CS 1400-03 Introduction to Programming and Problem Solving
Coding Practice #9
(Due: 11:59 PM, Friday, 4/16/2021)

Except Coding Practice #1, I will not grade your coding practice submissions. Instead, they will be treated as participation points. On blackboard, you will receive full points as long as you work on the exercises, which don't necessary mean they are all correct. Please check your own programs carefully and make sure they do generate the desired output.

Objectives:

- Be able to write complete Java programs with 2D arrays, classes, methods
- Be able to use `ArrayList`
- Be able to test and debug a program

Change your working directory to `cs1400/codingPractice` for this assignment.

Task #1 Two-Dimensional Arrays

(a) Write a utility class `TwoDarray` that provides the following static methods:

- `public static int getTotal(int[][] a)` – This method returns the total of all the values in the array.
- `public static double getAverage(int[][] a)` – This method returns the average of all the values in the array.
- `public static int getRowTotal(int[][] a, int row)` – This method returns the total of the values in the specified row.
- `public static int getColumnTotal(int[][] a, int col)` – This method returns the total of the values in the specified column.
- `public static int getHighestInRow(int[][] a, int row)` – This method returns the highest value in the specified row of the array.
- `public static int getLowestInColumn(int[][] a, int col)` – This method returns the lowest value in the specified column of the array.

(b) Write a driver program, called `TwoDarrayTest.java`, that demonstrates the `TwoDarray` utility class by doing the following:

- creating a 2D array using initialization list `{{2, 1, 9}, {7, 3, 4}}`
- calling appropriate methods in the utility class and generate the following output:

```
fcsang@fluffy ~/cs1400/codingPractice $ java TwoDarrayTest
Processing the int array.
Total: 26
Average: 4.333333333333333
Total of row 0: 12
Total of row 1: 14
Total of col 0: 9
Total of col 2: 13
Highest in row 0: 9
Highest in row 1: 7
Lowest in col 0: 2
Lowest in col 1: 1
```

Task #2 Phone Book `ArrayList`

Write a class named `PhoneBookEntry` that has two fields for a person's name and phone number. The class should have a two-argument constructor, appropriate setter and getter methods, and the `toString` method that returns a string containing person's name and phone number enclosed in a pair of parentheses and separated by a comma, e.g. (Daisy, 869-3469).

Then write a driver program `PhoneBookTest.java` that creates five `PhoneBookEntry` objects and stores them in an `ArrayList`. Use a loop to access and display the contents of each object in the `ArrayList`.

Now insert your name and phone number to the beginning of the `ArrayList` and display the contents of the phone book again. This time use the `toString` method defined in the `ArrayList` class (don't use a loop). At the end, remove the last entry in the phone book and display the result. The following are sample interactions when running the program, where user's input are shown in bold:

```
fcsang@garrison ~/cs1400/codingPractice $ java PhoneBookTest
I'm going to ask you to enter 5 names and phone numbers.
Enter a person's name: A Amamra
Enter that person's phone number: 869-3447
Enter a person's name: T Chen
Enter that person's phone number: 869-4842
Enter a person's name: M Husain
Enter that person's phone number: 869-2022
Enter a person's name: H Ji
Enter that person's phone number: 869-5521
Enter a person's name: A Raheja
Enter that person's phone number: 869-4412

Here's the data you entered:
(A Amamra, 869-3447)
(T Chen, 869-4842)
(M Husain, 869-2022)
(H Ji, 869-5521)
(A Raheja, 869-4412)

Insert my name to the beginning of phone book:
[(D Sang, 869-3469), (A Amamra, 869-3447), (T Chen, 869-4842), (M Husain,
869-2022), (H Ji, 869-5521), (A Raheja, 869-4412)]

Remove the last entry:
[(D Sang, 869-3469), (A Amamra, 869-3447), (T Chen, 869-4842), (M Husain,
869-2022), (H Ji, 869-5521)]
```

Submission:
Generate a `script` file `practice9.txt` with appropriate time stamps and the following steps visible:

1) a `pwd` to show the current working directory
2) a `ls -l` to show in long format the files in your `cs1400/codingPractice` directory
3) display both `TwoDarray.java` and `TwoDarrayTest.java`
4) compile `TwoDarrayTest.java`
5) run `TwoDarrayTest`
6) display both `PhoneBookEntry.java` and `PhoneBookTest.java`
7) compile `PhoneBookTest.java`
8) run `PhoneBookTest`

Submit the script file `practice9.txt` on Bb, under the Coding Practice Folder, Practice #9 link.