import java.util.Scanner;

import java.io.\*;

/\*\*

This program demonstrates a solution to the

Distance File programming challenge.

\*/

public class DistanceFile

{

public static void main(String[] args) throws IOException

{

double speed; // The vehicle's speed

int maxHours, // Max number of hours traveled

period; // To count time periods

// Create a Scanner object for keyboard input.

Scanner keyboard = new Scanner(System.in);

// Get the speed.

System.out.print("Enter the vehicle's speed: ");

speed = keyboard.nextDouble();

// Validate the speed.

while (speed < 0)

{

System.out.print("Enter 0 or greater for speed: ");

speed = keyboard.nextDouble();

}

// Get the number of hours.

System.out.print("Enter the number of hours the " +

"vehicle was in motion: ");

maxHours = keyboard.nextInt();

// Validate the hours.

while (maxHours < 1)

{

System.out.print("Enter 1 or greater for hours: ");

maxHours = keyboard.nextInt();

}

// Create the objects necessary to perform file output.

FileWriter fw = new FileWriter("DistanceReport.txt");

PrintWriter outFile = new PrintWriter(fw);

// Write the table header.

outFile.println("Hour\tDistance Traveled");

outFile.println("----------------------------------");

// Write the table of distances.

period = 1;

while (period <= maxHours)

{

// Display the distance for this period.

outFile.println(period + "\t\t" + period \* speed);

// Increment period.

period++;

}

// Close the file.

outFile.close();

System.out.println("Report written to DistanceReport.txt.");

}

}

import javax.swing.JOptionPane;

import java.io.\*;

/\*\*

This program demonstrates a solution to the

File Letter Counter programming challenge.

\*/

public class FileLetterCounter

{

public static void main(String[] args) throws IOException

{

String filename; // The file to read

String output; // To hold the output

String input; // To hold file input

char letter; // The letter to count

int num = 0; // Number of times the letter appears

// Get the file name from the user.

filename = JOptionPane.showInputDialog(

"Enter the name of a file.");

// Get the letter to count.

input = JOptionPane.showInputDialog(

"Enter a letter contained in " +

"the string.");

// Make sure the letter is uppercase.

input = input.toUpperCase();

// Retrieve the letter.

letter = input.charAt(0);

// Open the file.

FileReader fr = new FileReader(filename);

BufferedReader inFile = new BufferedReader(fr);

// Read the first line from the file.

input = inFile.readLine();

// Process the entire file.

while (input != null)

{

// Get the uppercase equivalent of the line.

input = input.toUpperCase();

// Count the number of times the letter appears

// in the line.

for (int i = 0; i < input.length(); i++)

{

if (input.charAt(i) == letter)

num++;

}

// Read the next line.

input = inFile.readLine();

}

// Close the file.

inFile.close();

// Format and display the count.

output = String.format("The letter %s " +

"appears %d times in " +

"the file.\n", letter, num);

JOptionPane.showMessageDialog(null, output);

System.exit(0);

}

}

/\*\*

This program demonstrates a solution to the

Average Rainfall programming challenge.

\*/

public class AverageRainfall

{

public static void main(String[] args)

{

final int NUM\_MONTHS = 12; // Months per year

int years; // Number of years

double monthRain; // Rain for a month

double totalRain = 0; // Rainfall accumulator

double average; // Average rainfall

// Create a Scanner object for keyboard input.

Scanner keyboard = new Scanner(System.in);

// Get the number of years.

System.out.print("Enter the number of years: ");

years = keyboard.nextInt();

// Validate the input.

while (years < 1)

{

System.out.print("Invalid. Enter 1 or greater: ");

years = keyboard.nextInt();

}

System.out.println("Enter the rainfall, in inches, " +

"for each month.");

for (int y = 1; y <= years; y++)

{

for (int m = 1; m <= NUM\_MONTHS; m++)

{

// Get the rainfall for a month.

System.out.print("Year " + y + " month " + m + ": ");

monthRain = keyboard.nextDouble();

// Validate the input.

while (monthRain < 0)

{

System.out.print("Invalid. Enter 0 or greater: ");

monthRain = keyboard.nextDouble();

}

// Accumulate the rainfall.

totalRain += monthRain;

}

}

// Calculate the average rainfall.

average = totalRain / (years \* NUM\_MONTHS);

// Display the statistics.

System.out.println("\nNumber of months: " + (years \* NUM\_MONTHS));

System.out.println("Total rainfall: " + totalRain + " inches");

System.out.printf("Average monthly rainfall: %.2f inches\n", average);

}

}

import java.util.Scanner;

import java.io.\*;

/\*\*

This program demonstrates a solution to the

Line Numbers programming challenge.

\*/

public class LineNumbers

{

public static void main(String[] args) throws IOException

{

String filename; // The name of the file

String input; // To hold file input

int lineCount; // To hold line numbers

// Create a Scanner object for keyboard input.

Scanner keyboard = new Scanner(System.in);

// Get the file name.

System.out.print("Enter the file name: ");

filename = keyboard.nextLine();

// Open the file.

File file = new File(filename);

Scanner inFile = new Scanner(file);

// Initialize the line counter to 1.

lineCount = 1;

// Display the lines with line numbers.

while (inFile.hasNext())

{

input = inFile.nextLine();

System.out.println(lineCount + ": " + input);

lineCount++;

}

// Close the file.

inFile.close();

}

}

import java.util.Scanner;

import java.io.\*;

/\*\*

This program demonstrates a solution to the

Uppercase File Converter programming challenge.

\*/

public class UppercaseFileConverter

{

public static void main(String[] args) throws IOException

{

String inFilename; // The name of the input file

String outFilename; // The name of the output file

String input; // To hold file input

String output; // To hold file output

// Create a Scanner object for keyboard input.

Scanner keyboard = new Scanner(System.in);

// Get the input file name.

System.out.print("Enter the input file name: ");

inFilename = keyboard.nextLine();

// Get the output file name.

System.out.print("Enter the output file name: ");

outFilename = keyboard.nextLine();

// Open the input file.

File file = new File(inFilename);

Scanner inFile = new Scanner(file);

// Open the output file.

PrintWriter outFile = new PrintWriter(outFilename);

// Process the files.

while (inFile.hasNext())

{

input = inFile.nextLine();

output = input.toUpperCase();

outFile.println(output);

}

// Close the files.

inFile.close();

outFile.close();

}

}