/\*\*

\* @package homework4

\*/

package homework4;

import java.awt.BorderLayout;

import javax.swing.\*;

/\*\*

\* Homework 4 CIS 211 Fall 2010

\* @author Neo Melonas <neo@neomelonas.com>

\* @version 2010-11-19

\*

\* Sort the names alphabetically (based on Last name) keeping all the student data

\* with each student during the sort. Concatenate all the student data to a string so it

\* can be printed, after all the items below have been added to the string. Note, Names are

\* entered in the starting array (First Last)

\*

\* Back in the main program compute the number of exams in the 70s 80s and 90s for students <150 lbs and >= 150 lbs. (6 results)

\* Print these results in pretty tabular form.

\*

\*/

public final class Homework4 extends JFrame {

/\*\*

\* OH NO A CONSTRUCTOR!

\*/

public Homework4() {

String[] name = {"Smith, Mary", "Smythe, John", "Clinton, William", "Klink, Debbie", "Theremin, Ralph"};

int[] temp = {34, 78, 78, 65, 45, 100, 90, 97, 56, 89, 78, 98, 74, 90, 98, 24, 45, 76, 89, 54, 12, 20, 22, 55, 66};

int[][] scores = new int[5][5];

int[] weight = {132, 165, 200, 155, 145};

float[] averages = new float[5];

int[][] sillyCount = new int[3][2];

JTextArea \_resultArea = new JTextArea(30, 100);

reorganizeArrays(name, temp, scores, weight);

String best = name[highestStudentExam(scores)];

int mostCommonScore = mostCommonScore(temp);

for (int a = 0; a < name.length; a++) averages[a] = calculateAverages(scores[a]);

for (int a = 0; a < 5; a++) {

for (int b = 0; b < 5; b++) {

int scoretest = 70;

for (int c = 0; c < 3; c++) {

int scoretest2 = scoretest + 10;

if (scores[a][b] >= scoretest && scores[a][b] < scoretest2)

if (weight[a] < 150) sillyCount[c][0]++;

else sillyCount[c][1]++;

scoretest += 10;

}

}

}

\_resultArea.setText(displayTime(name, scores, weight, averages, best, mostCommonScore, sillyCount));

JScrollPane scrollingArea = new JScrollPane(\_resultArea);

JPanel content = new JPanel();

content.setLayout(new BorderLayout());

content.add(scrollingArea, BorderLayout.CENTER);

this.setContentPane(content);

this.setTitle("Grade Report for a TERRIBLE CLASS");

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.pack();

}

/\*\*

\* This function sets the arrays, alphabetically by student last name.

\* @param n

\* @param temp

\* @param scores

\* @param w

\*/

public void reorganizeArrays(String[] n, int[] temp, int[][] scores, int[] w) {

int counter = 0;

String[][] trash = new String[5][7];

String[] holder = new String[6];

for (int a = 0; a < 5; a++) {

trash [a][0] = n[a];

for (int b = 1; b < 6; b++) {

trash[a][b] = Integer.toString(temp[counter]);

counter++;

}

trash [a][6] = Integer.toString(w[a]);

}

for (int x = 0; x < 5; x++) {

for (int y = 0; y < 4 - x; y++) {

if (trash[y][0].compareTo(trash[y + 1][0]) > 0) {

holder = trash[y];

trash[y] = trash[y + 1];

trash[y + 1] = holder;

}

}

}

for (int a = 0; a < 5; a++) {

n[a] = trash[a][0];

w[a] = Integer.parseInt(trash[a][6].trim());

for (int b = 1; b < 6; b++){

scores[a][b-1] = Integer.parseInt(trash[a][b].trim());

}

}

}

/\*\*

\* Call a function to get the name of the student with the highest score.

\* @param s

\* @return

\*/

public int highestStudentExam(int[][] s) {

int i = 0, test = 0;

for (int a = 0; a < 5; a++) {

for (int b = 0; b < 5; b++) {

if (s[a][b] > test) {i = a;test = s[a][b];}

}

}

return i;

}

/\*\*

\* Call a function that returns the most frequently occurring exam score.

\* @param s

\* @return

\*/

public int mostCommonScore(int[] s) {

int i = 0, mcs = 0;

int[] j = new int[101];

for (int a = 0; a < s.length; a++) {

if (j[s[a]] == 0) j[s[a]] = 1;

else j[s[a]]++;

}

for (int a = 0; a < j.length; a++) {

if (j[a] > i) { i = j[a]; mcs = a;}

}

return mcs;

}

/\*\*

\* Create another function that receives a small array that contains the 5 scores

\* for a person. Compute the average by eliminating the median and doubling the high and return this average

\* to the calling routine where it is stored in an array for future printing

\* @param s

\* @return a

\*/

public float calculateAverages(int[] s) {

float a = 0;

int test = 0;

float total = 0;

float len = (float) s.length;

int middle = s.length / 2;

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

total += (float) s[i];

}

sortInts(s);

if (len % 2 == 0) middle = (s.length / 2) + 1;

a = ((float) s[0] + (float) total - (float) s[middle]) / 5;

return a;

}

/\*\*

\* I might have needed to sort a 1d array more than once. I didn't, but whatever.

\* @param i

\*/

public void sortInts(int[] i) {

int[] s = new int[5];

for (int a = 0; a < 5; a++) {

int test = 0;

for (int b = 0; b < 5; b++) {

if (a == 0) {

if (i[b] > test) test = i[b];

}

else { if (i[b] > test && i[b] < s[a-1]) test = i[b]; }

}

s[a] = test;

i = s;

}

}

/\*\*

\* This function creates a fancy formated display. Or something like that.

\* @param n

\* @param s

\* @param w

\* @param a

\* @param b

\* @param mcs

\* @param sc

\* @return

\*/

public String displayTime(String[] n, int[][]s, int[] w, float[] a, String b, int mcs, int[][] sc) {

String output = "Student Name\tQ1\tQ2\tQ3\tQ4\tF \tWeight\tAverage\n\n";

int grade = 70;

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

output += n[i];

if (n[i].length() < 15) {

output += "\t";

}

for (int j = 0; j < 5; j++) {

output += "\t" + s[i][j];

}

output += "\t" + w[i] + "\t" + a[i] + "\n";

}

output += "\nMost Common Score:\t" + Integer.toString(mcs) +"\nBest Grade in class by " + b + "\n\n\t<150lbs\t>=150lbs";

for (int i = 0; i < 3; i ++) {

output += "\n" + grade + "\'s:\t ";

for (int j = 0; j < 2; j++) {

output += sc[i][j] + "\t ";

}

grade += 10;

}

return output;

}

/\*\*

\* MAIN METHOD FOR JAVA VICTORY.

\* @param args

\*/

public static void main(String[] args) {

JFrame win = new Homework4();

win.setVisible(true);

}

}

