1. **Write a Java program to get files with  specific extension from a specified folder.**

**CODE :**

import java.io.File;

import java.util.ArrayList;

import java.util.List;

public class FileExtensionFinder {

    public static List<File> getFilesWithExtension(String folderPath, String extension) {

        List<File> fileList = new ArrayList<>();

        File folder = new File(folderPath);

        File[] listOfFiles = folder.listFiles();

        if (listOfFiles != null) {

            for (File file : listOfFiles) {

                if (file.isFile() && file.getName().toLowerCase().endsWith(extension.toLowerCase())) {

                    fileList.add(file);

                }

            }

        }

        return fileList;

    }

    public static void main(String[] args) {

        String folderPath = "file1";

        String extension = ".txt";

        List<File> filesWithExtension = getFilesWithExtension(folderPath, extension);

        if (filesWithExtension.isEmpty()) {

            System.out.println("No files with the specified extension found.");

        } else {

            System.out.println("Files with extension '" + extension + "' in folder '" + folderPath + "':");

            for (File file : filesWithExtension) {

                System.out.println(file.getName());

            }

        }

    }

}

**2.  Write a Java program that reads a list of numbers from a file and throws an exception if any of the numbers are positive.**

**Output:**

**Content of test.txt: -1 -2 -3 4**

**Error: Positive number found: 4**

**CODE :**

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class PositiveNumberChecker {

    public static void checkForPositiveNumbers(String filePath) throws FileNotFoundException {

        File file = new File(filePath);

        Scanner scanner = new Scanner(file);

        System.out.print("Content of " + filePath + ": ");

        while (scanner.hasNext()) {

            int number = scanner.nextInt();

            System.out.print(number + " ");

            if (number > 0) {

                throw new IllegalArgumentException("Error: Positive number found: " + number + " in " + filePath);

            }

        }

        System.out.println();

    }

    public static void main(String[] args) {

        String filePath = "test.txt";

        try {

            checkForPositiveNumbers(filePath);

        } catch (FileNotFoundException e) {

            System.out.println("File not found: " + filePath);

        } catch (IllegalArgumentException e) {

            System.out.println(e.getMessage());

        }

    }

}

**3. You are given a directory path that contains a number of text files. Each text file contains words separated by spaces.**

**Your task is to write a Java program that finds the most common word across all the files. Consider a word as a sequence of characters separated by spaces. Ignore case sensitivity, meaning "hello" and "Hello" should be considered the same word.**

**Write a Java program that takes the directory path as input and outputs the most common word along with its frequency. If there are multiple words with the same highest frequency, output all of them.**

**Input:**

**Enter directory name : TextFolder**

**Output:**

**Word: world, Frequency: 3  
Word: java, Frequency: 2  
Word: hello, Frequency: 2  
Word: is, Frequency: 1  
Word: a, Frequency: 1  
Word: programming, Frequency: 1  
Word: language, Frequency: 1**

**CODE :**

import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

import java.util.HashMap;

import java.util.Map;

public class CommonWordFinder {

    public static void findMostCommonWord(String directoryPath) {

        File folder = new File(directoryPath);

        Map<String, Integer> wordFrequency = new HashMap<>();

        if (folder.isDirectory()) {

            for (File fileEntry : folder.listFiles()) {

                if (fileEntry.isFile()) {

                    try (BufferedReader br = new BufferedReader(new FileReader(fileEntry))) {

                        String line;

                        while ((line = br.readLine()) != null) {

                            String[] words = line.toLowerCase().split("\\s+");

                            for (String word : words) {

                                word = word.replaceAll("[^a-zA-Z]", "");

                                if (!word.isEmpty()) {

                                    wordFrequency.put(word, wordFrequency.getOrDefault(word, 0) + 1);

                                }

                            }

                        }

                    } catch (IOException e) {

                        e.printStackTrace();

                    }

                }

            }

        }

        int maxFrequency = 0;

        for (int frequency : wordFrequency.values()) {

            if (frequency > maxFrequency) {

                maxFrequency = frequency;

            }

        }

        System.out.println("Most common word(s) and their frequency:");

        for (Map.Entry<String, Integer> entry : wordFrequency.entrySet()) {

            if (entry.getValue() == maxFrequency) {

                System.out.println("Word: " + entry.getKey() + ", Frequency: " + entry.getValue());

            }

        }

    }

    public static void main(String[] args) {

        String directoryPath = "TextFolder";

        findMostCommonWord(directoryPath);

    }

}