**PRACTICAL – 6(1)**

**Aim: Write a program that will count the number of lines in each file that is**

**specified on the command line. Assume that the files are text files. Note**

**that multiple files can be specified, as in "java LineCounts file1.txt file2.txt**

**file3.txt". Write each file name, along with the number of lines in that file,**

**to standard output. If an error occurs while trying to read from one of the**

**files, you should print an error message for that file, but you should still**

**process all the remaining files.**

**SOURCE CODE:**

import java.io.\*;

import java.io.File;

import java.util.\*;

public class Practical\_6\_1 {

    public static *void* main(String[] *args*) {

*int* x;

        for (x = 0; x < *args*.length; x++) {

*int* y = 0;

            try {

                File fileobj = new File(*args*[x]);

                if (fileobj.exists()) {

                    System.out.println("Searching For " + *args*[x] + "......");

                    System.out.println(*args*[x] + "Found.");

                }

                Scanner sc = new Scanner(fileobj);

                while (sc.hasNextLine()) {

                    y++;

                    sc.nextLine();

                }

                System.out.println("There Are " + y + " lines in " + fileobj.getName() + "\n");

                sc.close();

            } catch (IOException e) {

                System.out.println("An error Occured While Finding " + *args*[x] + "file");

            }

            System.out.println("20DCE019-Yatharth Chauhan\n");

        }

    }

}

**OUTPUT:**

**Text

Description automatically generated**

**CONCLUSION:** In this practical we learnt how to perform file handling along with the syntax of file handling in java.