Laboratory Work

Subject: Java Technologies

Branch: B.Tech. (CE)

Semester: IV

Batch: A2

Student Roll No: CE030

Student Name: Paramar Lakhman

Department of Computer Engineering,

Faculty of Technology,

Dharmsinh Desai University, Nadiad – 387001.

Gujarat, INDIA.

LAB_1_4

LAB_1

Q.1

Write a Java program to display “Hello World”.

Ans.

```
package inheritance;

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

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Hello World
Process finished with exit code 0

Q.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.

Ans.

```
package inheritance;

import java.util.Scanner;
```

```
public class DivisibleByThreeAndFive {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the value of n: ");
        int n = scanner.nextInt();
        scanner.close();

        System.out.println("Numbers between 1 and " + n + " divisible by 3, 5, and both:");

        for (int i = 1; i <= n; i++) {
            if (i % 3 == 0 && i % 5 == 0) {
                System.out.println(i + " (divisible by 3 and 5)");
            } else if (i % 3 == 0) {
                System.out.println(i + " (divisible by 3)");
            } else if (i % 5 == 0) {
                System.out.println(i + " (divisible by 5)");
            }
        }
    }
}
```

Output:

Enter the value of n: 10
Numbers between 1 and 10 divisible by 3, 5, and both:
3 (divisible by 3)
5 (divisible by 5)
6 (divisible by 3)
9 (divisible by 3)
10 (divisible by 5)
Process finished with exit code 0

Q.3

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!".

Ans.

```
import java.util.Scanner;

public class Greeter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter your name: ");
        String userName = scanner.nextLine();
        scanner.close();
        System.out.println("Hello " + userName + "!");
    }
}
```

Output:

Enter your name: Hello lakhman!

Q.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.

Ans.

```
package inheritance;

import java.util.Scanner;

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

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

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

        System.out.print("Enter marks for 5 subjects (separated by spaces): ");
        int marks1 = scanner.nextInt();
        int marks2 = scanner.nextInt();
        int marks3 = scanner.nextInt();
        int marks4 = scanner.nextInt();
        int marks5 = scanner.nextInt();

        scanner.close();

        int totalMarks = marks1 + marks2 + marks3 + marks4 + marks5;
        double average = totalMarks / 5.0;

        System.out.println("Name: " + name);
        System.out.println("Roll No.: " + rollNo);
        System.out.println("Average: " + (int) average);

        if (average >= 90) {
            System.out.println("Grade: A");
        } else if (average >= 80) {
            System.out.println("Grade: B");
        } else if (average >= 70) {
            System.out.println("Grade: C");
        } else if (average >= 60) {
            System.out.println("Grade: D");
        } else {
            System.out.println("Grade: F");
        }
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA

Enter Name: lakhman
Enter Roll No.: 30
Enter marks for 5 subjects (separated by spaces): 80 90 96 89 97
Name: Lakhman
Roll No.: 30
Average: 90
Grade: A
Process finished with exit code 0

Q.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.

Test the functionalities using the main() method of the Tester class.

Sample Input and Output:

Sample Input	Sample Output
{68,79,86,99,23,2,41,100}	256
{1,2,3,4,5,6,7,8,9,10}	30

Ans.

```
package inheritance;

public class Tester {
    public static void main(String[] args) {
        int[] numbers1 = {68, 79, 86, 99, 23, 2, 41, 100};
        int[] numbers2 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

        int sum1 = calculateSumOfEvenNumbers(numbers1);
        int sum2 = calculateSumOfEvenNumbers(numbers2);

        System.out.println(sum1);
        System.out.println(sum2);
    }

    public static int calculateSumOfEvenNumbers(int[] numbers) {
        int sum = 0;
```

```
        for (int number : numbers) {
            if (number % 2 == 0) {
                sum += number;
            }
        }

        return sum;
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
256
30
Process finished with exit code 0

Q.6

1. 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.

Ans.

```
public class MatrixOperations {
    public static void main(String[] args) {
        int[][] matrix1 = {
            {1, 2, 3},
            {4, 5, 6},
            {7, 8, 9}
        };

        int[][] matrix2 = {
            {9, 8, 7},
            {6, 5, 4},
            {3, 2, 1}
        };
```

```
    System.out.println("Matrix 1:");
    displayMatrix(matrix1);

    System.out.println("\nMatrix 2:");
    displayMatrix(matrix2);

    System.out.println("\nMatrix Addition:");
    int[][] sumMatrix = addMatrices(matrix1, matrix2);
    displayMatrix(sumMatrix);

    System.out.println("\nMatrix Multiplication:");
    int[][] productMatrix = multiplyMatrices(matrix1, matrix2);
    displayMatrix(productMatrix);
}

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 rows1 = matrix1.length;
    int columns1 = matrix1[0].length;
    int columns2 = matrix2[0].length;
```

```
        int[][] resultMatrix = new int[rows1][columns2];

        for (int i = 0; i < rows1; i++) {
            for (int j = 0; j < columns2; j++) {
                for (int k = 0; k < columns1; k++) {
                    resultMatrix[i][j] += matrix1[i][k] * matrix2[k][j];
                }
            }
        }

        return resultMatrix;
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Matrix 1:
1 2 3
4 5 6
7 8 9
Matrix 2:
9 8 7
6 5 4
3 2 1
Matrix Addition:
10 10 10
10 10 10
10 10 10
Matrix Multiplication:
30 24 18
84 69 54
138 114 90
Process finished with exit code 0

Q.7

Practice Problems:

- Given an integer numRows, return the first numRows of Pascal's triangle using a jagged array. In Pascal's triangle, each number is the sum of the two numbers directly above it.

Sample Input and Output:

Sample Input	Sample Output
numRows = 5	[[1],[1,1],[1,2,1],[1,3,3,1],[1,4,6,4,1]]
numRows = 1	[[1]]

Ans.

```
public class PascalsTriangle {
    public static void main(String[] args) {
        int numRows = 5;
        int[][] pascalsTriangle = generatePascalsTriangle(numRows);

        for (int[] row : pascalsTriangle) {
            for (int num : row) {
                System.out.print(num + " ");
            }
            System.out.println();
        }
    }

    public static int[][] generatePascalsTriangle(int numRows) {
        if (numRows <= 0) {
            return new int[0][0];
        }

        int[][] triangle = new int[numRows][];
        for (int i = 0; i < numRows; i++) {
            triangle[i] = new int[i + 1];
            triangle[i][0] = 1;

            for (int j = 1; j < i; j++) {
                triangle[i][j] = triangle[i - 1][j - 1] + triangle[i - 1][j];
            }

            triangle[i][i] = 1;
        }
```

```
        return triangle;
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
1
1 1
121
1 3 31
1 4 6 41
Process finished with exit code 0

Q.8

- Write a Java program to convert the integer entered by the user into a roman numeral. Roman numerals are represented by seven different symbols: I, V, X, L, C, D and M.

Symbol	Value
I	1
V	5
X	10
L	50
C	100
D	500
M	1000

Ans.

```
import java.util.Scanner;

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

        System.out.print("Enter an integer: ");
```

```
        int num = scanner.nextInt();

        scanner.close();

        if (num < 1 || num > 3999) {
            System.out.println("Please enter an integer between 1 and 3999.");
        } else {
            String romanNumeral = intToRoman(num);
            System.out.println("Roman numeral: " + romanNumeral);
        }
    }

    public static String intToRoman(int num) {
        int[] values = {1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1};
        String[] symbols = {"M", "CM", "D", "CD", "C", "XC", "L", "XL", "X", "IX", "V", "IV", "I"};

        StringBuilder result = new StringBuilder();

        for (int i = 0; i < values.length; i++) {
            while (num >= values[i]) {
                num -= values[i];
                result.append(symbols[i]);
            }
        }

        return result.toString();
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Enter an integer: 200
Roman numeral: CC
Process finished with exit code 0

Q.9

- A group of MIT friends decide to run the Boston Marathon. Their names and times (in minutes) are below: Name Time (minutes) Elena 341 Thomas 273 Hamilton 278 Suzie 329 Phil 445 Matt 402 Alex 388 Emma 275 John 243 James 334 Jane 412 Emily 393 Daniel 299 Neda 343 Aaron 317 Kate 265 Find the fastest runner. Print the name and his/her time (in minutes).

Optional: Find the second fastest runner. Print the name and his/her time (in minutes).

Ans.

```
package inheritance;

public class BostonMarathon {
    public static void main(String[] args) {
        String[] names = {"Elena", "Thomas", "Hamilton", "Suzie", "Phil", "Matt", "Alex", "Emma", "John",
"James", "Jane", "Emily", "Daniel", "Neda", "Aaron", "Kate"};
        int[] times = {341, 273, 278, 329, 445, 402, 388, 275, 243, 334, 412, 393, 299, 343, 317, 265};

        String fastestRunner = findFastestRunner(names, times);
        System.out.println(fastestRunner);

        String secondFastestRunner = findSecondFastestRunner(names, times);
        System.out.println(secondFastestRunner);
    }

    public static String findFastestRunner(String[] names, int[] times) {
        String fastestRunner = "";
        int fastestTime = Integer.MAX_VALUE;

        for (int i = 0; i < times.length; i++) {
            if (times[i] < fastestTime) {
                fastestTime = times[i];
                fastestRunner = names[i];
            }
        }

        return "Fastest Runner: " + fastestRunner + ", Time: " + fastestTime + " minutes";
    }

    public static String findSecondFastestRunner(String[] names, int[] times) {
        String fastestRunner = "";
        int fastestTime = Integer.MAX_VALUE;
        String secondFastestRunner = "";
        int secondFastestTime = Integer.MAX_VALUE;
```

```
        for (int i = 0; i < times.length; i++) {
            if (times[i] < fastestTime) {
                secondFastestTime = fastestTime;
                secondFastestRunner = fastestRunner;

                fastestTime = times[i];
                fastestRunner = names[i];
            } else if (times[i] < secondFastestTime) {
                secondFastestTime = times[i];
                secondFastestRunner = names[i];
            }
        }

        return "Second Fastest Runner: " + secondFastestRunner + ", Time: " + secondFastestTime + "
minutes";
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Fastest Runner: John, Time: 243 minutes
Second Fastest Runner: Kate, Time: 265 minutes
Process finished with exit code 0

LAB_2

Q.1

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

Ans.

```
public class HiCount {

    public static int countHiOccurrences(String inputString) {
        String lowerCaseInput = inputString.toLowerCase();
        int count = 0;

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

        return count;
    }

    public static void main(String[] args) {
        String inputString = "hi there, hi how are you? Hi, this is a test.";
        int result = countHiOccurrences(inputString);
        System.out.println("The substring \"hi\" appears " + result + " times in the given string.");
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA Community Edition.app/Contents/lib/idea_rt.jar=57141:/Users/lakhman/Applications/IntelliJ IDEA Community Edition.app/Contents/bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/lakhman/Desktop/sem 4 coding/java/javabasics/out/production/java

basics lab2.HiCount
The substring "hi" appears 4 times in the given string.

Process finished with exit code 0

Q.2

1. 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].

Ans.

```
import java.util.Scanner;

public class PalindromeChecker {

    public static boolean isPalindrome(String inputString) {
        String cleanedInput = inputString.replaceAll("\\s", "").toLowerCase();
        String reversedString = "";

        for (int i = cleanedInput.length() - 1; i >= 0; i--) {
            reversedString += cleanedInput.charAt(i);
        }

        return cleanedInput.equals(reversedString);
    }

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

        System.out.print("Enter a string: ");
        String inputString = scanner.nextLine();

        if (isPalindrome(inputString)) {
            System.out.println("\"" + inputString + "\" is a palindrome string.");
        } else {
            System.out.println("\"" + inputString + "\" is not a palindrome string.");
        }

        scanner.close();
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Enter a string: yey
"yey" is a palindrome.
Process finished with exit code 0

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Enter a string: lakhman
"Zakhman" is not a palindrome.
Process finished with exit code 0

Q.3

1. 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.

Ans.

```
import java.util.Scanner;

public class NameAbbreviation {

    public static String abbreviateName(String fullName) {
        String[] names = fullName.split("\\s");
        StringBuilder abbreviation = new StringBuilder();

        for (int i = 0; i < names.length - 1; i++) {
            abbreviation.append(names[i].charAt(0)).append(".");
        }

        abbreviation.append(names[names.length - 1]);

        return abbreviation.toString();
    }
```

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

        System.out.print("Enter your full name: ");
        String fullName = scanner.nextLine();

        System.out.println("Abbreviation: " + abbreviateName(fullName));

        scanner.close();
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Enter your full name: parmar lakhman jivabhai
Abbreviation: p.l. jivabhai
Process finished with exit code 0

Q.4

1. 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.

Ans.

```
public class Tester {

    public static String removeWhiteSpaces(String str) {
        return str.replaceAll("\\s", "");
    }

    public static void main(String[] args) {
        String inputString = "This is a test string with white spaces.";
```

```
        System.out.println("Original String: " + inputString);
        System.out.println("Modified String: " + removeWhiteSpaces(inputString));
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Original String: This is a test string with white spaces.
Modified String: Thisisateststringwithwhitespaces.
Process finished with exit code 0

Q.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:

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

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.

Ans.

```
public class Student {
    private int rollNo;
    private String name;
    private int[] marks;

    public Student() {
        this.rollNo = 0;
        this.name = "";
        this.marks = new int[5];
    }

    public Student(int rollNo, String name) {
```

```
        this();
        this.rollNo = rollNo;
        this.name = name;
    }

    public Student(int rollNo, String name, int[] marks) {
        this(rollNo, name);
        this.marks = marks;
    }

    public double findAverage() {
        double sum = 0;
        for (int mark : marks) {
            sum += mark;
        }
        return sum / marks.length;
    }
}

public class TestStudent {
    public static void main(String[] args) {
        Student student1 = new Student();
        System.out.println("Student 1 Average: " + student1.findAverage());

        Student student2 = new Student(101, "John Doe");
        System.out.println("Student 2 Average: " + student2.findAverage());

        int[] marks = {90, 85, 78, 92, 88};
        Student student3 = new Student(102, "Jane Doe", marks);
        System.out.println("Student 3 Average: " + student3.findAverage());
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA
1
Lakhman

80 90 90 90 90
2
kunal
90 90 90 90 90
3
dev
89 90 89 78 99
Student with the highest average marks: kunal
Process finished with exit code 0

Practice Problems:

Q.6

- Write a Java program to find the second most frequent character in a given string.

E.g.

Input	Success
Output	c

Ans.

```
import java.util.*;

class Main {
    static void countFreq(String s, int[] freq) {
        for (int i = 0; i < s.length(); i++) {
            int temp = s.charAt(i);
            if (temp >= 97 && temp <= 122) {
                temp = temp - 32;
            }
            freq[temp - 65] = freq[temp - 65] + 1;
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] freq = new int[26];
```

```
        for (int i = 0; i < 26; i++) {
            freq[i] = 0;
        }
        String item = scanner.nextLine();
        countFreq(item, freq);
        int max = 0;
        int index = 0;
        for (int i = 0; i < 26; i++) {
            if (freq[i] > max) {
                max = freq[i];
                index = i;
            }
        }
        index += 65;
        char x = (char) index;
        System.out.print(x);
        System.out.print(" ");
    }
}
```

Output:

```
Finished in 141 ms
Enter a string:ccccccadfasdfcccc
Most frequent letter: c
Finished in 166 ms
Enter a string:AccccAAAAAAAWFEF
Most frequent letter: A
```

Q.7

- Write a program that prints the words of a sentence given as input in reverse order.

E.g.

Input	This is a java programming lab
Output	lab programming java a is This

Ans.

```
import java.util.Scanner;

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

        System.out.println("Enter a sentence: ");
        String sentence = scanner.nextLine();

        String[] words = sentence.split("\\s+"); // Split the sentence into words

        System.out.println("Reversed sentence:");
        for (int i = words.length - 1; i >= 0; i--) {
            System.out.print(words[i] + " ");
        }
    }
}
```

Output:

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -
javaagent:/Users/lakhman/Applications/IntelliJ IDEA
Enter a sentence: my name is lakhman
Reversed sentence: lakhman is name my
Process finished with exit code 0
```

Q.8

- Write a Java program to find the longest palindrome substring within a string.

E.g.

Input	thequickbrownfoxxofnworbquickthe
Output	The longest palindrome substring in the given string is: brownfoxxofnworb The length of the palindromic substring is: 16

Ans.

```
public class LongestPalindromeSubstring {
public static void main(String[] args) {
String input = "thequickbrownfoxxofnworbquickthe";

String longestPalindrome = findLongestPalindrome(input);

System.out.println("The longest palindrome substring in the given
string is:");
System.out.println(longestPalindrome);
System.out.println("The length of the palindromic substring is: " +

longestPalindrome.length());
}

private static String findLongestPalindrome(String input) {
int maxLength = 0;
String longestPalindrome = "";

for (int i = 0; i < input.length(); i++) {
for (int j = i + 1; j <= input.length(); j++) {
String sub = input.substring(i, j);
if (isPalindrome(sub) && sub.length() > maxLength) {
maxLength = sub.length();
longestPalindrome = sub;
}
}
}

return longestPalindrome;
}

private static boolean isPalindrome(String str) {
int left = 0;
int right = str.length() - 1;

while (left < right) {
if (str.charAt(left) != str.charAt(right)) {
return false;
}
left++;
```

```
right--;
}
return true;
}

}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA Community Edition.app/Contents/lib/idea_rt.jar=57152:/Users/lakhman/Applications/IntelliJ IDEA Community Edition.app/Contents/bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/lakhman/Desktop/sem 4 coding/java/javabasics/out/production/java basics lab2.LongestPalindromeSubstring

The longest palindrome substring in the given string is: brownfoxxofnworb
The length of the palindromic substring is: 16

Process finished with exit code

LAB_3

Q.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()]

Ans.

```
import java.util.*;
class NumberClass {
```

```
private int value;
public NumberClass(int value) {
this.value = value;
}
public boolean isPrimeNumber() {
if (value <= 1) {
return false;
} else {
for (int i = 2; i <= Math.sqrt(value); i++) {
if (value % i == 0) {
return false;
}
}
return true;
}
}
}
class Main {
public static void main(String[] args) {
Scanner scan=new Scanner(System.in);
int num1=scan.nextInt();
NumberClass numberObj = new NumberClass(num1);
if (numberObj.isPrimeNumber()) {
System.out.println(num1 + " is a prime number.");
} else {
System.out.println(num1 + " is not a prime number.");
}
}
}
```

Output:

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -
javaagent:/Users/lakhman/Applications/IntelliJ IDEA
23
23 is a prime number.
Process finished with exit code 0

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -
javaagent:/Users/lakhman/Applications/IntelliJ IDEA
```

22
22 is not a prime number.
Process finished with exit code 0

Q.2

1. 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.

Ans.

```
import java.util.Scanner;

class Person {
    private String name;
    private int age;

    public Person() {
        this.name = "";
        this.age = 0;
    }

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

    public String getName() {
        return name;
```

```
    }

    public int getAge() {
        return age;
    }

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

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

    public String toString() {
        return "Name: " + name + ", Age: " + age;
    }
}

class Student extends Person {
    private int rollno;
    private double[] marks;
    private static int count = 0;

    static {
        count = 0;
    }

    public Student() {
        super();
        this.rollno = 0;
        this.marks = new double[0];
        count++;
    }

    public Student(int rollno) {
        super();
        this.rollno = rollno;
        this.marks = new double[0];
        count++;
    }
```

```
public Student(int rollno, double[] marks) {
    super();
    this.rollno = rollno;
    this.marks = marks;
    count++;
}

public Student(int rollno, String name, int age, double[] marks) {
    super(name, age);
    this.rollno = rollno;
    this.marks = marks;
    count++;
}

public int getRollno() {
    return rollno;
}

public double[] getMarks() {
    return marks;
}

public void setRollno(int rollno) {
    this.rollno = rollno;
}

public void setMarks(double[] marks) {
    this.marks = marks;
}

public static String getCount() {
    return "Number of Student objects created: " + count;
}

public void displayDetails() {
    System.out.println("RollNo = " + rollno);
    System.out.println("Name = " + getName());
    System.out.println("Age = " + getAge());
    System.out.print("Marks = ");
    for (double mark : marks) {
        System.out.print(mark + " ");
    }
```

```
        System.out.println("\n");
    }

    public String toString() {
        return super.toString() + ", RollNo: " + rollno;
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        Student[] students = new Student[3];
        students[0] = new Student(100, "lakhman", 20, new double[]{78, 86, 88, 67, 92});
        students[1] = new Student(101, "shivansh", 21, new double[]{80, 75, 90, 60, 85});
        students[2] = new Student(102, "dev", 22, new double[]{85, 92, 78, 70, 88});

        Student highestAverageStudent = students[0];
        double highestAverage = students[0].getMarks().length > 0 ?
                calculateAverage(students[0].getMarks()) : 0;

        for (int i = 1; i < students.length; i++) {
            double avg = calculateAverage(students[i].getMarks());
            if (avg > highestAverage) {
                highestAverage = avg;
                highestAverageStudent = students[i];
            }
        }

        System.out.println("Student with the highest average marks:");
        highestAverageStudent.displayDetails();

        Student extraStudent = new Student(103);
        System.out.println(Student.getCount());

        // Close the scanner
        scan.close();
    }

    private static double calculateAverage(double[] marks) {
        double sum = 0;
        for (double mark : marks) {
            sum += mark;
```

```
        }
        return marks.length > 0 ? sum / marks.length : 0;
    }
}
```

Output:

LAB_4

Q.1

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.

Ans.

```
import java.util.Scanner;

public class DivideByZeroException {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");

        try {
            int number = scanner.nextInt();
            int result = number / 0;
            System.out.println("Result: " + result);
        } catch (ArithmeticException e) {
            System.out.println("ArithmeticException caught: Division by zero not allowed!");
        }
        scanner.close();
```

```
    }
}
```

Output:

java -cp /tmp/Idkbq5Vo27 DivideByZeroException
Enter a number: 20
ArithmeticException caught: Division by zero not allowed!

Q.2

Write a java program using multiple catch blocks. Create a class CatchExercise, inside the try block declare an array a[] with size of 5 elements and initialize with value a[5] =30/5 . Using Multiple catch blocks handle ArithmeticException and ArrayIndexOutOfBoundsException.

Ans.

```
public class CatchExercise {
    public static void main(String[] args) {
        try {
            int[] a = new int[5];
            a[5] = 30 / 7645;
        } catch (ArithmeticException e) {
            System.out.println("ArithmeticException caught: Division by zero not allowed!");
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("ArrayIndexOutOfBoundsException caught: Array index out of range!");
        }
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA Community

ArrayIndexOutOfBoundsException caught: Array index out of range!

Process finished with exit code 0

ArithmeticException caught: Division by zero not allowed!

Process finished with exit code 0

Q.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

Ans.

```
public class FinallyBlockDemo {
    public static void main(String[] args) {
        // Case A: No exception
        try {
            int result = 25 / 5;
            System.out.println("Result: " + result);
        } catch (NullPointerException e) {
            System.out.println("NullPointerException caught!");
        } finally {
            System.out.println("Case A: Finally block executed.");
        }

        // Case B: Exception occurs but not handled
        try {
            int result = 25 / 0; // This will throw an ArithmeticException
            System.out.println("Result: " + result);
        } catch (NullPointerException e) {
            System.out.println("NullPointerException caught!");
        } finally {
            System.out.println("Case B: Finally block executed.");
        }

        // Case C: Exception occurs and handled
```

```
        try {
            int result = 25 / 0; // This will throw an ArithmeticException
            System.out.println("Result: " + result);
        } catch (ArithmeticException e) {
            System.out.println("ArithmeticException caught: Division by zero not allowed!");
        } finally {
            System.out.println("Case C: Finally block executed.");
        }
    }
}
```

Output:

Result: 5
Case A: Finally block executed.
Case B: Finally block executed.
Exception in thread "main" java.lang.ArithmeticException: / by zero
at lab4.FinallyBlockDemo.main(FinallyBlockDemo.java:17)

Process finished with exit code 1

Q.4

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

Ans.

```
interface Account {
    void deposit(double amount);
    void withdraw(double amount) throws InsufficientFundsException;
}

class InsufficientFundsException extends Exception {
    public InsufficientFundsException(String message) {
```

```
        super(message);
    }
}

class SavingsAccount implements Account {
    private double balance;

    @Override
    public void deposit(double amount) {
        balance += amount;
        System.out.println("Deposited: " + amount);
    }

    @Override
    public void withdraw(double amount) throws InsufficientFundsException {
        if (amount > balance) {
            throw new InsufficientFundsException("Insufficient funds!");
        } else {
            balance -= amount;
            System.out.println("Withdrawn: " + amount);
        }
    }

    public double getBalance() {
        return balance;
    }
}

public class TestSavingsAccount {
    public static void main(String[] args) {
        SavingsAccount account = new SavingsAccount();
        account.deposit(1000);

        try {
            account.withdraw(1200); // Withdraw an amount greater than the balance
        } catch (InsufficientFundsException e) {
            System.out.println("InsufficientFundsException caught: " + e.getMessage());
        }
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA Community Edition.app/Contents/lib/idea_rt.jar=57169:/Users/lakhman/Applications/IntelliJ IDEA Community Edition.app/Contents/bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/lakhman/Desktop/sem 4 coding/java/javabasics/out/production/java basics lab4.TestSavingsAccount
Deposited: 1000.0
InsufficientFundsException caught: Insufficient funds!

Process finished with exit code 0

Q.5

A method named add() accepts an array of strings as an argument. It converts these to double values and returns their sum. The method generates a NumberFormatException if the element is incorrectly formatted. It can also create and throw a custom exception, RangeException, if an element is less than 0 or greater than 1. Write a program that illustrates how to declare and use this method. Invoke the method from main. Also provide the finally clause to thank the user for using the program

Ans.

```
import java.util.*;
class RangeException extends Exception {
    public RangeException(String message) {
        super(message);
    }
}

 class Main {
     static double add(String[] values) throws NumberFormatException, RangeException {
        double sum = 0.0;

        try {
            for (String value : values) {
                double num = Double.parseDouble(value);
```

```
                if (num < 0 || num > 1) {
                    throw new RangeException("Value out of range (0 to 1): " + num);
                }

                sum += num;
            }
        } catch (NumberFormatException e) {
            throw new NumberFormatException("Incorrectly formatted element!");
        }

        return sum;
    }

    public static void main(String[] args) {
        String[] inputs = {"0.5", "0.3", "a", "0.2"};

        try {
            double result = add(inputs);
            System.out.println("Sum of elements: " + result);
        } catch (NumberFormatException | RangeException e) {
            System.out.println("Exception caught: " + e.getMessage());
        } finally {
            System.out.println("Thank you for using the program!");
        }
    }
}
```

Output:

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:/Users/lakhman/Applications/IntelliJ IDEA Community
Exception caught: Incorrectly formatted element!
Thank you for using the program!

Process finished with exit code 0