

# Assignment #7

## CSE110 - Arizona State University

### *Topics*

- 2-Dimensional Arrays
- Classes
- Searching
- Reading from a file

### *Coding Guidelines:*

- Give identifiers semantic meaning and make them easy to read (examples numStudents, grossPay, etc).
- Keep identifiers to a reasonably short length.
- Use upper case for constants. Use title case (first letter is upper case) for classes. Use lower case with uppercase word separators for all other identifiers (variables, methods, objects).
- Use tabs or spaces to indent code within blocks (code surrounded by braces). This includes classes, methods, and code associated with ifs, switches and loops. Be consistent with the number of spaces or tabs that you use to indent.
- Use white space to make your program more readable.

### *Part #1: Written Exercises (0 pts)*

None.

### *Part #2 - Programming (20 pts)*

Write a program that reads customers' information from a file, and creates a movie theatre seating with a number of rows and columns specified by a user. Then it will attempt to assign each customer to a seat in a movie theatre.

You will be creating a class called `MovieSeating`. This class should be defined in a file named `MovieSeating.java`. The class `MovieSeating` will contain a 2 dimensional array called `seating` of `Customer` objects as its instance variable.

We will be using the following files. **These files are complete.** Download the following files (available on Blackboard) and use them for this assignment (do not change the content of the following files).

- `Assignment7.java`
- `Customer.java`
- `CustomerData.java`

The last file is an input file (text file) that will be read from the `Assignment7` class. Save all of the files in the same folder.

The class `MovieSeating` must include the following constructor and methods. (If your class does not contain any of the following methods, points will be deducted.)

- `public MovieSeating(int rowNum, int columnNum)` - It instantiates a two dimensional array of the size `rowNum` by `columnNum` specified by the parameters. Then it initializes each customer element of this array using the constructor of the class `Customer` without any parameter. So each customer will have default values for its instance variables.
- `private Customer getCustomerAt(int row, int col)` - It returns a customer at the indexes `row` and `col` (specified by the parameters of this method) of the array `seating`.
- `public boolean assignCustomerAt(int row, int col, Customer tempCustomer)` - The method attempts to assign `tempCustomer` to the seat at `row` and `col` (specified by the parameters of this method). If the seat has a default customer, i.e., a customer with the last name "???" and the first name "???", then we can assign the new customer `tempCustomer` to that seat and the method returns `true`. Otherwise, this seat is considered to be taken by someone else, the method does not assign the customer and returns `false`.
- `public boolean checkBoundaries(int row, int col)` - The method checks if the parameters `row` and `col` are valid. If at least one of the parameters `row` or `col` is less than 0 or larger than the last index of the array (note that the size of rows and columns can be different), then it returns `false`. Otherwise it returns `true`.
- `public String toString()` - Returns a `String` containing information of the `seating`. It should show the list of customers assigned to the seating using the `toString` method of the class `Customer` (it shows initials of each customer) and the following format:

```
The current seating
-----
C.B. ?.?. E.P.
??.?. ??.?. G.B.
B.C. H.C. ?.?.
```

Please see the sample output listed below.

### *Helpful Hints*

- Work on it in steps - write one method, test it with a test driver and make sure it works before going on to the next method.
- Always make sure your code compiles before you add another method.
- Your methods should be able to be called in any order.

### *Sample Output*

Make sure that your program works at least with this scenario (the inputs entered by a user are shown in bold).

Please enter a number of rows for a movie theatre seating

**3**

Please enter a number of columns for a movie theatre seating.

**3**

Please enter a file name

**customerData.txt**

A customer information is read from a file.

George Bush/11111111/3/4

Please enter a row number where the customer wants to sit.

**1**

Please enter a column number where the customer wants to sit.

**2**

The seat at row 1 and column 2 is assigned to the customer G.B.

The current seating

---

?.?. ?.?. ?.?.  
?.?. ?.?. G.B.  
?.?. ?.?. ?.?.

A customer information is read from a file.

Bill Clinton/22222222/6/5

Please enter a row number where the customer wants to sit.

**2**

Please enter a column number where the customer wants to sit.

**0**

The seat at row 2 and column 0 is assigned to the customer B.C.

The current seating

---

?.?. ?.?. ?.?.  
?.?. ?.?. G.B.  
B.C. ?.?. ?.?.

A customer information is read from a file.

Hilary Clinton/44444444/5/3

Please enter a row number where the customer wants to sit.

**2**

Please enter a column number where the customer wants to sit.

**1**

The seat at row 2 and column 1 is assigned to the customer H.C.

The current seating

---

?.?. ?.?. ?.?.  
?.?. ?.?. G.B.  
B.C. H.C. ?.?.

A customer information is read from a file.

Charlie Brown/33333333/4/3

Please enter a row number where the customer wants to sit.

**0**

Please enter a column number where the customer wants to sit.

**0**

The seat at row 0 and column 0 is assigned to the customer C.B.

The current seating

---

C.B. ?? ??.  
?? ?? G.B.  
B.C. H.C. ??.

A customer information is read from a file.  
David Beckham/555666777/4/5  
Please enter a row number where the customer wants to sit.  
**5**  
Please enter a column number where the customer wants to sit.  
**1**  
Row or column number is not valid.  
A customer David Beckham is not assigned a seat.

A customer information is read from a file.  
David Johnson/666888999/16/5  
Please enter a row number where the customer wants to sit.  
**2** Please enter a column number where the customer wants to sit.  
**0**

The seat is taken.

A customer information is read from a file.  
Snow White/777777777/43/23  
Please enter a row number where the customer wants to sit.  
**-1**  
Please enter a column number where the customer wants to sit.  
**0**

Row or column number is not valid.  
A customer Snow White is not assigned a seat.

A customer information is read from a file.  
Elvis Presley/888888888/2/4  
Please enter a row number where the customer wants to sit.  
**0**  
Please enter a column number where the customer wants to sit.  
**2**

The seat at row 0 and column 2 is assigned to the customer E.P.  
The current seating

---

C.B. ?? E.P.  
?? ?? G.B.  
B.C. H.C. ??.

## ***Submission***

- Go to the course web site (my.asu.edu), and then click on the on-line Submission tab.
- Submit your **Assignment7.java, MovieSeating.java, Customer.java, and customerData.txt** files on-line. Make sure to choose Hw7 from drop-down box.

**Important Note:** You may resubmit as many times as you like until the deadline, but we will only mark your last submission.

**NO LATE ASSIGNMENTS WILL BE ACCEPTED.**