COMP228– Array Practice

Assignment 3 (Bonus)
Array Practice

Due Week #9 (Friday November 13, 2015) @ midnight.

Value 10%

Array Practice **Maximum Mark: 24**

Overview: Using Java and Eclipse (or an IDE of your choice), you will write 2 small programs to demonstrate your understanding of Arrays and / or ArrayList structures.

Instructions :
(24 Marks: Functionality)

1. **Dice Rolling App**. Write an app to simulate the rolling of two dice and displaying how many times each result was rolled (10 Marks: Functionality).
    a. The app should randomly roll the first die and the second die (2 Marks: Functionality).
    b. The sum of the two values should then be calculated (2 Marks: Functionality).
    c. Each die can show an integer value from 1 to 6, so the sum of the values will vary from 2 to 12, with 7 being the most frequent sum and 2 and 12 the least frequent sums. The figure below shows the 36 possible combinations of the two dice. Your app should roll the dice **36,000 times** (2 Marks: Functionality)

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

    j.
    d. Use a one-dimensional array (or an ArrayList) to **tally** the number of times each possible sum appears (2 Marks: Functionality).
    e. Display the results in list format to the console (or using a GUI window) showing how many times each number was rolled (2 Marks: Functionality).

2. **Airline Reservations System**. A small airline has just purchased a computer for its new automated reservations system. You have been asked to develop the new system. You're to write an app (Console or GUI) to assign seats on each flight of the airline's only plane (capacity: 10 seats) **(14 Marks: Functionality).**
    a. Display the following alternatives in a Console-based menu (or appropriate UI component): **1 for First Class**, **2 for Economy, 3 for Exit**. You **must** use a switch statement and a while loop (4 Marks: Functionality)
    b. If the user selects option **1**, your app should assign a seat in the first-class section (seats 1–5). If the user selects option **2**, your app should assign a seat in the economy section (seats 6–10). Use a one-dimensional array of type bool (or an alternate ArrayList structure) to represent the seating chart of the plane (2 Marks: Functionality).
    c. **Initialize** all the elements of the array to **false** to indicate that all the seats are empty (or for an ArrayList structure, allocate the appropriate number of elements for First class seats and Economy seats.) (2 Marks: Functionality).
    d. As each seat is assigned, set the corresponding element of the array to **true** to indicate that the seat is no longer available (for an ArrayList structure, remove the seat from list). (2 Marks: Functionality).
    e. Your app should never assign a seat that has already been assigned. When the economy section is full, your app should ask the person if it's acceptable to be placed in the first-class section (and vice versa). If yes, make the appropriate seat assignment. If no, display the message "Next flight leaves in 3 hours." (4 Marks: Functionality).

SUBMITTING YOUR WORK

Your submission should include:

1. Individual zip archives of your project Files for Each program

This assignment is weighted **10%** of your total mark for this course.

Late submissions:

- 10% deducted for each additional day.

External code (e.g. from the internet or other sources) can be used for student submissions within the following parameters:

1. The code source (i.e. where you got the code and who wrote it) must be cited in your internal documentation.
2. It encompasses a maximum of 10% of your code (any more will be considered cheating).
3. You must understand any code you use and include documentation (comments) around the code that explains its function.
4. You must get written approval from me via email.