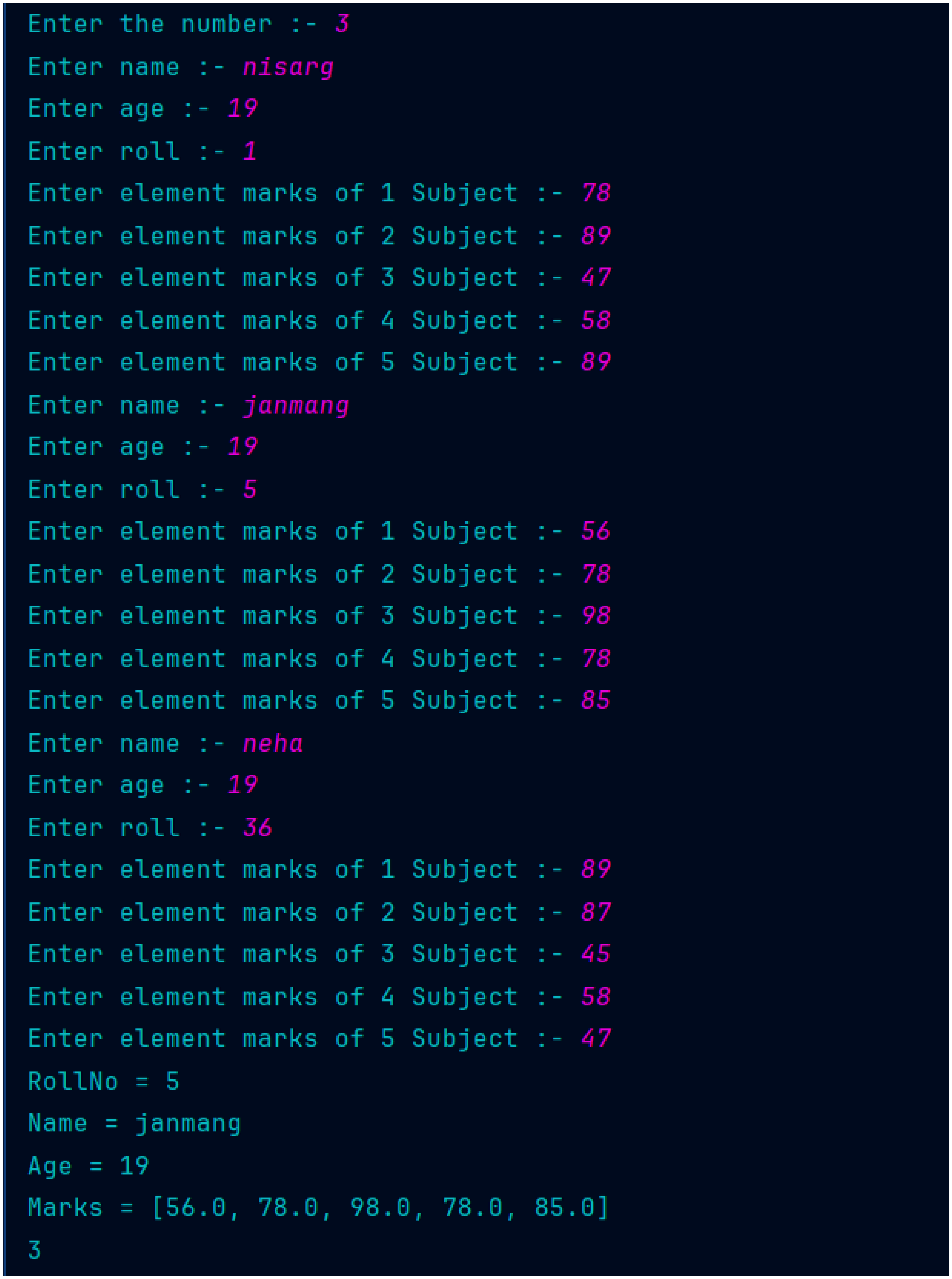
Student Class

```
import java.util.Arrays;

class Student extends Person{
    private int rollno ;
    private double[] marks ;
    public static int count ;
    static {
        count = 0;
    }
    public Student() {
        super();
        count++;
    }
    public Student(int rollno, double[] marks) {
        super();
        this.rollno = rollno;
        this.marks = marks;
        count++;
    }
    @Override
    public String toString() {
        return "Student [rollno=" + rollno + ", marks=" +
Arrays.toString(marks) + "]";
    }
    public int getRollno() {
        return rollno;
    }
    public void setRollno(int rollno) {
        this.rollno = rollno;
    }
    public double[] getMarks() {
        return marks;
    }
    public void setMarks(double[] marks) {
        this.marks = marks;
    }
    public Student(String name, int age, int rollno, double[] marks)
{
        super(name, age);
        this.rollno = rollno;
        this.marks = marks;
        count++;
    }

    void displayDetails()
    {
        System.out.println("RollNo = " + this.rollno + "\nName = "+
super.getName() + "\nAge = " + super.getAge() + "\nMarks = " +
Arrays.toString(this.marks));

    }
    static int  getCount()
    {
        return count;

    }
}
```

TestClass

```
import java.util.Scanner;

public class lab3_pg2 {

    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        System.out.print("Enter the number :- ");
        int n = obj.nextInt();
        Student[] s = new Student[n];
        int index  = 0;
        double highest = 0;

        for(int i = 0;i<n;i++)
        {
            obj.nextLine();
            System.out.print("Enter name :- ");
            String name = obj.nextLine();
            System.out.print("Enter age :- ");
            int age = obj.nextInt();
            System.out.print("Enter roll :- ");
            int roll = obj.nextInt();
            double[] marks = new double[5];
            double sum = 0;
            for(int j = 0;j<5;j++)
            {
                System.out.print("Enter element marks of " + (j+1) + "
Subject :- " );
                marks[j] = obj.nextDouble();
                sum += marks[j];
            }
            s[i] = new Student(name,age,roll,marks);
            if(sum > highest)
            {
                highest = sum;
                index = i;
            }
        }

        s[index].displayDetails();
        System.out.println(Student.getCount());


    }
}
```

Screenshot



```
Enter the number :- 3
Enter name :- nisarg
Enter age :- 19
Enter roll :- 1
Enter element marks of 1 Subject :- 78
Enter element marks of 2 Subject :- 89
Enter element marks of 3 Subject :- 47
Enter element marks of 4 Subject :- 58
Enter element marks of 5 Subject :- 89
Enter name :- janmang
Enter age :- 19
Enter roll :- 5
Enter element marks of 1 Subject :- 56
Enter element marks of 2 Subject :- 78
Enter element marks of 3 Subject :- 98
Enter element marks of 4 Subject :- 78
Enter element marks of 5 Subject :- 85
Enter name :- neha
Enter age :- 19
Enter roll :- 36
Enter element marks of 1 Subject :- 89
Enter element marks of 2 Subject :- 87
Enter element marks of 3 Subject :- 45
Enter element marks of 4 Subject :- 58
Enter element marks of 5 Subject :- 47
RollNo = 5
Name = janmang
Age = 19
Marks = [56.0, 78.0, 98.0, 78.0, 85.0]
3
```

LAB 04

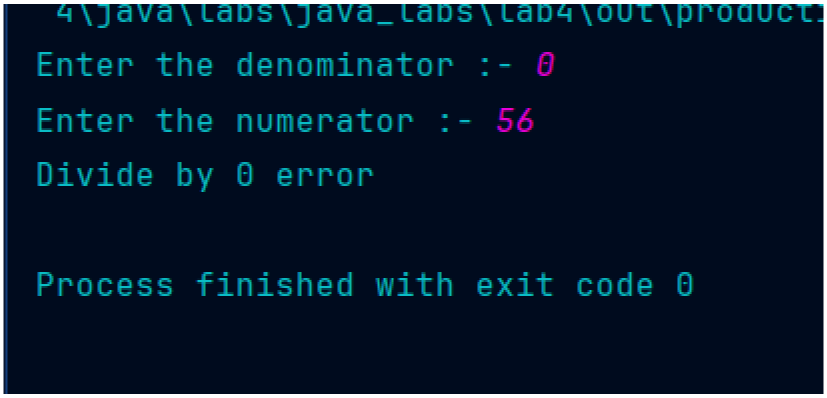
Question 1)

Write a program that catches the divide-by-zero exception using the try-catch mechanism. Take a numeric value and perform division by zero. Catch the ArithmeticException.

Solution

```
import java.util.Scanner;

public class lab4_pg1 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the denominator :- ");
        int denominator = sc.nextInt();
        System.out.print("Enter the numerator :- ");
        int numerator = sc.nextInt();
        try{
            int ans = numerator/denominator;
        }
        catch (ArithmeticException e) {
            System.out.println("Divide by 0 error");
        }
    }
}
```

Screenshot



```
 4\java\labs\java_labs\lab4\out\producti
Enter the denominator :- 0
Enter the numerator :- 56
Divide by 0 error

Process finished with exit code 0
```

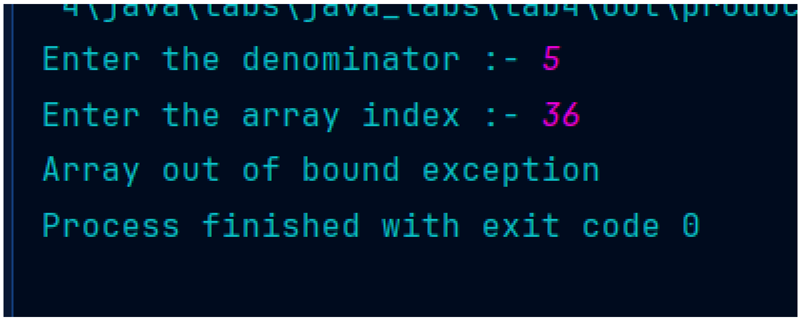
Question 2)

Write a java program using multiple catch blocks. Create a class CatchExercise inside the try block declare an array a[] and initialize with value a[5] =30/5; . In each catch block show Arithmetic exception and ArrayIndexOutOfBoundsException.

Solution

```
import java.util.Scanner;


public class CatchExercise {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        System.out.print("Enter the denominator :- ");
        int deno = obj.nextInt();
        System.out.print("Enter the array index :- ");
        int index = obj.nextInt();
        int[] arr  = new int[5];
        try{
            arr[index] = 30/5;
            int ans = 5/deno;
        }
        catch(ArithmeticException a)
        {
            System.out.print("Divide by 0 exception");
        }
        catch(ArrayIndexOutOfBoundsException aob)
        {
            System.out.print("Array out of bound exception ");
        }

    }
}
```

Screenshots



```
  4\java\labs\java_labs\lab4\out\produc
Enter the denominator :- 5
Enter the array index :- 36
Array out of bound exception
Process finished with exit code 0
```

Question 3)

Write a program that demonstrates use of finally block. Observe the output of your program for different cases as mentioned below.

● Case A: **exception does not occur. Perform 25/5 mathematical operation. Catch the NullPointerException.**

- **Case B: exception occurs but not handled. Perform 25/0 mathematical operation. Catch NullPointerException.**

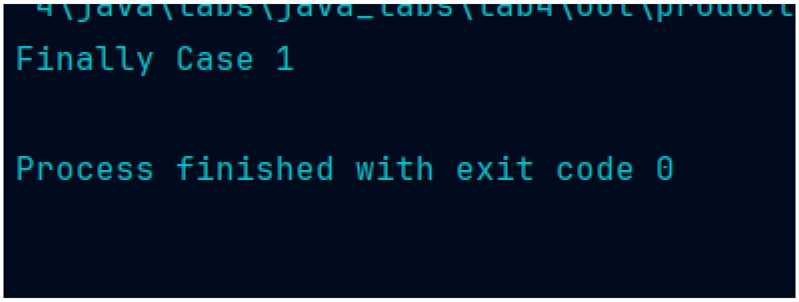
- **Case C: exception occurs and handled. Perform 25/0 mathematical operation. Catch ArithmeticException**

Solution

Case 1 :-

```
public class FinallyExercise_Case_1 {
    public static void main(String[] args) {
        try{
            int ans = 25/5;
        }
        catch(NullPointerException npe){
            System.out.println("Nullpointer Exception");

        }
        finally {
            System.out.println("Finally Case 1");
        }
    }

}
```

Screenshot case 1 :-

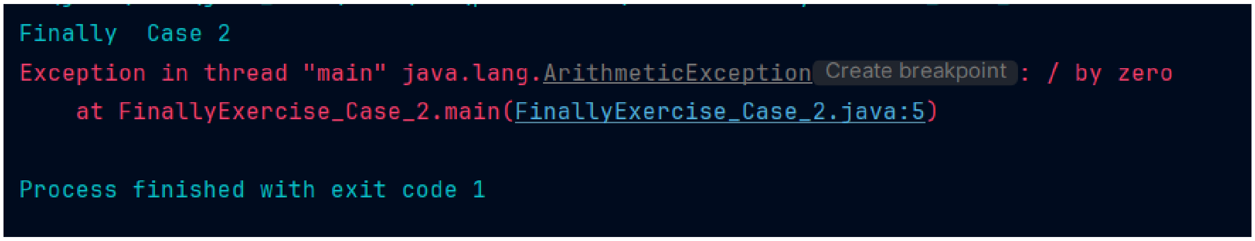


```
4\java\labs\java_labs\lab4\out\product
Finally Case 1

Process finished with exit code 0
```

Case 2 :-

```
public class FinallyExercise_Case_2 {
    public static void main(String[] args) {

        try{
            int ans = 25/0;
        }
        catch (NullPointerException nep) {
            System.out.println("NullPointer Exception");
        }
        finally {
            System.out.println("Finally  Case 2");
        }
    }
}
```

ScreenShot Case 2 :-

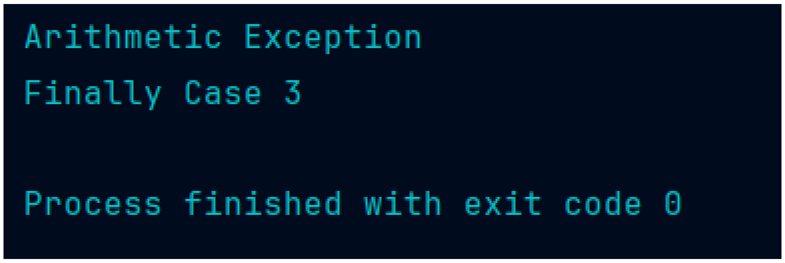


```
Finally  Case 2
Exception in thread "main" java.lang.ArithmeticException: / by zero
    at FinallyExercise_Case_2.main(FinallyExercise_Case_2.java:5)

Process finished with exit code 1
```

Case 3 :-

```
public class FinallyExercise_Case_3 {
    public static void main(String[] args) {
        try{
            int ans = 25/0;

        }
        catch (ArithmeticException ae)
        {
            System.out.println("Arithmetic Exception");
        }
        finally {
            System.out.println("Finally Case 3");
        }
    }
}
```

ScreenShot Case 3:-



```
Arithmetic Exception
Finally Case 3

Process finished with exit code 0
```

Question 4)

Create an interface Account with two methods: deposit and withdraw. Create class SavingsAccount which implements the interface. Write a custom Exception handler for SavingsAccount to handle the scenarios when the withdrawn amount is larger than the balance in the account.

Solution

Interface Account ➖

```
public interface Account {
    void deposite (int amount) ;

    void withdraw (int amount) throws Custom_Exception;
}
```

Class SavingsAccount :-

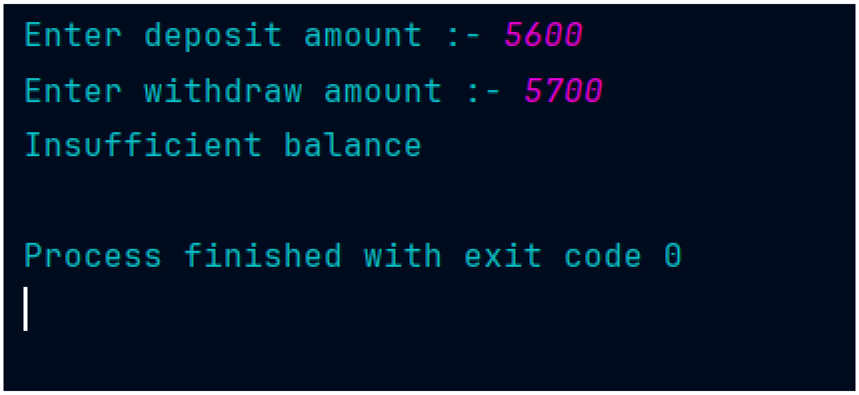
```
public class SavingsAccount implements Account{
    private int balance = 50000;

    @Override
    public void deposite(int amount) {
        this.balance = amount;
    }

    @Override
    public void withdraw(int amount) throws Custom_Exception{

        if(amount> this.balance)
        {
            throw new Custom_Exception("Insufficient balance");
        }
    }
}
```

Solution :-

```
import java.util.Scanner;
```

```
public class Lab4_pg4 {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        System.out.print("Enter deposit amount :- ");
        int amount = obj.nextInt();
        System.out.print("Enter withdraw amount :- ");
        int wamount = obj.nextInt();

        SavingsAccount a1 = new SavingsAccount();

        try{
            a1.deposite(amount);
            a1.withdraw(wamount);
        }
        catch (Custom_Exception c)
        {
            System.out.println(c.getMessage());
        }
    }
}
```

ScreenShots :-



```
Enter deposit amount :- 5600
Enter withdraw amount :- 5700
Insufficient balance


Process finished with exit code 0
```