CLASE 11:

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class RegularExpressionMatching {

public static void main(String[] args) {

String fileName = args[0]; // archivo de entrada

String regex = args[1]; // expresión regular

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

String line;

int lineNumber = 0;

Pattern pattern = Pattern.compile(regex);

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

lineNumber++;

Matcher matcher = pattern.matcher(line);

if (matcher.find()) {

System.out.println(lineNumber + " - Match found: " + line);

} else {

System.out.println(lineNumber + " - No match found: " + line);

}

}

} catch (IOException e) {

e.printStackTrace();

}

}

}

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class RegularExpressionMatchingWithGroups {

public static void main(String[] args) {

String fileName = args[0]; // archivo de entrada

String regex = args[1]; // expresión regular

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

String line;

int lineNumber = 0;

Pattern pattern = Pattern.compile(regex);

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

lineNumber++;

Matcher matcher = pattern.matcher(line);

if (matcher.find()) {

System.out.print(lineNumber + " - Match found: ");

for (int i = 1; i <= matcher.groupCount(); i++) {

System.out.print(matcher.group(i) + " ");

}

System.out.println();

} else {

System.out.println(lineNumber + " - No match found: " + line);

}

}

} catch (IOException e) {

e.printStackTrace();

}

}

}