import java.io.PrintWriter;

import java.util.\*;

import java.io.File;

public class PrintPlayerInfo

{

public static void printPlayerInfo(String[][] data, String player)

{

try

{

for(int i = 1; i < data.length; i++)

{

String p = player.toLowerCase();

String names = data[i][0].toLowerCase();

if(names.indexOf(p) != -1)

{

try

{

for(int j = 0; j < data[0].length; j++)

{

String stats = data[i][j];

{

System.out.printf("%21s: %s\n",data[0][j], stats);

}

}

System.out.printf("\n");

}

catch (Exception e)

{

break;

}

}

}

}

catch (Exception e)

{

System.out.printf("Not data has been loaded\n");

}

}

public static String[][] readSpreadsheet(String filename)

{

ArrayList<String> lines = readFile(filename);

if (lines == null)

{

return null;

}

String[][] result = new String[lines.size()][];

for (int i = 0; i < lines.size(); i++)

{

String line = lines.get(i);

result[i] = line.split(",");

}

return result;

}

import java.io.PrintWriter;

import java.util.\*;

import java.io.File;

public class SavePlayerInfo

{

public static void savePlayerInfo(String[][] data, String player, String output\_name)

{

PrintWriter out = null;

try

{

out = new PrintWriter(output\_name);

}

catch (Exception e)

{

System.exit(0);

}

try

{

for(int i = 1; i < data.length; i++)

{

String p = player.toLowerCase();

String names = data[i][0].toLowerCase();

if(names.indexOf(p) != -1)

{

try

{

for(int j = 0; j < data[0].length; j++)

{

String stats = data[i][j];

{

out.printf("%21s: %s\r\n",data[0][j], stats);

}

}

out.printf("\r\n");

}

catch (Exception e)

{

break;

}

}

}

}

catch (Exception e)

{

out.printf("Not data has been loaded\r\n");

}

out.close();

}

public static String[][] readSpreadsheet(String filename)

{

ArrayList<String> lines = readFile(filename);

if (lines == null)

{

return null;

}

String[][] result = new String[lines.size()][];

for (int i = 0; i < lines.size(); i++)

{

String line = lines.get(i);

result[i] = line.split(",");

}

return result;

}

import java.util.\*;

import java.io.File;

public class ColumnName

{

public static String columnName(String filename, int column)

{

File temp = new File(filename);

Scanner input;

try

{

input = new Scanner(temp);

}

catch (Exception e)

{

System.out.printf("Failed to open file %s\n", filename);

return null;

}

String line = input.nextLine();

String[] columns = line.split(",");

int n = column;

if(n > -1 && n < columns.length)

{

return columns[n];

}

return null;

}

public static int userInteger(String message)

{

Scanner in = new Scanner(System.in);

int result;

while (true)

{

System.out.printf(message);

String s = in.next();

if (s.equals("q"))

{

System.out.printf("Exiting...\n");

System.exit(0);

}

try

{

result = Integer.parseInt(s);

}

catch (Exception e)

{

System.out.printf("%s is not a valid number, try again.\n\n", s);

continue;

}

return result;

}

}

import java.util.\*;

import java.io.File;

public class ColumnSum

{

public static ArrayList<String> read\_file(String filename)

{

File temp = new File(filename);

Scanner input\_file;

try

{

input\_file = new Scanner(temp);

}

catch (Exception e)

{

System.out.printf("Failed to open file %s\n",

filename);

return null;

}

ArrayList<String> result = new ArrayList<String>();

while(input\_file.hasNextLine())

{

String line = input\_file.nextLine();

result.add(line);

}

input\_file.close();

return result;

}

public static String[][] read\_spreadsheet(String filename)

{

ArrayList<String> lines = read\_file(filename);

if (lines == null)

{

return null;

}

String[][] result = new String[lines.size()][];

for (int i = 0; i < lines.size(); i++)

{

String line = lines.get(i);

result[i] = line.split(",");

}

return result;

}

public static double columnSum(String filename, int column)

{

File temp = new File(filename);

Scanner input;

try

{

input = new Scanner(temp);

}

catch (Exception e)

{

System.out.printf("Failed to open file %s\n", filename);

return 0;

}

double d;

String[][] target = read\_spreadsheet(filename);

double sum = 0;

for(int i = 0; i < target.length; i++) //Row of spreadsheet

{

try

{

d = Double.parseDouble(target[i][column]); //Finds all double numbers

}

catch (Exception e)

{

continue; //Contines from elements that are not double numbers

}

if(d > 1000 || d < 0 )

{

return 0;

}

else

{

sum = sum + d;

}

}

return sum;

}

public static int getMonth(String date)

{

int result = 0;

String[] d = date.split(" "); // d = ["Sat","3/7/2015"]

String[] m = d[1].split("/"); // m = ["3","7","2015"]

result = Integer.parseInt(m[0]);

return result;

}

public static void printMonths(String filename)

{

ArrayList<String> lines = readFile(filename);

if (lines == null)

{

return;

}

for (int i = 1; i < lines.size(); i++)

{

String line = lines.get(i);

String[] columns = line.split(",");

String date = columns[0];

int line\_month = 0;

if(line.contains("/"))

{

line\_month = getMonth(line);

}

System.out.printf("row %d, month = %d\n", i, line\_month);

}

}

Monthly Average

public static String columnName(String filename, int column)

{

File temp = new File(filename);

Scanner input;

try

{

input = new Scanner(temp);

}

catch (Exception e)

{

System.out.printf("Failed to open file %s\n", filename);

return null;

}

String line = input.nextLine();

String[] columns = line.split(",");

int n = column;

if(n > -1 && n < columns.length)

{

return columns[n];

}

return null;

}

public static double monthlyAverage(String filename, int column, int month)

{

File temp = new File(filename);

Scanner input;

try

{

input = new Scanner(temp);

}

catch (Exception e)

{

System.out.printf("Failed to open file %s\n", filename);

return 0;

}

double average = 0;

int count = 0;

if(input.hasNextLine()) //Reads the first line that has the labels of the columns

{

input.nextLine();

}

while(input.hasNextLine())

{

String lines = input.nextLine();

String[] d;

String[] m;

int result;

try

{

d = lines.split(" ");

m = d[1].split("/");

result = Integer.parseInt(m[0]);

}

catch (Exception e)

{

continue;

}

if(result == month)

{

String[] columns = lines.split(",");

int n = column;

if(n > -1 && n < columns.length)

{

average += Double.parseDouble(columns[n]);

count++;

}

}

}

if (count == 0)

{

return -1;

}

average = average / count;

return average;

}

Phonebook

public static String[][] readPhonebook()

{

String filename = "phonebook.txt";

String[][] lines = readSpreadsheet(filename);

return lines;

}

public static void savePhonebook(String[][] data, String new\_name, String new\_number)

{

String filename = "phonebook.txt";

PrintWriter out = null;

try

{

out = new PrintWriter(filename);

}

catch (Exception e)

{

System.out.printf("Error: failed to open file %s. \n", filename);

return;

}

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

{

String g = data[i][0];

String h = data[i][1];

out.printf("%s, %s\r\n", g, h);

}

out.printf("%s, %s\r\n", new\_name, new\_number);

out.close();

}

public static void printSpreadsheet(String[][] data)

{

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

{

System.out.printf("%4d: %20s, %4s\n",i, data[i][0], data[i][1]);

}

}

public static void searchData(String[][] data)

{

Scanner in = new Scanner (System.in);

System.out.printf("Please enter a name: ");

String name = in.nextLine();

name = name.toLowerCase();

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

{

String n = data[i][0].toLowerCase();

if(n.indexOf(name) != -1)

{

System.out.printf("%20s, %s\n", data[i][0], data[i][1]);

}

}

}

public static String[][] inputNewEntry(String[][] data)

{

Scanner in = new Scanner(System.in);

System.out.printf("\nEnter a name: ");

String name = in.nextLine();

System.out.printf("\nEnter a number: ");

String number = in.nextLine();

savePhonebook(data, name, number);

data = readPhonebook();

return data;

}

public static String[][] processOption(String[][] data, String option)

{

if (option.equals("1"))

{

printSpreadsheet(data);

}

else if (option.equals("2"))

{

data = inputNewEntry(data);

}

else if (option.equals("3"))

{

searchData(data);

}

else if (option.equals("q"))

{

System.out.printf("Exiting...\n");

System.exit(0);

}

else

{

System.out.printf("Unrecognized option %s.\n", option);

}

return data;

}

public static String askOption()

{

Scanner in = new Scanner(System.in);

System.out.printf("\n1: Print phonebook.\n");

System.out.printf("2: Input a new entry.\n");

System.out.printf("3: Search by name.\n");

System.out.printf("q: Quit program.\n");

System.out.printf("Please enter an option: ");

String option = in.next();

return option;

}