# Assignment 3

*ArrayLists instead of arrays*

## Background:

You have been asked to fix and modify a program which was created to track theatre seats. The program has a few errors and is using arrays. The client wants you to use ArrayLists in your solution. This will be a challenging exercise, but provide you with a lot of practice working with ArrayLists.

## Requirement 1:

The solution is not displaying the price as requested when the program asks for the name of the person requesting the ticket.

## Requirement 2:

Fix any syntax or logic issues that are causing the program to work incorrectly. Would want to check the switch statement and the do..while in the main method.

## Requirement 3:

Modify the program so that there is another class called Row. This class should contain an attribute to hold the  row number and an attribute of type ArrayList of Seats. The rows in this ArrayList will contain an element for the rows in the theatre. These rows do not have to be initialized to a lot of empty rows. As rows are needed, they can be added to the ArrayList.

A cost can be added to each row since the price is determined by the row number. Ie. The first five rows would be $100 and the others$70.

The Theatre class should have an ArrayList of Rows. Each time the user requests to reserve a seat, the code would have to loop through the ArrayList of Rows to see if there is an entry already for that row number. A boolean flag could be used to allow you to track whether or not an entry for the requested row was found.

If the row was not found, then it is a given that the seat would not be found. A new Row would have to be created and added to the collection of rows. The new row would be used in further processing.

If there was an entry found for the row, there should be code to loop through the Seats in the row to see if there is a Seat instance that matches the seat number being requested. The Seat class will need to have an additional attribute to hold the seat number. Again, another boolean flag could be used to determine whether or not the seat was found. If the seat is found in the ArrayList of Seats, then it is known that that seat is already booked. If no seat is found, then

the seat is not already reserved and a new Seat would have to be created and added to the ArrayList.  This Seat would then show up next time that the user tries to reserve a seat.

ArrayLists are the preferred method since the client has no knowledge of other collection types. .  Have to move away from arrays for this solution.

# Requirement 4:

You can test your solution by invoking the showTheatre method (which has been created in the main class.

Fixes to showTheatre
- Input parameter should be a Theatre instance not an ArrayList of Theatre types
- Fix the constant name for the max number of rows in a theatre.
- It is called from the static main method so a change may have to be made to accomodate this.

This method should work with your Theatre class after you have made the modifications above. This method should not be changed further.  This method expects to receive a Theatre instance and use this class' attributes to display the theatre information.

The showTheatre method in the main method should be invoked when the user selects s.

The showTotal method will also have to be modified to work with the ArrayLists.