Programming fundamentals

c-cs112

(spring 2024)

Team Members

Student Name	Student ID
Omar Mohamed	23-101149
Maroska Osama	23-101209
Youssef Ihab	23-101138
Hamdy El Saeed	23-101232
Omar El Sayed	23-101091

Table of contents

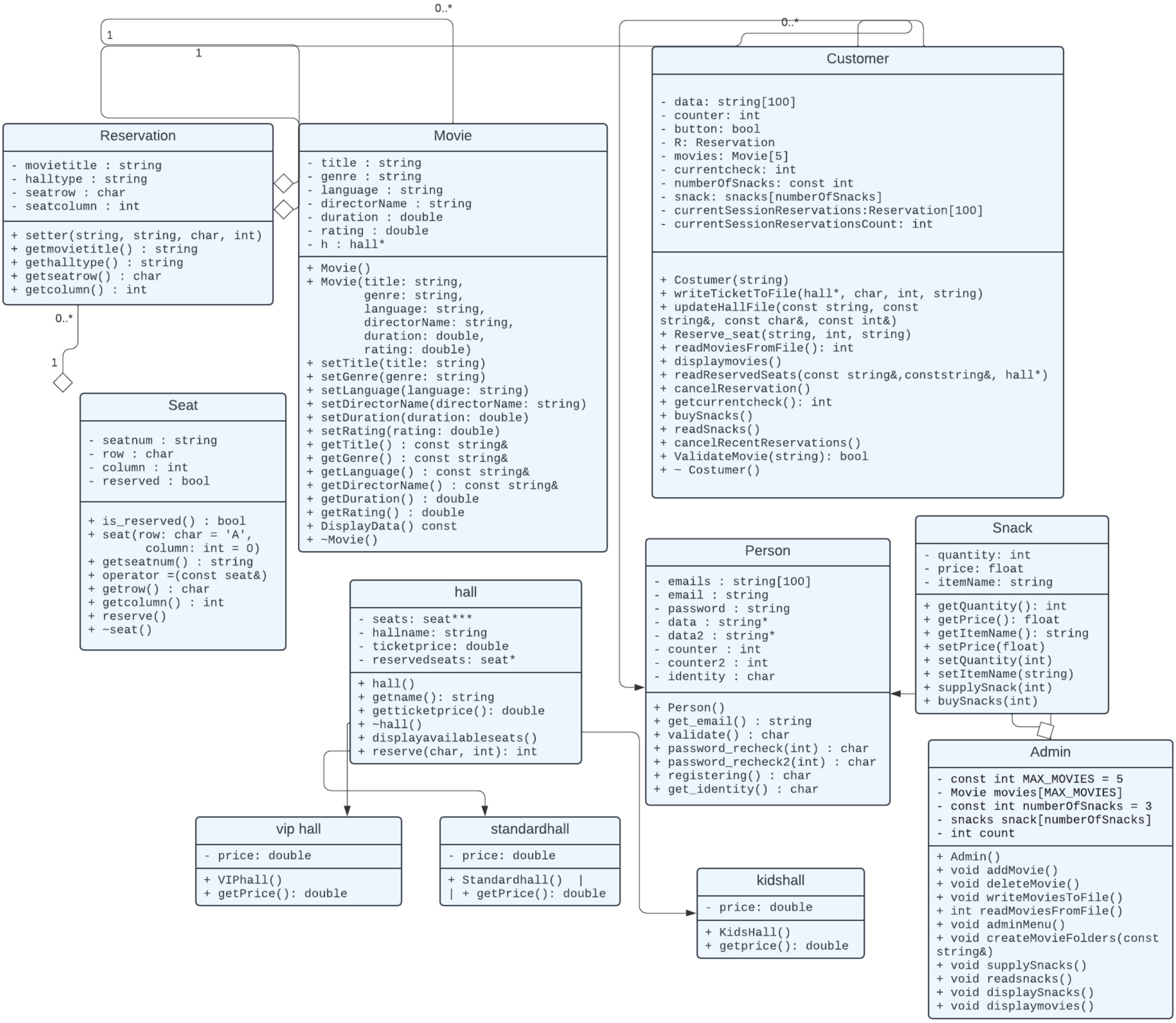
The abstract:

This report outlines how a computer software system was designed and developed using the programming language C++. Our program provides 2 modes for the ticket reservation system in the cinema, the first one is the admin mode which provide some features that are only available for the admins such as: providing a menu for adding movies, deleting movies, and also to save them in files. the other mode is for the customers to provide for the full experience in reserving a place for the movie of their choice by selecting the movie name, and the seat number also our program provides an option for the user to view the movie details.

The introduction:

our main target is to provide a full experience for the customers in reserving their cinema tickets that make it easier for them to choose between the variety of movies the cinema provides for them. Not only that but also to allow the admins to supply the system with many helpful features with plenty of validations.

The system UML:



The admin class

Functions break down

Function	Description
Admin	Constructor that initializes the Admin object and calls `adminMenu` and `readMoviesFromFile`.
addMovie	Adds a new movie to the list of movies and creates necessary directories and files for it.
deleteMovie	Deletes a movie from the list of movies and removes its directory.
writeMoviesToFile	Writes the current list of movies to the `movies.txt` file.
readMoviesFromFile	Reads the list of movies from the `movies.txt` file and updates the movie array.
adminMenu	Displays the admin menu and handles user input for different admin operations.
createMovieFolders	Creates necessary directories and files for a new movie.
supplySnacks	Supplies additional units of a specified snack and updates the `snacks.txt` file.
readsnacks	Reads the current list of snacks from the `snacks.txt` file.
displaySnacks	Displays the list of available snacks with their prices and quantities.
displaymovies	Displays the list of movies currently available.
deleteDirectory	Utility function to delete a directory and its contents.

the purpose of admin class:

it allows the admin to have control over the movies by adding /deleting movies in addition to entering the movie details, and also allows the admin to have control over the snacks by monitoring the stock of the snacks and displaying them.

the customer class:

the function breakdown:

Function	Description
Costumer(string)	Constructor that initializes the `Costumer` object with an email
writeTicketToFile(hall*, char, int, string)	Writes ticket information to a file
updateHallFile(const string, const string&, const char&,const int&)	Updates hall file with reservation information
Reserve_seat(string, int, string)	Reserves seats for a movie in a specific hall based on type and quantity
readMoviesFromFile(): int	Reads movie information from a file and initializes the `movies` array
displaymovies()	Displays the available movies
readReservedSeats(const string&, const string&, hall*)	Reads reserved seats information from a file into the hall object
cancelReservation()	Cancels a reservation based on user input
getcurrentcheck(): int	Returns the current check amount
buySnacks()	Allows the customer to buy snacks
readSnacks()	Reads snack information from a file into the `snack` array
cancelRecentReservations()	Cancels recent reservations made in the current session
ValidateMovie(string): bool	Validates if a movie exists based on its title
~Costumer()	Destructor to clean up memory

The purpose of the costumer class:

The system oversees multiple facets associated with client communications, including movie ticket sales, snack purchases, and reservation cancellations. It also manages reading and writing movie, reservation, ticket, and snack data to and from files.

the hall class:

the function breakdown:

Function	Description
hall()	Constructor for the `hall` class. Initializes attributes, including `seats`, `hallname`, `ticketprice`, and `reservedseats`.
getname(): string	Retrieves the name of the hall.
getticketprice(): double	Retrieves the ticket price for the hall.
~hall()	Destructor for the `hall` class. Cleans up memory allocated for `seats` and `reservedseats`.
displayavailableseats()	Displays the available seats in the hall.
reserve(char, int): int	Reserves a seat in the hall based on the provided row and column numbers. Returns 1 if successful, 0 otherwise.

the purpose of this code:

The purpose of the hall class is to provide a representation of movie halls in a cinema allowing for seat reservation, information retrieval, and display of available seats.

The kids hall class:

The function breakdown:

Function	Description
KidsHall()	Constructor for the `KidsHall` class. It initializes the `price` member variable.
getprice()	Public member function that returns the price of the kids hall. It overrides the base class function.

The purpose of this class:

provide a specialized representation of a hall suitable for children's events, offering customized features and functionality while inheriting common traits from a hall class.

The standard hall class:

The function breakdown:

Function	Description
Standardhall()	Constructor for the `Standardhall` class. It initializes the `price` member variable.
getPrice()	Public member function that returns the price of the standard hall. It overrides the base class function.

The purpose of this class:

class provides a basic representation of a hall that has a lower price range

The VIP hall class:

the function breakdown:

Function	Description
VIPhall()	Constructor for the `VIPhall` class. It initializes the `price` member variable.
getPrice()	Public member function that returns the price of the VIP hall. It overrides the base class function.

the purpose of the this class:

the VIPhall class serves the purpose of providing a specialized and premium representation of a hall, designed to meet the needs and expectations of VIP guests or events.

Movie class:

The function breakdown:

Function	Purpose
`Movie()`	Default constructor for creating a Movie object. Initializes member variables to default values.
`Movie(title, genre, language, directorName, duration, rating)`	Constructor for creating a Movie object with specified attributes.
`setTitle(title)`	Sets the title of the movie.
`setGenre(genre)`	Sets the genre of the movie.
`setLanguage(language)`	Sets the language of the movie.
`setDirectorName(directorName)`	Sets the name of the director of the movie.
`setDuration(duration)`	Sets the duration of the movie (in minutes).
`setRating(rating)`	Sets the rating of the movie.
`getTitle()`	Retrieves the title of the movie.
`getGenre()`	Retrieves the genre of the movie.
`getLanguage()`	Retrieves the language of the movie.
`getDirectorName()`	Retrieves the name of the director of the movie.
`getDuration()`	Retrieves the duration of the movie.
`getRating()`	Retrieves the rating of the movie.
`DisplayData()`	Displays all the data related to the movie (title, genre, language, director's name, duration, rating).
`~Movie()`	Destructor for properly deallocating resources associated with a Movie object.

The purpose of the class:

This class allows the admin to set the details for each movie and display them when needed at the end.,such as name , director ,duration and the rating.

The person class:

The function breakdown:

Function	Purpose
`Person()`	Default constructor for creating a `Person` object. Initializes member variables to default values.
`get_email()`	Retrieves the email address associated with the `Person` object.
`validate()`	Validates the email address entered by the user. Returns a character indicating the validation result.
`password_recheck(int)`	Checks the validity of the password based on certain criteria (e.g., length, complexity). Returns a character indicating the validation result.
`password_recheck2(int)`	Checks the validity of the password again. Similar to `password_recheck(int)`. Returns a character indicating the validation result.
`registering()`	Handles the registration process for the `Person`.
`get_identity()`	Retrieves the identity (e.g., role) associated with the `Person` object.

The purpose of the class:

the purpose of the Person class is to revolve around managing account-related data, handling registration and authentication processes, and managing additional information associated with each person's account.

The seat class:

The function breakdown:

Function	Purpose
`is_reserved()`	Checks whether the seat is reserved or not.
`seat(row: char = 'A', column: int = 0)`	Constructor for creating a `seat` object with the specified row and column. Defaults to row 'A' and column 0 if not provided.
`getseatnum()`	Retrieves the seat number.
`operator =(const seat&)`	Assignment operator overload, assigns the values of one `seat` object to another.
`getrow()`	Retrieves the row of the seat.
`getcolumn()`	Retrieves the column of the seat.
`reserve()`	Marks the seat as reserved.
`~seat()`	Destructor for cleaning up resources associated with a `seat` object.

The purpose of the class:

The function of this class is to reserve a seat and get its coordinates for the columns and the rows and check if the seat is reserved or not.

The reservation class:

The function breakdown:

Function	Purpose
`setter(string, string, char, int)`	Sets the values for the reservation, including movie title, hall type, seat row, and seat column.
`getmovietitle()`	Retrieves the movie title associated with the reservation.
`gethalltype()`	Retrieves the hall type associated with the reservation.
`getseatrow()`	Retrieves the row of the reserved seat.
`getcolumn()`	Retrieves the column of the reserved seat.

The purpose of the class:

The reason for this class is to get the reservation details such as the movie title, the hall type, and the reserved seat information.

The snacks class:

The function breakdown:

Function	Description
getQuantity()	Retrieves the quantity of the snack items in stock.
getPrice()	Retrieves the price of the snack item.
getItemName()	Retrieves the name of the snack item.
setPrice(float)	Sets the price of the snack item.
setQuantity(int)	Sets the quantity of the snack items in stock.
setItemName(string)	Sets the name of the snack item.
supplySnack(int)	Increases the quantity of snack items in stock by the specified amount.
buySnacks(int)	Decreases the quantity of snack items in stock by the specified amount.

The purpose of the class:

The snacks class is like a toolbox specifically designed to handle information about snack items. It keeps track of how many snacks there are, how much they cost, and what they're called.

Workload distribution:

Youssif Ihab: Classes Hall and 50% from admin, user and file handling

Omar Mohamed: User Menus, validations admin, 50% from admin, user, class snacks,file handeling

Maroska Osama: Classes Movie, person, report, and ppt

Hamdy El Saeed: Classes Costumer, Admin, file handling

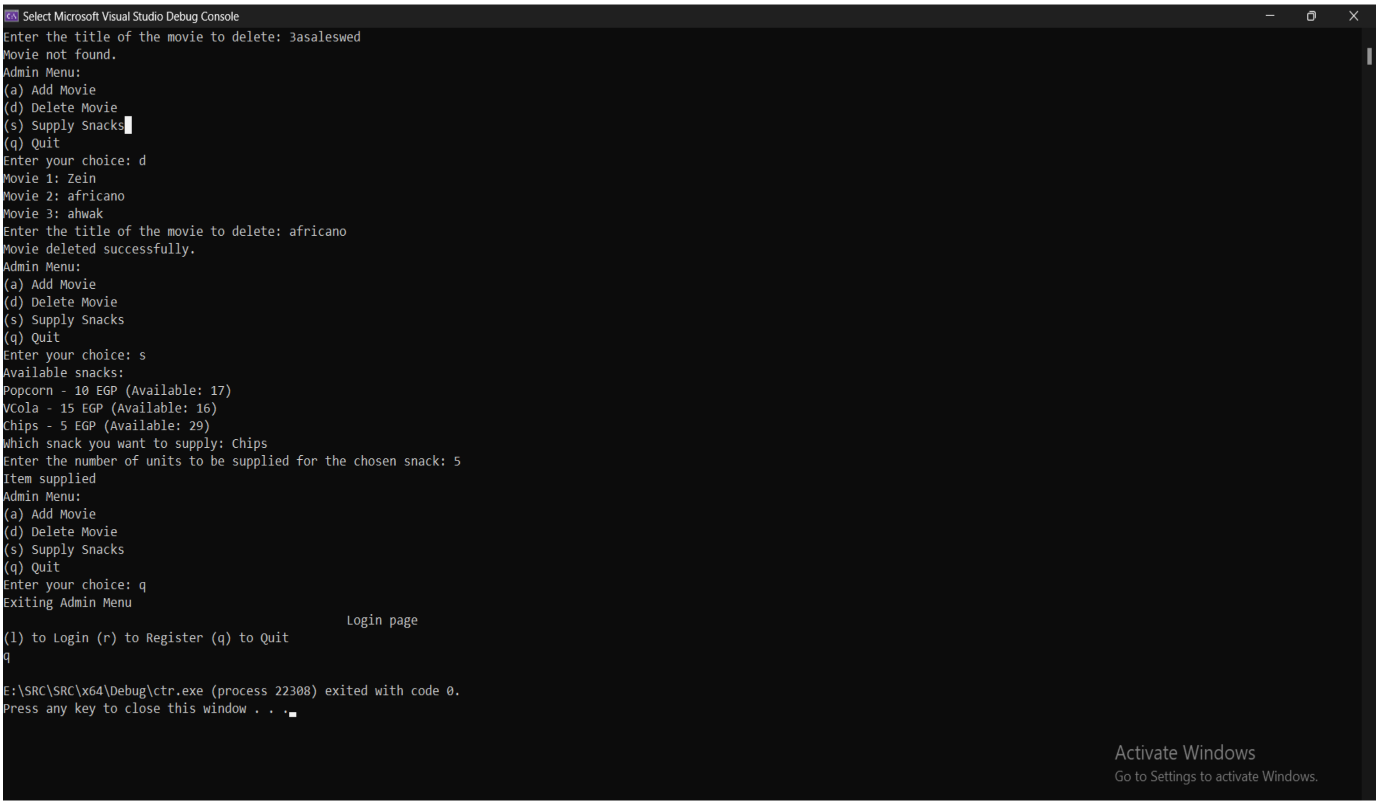
Omar El Sayed: Classes Seats and Reservation

Challenges and conclusions:

The main challenge that we have faced is to link the classes together, as when we linked them the main error was the functions weren't running correctly beside the validation errors in the classes. The biggest challenge is to make the files work as it destroyed some of the code but at the end, it worked successfully. In addition to linking the qt GUI with our classes.

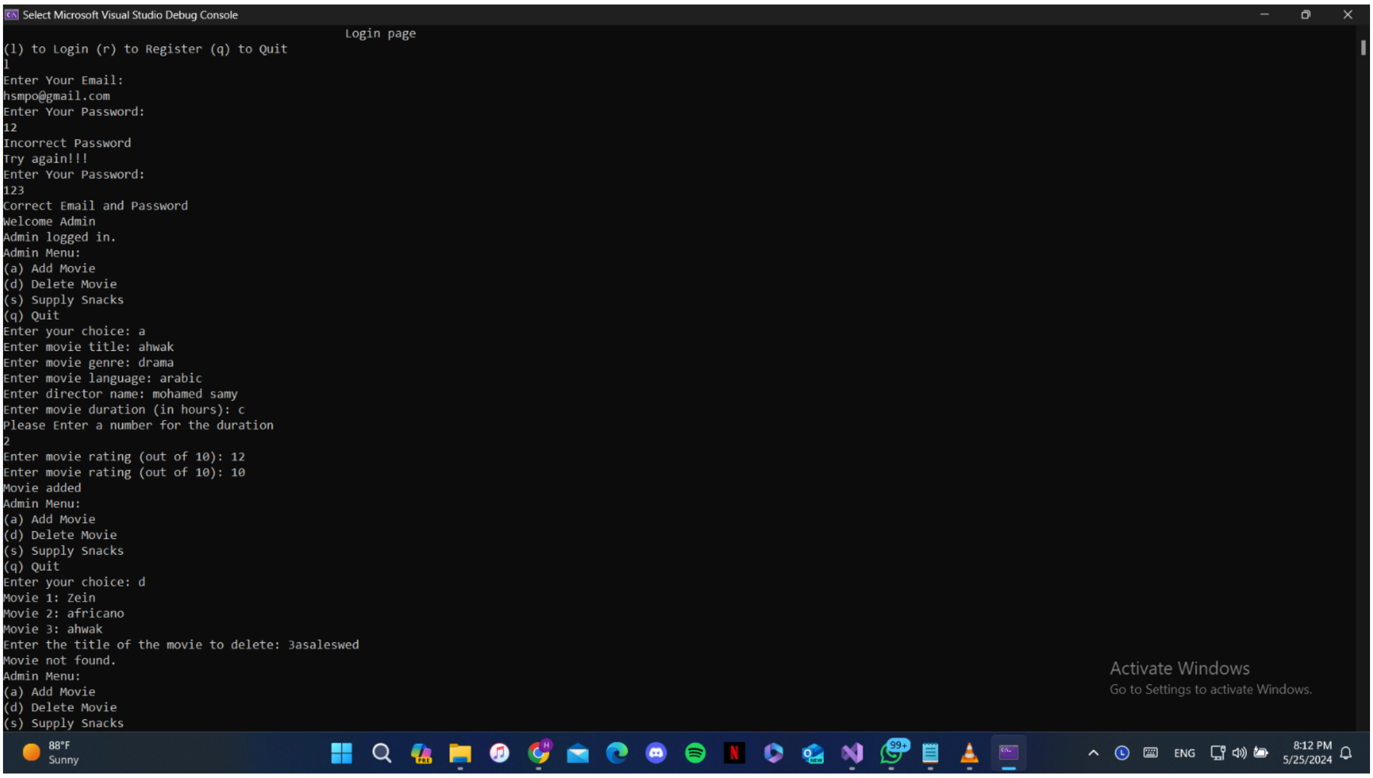
Our future goal is to create a successful GUI with plenty of features to allow the user and the admin to interact with it. And also to create a schedule for each movie.

Validation:

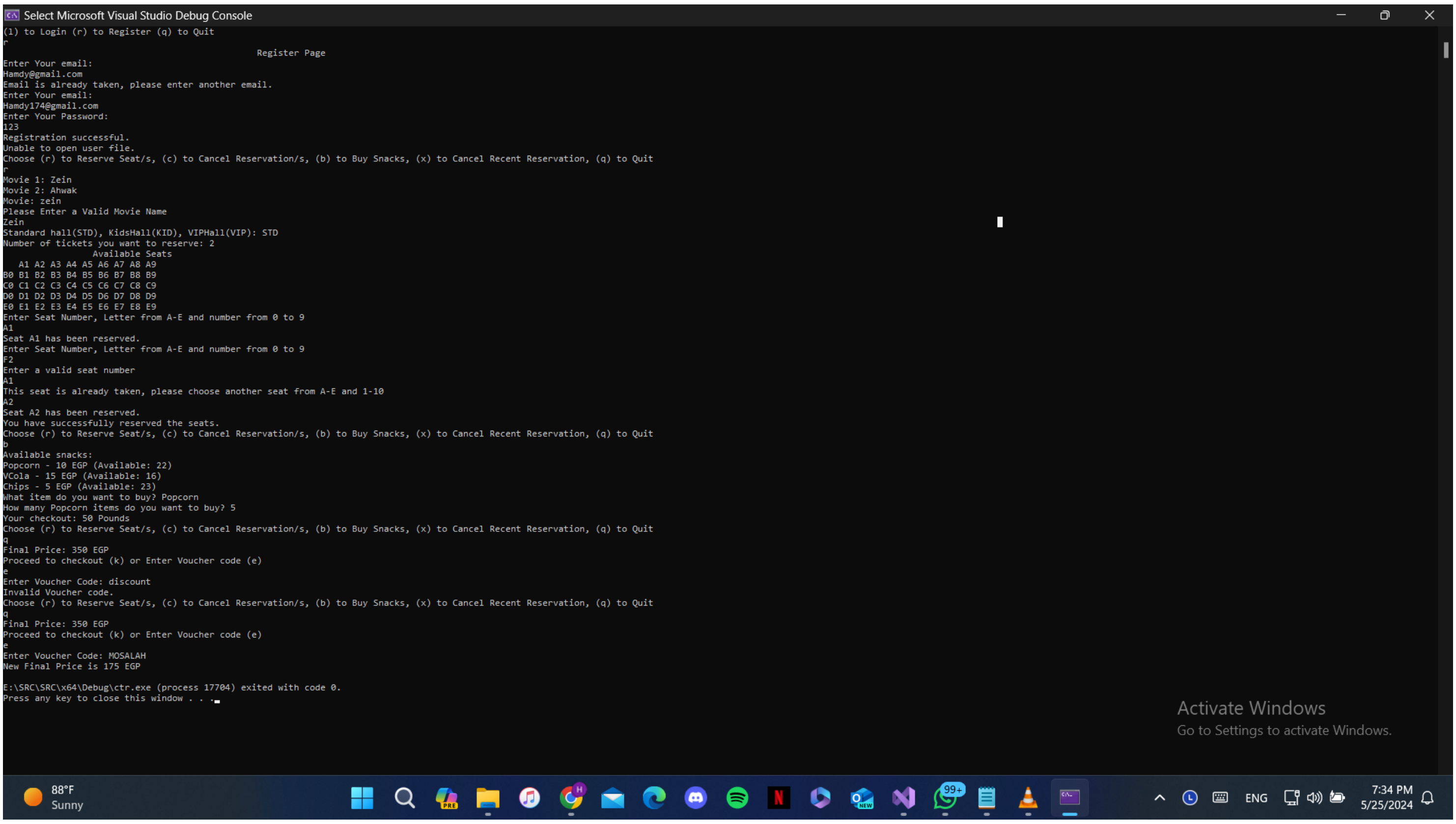

Select Microsoft Visual Studio Debug Console
Activate Windows
Go to Settings to activate Windows.

```
Enter the title of the movie to delete: 3asaleswed
Movie not found.
Admin Menu:
(a) Add Movie
(d) Delete Movie
(s) Supply Snacks
(q) Quit
Enter your choice: d
Movie 1: Zein
Movie 2: africano
Movie 3: ahwak
Enter the title of the movie to delete: africano
Movie deleted successfully.
Admin Menu:
(a) Add Movie
(d) Delete Movie
(s) Supply Snacks
(q) Quit
Enter your choice: s
Available snacks:
Popcorn - 10 EGP (Available: 17)
VCola - 15 EGP (Available: 16)
Chips - 5 EGP (Available: 29)
Which snack you want to supply: Chips
Enter the number of units to be supplied for the chosen snack: 5
Item supplied
Admin Menu:
(a) Add Movie
(d) Delete Movie
(s) Supply Snacks
(q) Quit
Enter your choice: q
Exiting Admin Menu
                                                Login page
(l) to Login (r) to Register (q) to Quit
q

E:\SRC\SRC\x64\Debug\ctr.exe (process 22308) exited with code 0.
Press any key to close this window . . .
```


Select Microsoft Visual Studio Debug Console
Activate Windows
Go to Settings to activate Windows.
88°F
Sunny
ENG
8:12 PM
5/25/2024

```
                                                Login page
(l) to Login (r) to Register (q) to Quit
l
Enter Your Email:
hsmpo@gmail.com
Enter Your Password:
12
Incorrect Password
Try again!!!
Enter Your Password:
123
Correct Email and Password
Welcome Admin
Admin logged in.
Admin Menu:
(a) Add Movie
(d) Delete Movie
(s) Supply Snacks
(q) Quit
Enter your choice: a
Enter movie title: ahwak
Enter movie genre: drama
Enter movie language: arabic
Enter director name: mohamed samy
Enter movie duration (in hours): c
Please Enter a number for the duration
2
Enter movie rating (out of 10): 12
Enter movie rating (out of 10): 10
Movie added
Admin Menu:
(a) Add Movie
(d) Delete Movie
(s) Supply Snacks
(q) Quit
Enter your choice: d
Movie 1: Zein
Movie 2: africano
Movie 3: ahwak
Enter the title of the movie to delete: 3asaleswed
Movie not found.
Admin Menu:
(a) Add Movie
(d) Delete Movie
(s) Supply Snacks
```


Select Microsoft Visual Studio Debug Console
Activate Windows
Go to Settings to activate Windows.
88°F
Sunny
ENG
7:34 PM
5/25/2024

```
(l) to Login (r) to Register (q) to Quit
r
                                                Register Page
Enter Your email:
Hamdy@gmail.com
Email is already taken, please enter another email.
Enter Your email:
Hamdy174@gmail.com
Enter Your Password:
123
Registration successful.
Unable to open user file.
Choose (r) to Reserve Seat/s, (c) to Cancel Reservation/s, (b) to Buy Snacks, (x) to Cancel Recent Reservation, (q) to Quit
r
Movie 1: Zein
Movie 2: Ahwak
Movie: zein
Please Enter a Valid Movie Name
Zein
Standard hall(STD), KidsHall(KID), VIPHall(VIP): STD
Number of tickets you want to reserve: 2
                 Available Seats
   A1 A2 A3 A4 A5 A6 A7 A8 A9
B0 B1 B2 B3 B4 B5 B6 B7 B8 B9
C0 C1 C2 C3 C4 C5 C6 C7 C8 C9
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9
E0 E1 E2 E3 E4 E5 E6 E7 E8 E9
Enter Seat Number, Letter from A-E and number from 0 to 9
A1
Seat A1 has been reserved.
Enter Seat Number, Letter from A-E and number from 0 to 9
F2
Enter a valid seat number
A1
This seat is already taken, please choose another seat from A-E and 1-10
A2
Seat A2 has been reserved.
You have successfully reserved the seats.
Choose (r) to Reserve Seat/s, (c) to Cancel Reservation/s, (b) to Buy Snacks, (x) to Cancel Recent Reservation, (q) to Quit
b
Available snacks:
Popcorn - 10 EGP (Available: 22)
VCola - 15 EGP (Available: 16)
Chips - 5 EGP (Available: 23)
What item do you want to buy? Popcorn
How many Popcorn items do you want to buy? 5
Your checkout: 50 Pounds
Choose (r) to Reserve Seat/s, (c) to Cancel Reservation/s, (b) to Buy Snacks, (x) to Cancel Recent Reservation, (q) to Quit
q
Final Price: 350 EGP
Proceed to checkout (k) or Enter Voucher code (e)
e
Enter Voucher Code: discount
Invalid Voucher code.
Choose (r) to Reserve Seat/s, (c) to Cancel Reservation/s, (b) to Buy Snacks, (x) to Cancel Recent Reservation, (q) to Quit
q
Final Price: 350 EGP
Proceed to checkout (k) or Enter Voucher code (e)
e
Enter Voucher Code: MOSALAH
New Final Price is 175 EGP

E:\SRC\SRC\x64\Debug\ctr.exe (process 17704) exited with code 0.
Press any key to close this window . . .
```

References:

YouTube channel called code beauty

Stack Overflow - Where Developers Learn, Share, & Build Careers

Starting out with c++ 9th edition