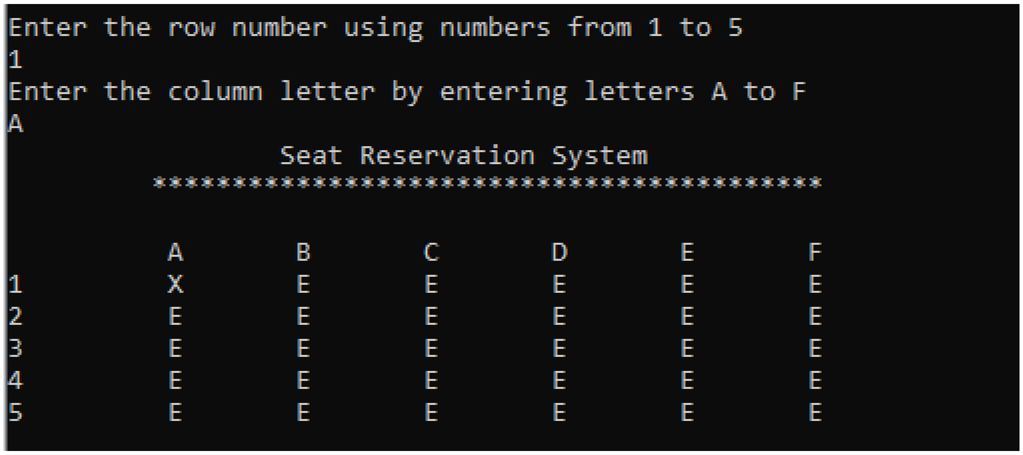
Lab Exercise 4- Advance OOP Concepts

General Description:

The program is a seat reservation system where users reserve a seat. There are two types of seat the matinee seats and the regular seats both having different prices, 20 for matinee and 10 for regular. The program will display the number of seats bought and the total price.

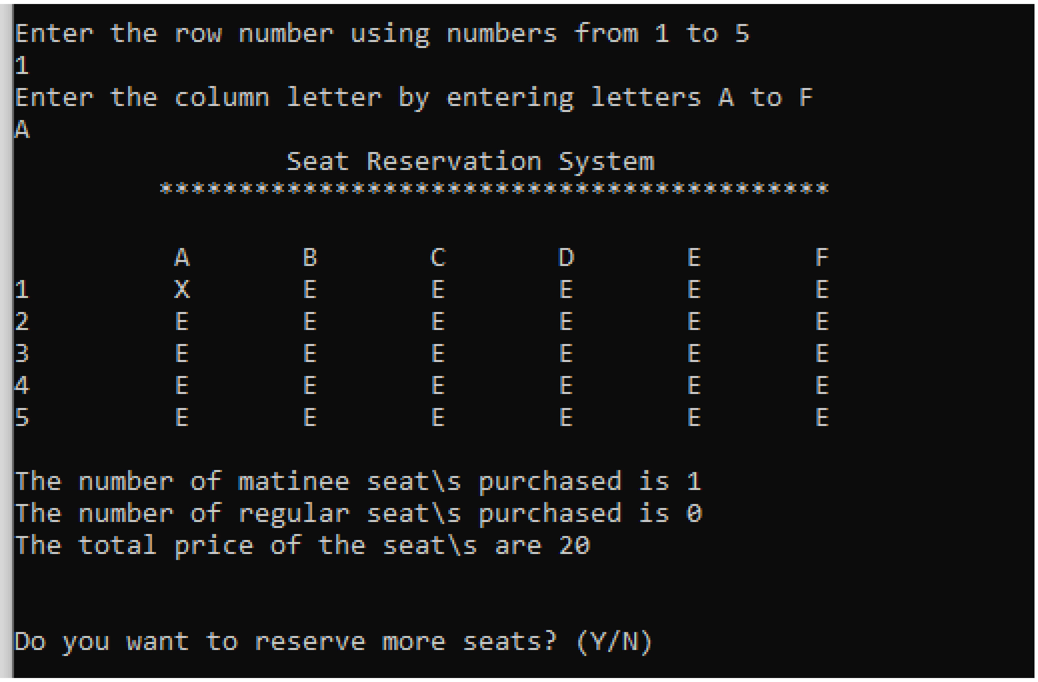
Sample output:

1. The program will prompt the user to enter the rows from 1 to 5.
2. If the user enters an input that is not 1 to 5, the program will prompt the user to enter an input again.
3. The program will prompt the user to enter the columns from A to F.
4. If the user enters an input that is not compliant to number 3, the program will prompt the user to enter an input again.
5. The program will then mark the seats using the following output:



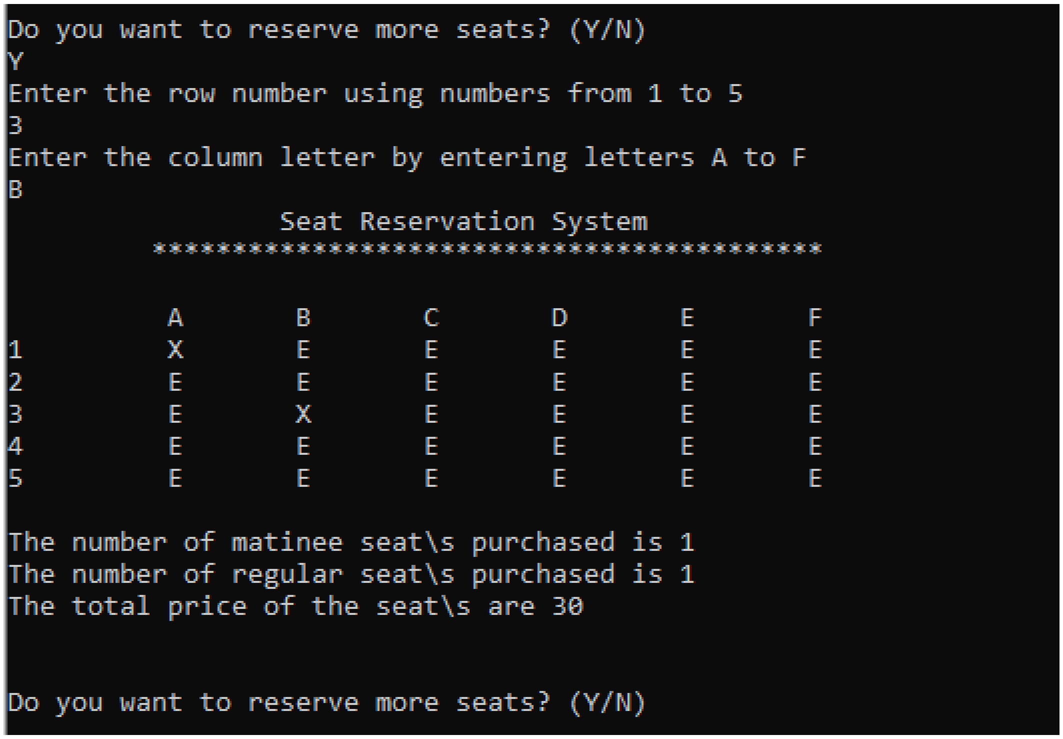
```
Enter the row number using numbers from 1 to 5
1
Enter the column letter by entering letters A to F
A
                 Seat Reservation System
         ******************************************

          A       B       C       D       E       F
1         X       E       E       E       E       E
2         E       E       E       E       E       E
3         E       E       E       E       E       E
4         E       E       E       E       E       E
5         E       E       E       E       E       E
```

6. At the end of the matrix, the program will also compute the number of regular or matinee seats purchased by the user.



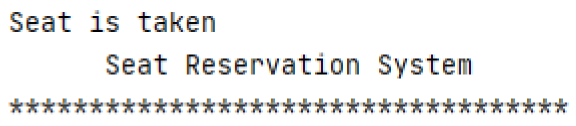
```
Enter the row number using numbers from 1 to 5
1
Enter the column letter by entering letters A to F
A
                 Seat Reservation System
         ******************************************

          A       B       C       D       E       F
1         X       E       E       E       E       E
2         E       E       E       E       E       E
3         E       E       E       E       E       E
4         E       E       E       E       E       E
5         E       E       E       E       E       E

The number of matinee seat\s purchased is 1
The number of regular seat\s purchased is 0
The total price of the seat\s are 20


Do you want to reserve more seats? (Y/N)
```

7. The program will then prompt the user if he or she wants to make more reservations.
8. The program should be able to validate the input for number 7 and if it is wrong should prompt the user accordingly.
9. If the user says, yes, the program will again prompt the user to input a row and column and mark the appropriate seat accordingly.



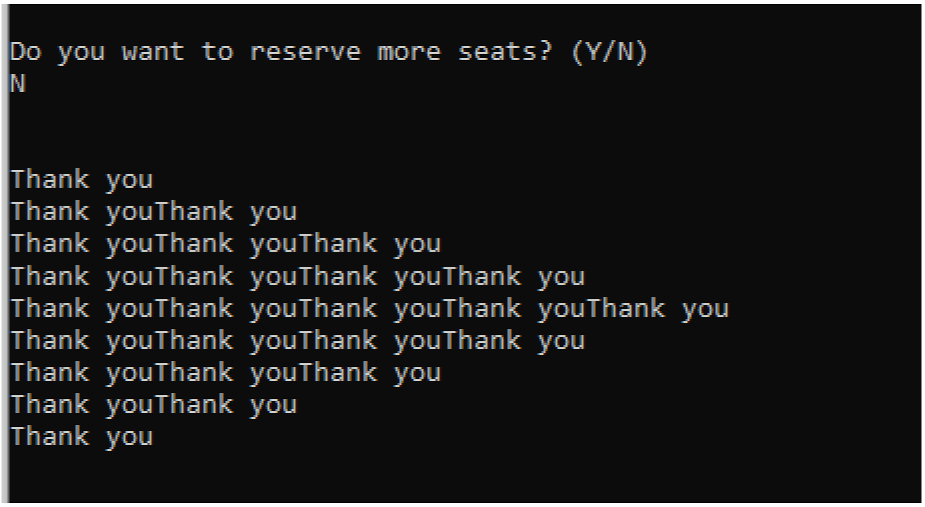
```
Do you want to reserve more seats? (Y/N)
Y
Enter the row number using numbers from 1 to 5
3
Enter the column letter by entering letters A to F
B
                 Seat Reservation System
         ******************************************

          A       B       C       D       E       F
1         X       E       E       E       E       E
2         E       E       E       E       E       E
3         E       X       E       E       E       E
4         E       E       E       E       E       E
5         E       E       E       E       E       E

The number of matinee seat\s purchased is 1
The number of regular seat\s purchased is 1
The total price of the seat\s are 30


Do you want to reserve more seats? (Y/N)
```

10. If the user enters seat that is already taken, the program will prompt the user that that seat is taken.



```
Seat is taken
      Seat Reservation System
***********************************
```

11. If the user says no, the following will be displayed:



```
Do you want to reserve more seats? (Y/N)
N


Thank you
Thank youThank you
Thank youThank youThank you
Thank youThank youThank youThank you
Thank youThank youThank youThank youThank you
Thank youThank youThank youThank you
Thank youThank youThank you
Thank youThank you
Thank you
```

General Guidelines

1. The program should have 5 classes and a two-dimensional array
2. The first class will be called the **mainapp**, second is **seats**, third is **matinee**, fourth is **regular** and the fifth is an interface named **thank_you**.
3. The seats class will have the following purpose and specifications:
   a. Initializes the array for output. This means putting E for all seats.
   b. Display the initial array.
   c. Marks the X for seats reserved.
   d. It should have a constructor that accepts the row and column values.
   e. Displays the seat plan again once a seat is reserved.
4. The matinee class inherits from the seats class and will have the following purpose and specifications:
   a. The first row of the seat plan (row 1, Columns A to F) are considered matinee
   b. Their ticket price is 20 dollars per seat.
   c. The matinee class should be able to count the number of matinee seats and compute the total matinee seats purchased.
5. The regular class inherits from the seats class and will have the following purpose and specifications:
   a. All the other seats besides the matinee are considered regular.
   b. The ticket price for regular is 10 dollars per seat.

    c. The regular class should be able to count the number of regular seats and compute the total regular seats purchased.

6. The mainapp class will have the following purpose and specifications:
    a. Accept entry of row and column form the user and pass these values to the respective classes.
    b. All columns and rows should have proper validations. This means it is going to prompt the user again and again if the user enters an incorrect value.
    c. Instantiate the necessary objects.
    d. Compute the total bill of the user.
    e. Ask the user for reserving more seats.
    f. Call the class methods that are needed.
    g. Implement the thank you class.
7. The Thankyou is an interface. It will have one abstract method called display. The mainapp class is going to implement the display method which will display the group of "thank you" as shown in the picture. You must use a loop to complete this.
8. Be able to utilize exception handling in validations.
9. The program should be running and output correct.
10. Adequate prompts and documentation should be provided.