NAME: Pratham Srivastava

Roll No: 2401330130192

Branch: IT

Section: C

Workshop Lab Number : D-203

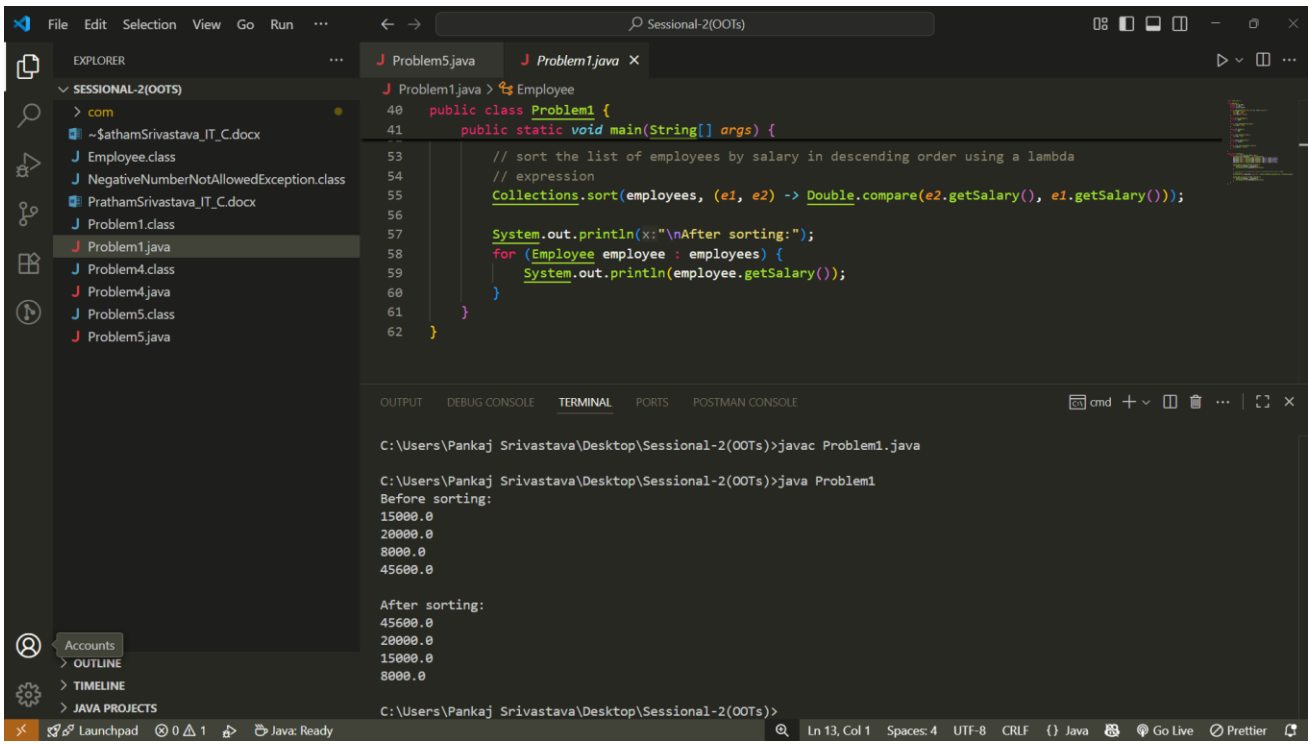
Problem-1: SOURCE CODE

1. Main Class-

```
2. import java.util.*;
3.
4. class Employee {
5.     private String name;
6.     private int age;
7.     private double salary;
8.
9.     public Employee(String name, int age, double salary) {
10.        this.name = name;
11.        this.age = age;
12.        this.salary = salary;
13.    }
14.
15.    public String getName() {
16.        return name;
17.    }
18.
19.    public void setName(String name) {
20.        this.name = name;
21.    }
22.
23.    public int getAge() {
24.        return age;
25.    }
26.
27.    public void setAge(int age) {
28.        this.age = age;
29.    }
30.
31.    public double getSalary() {
32.        return salary;
33.    }
34.
35.    public void setSalary(double salary) {
36.        this.salary = salary;
37.    }
38.
39.}
40.
41.public class Problem1 {
42.    public static void main(String[] args) {
43.        List<Employee> employees = new ArrayList<>();
44.        employees.add(new Employee("Random1", 30, 15000));
45.        employees.add(new Employee("Random2", 25, 20000));
46.        employees.add(new Employee("Random3", 40, 8000));
```

```
47.         employees.add(new Employee("Random4", 21, 45600));
48.
49.         System.out.println("Before sorting:");
50.         for (Employee employee : employees) {
51.             System.out.println(employee.getSalary());
52.         }
53.
54.         Collections.sort(employees, (e1, e2) ->
    Double.compare(e2.getSalary(), e1.getSalary()));
55.
56.         System.out.println("\nAfter sorting:");
57.         for (Employee employee : employees) {
58.             System.out.println(employee.getSalary());
59.         }
60.     }
61. }
```

Problem-1: OUTPUT



Problem-2: SOURCE CODE

Main Class –

```
package com.main;

import com.maths.*;
import com.maths.operations.Calculator;


public class Main {
    public static void main(String[] args) {
        Calculator calculator = new Calculator();

        int result1 = calculator.add(5, 3);
        System.out.println("5 + 3 = " + result1);

        int result2 = calculator.subtract(7, 4);
        System.out.println("7 - 4 = " + result2);

        int result3 = calculator.multiply(8, 2);
        System.out.println("8 * 2 = " + result3);

        double result4 = calculator.divide(16, 4);
        System.out.println("16 / 4 = " + result4);
    }
}
```

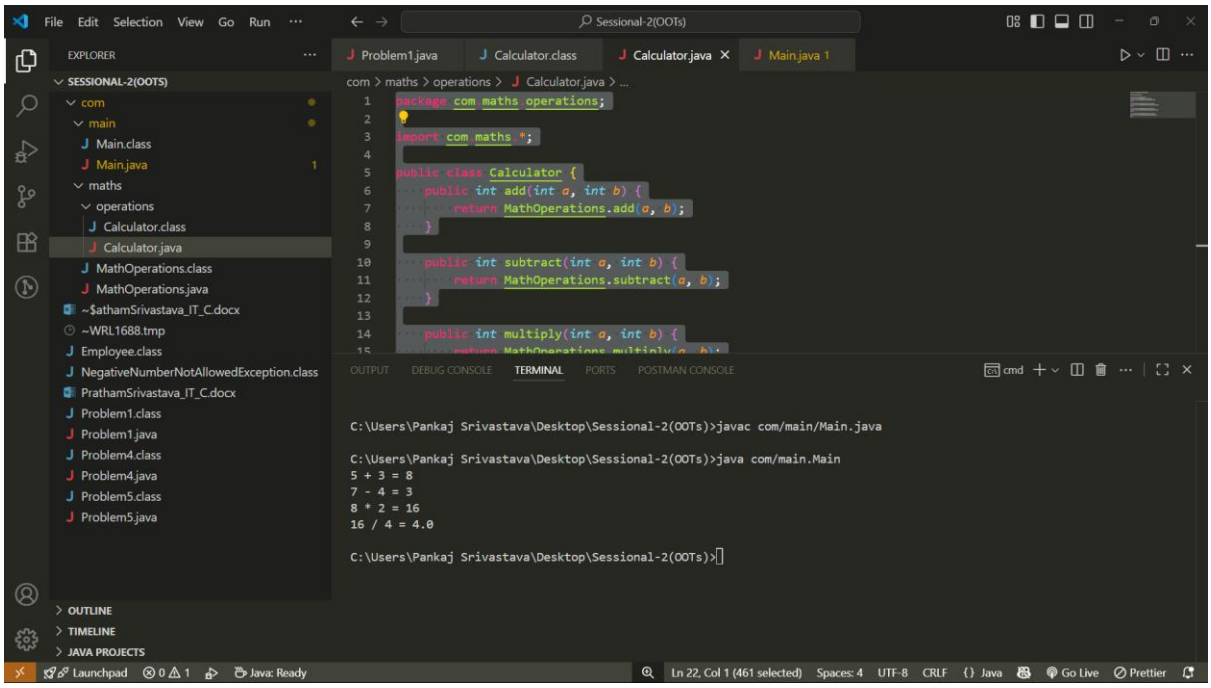
MathOperations Class –

```
package com.maths;


public class MathOperations {
    public static int add(int x, int y) {
        return x + y;
    }

    public static int subtract(int x, int y) {
        return x - y;
    }

    public static int multiply(int x, int y) {
        return x * y;
    }

    public static int divide(int x, int y) {
        return x / y;
    }
}
```

Calculator Class –

```
package com.maths.operations;

import com.maths.*;

public class Calculator {
    public int add(int a, int b) {
        return MathOperations.add(a, b);
    }

    public int subtract(int a, int b) {
        return MathOperations.subtract(a, b);
    }

    public int multiply(int a, int b) {
        return MathOperations.multiply(a, b);
    }

    public double divide(int a, int b) {
        return MathOperations.divide(a, b);
    }
}
```

Problem-2: OUTPUT

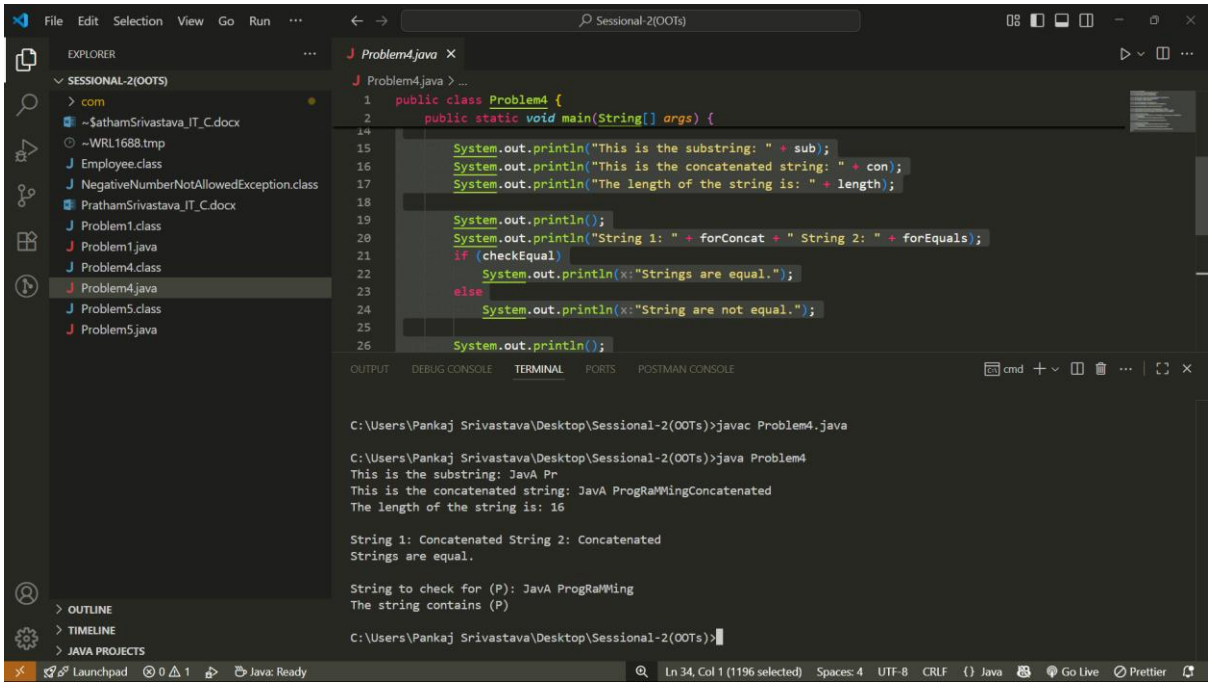


Problem-4: SOURCE CODE

```
public class Problem4 {
    public static void main(String[] args) {
        String test = "JavA ProgRaMMing";
        String forConcat = "Concatenated";

        String sub = test.substring(0, 7);
        String con = test.concat(forConcat);
        int length = test.length();

        String forEquals = "Concatenated";
        boolean checkEqual = forConcat.equals(forEquals);

        boolean forContains = test.contains("P");

        System.out.println("This is the substring: " + sub);
        System.out.println("This is the concatenated string: " + con);
        System.out.println("The length of the string is: " + length);

        System.out.println();
        System.out.println("String 1: " + forConcat + " String 2: " +
forEquals);
        if (checkEqual)
            System.out.println("Strings are equal.");
```

```
        else
            System.out.println("String are not equal.");

        System.out.println();
        System.out.println("String to check for (P): " + test);
        if (forContains)
            System.out.println("The string contains (P)");
        else
            System.out.println("The string does not contain (P)");
    }
}
```

Problem-4: OUTPUT



Problem-5: SOURCE CODE

```
import java.util.Scanner;

class NegativeNumberNotAllowedException extends Exception {
}

public class Problem5 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        int n = 0;
        System.out.print("Enter the size of the array: ");
        n = scanner.nextInt();

        int[] numbers = new int[n];
        int index = 0;

        while (index < numbers.length) {
            System.out.print("Enter a number: ");
            int input = scanner.nextInt();

            try {
                int num = input;
                if (num < 0) {
```

```
                throw new NegativeNumberNotAllowedException();
            }
            numbers[index] = num;
            index++;
        } catch (NumberFormatException e) {
            System.out.println("Invalid input, please enter a valid
number.");
        } catch (NegativeNumberNotAllowedException e) {
            System.out.println("Negative numbers are not allowed.");
        }
    }
    scanner.close();

    for (int i = 0; i < index; i++) {
        System.out.print(numbers[i] + " ");
    }
  }
}
```

PROBLEM-5: OUTPUT