**Hotel Room Booking System**

**Overview :**

This application is designed to manage **hotel room availability and reservations**. It reads hotel and booking data from JSON files and provides two key operations:

1. **Availability Command** - Checks room availability for a specific hotel, date range, and room type.
2. **Search Command** - Finds available date ranges for a specified hotel, room type, and number of days ahead.
3. **Folder Structure**

HotelBookingSystem/

├── Data/

│ ├── hotels.json

│ ├── bookings.json

├── Models/

│ ├── Model.CS

├── Services/

│ ├── HotelService.cs

├── Utilities/

│ ├── CommandHandler.cs

│ ├──DateHelper.cs

├── Program.cs

├── README.md

**How It Works:**

**1. Availability Command**

**Example Input:**

Availability(H1, 20240901, SGL)

Availability(H1, 20240901-20240903, DBL)

**Expected Output:**

Available SGL rooms: 2

Available DBL rooms: -1 (Overbooked)

* The program calculates availability by checking the number of total rooms of a type **minus** booked rooms.
* If the number is negative, the hotel is **overbooked**.

**2. Search Command**

**Example Input:**

Search(H1, 365, SGL)

**Expected Output:**

(20241101-20241103, 2), (20241203-20241210, 1)

* The program checks room availability for the next **365 days**.
* It returns **continuous date ranges** where the room type is available.
* If no availability is found, it returns an empty line.

**Implementation Details:**

**JSON Files-**

**hotels.json**

[

{

"id": "H1",

"name": "Hotel California",

"roomTypes": [

{ "code": "SGL", "description": "Single Room", "amenities": ["WiFi", "TV"] },

{ "code": "DBL", "description": "Double Room", "amenities": ["WiFi", "TV", "Minibar"] }

],

"rooms": [

{ "roomType": "SGL", "roomId": "101" },

{ "roomType": "DBL", "roomId": "201" }

]

}

]

**bookings.json**

[

{ "hotelId": "H1", "arrival": "20240901", "departure": "20240903", "roomType": "DBL" }

]

**Key Business Logic:**

* **HotelService.cs: Contains logic for CheckAvailability and SearchAvailability.**
* **DateHelper.cs: Handles date parsing and range calculations.**

**Running the Application**

1. **Clone the repository or copy the project files.**
2. **Ensure .NET is installed.**
3. **Run the application using below way:**

**i)Click on HotelReservationSystem.exe inside /bin folder**

**ii)dotnet run --hotels Data/hotels.json --bookings Data/bookings.json**

1. **Enter commands in the console as described above.**

**Screenshots –**

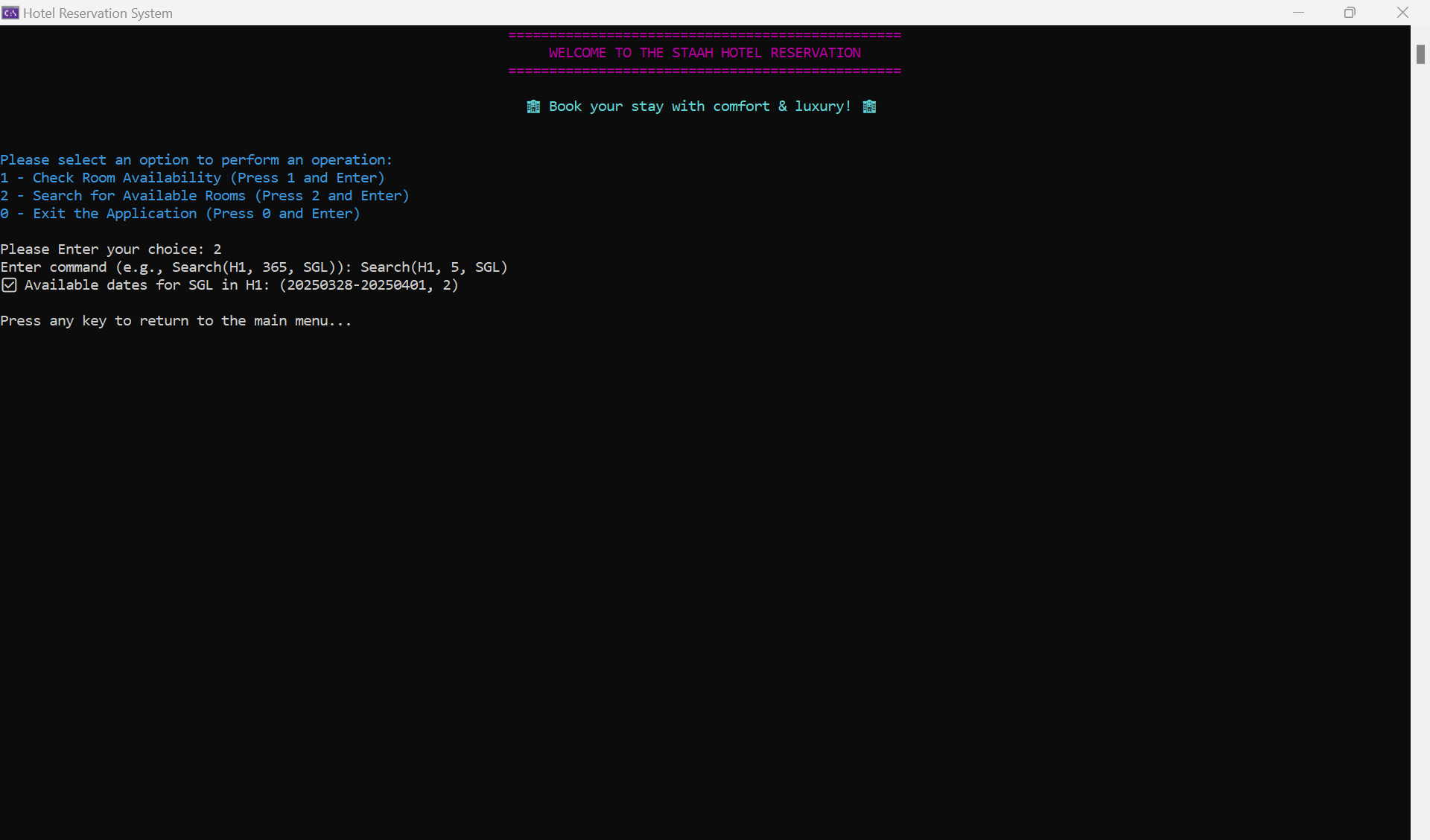
**1.Welcome Window:**

A screenshot of a computer

AI-generated content may be incorrect.

**2.Availability Command:**A screenshot of a computer

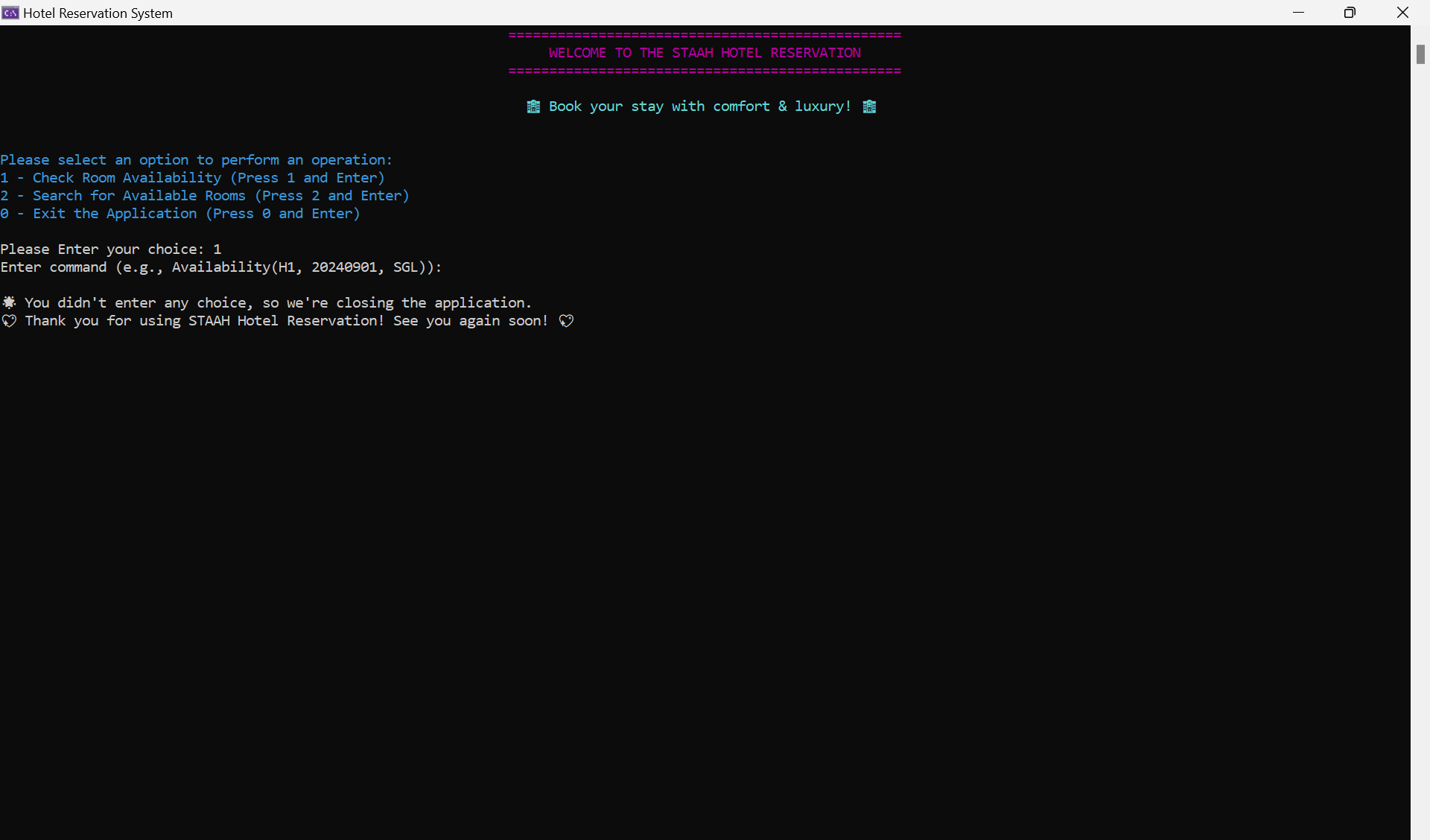
AI-generated content may be incorrect.

**3. Search Command:  
**

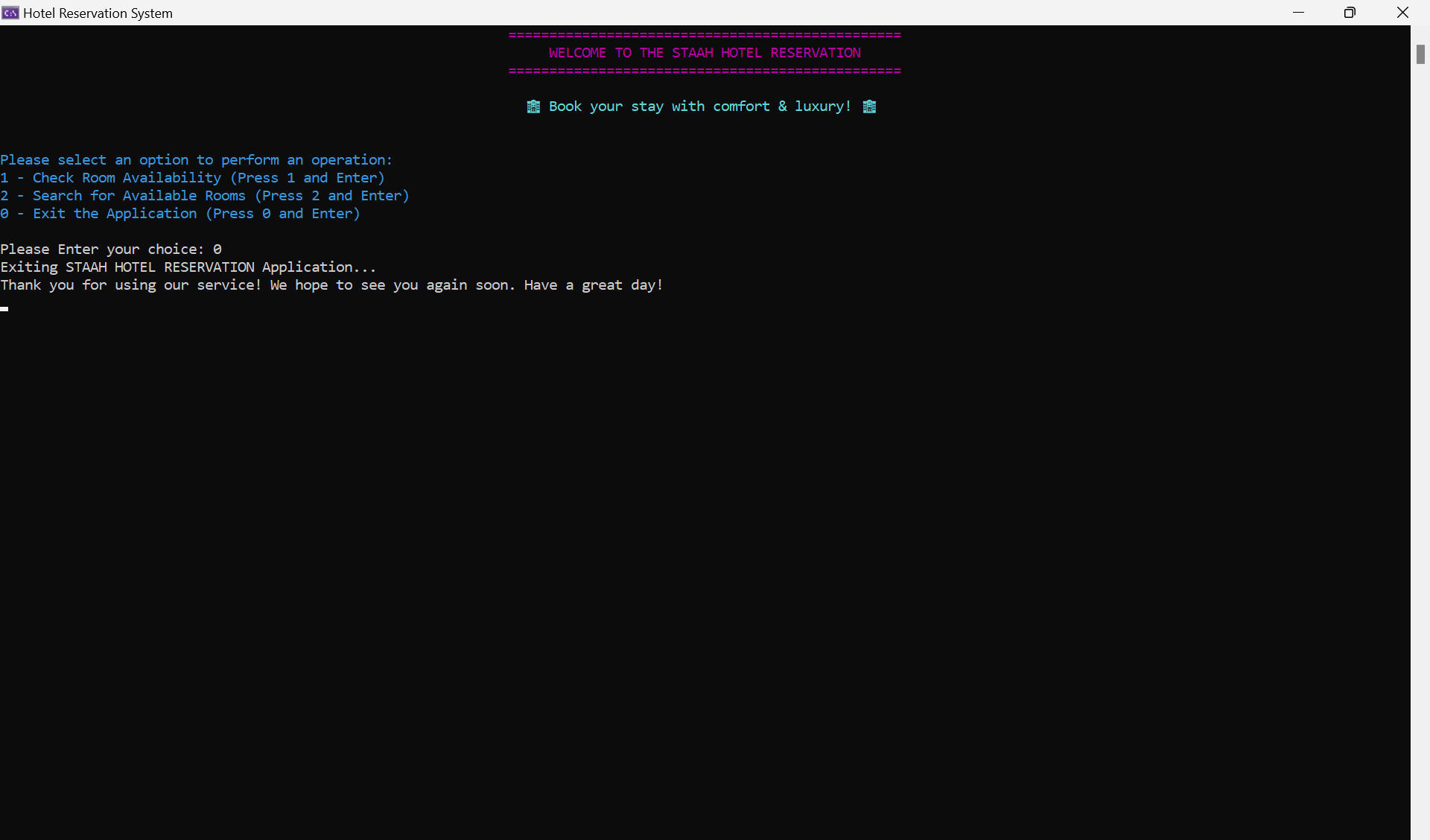
**3.Return to Main Menu-**

A screenshot of a computer

AI-generated content may be incorrect.

**4.If user enter blank line or enter without command application will be exist -  
**

**5.To Exit Application-**



**Conclusion**

This hotel booking system effectively manages room availability and reservations using structured JSON data. With its **availability and search commands**, it ensures efficient room allocation while handling overbookings.

This project demonstrates **clean code structuring**, **efficient data handling**, and **scalable design**, making it a solid foundation for real-world applications.

**Kunal R. Bhamare,**

**Thank You.**

**28-Mar-2025.**