LAB SUBMISSION 4 - 221047012

1.Familiarize with the File Operations in Java. Develop a program to read the content of one file and copy the content to another file.

<u>Code:</u>

```java
package Ph_221047012;


import java.io.File;
import java.io.IOException;
import java.io.FileWriter;
import java.util.Scanner;
import java.io.FileNotFoundException;
import java.io.FileReader;
    public class Filehandling {
        public static void main(String[] args) {
            try {
                File myObj1 = new
File("C:\Users\User\IdeaProjects\\Lab4\\file1.txt");
                if (myObj1.createNewFile()) {
                    System.out.println("File created: " +
myObj1.getName());
                } else {
                    System.out.println("File already exists.");
                }
            } catch (IOException e) {
                System.out.println("An error occurred.");
                e.printStackTrace();
            }
            try {
                FileWriter myWriter = new
FileWriter("C:\Users\User\IdeaProjects\Lab4\file1.txt");
                myWriter.write(" My name is Sahana P H from cloud
computing");
                myWriter.close();
                System.out.println("Input given and inserted
successfully.");
            } catch (IOException e) {
                System.out.println("An error occurred.");
                e.printStackTrace();
            }
            try {
                File myObj2 = new
File("C:\\Users\\MSIS\\Documents\\Java\\file2.txt");
                if (myObj2.createNewFile()) {
                    System.out.println("File created: " +
myObj2.getName());
                } else {
                    System.out.println("File already exists.");
                }
            } catch (IOException e) {
                System.out.println("An error occurred.");
                e.printStackTrace();
            }
            try {
                FileReader f1 = new FileReader("file1.txt");

                File myObj = new File("
C:\Users\User\IdeaProjects\\Lab4\\file1.txt");
                FileWriter myWriter = new FileWriter("
```

```
C:\Users\User\IdeaProjects\\Lab4\\file2.txt");
            Scanner myReader = new Scanner(myObj);
            int i; String s="";
            while ((i=f1.read())!=-1)
                s+=(char)i;
            myWriter.write(s);
            myWriter.close();
            System.out.println("success.");
            myReader.close();
            f1.close();
        } catch (FileNotFoundException e) {
            System.out.println("An error occurred.");
            e.printStackTrace();

        } catch (IOException e) {
            System.out.println("An error occurred.");
            e.printStackTrace();
        }
    }
}
```

Output:

file1 - Notepad
File Edit Format View Help

My name is Sahana P H (221047012) from cloud computing

file2 - Notepad
File Edit Format View Help

My name is Sahana P H (221047012) from cloud computing

2.Write a Calculator Utility that does the that performs the set of operations defined in an interface called MyOperations. Develop the Application Program to utilize the operations defined in MyOperations. Your Application should have proper Exception Handling and your operation should utilize at least two Custom Exceptions. Record of your operations should be recorded in a file

Code:

```java
package Ph_221047012;

import java.io.*;
import java.nio.file.Paths;
import java.util.Scanner;

public class Calculator {

    static boolean alive = true;

    public static void main(String[] args){

        int number1;
        int number2;
        char mathchoice;
        int result;
        char userchoice;

        final File file = Paths.get("result.txt").toFile();

        FileOutputStream fileOutputStream = null;
        try {
            fileOutputStream = new FileOutputStream(file);
            file.createNewFile();
            System.setOut(new PrintStreamFileForwarder(System.out,
fileOutputStream));
        } catch (IOException e) {
            e.printStackTrace();
        }

        Scanner choice = new Scanner(new InputStreamFileForwarder(System.in,
fileOutputStream));

        do {

            System.out.println("Enter the first operand...");
            number1 = choice.nextInt();
            System.out.println("Choose the operator:\n+\n-\n*\n/\n%\n");
            mathchoice = choice.next().charAt(0);
            System.out.println("Enter the second operand...");
            number2 = choice.nextInt();

            if (mathchoice == '+') {
                result = number1 + number2;
                System.out.println("Result = " + result);
            } else if (mathchoice == '-') {
                result = number1 - number2;
                System.out.println("Result =  " + result);
            } else if (mathchoice == '*') {
                result = number1 * number2;
                System.out.println("Result = " + result);
            } else if (mathchoice == '/') {
                if (number2 == 0) {
                    throw new java.lang.ArithmeticException("Number cannot be divided
by zero");
                } else {
```

```java
                        result = number1 / number2;
                        System.out.println("Result = " + result);
                    }

                } else if (mathchoice == '%') {
                    result = number1 % number2;
                    System.out.println("Result = " + result);
                } else if (mathchoice == '^') {
                    result = (int) Math.pow(number1, number2);
                    System.out.println("Result = " + result);
                } else {
                    System.out.println("Wrong choice");

                }
                System.out.println("Want to continue? Press y");
                userchoice = choice.next().charAt(0);

            } while (userchoice == 'y');
            alive = false;
        }

    public static class InputStreamFileForwarder extends InputStreamReader {

        private final FileOutputStream fileOutputStream;

        public InputStreamFileForwarder(InputStream console, FileOutputStream
fileOutputStream) {
            super(console);
            this.fileOutputStream = fileOutputStream;
        }

        @Override
        public int read(char[] cbuf, int offset, int length) throws IOException {
            int read = super.read(cbuf, offset, length);
            if(read > 0) {
                char[] allRead = new char[read];
                System.arraycopy(cbuf, offset, allRead, 0, read);
                fileOutputStream.write(new String(allRead).getBytes());
            }
            return read;
        }
    }

    public static class PrintStreamFileForwarder extends PrintStream {

        private final FileOutputStream fileOutputStream;

        public PrintStreamFileForwarder(PrintStream console, FileOutputStream
fileOutputStream) {
            super(console);
            this.fileOutputStream = fileOutputStream;
        }

        @Override
        public void write(byte[] buf, int off, int len) {
            super.write(buf, off, len);
            try {
                fileOutputStream.write(buf, off, len);
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}
```
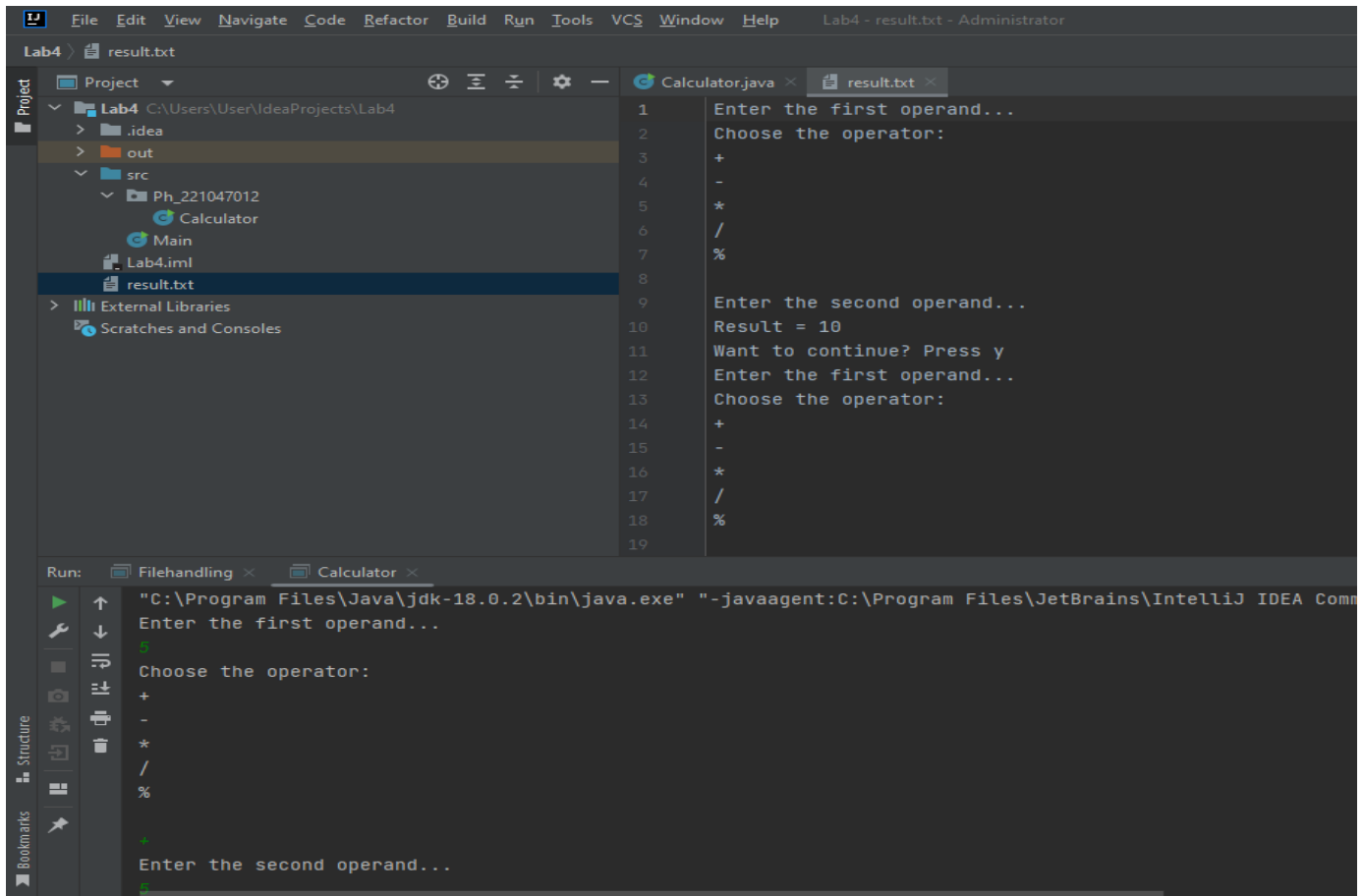
outputs: