CSC 1051 - Lab 14

Objectives:
Exercises on arrays of objects and two-dimensional arrays

1) Arrays of objects
Suppose you have a Person class as follows:

```java
public class Person {
    private String name;
    private int age;

    public Person (String name, int age) {
        this.name = name;
        this.age = age;
    }

    // increases a Person’s age
    public void grow(int n)

    // returns a string with the Person’s info
    public String toString ()

    // like toString(), but **prints** the Person’s info
    public void printInfo ()
}
```

Suppose you want to use the Person class in a program that stores information about people. Implement your class and experiment with it until you are sure how to accomplish each of the following:

a) Declare (but not instantiate) an array named `people` that stores `Person` objects

b) Instantiate an array of 5 `Person` objects and assign it to `people`
c) Create a `Person` object for someone named Maddie who is 14, and store it as the first element in the array `people`.

d) Supposing that the array `people` contains references to actual `Person` objects (like the one for Maddie), show how to use the `toString()` method to print out the name and age of the last person in the array.

e) Repeat using `printInfo()`.

f) Grow everyone in `people` by 3 years. (write code fragment here)

g) Suppose the class also contains a `getAge()` method. Write a code fragment that uses this to compute and print the average age of the five `Person` objects stored in `people`. 
h) Write a method that takes an array of Person as parameter and “grows” each person by one year. The method should not return anything. It should work with any array of Person, of any size. (To test it, implement it as a static method in the driver class.)

i) Show how to invoke the method above to grow everyone in the array people. (Assume it is a static method in the same class.)

2) 2D Arrays
Create a test program to experiment and to test each of the following code fragments and show the array contents after execution. Note that you will need to write some extra code to display the array contents - you can see how this is done in the example. You will need to adapt this code to print out the array contents of arrays of different sizes.

EXAMPLE:
```java
int count = 1;
int[][] table = new int[2][3];
for (int i=0; i < 2; i++)
    for (int j=0; j < 3; j++)
    {
        table[i][j] = count;
        count++;
    }

//print out the array contents
for (int i=0; i < 2; i++) // processing rows
{
    for (int j=0; j < 3; j++) //processing columns
        System.out.print( table[i][j] + " ");
    System.out.println(); // done with row, go to new line
}
```

ANSWER:
```
table
```
a)

```java
int[][] table = new int[4][4];
for (int i=0; i < 4; i++)
    table[i][i] = i;
```

b)

```java
char[][] table = new char[4][4];
String sample = "OPEN THIS FIRST";
for (int i=0; i < 4; i++)
    for (int j=0; j < 4; j++)
        table[i][j] = sample.charAt(i+j);
```

c)

```java
int[][] table = new int[3][4];
for (int i=0; i < table.length; i++)
    for (int j=0; j < table[i].length; j++)
        table[i][j] = i * 2 + j;
```