



**AMCS1013  
Problem Solving and Programming**

**Assignment  
2024/25**

Programme : DFT  
Tutorial Group : DFT1S1G5  
Tutor Name : Lim Kah Lok  
Date Submitted to Tutor : 17-SEP-2024

**Team Members:**

No	Student Name	Student ID
1.	Lai Jia Tong	24WMD03620
2.	Chong Zhi Yi	24WMD03561
3.	Ng Jia Hao	24WMD03590
4.	Low Kai Qin	24WMD03783

## Problem Solving and Programming

---

No.	Team Member	Task(s) Completed (in detailed including mention the programming and reporting tasks)	Overall Contribution (%)
1.	Lai Jia Tong	<p><b>Programming:</b></p> <ul style="list-style-type: none"> <li>Module Location Number</li> <li>Module Number of Pax</li> <li>Module Call Room Table</li> <li>Module Return Struct</li> <li>Module Name Input</li> <li>Module Input Phone Number</li> <li>Module Display Overview</li> <li>Module Display Invoice</li> <li>Module Cancellation</li> <li>Module Check In &amp; Check Out</li> <li>Menu List</li> </ul> <p><b>Reporting:</b></p> <ul style="list-style-type: none"> <li>Structure Chart</li> <li>Handle variables, constants, and pseudocode in the module above.</li> <li>Program Testing &amp; Output</li> </ul>	32
2.	Chong Zhi Yi	<p><b>Programming:</b></p> <ul style="list-style-type: none"> <li>Module Check in and Check Out Date (all sub function)</li> <li>Module Call Room Table</li> <li>All Table Format</li> <li>Module Feedback</li> <li>Menu List</li> </ul> <p><b>Reporting:</b></p> <ul style="list-style-type: none"> <li>Brief Description / Purpose</li> <li>Handle variables, constants, and pseudocode in the module above.</li> <li>Additional Functionalities</li> <li>Program Listing</li> </ul>	28

## Problem Solving and Programming

---

3.	Ng Jia Hao	<p><b>Programming:</b></p> <p>Module Member Login Module Voucher Module Report Module Feedback Module Staff Login</p> <p><b>Reporting:</b></p> <p>Handle variables, constants, and pseudocode in the module above. Program Testing &amp; Output</p>	28
4.	Low Kai Qin	<p><b>Programming:</b></p> <p>Module Feedback Module Check In &amp; Check Out</p> <p><b>Reporting:</b></p> <p>Program Testing &amp; Output</p>	12
<b>Total</b>			100

**Coursework Originality Declaration**

We confirm that we have read and shall comply with all the terms and conditions of TAR University of Management and Technology plagiarism policy.

We declare that this assignment is free from all forms of plagiarism and for all intents and purposes is our own properly derived work.

Name	:	Lai Jia Tong	Chong Zhi Yi	Ng Jia Hao	Low Kai Qin
Photo	:				
Signature	:				

## Table of Contents

Brief Description / Purpose.....	8
Constants & variables .....	11
Constants .....	11
Variables.....	16
Overall Program Design .....	23
Structure Chart .....	23
Pseudocode.....	24
1.     Menu List.....	24
Case 1: Order Room.....	25
Case 2: Check-In & Check-Out.....	31
Case 3: Cancellation.....	31
Case 4: Report (Staff).....	32
Case 5: Feedback .....	32
Case 6: Exit.....	33
2.     Location Number .....	34
3.     Number of Pax .....	36
4.     Call Room Table .....	37
5.     Return Struct bookingList .....	38
6.     Name .....	40
7.     Phone Number.....	41
8.     Overview Booking .....	42
9.     Invoice.....	46
10.    Validation Pin .....	51
11.    Validation Integer Input.....	52
12.    Validation Phone Number .....	53
13.    Validation IC Number .....	54
14.    Check In & Check Out .....	55
15.    Cancellation .....	60
16.    Staff login .....	62
17.    Voucher .....	65
18.    Member Login .....	67
19.    Membership Structure .....	68
20.    Display summary report .....	71
21.    Report .....	72
22.    Feedback.....	75

Additional functionalities.....	77
Check-In & Check-Out .....	77
Feedback.....	79
Cancellation.....	80
Program Testing & Output.....	81
Scenario 1:.....	81
Scenario 2:.....	94
Scenario 3:.....	104
Scenario 4:.....	113
Scenario 5:.....	119
Scenario 6:.....	121
Scenario 7:.....	123
Scenario 8:.....	124
Scenario 9:.....	125
Scenario 10:.....	128
Program Listing .....	132
1. Header Files.....	132
1.1 BookingSystem.....	132
1.2 getvalid .....	132
2. Source Files.....	133
2.1 28 days Table.....	133
2.2 29 days Table.....	133
2.3 30 days Table.....	133
2.4 31 days Table.....	133
2.5 Add on service Table .....	134
2.6 CallRoomTable.....	134
2.7 CancellationBooking .....	134
2.8 Check In & Check Out .....	135
2.9 DisplaySummaryReport .....	137
2.10 Feedback.....	138
2.11 Invoice .....	140
2.12 Kuala Lumpur Table.....	143
2.13 Location Number .....	144
2.14 Logo.....	145
2.15 Melaka Table .....	145
2.16 Member Login .....	146
2.17 Menu List.....	148

2.18	Menulisttable .....	151
2.19	Month Year Table.....	152
2.20	Name.....	152
2.21	Number of Days.....	152
2.22	Number of Pax.....	153
2.23	overviewbooking .....	153
2.24	Payment Method Table.....	154
2.25	Penang Table .....	155
2.26	PhoneNumber .....	156
2.27	Pin Validation .....	156
2.28	Report .....	157
2.29	return struct BookingList.....	158
2.30	Sabah Table .....	159
2.31	Sarawak Table .....	160
2.32	Selangor Table.....	161
2.33	StaffLogin.....	162
2.34	Validate Integer Input .....	162
2.35	Validation IC .....	163
2.36	Validation PhoneNumber .....	163
2.37	Validation String Input .....	163
2.38	Voucher .....	164

## Brief Description / Purpose

Our selected Airbnb ordering system is designed to manage customer needs and service handling as effectively as possible. The main purpose is to enhance the customer experience by providing a convenient way to order services through various functions.

The system begins by presenting a comprehensive services catalogue that displays a list of menu options based on the customer's needs: room orders, cancellations, feedback, check-in/check-out (logged in by staff), and reports (logged in by staff). This catalogue allows customers to easily browse and select services.

Customers can place orders by selecting room service, specifying their location, and entering check-in and check-out dates, along with the number of guests. The system then provides detailed information about room options, including type, size, bed type, price, and available services, all organized in a clear and detailed table format.

In addition to room details, customers can request additional services such as breakfast, mid-stay cleaning, airport shuttle service, and other options. Each service can be added based on customer preferences.

The system then generates a detailed summary of the customer's order, including an overview of their booking details and a clear breakdown of all selected services and reservations. Customers can view this summary and apply any applicable voucher codes if logged in as a member.

Following this, an invoice is produced, detailing all associated costs, such as service taxes, the total amount, and any additional charges. This provides customers with a transparent view of their expenses, allowing them to review their choices and total costs before finalizing their order. Payment is processed, with customers selecting their preferred payment method to complete the transaction. Once payment is made, a booking confirmation is generated.

Furthermore, customers can proceed with cancellations if they find errors in their order or dates. Customers can also view feedback from the menu list and provide feedback after their stay through our booking system.

Additionally, our system features a staff login interface for check-in/check-out and reporting functions. When a customer checks in, staff need to enter the booking number, and the system will verify if the booking number is valid. If it is valid, staff must enter the customer's name and IC number. During check-out, staff will enter the check-out date and check for any delays. If delays are present, staff will record the number of hours late, with a delay fine of RM 20 per hour. The report function generates total sales for each location and the quantity of each item at each location.

This system aims to provide a comprehensive solution for managing customer orders and staff operations, enhancing both operational efficiency and overall customer satisfaction.

## Customer Functions:

### 1. Service Catalogue:

- **Function:** Displays a comprehensive list of available services and menu options.
- **Purpose:** To provide customers with an overview of all services and options, allowing them to easily browse, compare, and select services according to their needs.

### 2. Order Placement:

- **Function:** Allows customers to select room service, specify their location, and enter check-in and check-out dates along with the number of guests.
- **Purpose:** To facilitate the ordering process by gathering essential details for room reservations and service requests, ensuring accurate and tailored service.

### 3. Additional Services:

- **Function:** Provides options for customers to request extra services such as breakfast, mid-stay cleaning, or airport shuttle service.
- **Purpose:** To offer customers the ability to customize their stay with additional amenities, enhancing their overall experience.

### 4. Order Summary:

- **Function:** Generates a detailed summary of the customer's order, including booking details and selected services. Allows customers to view and apply voucher codes if logged in as a member.
- **Purpose:** To ensure customers can review their selections, confirm the accuracy of their order, and apply any applicable discounts before finalizing their booking.

### 5. Invoice Generation:

- **Function:** Produces an invoice that details all costs, including service taxes, total amount, and additional charges.
- **Purpose:** To provide a transparent view of all expenses, allowing customers to understand their financial commitment and review costs before payment.

### 6. Payment Processing:

- **Function:** Handles the payment transaction and allows customers to select their preferred payment method.
- **Purpose:** To securely complete the financial transaction and generate a booking confirmation, finalizing the reservation process.

### 7. Cancellation:

- **Function:** Allows customers to cancel their bookings if they identify errors or need to change their plans.
- **Purpose:** To offer flexibility for customers to correct mistakes or modify their reservations, ensuring their satisfaction and accommodating changes.

### 8. Feedback:

- **Function:** Enables customers to view feedback from previous users and submit their own feedback after their stay.
- **Purpose:** To gather insights into customer experiences, improve service quality, and build trust through transparency and user-generated reviews.

## Staff Functions:

### (a) Check-In Management:

- **Function:** Staff enter the booking number to verify its validity and input the customer's name and IC number upon confirmation.
- **Purpose:** To ensure secure and accurate check-in processing, verifying bookings and recording customer details for a smooth check-in experience.

### (b) Check-Out Management:

- **Function:** Staff enter the check-out date, assess any delays, and record the number of hours late, calculating a fine of RM 20 per hour if applicable.
- **Purpose:** To manage the check-out process efficiently, handle any delays, and enforce hotel policies on late check-outs.

### (c) Reporting:

- **Function:** Generates reports on total sales and quantities of services for each location.
- **Purpose:** To monitor and analyze performance, manage resources, and make informed business decisions based on comprehensive sales data.

## System-Wide Functions:

### 1. User Authentication:

- **Function:** Manages access to system functions based on user roles (customers and staff).
- **Purpose:** To ensure secure access and personalized experiences by allowing users to access only the functions relevant to their role.

### 2. Error Handling and Validation:

- **Function:** Validates data inputs and manages errors in booking, payment, and other processes.
- **Purpose:** To maintain the integrity and reliability of the system by preventing errors and ensuring accurate data handling.

## Constants & variables

### Constants

Identifier	Value	Data Type	Purpose
<b>roomsKualaLumpur</b>			
roomNo	1,2,3,4,5	integer	Unique identifier for the room number.
hotelName	"Cubic Botanical", "Mont Kiara", "Urban Retreat", "Luxe Residence", "Panorama Hotel"	string	Name of the hotel where the room is located.
price	650.00, 200.00, 450.00, 450.00, 450.00	double	Price of the room per night.
service	"Wifi & Breakfast", "Wifi & Breakfast", "Wifi, Breakfast & Gym", "Wifi, Breakfast & Lounge", "Wifi, Breakfast & Pool"	string	Services included with the room.
<b>roomsMelaka</b>			
roomNo	1,2,3,4,5	integer	Unique identifier for the room number.
hotelName	"Heritage Haven", "Melaka River Inn", "Coastal Retreat", "Grand Melaka Resort ", "Historical Lodge"	string	Name of the hotel where the room is located.
price	300.00, 250.00, 350.00, 500.00, 200.00	double	Price of the room per night.
service	"Wifi & Breakfast", "Wifi, Breakfast & Shuttle Service", "Wifi, Breakfast, Gym & Spa", "Wifi, Breakfast, Lounge & Business Center", "Wifi, Breakfast & Laundry Service"	string	Services included with the room.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Value</b>	<b>Data Type</b>	<b>Purpose</b>
<b>roomsPenang</b>			
roomNo	1,2,3,4,5	integer	Unique identifier for the room number.
hotelName	"Bayview Grand Hotel", "Georgetown Inn", "Seaside Resort", "Penang Heritage Lodge", "Bayview Penang"	string	Name of the hotel where the room is located.
price	300.00, 250.00, 400.00, 300.00, 500.00	double	Price of the room per night.
service	"Wifi & Breakfast", "Wifi, Breakfast & Shuttle Service", "Wifi, Breakfast, Gym & Spa", "Wifi, Breakfast & Lounge", "Wifi, Breakfast & Business Center"	string	Services included with the room.
<b>roomsSelangor</b>			
roomNo	1,2,3,4,5	integer	Unique identifier for the room number.
hotelName	"Metro Plaza Hotel", "City View Resort", "Elite Comfort Hotel", "Urban Haven Hotel", "Selangor Grand"	string	Name of the hotel where the room is located.
price	420.00, 550.00, 480.00, 390.00, 350.00	double	Price of the room per night.
service	"Wifi, Gym, Spa & Lounge", "Wifi, Dinner, Free Parking & Airport Shuttle", "Wifi, Breakfast, Conference Room & Laundry Service", "Wifi, Breakfast, Afternoon Tea & Business Center", "Wifi, Breakfast, 24 Hours Front Desk & Lounge"	string	Services included with the room.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Value</b>	<b>Data Type</b>	<b>Purpose</b>
<b>roomsSabah</b>			
roomNo	1,2,3,4,5	integer	Unique identifier for the room number.
hotelName	"Borneo Bliss Hotel", "Kinabalu View", "Sunset Coastal Resort", "Sabah Serenity Lodge", "Horizon Hotel"	string	Name of the hotel where the room is located.
price	320.00, 270.00, 380.00, 290.00, 450.00	double	Price of the room per night.
service	"Wifi, Breakfast, Pool & Parking", "Wifi, Breakfast, Shuttle Service & Airport Transfer", "Wifi, Breakfast, Restaurant & Spa", "Wifi, Breakfast, Concierge", "Wifi, Breakfast, Business Center & Conference Rooms"	string	Services included with the room.
<b>roomsSarawak</b>			
roomNo	1,2,3,4,5	integer	Unique identifier for the room number.
hotelName	"Riverfront Resort", "Heritage City Hotel", "Borneo Vista Lodge", "Tropical Haven Hotel", "Kuching Grand Inn"	string	Name of the hotel where the room is located.
price	350.00, 580.00, 400.00, 460.00, 310.00	double	Price of the room per night.
service	"Wifi, Dinner, Spa & Free Parking", "Wifi, Breakfast, Pool & Laundry", "Wifi, Breakfast, Gym & Spa", "Wifi, Breakfast, Afternoon Tea", "Wifi, Breakfast, Business Center & Lounge"	string	Services included with the room.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Value</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Invoice / Overview Booking</b>			
numRoomsKualaLumpur	5	integer	Number of rooms available in Kuala Lumpur.
numRoomsMelaka	5	integer	Number of rooms available in Melaka.
numRoomsPenang	5	integer	Number of rooms available in Penang.
numRoomsSelangor	5	integer	Number of rooms available in Selangor.
numRoomsSabah	5	integer	Number of rooms available in Sabah.
numRoomsSarawak	5	integer	Number of rooms available in Sarawak.
MAX_BOOKINGS	100	integer	Limits the maximum number of bookings that can be stored in the BookingList
<b>Invoice</b>			
SSTRATE	0.06	double	SST (Sales and Services Tax) rate
<b>Member Login</b>			
NUM_MEMBERS	3	integer	Defines the total number of members allowed in the system.
correctMemberIDs	"member1", "member2", "member3"	string	Stores the valid member IDs.
correctPasswords	"password1", "password2", "password3"	string	Stores the valid passwords for corresponding member IDs.
<b>Report</b>			
kualaLumpurTotalUnits	78	integer	Initial total units for Kuala Lumpur
kualaLumpurTotalAmount	41000	double	Initial total amount for Kuala Lumpur
melakaTotalUnits	65	integer	Initial total units for Melaka
melakaTotalAmount	20400	double	Initial total amount for Melaka
penangTotalUnits	99	integer	Initial total units for Penang
penangTotalAmount	50880	double	Initial total amount for Penang
selangorTotalUnits	56	integer	Initial total units for Selangor
selangorTotalAmount	20580	double	Initial total amount for Selangor
sabahTotalUnits	101	integer	Initial total units for Sabah

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Value</b>	<b>Data Type</b>	<b>Purpose</b>
sabahTotalAmount	33840	double	Initial total amount for Sabah
sarawakTotalUnits	77	integer	Initial total units for Sarawak
sarawakTotalAmount	35600	double	Initial total amount for Sarawak
<b>Staff Login</b>			
NUM_STAFFS	3	integer	Defines the number of staff members allowed to login.
correctStaffIDs	"staff1" "staff2" "staff3"	string	Stores the valid staff IDs for login.
correctPasswords	"password1", "password2", "password3"	string	Stores the valid passwords corresponding to the staff IDs.
<b>Voucher</b>			
vouchers	"CNY8888", "HR4444", "DP6666", "MC1225", "V2141314", "WELCOME123"	string	Stores the valid voucher codes.
discounts	0.08, 0.10, 0.12, 0.15, 0.18, 0.20	double	Stores the discount rates corresponding to each voucher code.
<b>Check In and Check Out</b>			
FINESPERHOURS	20	double	The fine amount charged per hour of delay in checkout.
<b>Feedback</b>			
MAX_FEEDBACK	5	integer	This constant defines the maximum number of customer feedback entries that the program can store and display.

**Variables**

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Location Number</b>		
locationNo	integer	This number allows customers to select their desired hotel area. We will use this number to display the corresponding area table with available hotels.
<b>Number of Pax</b>		
adult	Integer	Customer can input the number of adults (13 years and older).
children	integer	Customer can input the number of children (2-12 years old).
infants	Integer	Customer can key in the number of pax for infants
totalPax	integer	To calculate the total number of people, including adults, children, and infants.
<b>Return Struct Booking List</b>		
nextBookingNumber	integer	Keeps track of the next booking number to be assigned. It starts at 1001 and increments with each new booking.
roomNo	integer	This represents the room number selected by the customer. It allows the system to reference the specific room being booked. This value can be used to look up details about the room in a BookingRoom structure
quantity	integer	This represents the number of rooms booked in a single booking. It is used to calculate the total price to be paid for the booking, as the price is typically determined by multiplying the quantity by the cost per room.
<b>Name</b>		
name	string	Used to collect and display the customer's details in both the booking overview and the invoice.
<b>Phone Number</b>		
phoneNumber	String	Used to collect and validate the phone number, ensuring that only valid phone numbers consisting solely of digits are accepted.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Overview Booking &amp; Invoice</b>		
Rooms	const BookingRoom*	Pointer to the array of BookingRoom structures for the selected location, used for room lookup.
totalRooms	integer	Stores the total number of rooms available in the selected location.
totalPrice	double	Accumulates the total price for all rooms booked in the latest booking.
latestBookingIndex	integer	Index of the latest booking in the BookingList to retrieve booking details.
roomNo	integer	Room number for the latest booking, used to find the room details in the selected location's room array.
quantity	integer	Quantity of rooms booked in the latest booking, used to calculate the total price
bookingNo	integer	Booking number for the latest booking, used to display booking details in the overview.
roomPrice	double	Price per room for the specific room number, used to calculate the total price for the booking.
bookingTotalPrice	double	Total price for the specific booking based on the room price and quantity.
totalPriceAlldays	double	Total price for all days of the booking period, calculated by multiplying totalPrice by numberOfDays.
service	character	Indicates if add-on service is needed (Y/N)
serviceNo	integer	Select add-on service number (1-5)
servicePrice	double	Price of the selected add-on service
serviceName	string	Name of the selected add-on service
gotvoucher	character	Indicates if a voucher code is applied (Y/N)
SST	double	Calculated SST value
grandtotal	double	Final grand total amount after discounts, taxes, and services
paymentMethod	integer	Selected payment method (1-5)
accountNo	integer	Account number for bank transfer
cardNumber	integer	Credit/Debit card number
PIN	integer	Personal Identification Number for payment validation
cashAmount	double	Amount of cash paid in case of cash payment
change	double	Change to be returned in case of cash payment

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Cancellation</b>		
bookingNumberToCancel	integer	Stores the booking number that the user wants to cancel.
bookingFound	Boolean	Indicates whether the booking was found and successfully cancelled.
<b>BookingRoom Struct</b>		
roomNo	integer	The number assigned to the room.
hotelName	string	The name of the hotel.
price	double	The price of the room.
service	string	The service provided with the room.
<b>Booking Struct</b>		
bookingNumber	integer	Unique identifier for the booking.
roomNo	integer	The room number that was booked.
quantity	integer	The number of rooms booked.
<b>BookingList Struct</b>		
bookings	Booking[]	An array of Booking structs.
count	integer	The current number of bookings in the list.
<b>RebateValue</b>		
rebate	double	The rebate amount.
selectedvoucher	string	The voucher selected for the rebate.
<b>MenuList</b>		
option	integer	To let user input the menu option.
continueOption	character	This is to control whether the loop continue based on user input Y or N.
locationNo	integer	Show the location number for room booking.
grandtotal	double	Show the total price for all bookings.
loggedin	boolean	Indicates whether the staff login was successful or not.
startMonth	integer	By display a month & year table, this is to let user input check-in month.
startYear	integer	By display a month & year table, this is to let user input check-in year.
CheckInDay	integer	By display a total day in month table depends on user selected month, this is to let user input check-in date.
endMonth	integer	By display a month & year table, this is to let user input check-out month.
endYear	integer	By display a month & year table, this is let user input check-out year.
CheckOutDay	integer	By display a total day in month table depends on user selected month, this is to let user input check-out date.
numberOfDays	integer	To calculate the number of days between check-in date and check-out date as the booking days.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Number of days</b>		
year	integer	Input parameter representing the year to determine whether the year is a leap year. To determine if February has 28 or 29 days based on whether it is a leap year.
month	integer	Represents the current month which to determine the number of days of month.
day	integer	Represent the specific day of the given month which to calculate total number of days start from 1 Jan.
<b>Member Login</b>		
correctMemberIDs	string	Array to store the valid member IDs.
correctPasswords	string	Array to store the corresponding passwords for each valid member ID.
name	string	Stores the name of the new member during registration.
phoneNumber	string	Stores the phone number of the new member during registration.
email	string	Stores the email address of the new member during registration.
inputMemberID	string	Temporarily holds the member ID entered during login.
inputPassword	string	Temporarily holds the password entered during login.
isAuthenticated	boolean	Indicates if the member login was successful.
idExists	boolean	Indicates whether the entered member ID exists in the system.
choice_membership	integer	Stores the user's choice of whether they are a member or a new member during the login process.
valid_input	boolean	Tracks if the user's input during login or registration is valid.
result	RebateValue	Stores the result from the Voucher function.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Report</b>		
totalunit	integer	Stores the total number of units from bookings in a location.
grandtotal	double	Stores the total amount of bookings for a specific location.
locationNo	integer	Stores the location number selected by the user.
BL.count	integer	Stores the total count of bookings in the BookingList object.
BL.bookings[].quantity	integer	Stores the quantity of bookings in a specific location.
staffChoice	integer	Stores the choice selected by the staff in the report menu.
kualaLumpurTotalUnits	integer	Tracks total units for Kuala Lumpur during the program.
kualaLumpurTotalAmount	double	Tracks total amount for Kuala Lumpur during the program.
melakaTotalUnits	integer	Tracks total units for Melaka during the program.
melakaTotalAmount	double	Tracks total amount for Melaka during the program.
penangTotalUnits	integer	Tracks total units for Penang during the program.
penangTotalAmount	double	Tracks total amount for Penang during the program.
selangorTotalUnits	integer	Tracks total units for Selangor during the program.
selangorTotalAmount	double	Tracks total amount for Selangor during the program.
sabahTotalUnits	integer	Tracks total units for Sabah during the program.
sabahTotalAmount	double	Tracks total amount for Sabah during the program.
sarawakTotalUnits	integer	Tracks total units for Sarawak during the program.
sarawakTotalAmount	double	Tracks total amount for Sarawak during the program.
<b>Staff Login</b>		
inputStaffID	string	Stores the staff ID entered by the user.
inputPassword	string	Stores the password entered by the user.
staffchoice	character	Stores the user's choice for continuing or exiting the staff interface.
isAuthenticated	boolean	Tracks whether the user is successfully authenticated.
idExists	boolean	Tracks whether the entered staff ID exists in the valid staff IDs list.

## Problem Solving and Programming

---

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Voucher</b>		
vouchername	string	Stores the voucher code entered by the user.
finalprice	double	Holds the final price after applying the discount.
rebate	double	Calculates and stores the rebate amount based on the discount.
voucherFound	boolean	Tracks whether a valid voucher code was entered.
x	RebateValue	Holds the result with the rebate amount and voucher code for return.
<b>Check In and Check Out</b>		
selection	integer	Show integer value of the user's menu selection for check-in, check-out, or exit.
bookingNumberToCheckIn	integer	Show integer value of the booking number entered by the user for check-in.
name	string	Enter the name of the customer for check-in.
ic_num	string	Enter the IC number of the customer for check-in.
bookingFoundCheckIN	boolean	Flag to indicate if the booking number for check-in was found.
bookingNumberToCheckOut	integer	Stores the booking number entered by the user for check-out.
delayed_time	integer	Stores the number of hours the customer is delayed for check-out.
fines	double	Stores the calculated fines based on the delayed time.
paymentMethod	integer	Stores the selected payment method for the fines.
accountNo	integer	Stores the account number for bank transfer payment method.
cardNumber	integer	Stores the card number for credit/debit card payment method.
PIN	integer	Stores the PIN for various payment methods.
paymentMethodName	string	Stores the name of the payment method selected by the user.
cashAmount	double	Stores the amount of cash provided by the customer, if applicable.
change	double	Stores the change to be given back to the customer if the cash amount is more than the fines.

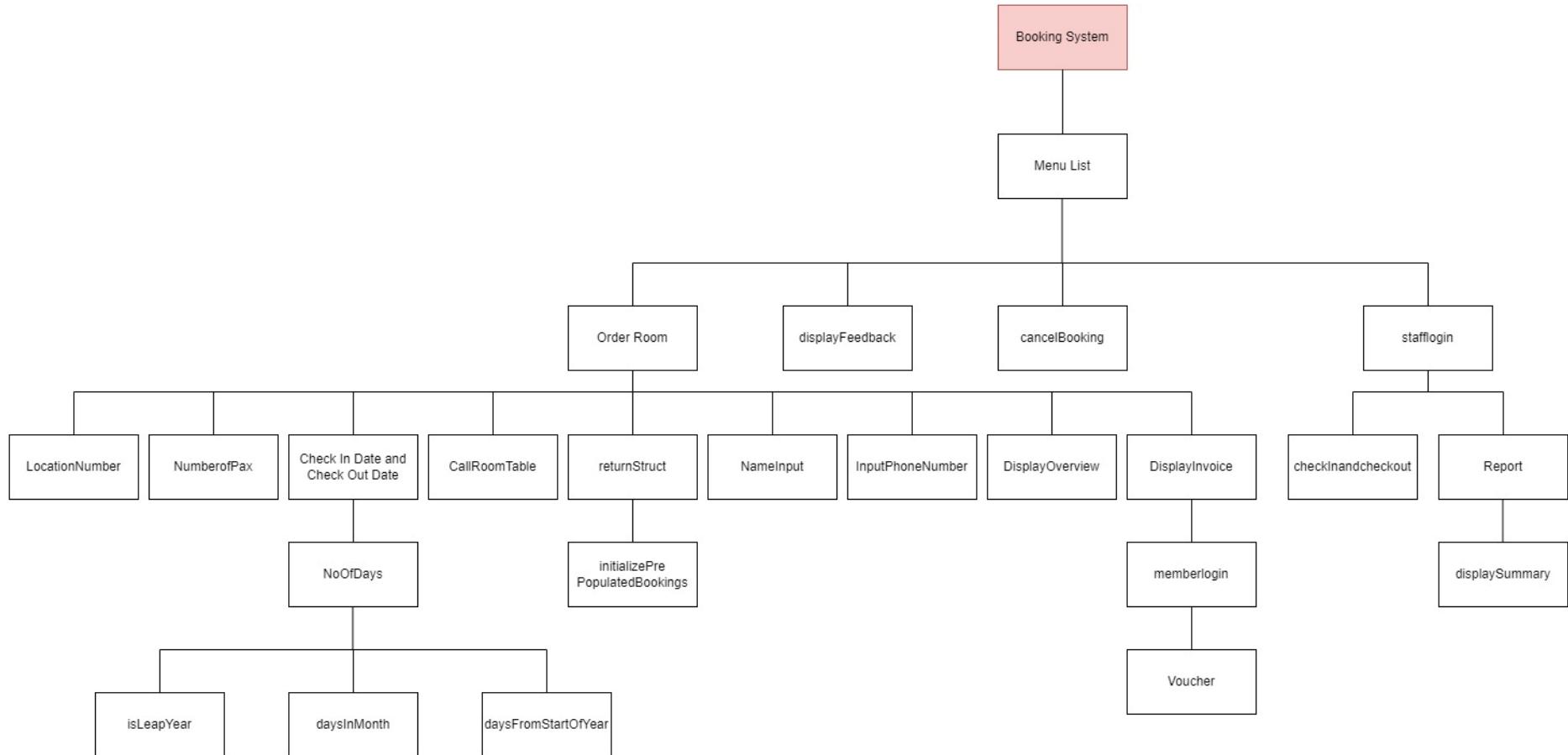
## Problem Solving and Programming

---

<b>Identifier</b>	<b>Data Type</b>	<b>Purpose</b>
<b>Feedback</b>		
ratings	integer	An array that stores the star ratings provided by each customer. The size of the array is defined by the constant MAX_FEEDBACK.
names	string	An array that stores the names of the customers providing feedback. Each name corresponds to the associated rating and comment.
comments	string	An array that stores the feedback comments provided by the customers. Each comment corresponds to the associated rating and name.
feedbackCount	integer	This variable tracks the number of feedback entries currently stored. It ensures feedback entries do not exceed MAX_FEEDBACK.
decision	character	Stores the customer's decision on whether or not to leave feedback. Accepts 'y' for yes and 'n' for no.
rating	integer	Temporarily stores the customer's rating (between 1 and 5 stars) before it is added to the ratings array.

# Overall Program Design

## Structure Chart



## Pseudocode

### 1. Menu List

1. **Initialization:**
  - a) **Option:** Handle the user's menu choice.
  - b) **continueOption:** Determines whether the user wants to continue with the process or exit.
  - c) **locationNo:** Stores the location number for booking.
  - d) **grandtotal:** Accumulates the total amount from bookings.
  - e) **loggedIn:** Indicates whether a staff member is logged in.
2. **Pre-populated Bookings:** Call a function named initializePrePopulatedBookings() that creates and returns a list of bookings that have been pre-loaded for the system. The list of bookings is then assigned to myBookings, which is used for further operations in the current session.
3. **Option input:** After display logo, welcome message and menu options. Read user input to determine which menu option was selected.
4. **Validate option input:** To make sure that user's option is between 1 to 6. If not, will prompt the user to enter a valid option.
5. **Process Menu Option:** Use switch statement to determine user selection of menu option.

```

START
FUNCTION main()
    INITIALIZE option TO 0
    INITIALIZE continueOption TO 'Y' // Initialize to 'Y' to enter the loop !
    INITIALIZE locationNo TO 0
    INITIALIZE grandtotal TO 0
    INITIALIZE loggedIn TO false

    // Initialize pre-populated bookings
    prePopulatedBookings = initializePrePopulatedBookings()
    myBookings = prePopulatedBookings // Use the initialized bookings for this session

    logo()

    REPEAT
        PRINT "Welcome to Elegant Hideaway, select an option to proceed"
        MenuListtable() // Display menu P
        RINT "Your option: "
        READ option

        // Validate option input
        WHILE option IS NOT BETWEEN 1 AND 6
            CLEAR error
            IGNORE unwanted characters
            PRINT "Invalid option. Please enter a number between 1 - 6"
            MenuListtable()
            PRINT "Your option: "
            READ option
        END WHILE
    
```

### Case 1: Order Room

- a) **Print selection:** Inform the user that the order room option has been selected.
- b) **Call Function and Initialization:** Call the LocationNumber function to get the location number, store the location number in locationNo. Call function number of pax to get the number of people booking the room. Initialize variables to store check-in and check-out date and number of days.
- c) **Date selection process:** Start with a loop that will continue until the user provides valid dates, prompt the user to select their check-in and check-out dates.
  - **Input validation for Month:** Ensure the start month and end month entered by the user is valid which is between 1 to 12. If the user enters an invalid month, display an error message and prompt user to enter a valid month.
  - **Input validation for Year:** Ensure the start year and end year entered by the user is either 2024 or 2025. If the year is not one of these two values, display an error message and prompt the user to enter a valid year.
  - **Check-In & Check-Out Date Selection and Validation:** Prompt the user to select valid check-in date and check-out date, validate it based on the month and year.
    - i. **For February 2024:** Call days29() which handles February with 29 days and validate the check-in day is between 1 and 29.
    - ii. **For February 2025:** Call days28() which handles February with 28 days and validate the check-in day is between 1 and 28.
    - iii. **Months with 31 days:** Call days31() which handles months with 31 days and validate that the check-in day is between 1 and 31.
    - iv. **Months with 30 days:** Call days30() which handles months with 30 days and validate that the check-in day is between 1 and 30.
  - **Calculate number of days:** Calculate the number of days between the check-in and check-out dates based on number of days function.
  - **Validate number of days:** Then, display the number of days as booking days. Make sure number of days is valid which should be greater than 0. If valid, confirm the dates. If not, prompt the user to re-enter the dates.
- d) **Call Function:** The process of booking rooms
  - **Room Table Function:** Call with a parameter locationNo, which represents a location identifier. To display information about available rooms at the specified location.
  - **Customer Booking Process:** Initializes updated booking information that is assigned to myBookings. This prepares the current state of bookings for further processing.
  - **Input Name and Phone Number:** Call those functions to collect user information. The input name and phone number will store in variables name and phoneNumber, used for use in booking details.
  - **OverBooking:** This function calculates and displays an overview of the booking. It takes multiple parameters related to the booking and calculates the total price for the stay. The total price will be stored for further use.
  - **Invoice:** This function will generate and update the invoice for the booking.
- e) **Prompt for next order:** Prompt the user to decide whether they want to continue with another booking or done their current order.
- f) **Validate User Input:** Validates the continueOption to make sure it is either 'Y' for yes or 'N' for no. If the input is invalid, it will prompt again until a valid input.

```
SWITCH option
CASE 1:
    PRINT "You selected Order Room"

    // Call Function LocationNumber
    locationNo = LocationNumber()

    REPEAT
        // Call Function Number of Pax
        totalPax = NumberofPax()

        // Selection Date Month Year
        INITIALIZE startMonth, endMonth, startYear, endYear, checkInDay,
        checkOutDay, numberOfDays TO 0

        REPEAT
            PRINT "Select your Check-In & Check-Out Date"
            monthYear()

            PRINT "Start Month: "
            READ startMonth

            WHILE startMonth IS NOT BETWEEN 1 AND 12
                CLEAR error
                IGNORE unwanted characters
                PRINT "Invalid month. Please enter a number between 1 - 12"
                PRINT "Start Month: "
                READ startMonth
            END WHILE

            PRINT "End Month: "
            READ endMonth

            WHILE endMonth IS NOT BETWEEN 1 AND 12
                CLEAR error
                IGNORE unwanted characters
                PRINT "Invalid month. Please enter a number between 1 - 12"
                PRINT "End Month: "
                READ endMonth
            END WHILE

            PRINT "Start Year: "
            READ startYear

            WHILE startYear IS NOT 2024 OR 2025
                CLEAR error
                IGNORE unwanted characters
                PRINT "Not available year. Please enter 2024 or 2025"
                PRINT "Start Year: "
                READ startYear
            END WHILE
```

## Problem Solving and Programming

```
PRINT "End Year: "
READ endYear

WHILE endYear IS NOT 2024 OR 2025
    CLEAR error
    IGNORE unwanted characters
    PRINT "Not available year. Please enter 2024 or 2025"
    PRINT "End Year: "
    READ endYear
END WHILE

PRINT "Select your Check-In Date, __/startMonth/startYear"

IF startYear IS 2024 AND startMonth IS 2 THEN
    days29()

    PRINT "Check-In Date: "
    READ checkInDay

    WHILE checkInDay IS NOT BETWEEN 1 AND 29
        CLEAR error
        IGNORE unwanted characters
        PRINT "Not available date. Please enter a number between 1 - 29"
        PRINT "Check-In Date: "
        READ checkInDay
    END WHILE

    PRINT checkInDay, "/", startMonth, "/", startYear, ", Check-In Date selected
successfully."

ELSE IF startYear IS 2025 AND startMonth IS 2 THEN
    days28()

    PRINT "Check-In Date: "
    READ checkInDay

    WHILE checkInDay IS NOT BETWEEN 1 AND 28
        CLEAR error
        IGNORE unwanted characters
        PRINT "Not available date. Please enter a number between 1 - 28"
        PRINT "Check-In Date: "
        READ checkInDay
    END WHILE

    PRINT checkInDay, "/", startMonth, "/", startYear, ", Check-In Date selected
successfully."

ELSE IF startMonth IS ONE OF {1, 3, 5, 7, 8, 10, 12} THEN
    days31()

    PRINT "Check-In Date: "
    READ checkInDay
    WHILE checkInDay IS NOT BETWEEN 1 AND 31
```

```
CLEAR error
IGNORE unwanted characters
PRINT "Not available date. Please enter a number between 1 - 31"
PRINT "Check-In Date: "
READ checkInDay
END WHILE

PRINT checkInDay, "/", startMonth, "/", startYear, ", Check-In Date
selected successfully.

ELSE
days30()

PRINT "Check-In Date: "
READ checkInDay

WHILE checkInDay IS NOT BETWEEN 1 AND 30
CLEAR error
IGNORE unwanted characters
PRINT "Not available date. Please enter a number between 1 - 30"
PRINT "Check-In Date: "
READ checkInDay
END WHILE

PRINT checkInDay, "/", startMonth, "/", startYear, ", Check-In Date
selected successfully.

END IF

PRINT "Select your Check-Out Date, __/endMonth/endYear"

IF endYear IS 2024 AND endMonth IS 2 THEN
days29()

PRINT "Check-Out Date: "
READ checkOutDay

WHILE checkOutDay IS NOT BETWEEN 1 AND 29
CLEAR error
IGNORE unwanted characters
PRINT "Not available date. Please enter a number between 1 - 29"
PRINT "Check-Out Date: "
READ checkOutDay
END WHILE

PRINT checkOutDay, "/", endMonth, "/", endYear, ", Check-Out Date
selected successfully.

ELSE IF endYear IS 2025 AND endMonth IS 2 THEN
days28()

PRINT "Check-Out Date: "
READ checkOutDay
WHILE checkOutDay IS NOT BETWEEN 1 AND 28
```

```

CLEAR error
IGNORE unwanted characters
PRINT "Not available date. Please enter a number between 1 - 28"
PRINT "Check-Out Date: "
READ checkOutDay
END WHILE

PRINT checkOutDay, "/", endMonth, "/", endYear, ", Check-Out Date
selected successfully."
ELSE IF endMonth IS ONE OF {1, 3, 5, 7, 8, 10, 12} THEN
    days31()

PRINT "Check-Out Date: "
READ checkOutDay

WHILE checkOutDay IS NOT BETWEEN 1 AND 31
    CLEAR error
    IGNORE unwanted characters
    PRINT "Not available date. Please enter a number between 1 - 31"
    PRINT "Check-Out Date: "
    READ checkOutDay

WHILE checkOutDay IS NOT BETWEEN 1 AND 30
    CLEAR error
    IGNORE unwanted characters
    PRINT "Not available date. Please enter a number between 1 - 30"
    PRINT "Check-Out Date: "
    READ checkOutDay
END WHILE

PRINT checkOutDay, "/", endMonth, "/", endYear, ", Check-Out Date
selected successfully."
END IF

numberOfDays = NoOfDays(startMonth, endMonth, startYear, endYear,
checkInDay, checkOutDay)
PRINT "Number of your booking days is ", numberOfDays, "."

IF numberOfDays > 0 THEN
    PRINT "Check-In Date ", checkInDay, "/", startMonth, "/", startYear, ",
    Check-Out Date ", checkOutDay, "/", endMonth, "/", endYear, " selected
    successfully."
ELSE
    PRINT "Invalid date range. Please select your Check-In & Check-Out Date
    again."
END IF
UNTIL numberOfDays > 0

// Call Room Table Function
CallRoomTable(locationNo)

```

```
// Customer booking process
myBookings = returnStruct(myBookings)

// Call Function Input Name & Phone Number
name = NameInput()
phoneNumber = InputPhoneNumber()

// Call Function OverBooking
totalPriceAlldays = displayOverview(locationNo, totalPax, name,
phoneNumber, startMonth, endMonth, startYear, endYear, checkInDay,
checkOutDay, numberOfDays, myBookings)

// Call Function Invoice
grandtotal += displayInvoice(locationNo, totalPax, name, phoneNumber,
startMonth, endMonth, startYear, endYear, checkInDay, checkOutDay,
numberOfDays, totalPriceAlldays, myBookings)

PRINT "Do you want to continue order? (Y/N):"
READ continueOption

// Validate continueOption input
WHILE continueOption IS NOT ONE OF {'Y', 'y', 'N', 'n'}
    PRINT "Invalid input. Please enter 'Y' for Yes or 'N' for No: "
    READ continueOption
END WHILE
UNTIL continueOption IS 'N' OR continueOption IS 'n'
BREAK
```

### Case 2: Check-In & Check-Out

- a) **Print selection:** Inform the user that the check-in and check-out option has been selected.
- b) **Call Function:** Call stafflogin() which is to handle staff login.
- c) **Check staff login:** If the staff login is successful, the checkInandcheckout(myBookings) function is to call process check-in and check-out actions using the myBookings data.

```
CASE 2:  
PRINT "You selected Check-In & Check-Out"  
  
loggedin = Call stafflogin function  
  
IF loggedin IS TRUE THEN  
    CALL checkInandcheckout function with myBookings as argument  
END IF  
  
Set continueOption to 'N'  
BREAK
```

### Case 3: Cancellation

- a) **Print selection:** Inform the user that the cancellation option has been selected.
- b) **Handle cancellation of bookings:** Start with a loop that allow user to cancel bookings. Call the function to handle the cancellation of bookings.
- c) **Another cancellation:** Prompt the user to continue with another cancellation or stop.
- d) **Validate User Input:** Validates the continueOption to make sure it is either 'Y' for yes or 'N' for no. If the input is invalid, it will prompt again until a valid input.

```
CASE 3:  
PRINT "You selected Cancellation"  
REPEAT  
    cancelBooking(myBookings) // Pass the bookings to cancel  
    PRINT "Do you want to continue cancellation? (Y/N): "  
    READ continueOption  
    // Validate continueOption input  
    WHILE continueOption IS NOT ONE OF {'Y', 'y', 'N', 'n'}  
        PRINT "Invalid input. Please enter 'Y' for Yes or 'N' for No: "  
        READ continueOption  
    END WHILE  
    UNTIL continueOption IS 'N' OR continueOption IS 'n'  
END CASE
```

#### Case 4: Report (Staff)

- a) **Print selection:** Inform the user that report option has been selected.
- b) **Prompt Input:** To prompt for report generation until the user not to continue by input 'N' or 'n'.
- c) **Call Function:** stafflogin() function is to make sure is for staff login.
- d) **Report Generation:** If the login successful, it will proceed to generate report. Call report function to generate report using the given parameters which is locationNo, grandtotal, myBookings. If unsuccessful, report cannot be generated due to fail login.
- e) **Validate User Input:** Validates the continueOption to make sure it is either 'Y' for yes or 'N' for no. If the input is invalid, it will prompt again until a valid input.

CASE 4:

```

PRINT "You selected Report (Staff)"
REPEAT
    loggedIn = stafflogin()
    IF loggedIn THEN
        PRINT "Staff logged in successfully. Generating report..."
        Report(locationNo, grandtotal, myBookings)
    ELSE
        PRINT "Login failed. Cannot generate report."
    END IF
    PRINT "Do you want to continue Report? (Y/N): "
    READ continueOption
    // Validate continueOption input
    WHILE continueOption IS NOT ONE OF {'Y', 'y', 'N', 'n'}
        PRINT "Invalid input. Please enter 'Y' for Yes or 'N' for No: "
        READ continueOption
    END WHILE
    UNTIL continueOption IS 'N' OR continueOption IS 'n'
BREAK

```

#### Case 5: Feedback

- a) **Print selection:** Inform the user that the feedback option has been selected.
- b) **Call Function:** To handle the feedback collection and processing, collect feedback from users.
- c) **End the loop:** If continueOption is 'N' will end the loop. To make sure that the main loop will stop running after displaying the message.

CASE 5:

```

PRINT "You selected Feedback"
CALL displayFeedback() // Collect and process feedback
continueOption TO 'N'
EXIT CASE

```

### Case 6: Exit

- a) **Print selection:** Display a thank you message to user to thanks for their visit.
- b) **End the loop:** If continueOption is 'Y' will end the loop. To make sure that the main loop will stop running after displaying the message.
- c) **Default:** Prompt the user to provide a valid option and ensure they can correct their input.

CASE 6:

```
PRINT "Thank you for visiting Elegant Hideaway, see you next time!"
```

```
continueOption = 'Y'
```

```
BREAK
```

```
DEFAULT:
```

```
PRINT "Invalid option. Please try again."
```

```
BREAK
```

```
END SWITCH
```

```
UNTIL continueOption IS 'N' OR continueOption IS 'n'
```

```
END FUNCTION
```

```
END
```

### 2. Location Number

- (a) **Initialization:** Start by setting the variable locationNo to 0 to initialize it for later use.
- (b) **Display Options:** Present a list of travel locations to the user, such as Kuala Lumpur, Melaka, Penang, and others, and prompt them to select one by entering a corresponding number.
- (c) **Input Validation Loop:** Use a loop to repeatedly ask the user to enter a number for the location until they provide a valid input between 1 and 6. If the input is outside this range or invalid, show an error message and prompt the user again.
- (d) **Location Confirmation:** Once the user provides a valid number, display a message confirming their selected location based on their input.
- (e) **Return:** Return the valid location number that the user selected so it can be used for further processing in the program.
- (f) **Pseudocode:**

```

START
FUNCTION LocationNumber()
    // Initialize the location number
    SET locationNo = 0

    // Display welcome message and location options
    PRINT "Welcome to Elegant Hideaway Airbnb"
    PRINT "Please let us know the location you wish to travel to at your convenience"
    PRINT "-----"
    PRINT "1. Kuala Lumpur"
    PRINT "2. Melaka"
    PRINT "3. Penang"
    PRINT "4. Selangor"
    PRINT "5. Sabah"
    PRINT "6. Sarawak"
    PRINT "Please enter the number of the location: "
    READ locationNo

    // Validate location input
    WHILE locationNo IS NOT a valid integer OR locationNo IS LESS THAN 1 OR locationNo IS
    GREATER THAN 6
        PRINT "Error: Please enter a number between 1 and 6."
        PRINT "Welcome to Elegant Hideaway Airbnb"
        PRINT "Please let us know the location you wish to travel to at your convenience"
        PRINT "-----"
        PRINT "1. Kuala Lumpur"
        PRINT "2. Melaka"
        PRINT "3. Penang"
        PRINT "4. Selangor"
        PRINT "5. Sabah"
        PRINT "6. Sarawak"
        PRINT "Please enter the number of the location: "
        READ locationNo
    END WHILE

```

```
// Confirm the selected location
SWITCH locationNo
    CASE 1:
        PRINT "You have selected Kuala Lumpur."
        BREAK
    CASE 2:
        PRINT "You have selected Melaka."
        BREAK
    CASE 3:
        PRINT "You have selected Penang."
        BREAK
    CASE 4:
        PRINT "You have selected Selangor."
        BREAK
    CASE 5:
        PRINT "You have selected Sabah."
        BREAK
    CASE 6:
        PRINT "You have selected Sarawak."
        BREAK
    DEFAULT:
        PRINT "Invalid selection."
        BREAK
END SWITCH

RETURN locationNo
END FUNCTION
END
```

### 3. Number of Pax

- (a) **Prompt User:** The function starts by calling getValidatedInput three times to collect the number of passengers in each category: adults, children, and infants. Each call to getValidatedInput displays a prompt to the user, ensuring that the input received is a valid non-negative integer.
- (b) **Calculate Total Pax:** After obtaining the valid numbers for adults, children, and infants, the function calculates the total number of passengers by adding these three numbers together. This gives the overall count of passengers.
- (c) **Display Total Pax:** The function then outputs the total number of passengers to the user, providing clear information about the total count.
- (d) **Return:** Finally, the function returns the total number of passengers. This returned value can be used for further processing or operations in other parts of the program.
- (e) **Pseudocode :**

```

START

DEFINE FUNCTION getValidatedInput(prompt)

// Function to get and validate the number of pax
FUNCTION NumberofPax()

    // Print a separator line
    PRINT "-----"

    // Call getValidatedInput to get the number of adults, children, and infants
    adult = CALL getValidatedInput("Please enter number of pax adult (13 years and older)")
    children = CALL getValidatedInput("Please enter number of pax children (2-12 years old)")
    infants = CALL getValidatedInput("Please enter number of pax infants (under 2 years old)")

    // Calculate the total number of pax
    totalPax = adult + children + infants

    // Print the total number of pax
    PRINT "-----"
    PRINT "The total number of pax is " + totalPax
    PRINT "-----"

    // Return the total number of pax
    RETURN totalPax
END FUNCTION

END

```

#### 4. Call Room Table

- (a) **Input Handling:** It takes a single parameter, locationNo, which represents the user's choice of location, mapped to a specific area.
- (b) **Decision Making:** The function uses a switch statement to determine which area function to call based on the value of locationNo.
- (c) **Handling Invalid Input:** If locationNo does not match any of the predefined cases (i.e., it is not between 1 and 6), the function outputs an error message indicating that the location number is invalid.
- (d) **Pseudocode:**

```

START
// Function declarations
DEFINE FUNCTION KualaLumpurArea()
DEFINE FUNCTION SelangorArea()
DEFINE FUNCTION MelakaArea()
DEFINE FUNCTION PenangArea()
DEFINE FUNCTION SabahArea()
DEFINE FUNCTION SarawakArea()

// Function to call the appropriate room table based on location number
FUNCTION CallRoomTable(locationNo)
    SWITCH (locationNo)
        CASE 1:
            CALL KualaLumpurArea()
            BREAK
        CASE 2:
            CALL MelakaArea()
            BREAK
        CASE 3:
            CALL PenangArea()
            BREAK
        CASE 4:
            CALL SelangorArea()
            BREAK
        CASE 5:
            CALL SabahArea()
            BREAK
        CASE 6:
            CALL SarawakArea()
            BREAK
        DEFAULT:
            // Handle invalid locationNo if necessary
            PRINT "Invalid location number"
    END SWITCH
END FUNCTION

END

```

### 5. Return Struct bookingList

- (a) **Global Variable nextBookingNumber:** This global variable is initialized to 1001 and is used to generate unique booking numbers for each new booking. Each time a new booking is made, this number is incremented to ensure that every booking has a distinct number.
- (b) **initializePrePopulatedBookings()** ensures there is a default booking for initial testing or reporting purposes.
  - a. **Initial Booking:** Sets the first booking entry with a booking number of 1001, a room number of 1, and a quantity of 0. This is primarily for testing or to ensure that other modules, like a reporting module, can run even when no real bookings have been made yet.
  - b. **Booking Number:** The nextBookingNumber is incremented so that subsequent bookings will have unique numbers.
- (c) **returnStruct(bookedList)** takes user inputs for room number and quantity, validates them, and updates the booking list. It also handles cases where the list might be full and ensures the booking list reflects the most recent updates.
  - a. **Prompt for Room Number:** The function asks the user to enter a room number between 1 and 5. It checks if the input is within this range and valid. If the user enters an invalid number or something not in the range, the function repeatedly prompts the user until a valid room number is provided.
  - b. **Prompt for Quantity:** The function then requests the user to enter the quantity of rooms they wish to book. It ensures that this quantity is a positive integer and does not exceed the allowed maximum (e.g., 10). If the input is invalid, the function prompts the user again until a valid quantity is given.
  - c. **Check Space:** The function first checks if there is space available in the bookingList by comparing the current number of bookings with the maximum allowed (MAX\_BOOKINGS).
  - d. **Add Booking:** If there is space available, the function assigns the current nextBookingNumber to the new booking, updates the bookingList with the provided room number and quantity, increments nextBookingNumber for future bookings, and updates the count of bookings.
  - e. **Handle Full List:** If the booking list is full, the function displays a message to inform the user that no more bookings can be added.

**(d) Pseudocode:**

```

START
    DEFINE FUNCTION getValidatedInput(prompt)
        SET nextBookingNumber = 1001

        FUNCTION initializePrePopulatedBookings()
            CREATE bookingList
            SET bookingList.count = 0

            IF bookingList.count < MAX_BOOKINGS
                SET bookingList.bookings[bookingList.count].bookingNumber = nextBookingNumber
                INCREMENT nextBookingNumber
                SET bookingList.bookings[bookingList.count].roomNo = 1
                SET bookingList.bookings[bookingList.count].quantity = 0
                INCREMENT bookingList.count
            END IF
            RETURN bookingList
        END FUNCTION

        FUNCTION returnStruct(bookingList)
            SET roomNo = 0
            SET quantity = 0

            DISPLAY "Room Number (1-5):"
            READ roomNo

            WHILE roomNo is invalid:
                DISPLAY "Invalid number. Please select a valid room number (1-5)"
                READ roomNo
            END WHILE

            quantity = CALL getValidatedInput("Quantity")

            WHILE quantity is not valid
                DISPLAY "Sorry! The maximum number of rooms that can be booked is 10."
                SET quantity = CALL getValidatedInput("Quantity")
            END WHILE

            IF bookingList.count < MAX_BOOKINGS
                SET bookingList.bookings[bookingList.count].bookingNumber = nextBookingNumber
                INCREMENT nextBookingNumber
                SET bookingList.bookings[bookingList.count].roomNo = roomNo
                SET bookingList.bookings[bookingList.count].quantity = quantity
                INCREMENT bookingList.count
            ELSE
                DISPLAY "Booking limit reached. Cannot add more rooms."
            END IF

            RETURN bookingList
        END FUNCTION
    END

```

### 6. Name

- (a) **Prompt for Name:** Start a loop where the user is repeatedly prompted to input their name.
- (b) **Validate Input:** For each input, check if the name is valid using the validateName function. If the name does not meet the validation criteria, display an error message telling the user to try again. Continue this process until the user provides a valid name.
- (c) **Return Valid Name:** Once the user enters a valid name, exit the loop and return the valid name as the output of the function.
- (d) **Pseudocode:**

```
START

DEFINE FUNCTION validateName(name)

DEFINE FUNCTION NameInput()
    // Display instructions to the user
    DISPLAY "Please enter your information below"
    DISPLAY "-----"

    // Clear any leftover input in the buffer
    CALL cin.ignore()

START DO
    // Prompt the user to enter their name
    DISPLAY "Name\t\t: "
    READ name

    // Validate the name
    IF validateName(name) IS FALSE THEN
        // If the name is invalid, display an error message
        DISPLAY "Invalid name : Please try again."
    END IF
    WHILE validateName(name) IS FALSE
        END DO WHILE

    // Return the valid name
    RETURN name
END FUNCTION

END
```

### 7. Phone Number

- (a) **Prompt for Phone Number:** Continuously prompt the user to enter their phone number until they provide a valid input.
- (b) **Validate Input:** Check if the entered phone number consists only of digits using the validatePhoneNumber function, and display an error message if it does not.
- (c) **Return Valid Phone Number:** Once a valid phone number is entered, return this phone number as the output of the function.
- (d) **Pseudocode:**

```
START
DEFINE FUNCTION validatePhoneNumber(phoneNumber)

DEFINE FUNCTION InputPhoneNumber()
    START DO
        // Prompt the user to enter their phone number
        DISPLAY "Phone Number\t: "
        READ phoneNumber

        // Validate the phone number
        IF validatePhoneNumber(phoneNumber) IS FALSE THEN
            // If the phone number is invalid, display an error message
            DISPLAY "Error: Please enter a valid phone number consisting only of digits."
        END IF
    WHILE validatePhoneNumber(phoneNumber) IS FALSE
    END DO WHILE

    // Return the valid phone number
    RETURN phoneNumber
END FUNCTION
END
```

### 8. Overview Booking

- (a) **Define and Initialize Room Data:** Set up arrays containing hotel room details for various locations, which include room numbers, hotel names, prices, and services offered. These arrays store the structured data for each location's rooms.
- (b) **Select Room Array:** Based on the user's location selection (locationNo), assign the appropriate room array to a variable and determine the total number of rooms available in that array.
- (c) **Display Booking Overview:** Present a summary of the booking details by printing the guest's name, phone number, check-in and check-out dates, total number of passengers, and the duration of stay.
- (d) **Calculate and Display Costs:** Use the BookingList struct to access the latest booking details, compute the total cost by multiplying the room price by the quantity and number of days, and then display the detailed pricing information along with the final total before taxes.

## Problem Solving and Programming

```
START
DEFINE FUNCTION displayOverview(locationNo, totalPax, name, phoneNumber, startMonth,
endMonth, startYer, endYear, checkInDay, checkOutDay, numberOfDays, BL)

// Define and initialize hotel rooms for Kuala Lumpur
DEFINE roomsKualaLumpur AS ARRAY OF BookingRoom
INITIALIZE roomsKualaLumpur WITH hotel details and prices

// Define and initialize hotel rooms for Melaka
DEFINE roomsMelaka AS ARRAY OF BookingRoom
INITIALIZE roomsMelaka WITH hotel details and prices

// Define and initialize hotel rooms for Penang
DEFINE roomsPenang AS ARRAY OF BookingRoom
INITIALIZE roomsPenang WITH hotel details and prices

// Define and initialize hotel rooms for Selangor
DEFINE roomsSelangor AS ARRAY OF BookingRoom
INITIALIZE roomsSelangor WITH hotel details and prices

// Define and initialize hotel rooms for Sabah
DEFINE roomsSabah AS ARRAY OF BookingRoom
INITIALIZE roomsSabah WITH hotel details and prices

// Define and initialize hotel rooms for Sarawak
DEFINE roomsSarawak AS ARRAY OF BookingRoom
INITIALIZE roomsSarawak WITH hotel details and prices

// Choose the room array based on locationNo
IF locationNo = 1 THEN
    SET rooms = roomsKualaLumpur
    SET totalRooms = number of rooms in roomsKualaLumpur
ELSE IF locationNo = 2 THEN
    SET rooms = roomsMelaka
    SET totalRooms = number of rooms in roomsMelaka
ELSE IF locationNo = 3 THEN
    SET rooms TO roomsPenang
    SET totalRooms = number of rooms in roomsPenang
ELSE IF locationNo = 4 THEN
    SET rooms TO roomsSelangor
    SET totalRooms = number of rooms in roomsSelangor
ELSE IF locationNo = 5 THEN
    SET rooms = roomsSabah
    SET totalRooms = number of rooms in roomsSabah
ELSE IF locationNo = 6 THEN
    SET rooms = roomsSarawak
    SET totalRooms = number of rooms in roomsSarawak
ELSE
    DISPLAY "Invalid location number."
    RETURN 1
END IF
```

```
// Display overview of bookings
DISPLAY "Overview of Bookings"
DISPLAY formatted details of name, phoneNumber, checkIn and checkOut dates, totalPax,
numberOfDays

// Initialize totalPrice
SET totalPrice = 0

// Retrieve the latest booking details
SET latestBookingIndex = BL.count - 1
SET roomNo = BL.bookings[latestBookingIndex].roomNo
SET quantity = BL.bookings[latestBookingIndex].quantity
SET bookingNo = BL.bookings[latestBookingIndex].bookingNumber

// Loop through the rooms to find and display booking details
FOR j = 0; j < totalRooms; ++j
IF rooms[j].roomNo = roomNo THEN
    SET roomPrice = rooms[j].price
    SET bookingTotalPrice = roomPrice * quantity
    ADD bookingTotalPrice TO totalPrice
END IF
END FOR

DISPLAY booking details including bookingNo, roomNo, hotelName, quantity, roomPrice,
and bookingTotalPrice
DISPLAY services offered

// Calculate total price for all days
SET totalPriceAlldays = totalPrice * numberOfDays
DISPLAY totalPriceAlldays

RETURN totalPriceAlldays
END FUNCTION
END
```



### 9. Invoice

- (a) **Function Definitions:** Implement ADDONSERVICE() to display additional services, paymentMethodTable() to display payment options, getValidatedInput(prompt) for user input validation, and pinValidation(prompt) for verifying PINs.
- (b) **Define and Initialize Room Data:** Set up arrays containing hotel room details for various locations, which include room numbers, hotel names, prices, and services offered. These arrays store the structured data for each location's rooms.
- (c) **Select Room Array:** Based on the user's location selection (locationNo), assign the appropriate room array to a variable and determine the total number of rooms available in that array.
- (d) **Add-On Service:** Ask the user if they want additional services, calculate the cost for the selected service if they choose 'Yes', or set the cost to zero if 'No'.
- (e) **Voucher Code:** Inquire if the user wants to apply a voucher, apply the voucher to calculate the rebate if chosen, or indicate no voucher was used.
- (f) **Invoice Calculation:** Calculate the total cost including room charges, taxes, service charges, and discounts, and print an invoice with all details and cost breakdowns.
- (g) **Payment Method:** Present payment options, handle the selected method to process payment, and ensure sufficient funds if paying by cash.
- (h) **Print Confirmation and Summary:** After processing, print a booking confirmation, payment summary, and provide check-in/check-out information.

## Problem Solving and Programming

```
START
    DEFINE FUNCTION ADDONSERVICE();
    DEFINE FUNCTION paymentMethodTable();
    DEFINE FUNCTION getValidatedInput(prompt);
    DEFINE FUNCTION pinValidation(prompt);

Function DisplayInvoice(locationNo, totalPax, name, phoneNumber, startMonth, endMonth, startYear,
endYear, checkInDay, checkOutDay, numberOfDays, totalPriceAllDays, BookingList)

    // Define and initialize hotel rooms for Kuala Lumpur
    DEFINE roomsKualaLumpur AS ARRAY OF BookingRoom
    INITIALIZE roomsKualaLumpur WITH hotel details and prices

    // Define and initialize hotel rooms for Melaka
    DEFINE roomsMelaka AS ARRAY OF BookingRoom
    INITIALIZE roomsMelaka WITH hotel details and prices

    // Define and initialize hotel rooms for Penang
    DEFINE roomsPenang AS ARRAY OF BookingRoom
    INITIALIZE roomsPenang WITH hotel details and prices

    // Define and initialize hotel rooms for Selangor
    DEFINE roomsSelangor AS ARRAY OF BookingRoom
    INITIALIZE roomsSelangor WITH hotel details and prices

    // Define and initialize hotel rooms for Sabah
    DEFINE roomsSabah AS ARRAY OF BookingRoom
    INITIALIZE roomsSabah WITH hotel details and prices

    // Define and initialize hotel rooms for Sarawak
    DEFINE roomsSarawak AS ARRAY OF BookingRoom
    INITIALIZE roomsSarawak WITH hotel details and prices

// Choose the room array based on locationNo
IF locationNo = 1 THEN
    SET rooms = roomsKualaLumpur
    SET totalRooms = number of rooms in roomsKualaLumpur
ELSE IF locationNo = 2 THEN
    SET rooms = roomsMelaka
    SET totalRooms = number of rooms in roomsMelaka
ELSE IF locationNo = 3 THEN
    SET rooms TO roomsPenang
    SET totalRooms = number of rooms in roomsPenang
ELSE IF locationNo = 4 THEN
    SET rooms TO roomsSelangor
    SET totalRooms = number of rooms in roomsSelangor
ELSE IF locationNo = 5 THEN
    SET rooms = roomsSabah
    SET totalRooms = number of rooms in roomsSabah
ELSE IF locationNo = 6 THEN
    SET rooms = roomsSarawak
    SET totalRooms = number of rooms in roomsSarawak
ELSE
    DISPLAY "Invalid location number."
    RETURN 1
END IF
```

```
// Process add-on service
SET serviceName = ""
SET servicePrice = 0.00

DISPLAY "Do you need any add-on service (Y/N): "
READ service

IF userInput IS 'Y' OR 'y' THEN
    CALL ADDONSERVICE()

DISPLAY "Please select the service you need (1-5): "
READ serviceNo

// Validate the serviceNo
WHILE serviceNo IS NOT BETWEEN 1 AND 5
    DISPLAY "Error: Please enter a number between 1 and 5."
    DISPLAY "Please select the service you need (1-5): "
    READ serviceNo INTO userInput
END WHILE

// Determine service name and price
SWITCH serviceNo
    CASE 1:
        SET serviceName = "High Speed Wi-Fi, RM 30.00 per day"
        SET servicePrice = 30.00 * numberOfDays
    CASE 2:
        SET serviceName = "Breakfast, RM 50.00 per day"
        SET servicePrice = 50.00 * numberOfDays
    CASE 3:
        SET serviceName = "Airport Shuttle, RM 150.00 per day"
        SET servicePrice = 150.00 * numberOfDays
    CASE 4:
        SET serviceName = "Room Upgrade, RM 300.00 per day"
        SET servicePrice = 300.00 * numberOfDays
    CASE 5:
        SET serviceName = "Mid-Stay Cleaning, RM 100.00 per day"
        SET servicePrice = 100.00 * numberOfDays
END SWITCH

ELSE IF userInput IS 'N' OR 'n' THEN
    SET serviceName = "NO EXTRA SERVICE"
    SET servicePrice = 0.00
ELSE
    DISPLAY "Invalid input. Please enter 'Y' for Yes or 'N' for No."
    RETURN TO service processing step
END IF
```

```

// Process voucher code
DISPLAY "Do you want to apply a voucher code (Y/N): "
READ gotvoucher

IF userInput IS 'Y' OR 'y' THEN
    CALL login(totalPriceAlldays) TO GET finalresult
ELSE IF userInput IS 'N' OR 'n' THEN
    DISPLAY "NO VOUCHER"
    SET finalresult.selectedvoucher TO "NO VOUCHER"
    SET finalresult.rebate TO 0.00
ELSE
    DISPLAY "Invalid input. Please enter 'Y' for Yes or 'N' for No."
    RETURN TO voucher processing step
END IF

// Get latest booking details
SET latestBookingIndex = BL.count - 1
SET roomNo = BL.bookings[latestBookingIndex].roomNo
SET quantity = BL.bookings[latestBookingIndex].quantity
SET bookingNo = BL.bookings[latestBookingIndex].bookingNumber

// Loop through rooms to find details
FOR j = 0; j < totalRooms; ++j
IF rooms[j].roomNo == roomNo THEN
    SET roomPrice = rooms[j].price
    SET bookingTotalPrice = roomPrice * quantity
    ADD bookingTotalPrice = totalPrice
END IF
END FOR

// Display invoice
DISPLAY booking details including bookingNo, roomNo, hotelName, quantity, roomPrice,
and bookingTotalPrice,services offered

// Calculate taxes, discounts, and grand total
DEFINE SSTRATE AS 0.06 // 6%
SET SST = totalPriceAlldays * SSTRATE
SET grandtotal = (totalPriceAlldays + SST + servicePrice) - finalresult.rebate

// Select payment method
CALL paymentMethodTable()
DISPLAY "Please select payment method (1-5): "
READ paymentMethod
// Validate payment method
WHILE paymentMethod IS NOT BETWEEN 1 AND 5
    DISPLAY "Error: Please enter a number between 1 and 5."
    DISPLAY "Please select payment method (1-5): "
    READ paymentMethod
END WHILE

```

```

// Handle payment details
SWITCH paymentMethod
CASE 1:
    DISPLAY "Credit/Debit Cards"
    READ cardNumber
    READ PIN
    CALL pinValidation("Enter PIN")
    SET cashAmount = grandtotal
CASE 2:
    DISPLAY "Bank Transfer"
    READ accountNo
    READ PIN
    CALL pinValidation("Enter PIN")
    SET cashAmount = grandtotal
CASE 3:
    DISPLAY "Touch N Go"
    READ PIN
    CALL pinValidation("Enter PIN")
    SET cashAmount TO grandtotal
CASE 4:
    DISPLAY "Apple Pay"
    READ PIN
    CALL pinValidation("Enter PIN")
    SET cashAmount TO grandtotal
CASE 5:
    DISPLAY "Cash Payment"
    DO
        DISPLAY "Please enter cash amount:"
        READ cashAmount
        IF cashAmount >= grandtotal THEN
            SET change = cashAmount - grandtotal
        ELSE
            DISPLAY "Insufficient amount. Please make payment again."
        END IF
        WHILE cashAmount >= grandtotal
        END DO WHILE
    END SWITCH

    // Print booking confirmation and payment summary
    DISPLAY "Booking Confirmation"
    DISPLAY "Total before taxes: RM " + totalPriceAlldays
    DISPLAY "SST (6%): RM " + SST
    DISPLAY "Service Cost: RM " + servicePrice
    DISPLAY "Voucher Discount: RM " + finalresult.rebate
    DISPLAY "Grand Total: RM " + grandtotal
    DISPLAY "Payment Method: " + paymentMethodName
    DISPLAY "Payment Amount: RM " + cashAmount
    DISPLAY "Change: RM " + change
    DISPLAY "Check-in/Check-out Information"

RETURN grandtotal
END FUNCTION
END

```

## 10. Validation Pin

- (a) **Prompt Display:** It shows a prompt message to the user asking for the PIN input.
- (b) **Input Validation:** It ensures the input is an integer and checks if it is between 1000 and 999999, thus covering 4-to-6-digit numbers.
- (c) **Length Check:** It verifies that the length of the PIN is between 4 and 6 digits.
- (d) **Error Handling:** It clears any error state and ignores invalid input to avoid infinite loops on invalid data.
- (e) **User Feedback:** It provides feedback and prompts the user again if the input is invalid.

```
FUNCTION pinValidation(prompt)
    Display the prompt message followed by a colon

    WHILE (True)
        Try to read an integer input as PIN into variable `value`

        IF the input `value` is valid and within the range 1000 to 999999
            CONVERT `value` to a string
            Calculate the length of this string
            IF the length of the string is between 4 and 6 characters
                EXIT the loop
            ELSE
                Display an error message: "PIN must be between 4 and 6 digits"
            ELSE
                CLEAR any input errors
                IGNORE any remaining input until the end of the line

                DISPLAY "Please enter a valid PIN number with 4 to 6 digits"

                DISPLAY the prompt message again for user input
        END WHILE

        RETURN the valid PIN
    END FUNCTION
END
```

## 11. Validation Integer Input

- (a) **Displaying the Prompt:** The function first displays a prompt message, which is formatted to be left-aligned with a width of 56 characters, followed by a colon. This helps ensure that the prompt is neatly aligned in the output.
- (b) **Input Validation Loop:** The function enters a loop where it attempts to read an integer from the user. The loop condition checks if the input operation fails or if the number entered is negative.
- (c) **Handling Invalid Input:** If the input is invalid, the function clears any error flags that were set due to the invalid input. It then discards the rest of the input until the end of the current line to prepare for the next input attempt.
- (d) **Error Message and Re-Prompt:** After clearing the invalid input, the function prints an error message indicating that the user should enter a valid number. It then re-displays the prompt to ask for input again.
- (e) **Returning Valid Input:** Once a valid, non-negative integer is entered, the loop terminates. The function then returns this valid integer value to the caller.

```

START
FUNCTION getValidatedInput(prompt)
    INITIALIZE variable value as Integer

    // Display the prompt to the user
    PRINT prompt followed by a colon, right-aligned with width 56

    // Loop until valid input is received
    WHILE true
        READ user input into variable value

        // Check if the input is valid
        IF input is not an integer or value is less than 0
            CLEAR the error state
            IGNORE the remaining characters in the input buffer

            // Inform the user of the error and prompt again
            PRINT "Error: Please enter a valid number."
            PRINT prompt followed by a colon, right-aligned with width 56
        ELSE
            // Valid input received, exit the loop
            Break
        END IF
    END WHILE

    RETURN value
END FUNCTION
END

```

### 12. Validation Phone Number

- (a) **Function Definition:** The validatePhoneNumber function checks if a given string, phoneNumber, is a valid phone number. It returns true if the phone number meets the criteria, and false otherwise.
- (b) **Check for Empty String:** If phoneNumber is empty, it returns false.
- (c) **Check for Non-Digit Characters:** It verifies that all characters in phoneNumber are digits. If any character is not a digit, it returns false.
- (d) **Check Length:** It ensures that phoneNumber has at least 10 digits. If it has fewer than 10 digits, it returns false.
- (e) **Return True if Valid:** If phoneNumber passes all checks not empty, all digits, and at least 10 digits long, it returns true.

```
FUNCTION validatePhoneNumber(phoneNumber)
    // Check if the phone number is empty
    If phoneNumber is empty
        Return false

    // Check if the phone number contains only digits
    For each character in phoneNumber
        If character is not a digit
            Return false

    // Check if the length of the phone number is at least 10 digits
    If length of phoneNumber is less than 10
        Return false

    // If all checks pass, the phone number is valid
    Return true
End Function
```

### 13. Validation IC Number

- (a) **Function Definition:** The function validateICNumber takes a string ic\_num and returns a boolean indicating whether the string is a valid 12-digit IC number.
- (b) **Check Length:** It first verifies that the length of ic\_num is exactly 12 characters. If not, it returns false.
- (c) **Check Digits:** It then iterates through each character in ic\_num and ensures each character is a digit. If any character is not a digit, it returns false.
- (d) **Return True:** If both conditions are satisfied (the length is 12 and all characters are digits), the function returns true.

```
FUNCTION validateICNumber(ic_num)
    // Check if the length of ic_num is exactly 12 characters
    IF length of ic_num is not equal to 12
        RETURN false
    END IF

    // Check if all characters in ic_num are digits
    FOR each character c in ic_num
        IF c is not a digit
            Return false
        END IF
    END FOR

    // If all checks pass, the IC number is valid
    RETURN true
END Function
```

### 14. Check In & Check Out

- (a) **Purpose:** The checkInAndCheckout function is designed to handle check-in and check-out operations for bookings in a booking system, using a BookingList object.
- (b) **Initialization:** The function begins by declaring an integer variable named selection to capture the user's menu choice.
- (c) **Menu Loop:** It uses a DO WHILE loop to continually display a menu with options to "CHECK IN", "CHECK OUT", or "EXIT", and processes the user's selection until they choose to exit.
- (d) **Selection Validation:** The user's menu selection is validated to ensure it is a number between 1 and 3. The invalid input prompts the user to enter a valid number.
- (e) **Check-In Process:**
  - a. When the user selects "1" for CHECK IN, they enter a booking number, which is validated and checked against the BookingList.
  - b. If the booking is found, the user is then asked for the customer's name and IC number, which must be validated.
  - c. Upon successful check-in, a confirmation message is displayed.
- (f) **Check-Out Process:**
  - a. When the user selects "2" for CHECK OUT, they enter a booking number, which is validated and checked.
  - b. If the booking is found, the user provides the delayed time, and fines are calculated based on the delay.
  - c. The user is then asked to choose a payment method from several options and provide the necessary payment details.
  - d. If the payment is processed successfully, or if no fines are incurred, a confirmation message is shown.
- (g) **Exit Process:** When the user selects "3" for EXIT, the function displays a message indicating that the system is exiting and then terminates.
- (h) **Invalid Selection:** If the user inputs a value that is not between 1 and 3, an error message is shown, and the user is prompted to enter a valid choice.
- (i) **Loop Continuation:** The loop continues to prompt the user with the menu options and processes their choices until they select the exit option.

## Problem Solving and Programming

```
START

// Declare Validation Function
DEFINE FUNCTION validateName(name);
DEFINE FUNCTION getValidatedInput(prompt);
DEFINE FUNCTION validateICNumber(ic_num);
DEFINE FUNCTION pinValidation(prompt);

Function checkInandcheckout(BookingList BL)
DO
    // Display the menu
    PRINT "WELCOME TO STAFF INTERFACE"
    PRINT "-----"
    PRINT "1. CHECK IN"
    PRINT "2. CHECK OUT"
    PRINT "3. EXIT"
    PRINT "-----"
    PRINT "Please select 1, 2, or 3: "
    READ selection

    // Input validation for selection
    WHILE (selection is not a number between 1 and 3)
        PRINT "Error: Please enter a number between 1 - 3"
        PRINT "Your selection: "
        READ selection
    END WHILE

    IF (selection == 1) THEN // CHECK IN
        PRINT "Enter the booking number you want to Check In: "
        READ bookingNumberToCheckIn

        WHILE (bookingNumberToCheckIn is not valid)
            PRINT "Error: Please enter a valid booking number"
            PRINT "Booking Number : "
            READ bookingNumberToCheckIn
        END WHILE

        SET bookingFoundCheckIN = False

        WHILE bookingFoundCheckIN is False
            FOR each booking in BL.bookings
                IF (booking.bookingNumber == bookingNumberToCheckIn) THEN
                    Print "Booking number " + bookingNumberToCheckIn + " has been Checked In."
                    SET bookingFoundCheckIN = True
                    BREAK
                END IF
            END FOR

            IF (bookingFoundCheckIN is False) THEN
                Print "Booking number " + bookingNumberToCheckIn + " not found."
                Print "Enter the booking number you want to Check In: "
                Read bookingNumberToCheckIn
            END IF
        END WHILE
    END IF
END
```

## Problem Solving and Programming

```
// Enter the customer Name
DO
    PRINT "Customer Name: "
    READ name
    IF name is invalid THEN
        PRINT "Invalid name: Please try again."
    END IF
    WHILE name is invalid
    END DO WHILE

// Enter the IC number
SET icValid = False
WHILE icValid is False
    PRINT "Customer IC number: "
    READ ic_num
    IF (ic_num is invalid) THEN
        PRINT "Invalid IC number: Please enter a valid IC number."
    ELSE
        SET icValid to True
    END IF
END WHILE

PRINT "-----"
PRINT "Check-in successful!"
PRINT "-----"

ELSE IF (selection == 2) THEN // CHECK OUT
    PRINT "Enter the booking number you want to Check Out: "
    READ bookingNumberToCheckOut

    WHILE bookingNumberToCheckOut is not valid
        PRINT "Error: Please enter a valid booking number"
        PRINT "Booking Number : "
        READ bookingNumberToCheckOut
    END WHILE

    SET bookingFoundCheckOUT = False
    SET FINESPERHOURS = 20.00
    SET fines = 0.00
    SET change= 0.00

    WHILE (bookingFoundCheckOUT is False)
        FOR each booking in BL.bookings
            IF (booking.bookingNumber == bookingNumberToCheckOut) THEN
                PRINT "Customer delayed time (in hours): "
                READ delayed_time

                WHILE (delayed_time is not valid)
                    PRINT "Error: Please enter a valid number for delayed time."
                    PRINT "Customer delayed time (in hours): "
                    READ delayed_time
                END WHILE
```

## Problem Solving and Programming

```
IF (delayed_time > 0) THEN
    SET fines to delayed_time * FINESPERHOURS
    PRINT "Fine for Customer: RM" + fines
    PRINT "Please select the customer payment method"
    CALL paymentMethodTable()
    PRINT "Please select payment method (1-5):"
    READ paymentMethod

    WHILE (paymentMethod is not between 1 and 5)
        PRINT "Error: Please enter a number between 1 and 5."
        PRINT "Please select payment method (1-5):"
        READ paymentMethod
    END WHILE

    SWITCH paymentMethod
        CASE 1:
            SET paymentMethodName = "Credit/Debit Cards"
            SET cardNumber = getValidatedInput("Card Number")
            SET PIN = pinValidation("PIN")
            SET cashAmount = fines
        CASE 2:
            SET paymentMethodName = "Bank Transfer"
            SET accountNo = getValidatedInput("Account Number")
            SET PIN = pinValidation("PIN")
            SET cashAmount = fines
        CASE 3:
            SET paymentMethodName = "Touch N Go"
            SET PIN = pinValidation("PIN")
            SET cashAmount = fines
        CASE 4:
            SET paymentMethodName = "Apple Pay"
            SET PIN = pinValidation("PIN")
            SET cashAmount = fines
        CASE 5:
            DO
                SET paymentMethodName = "Cash Payment"
                SET cashAmount = getValidatedInput("Payment Amount")
                IF (cashAmount >= fines) THEN
                    SET change to cashAmount - fines
                ELSE
                    PRINT "Insufficient amount. Please make payment again."
                END IF
                WHILE (cashAmount < fines)
                END DO
            END SWITCH
        ELSE
            Print "-----"
            Print "No fines incurred."
        END IF

        Print "-----"
        Print "Check-out complete!"
        Print "Booking number " + bookingNumberToCheckOut + " has been Checked Out."
        Set bookingFoundCheckOUT to True
        BREAK
    END IF
END FOR
```

## Problem Solving and Programming

---

```
IF (bookingFoundCheckOUT is False) THEN
    PRINT "Booking number " + bookingNumberToCheckOut + " not found."
    PRINT "Enter the booking number you want to Check Out: "
    READ bookingNumberToCheckOut
END IF
END WHILE

ELSE IF (selection == 3) THEN // EXIT
    PRINT "Exiting the system..."

ELSE
    PRINT "Invalid selection. Please choose 1, 2, or 3."
END IF

UNTIL (selection == 3) // Continue until user selects exit
END FUNCTION
END
```

### 15. Cancellation

- (a) **Prompt for Booking Number:** The function starts by asking the user to input the booking number they wish to cancel.
- (b) **Initialize Booking Status:** It initializes bookingFound to false to keep track of whether the specified booking exists in the list.
- (c) **Print Current Bookings:** To help with debugging and verification, the function prints out the list of current bookings before attempting to cancel any.
- (d) **Find and Remove the Booking:** The function searches through the list for the booking with the number entered by the user. If it finds the booking, it removes it by shifting all subsequent bookings one position to the left and decreases the booking count by one. A confirmation message is displayed to indicate that the booking has been successfully cancelled.
- (e) **Handle Booking Not Found:** If the function does not find a booking with the specified number, it informs the user that the booking number was not found, ensuring the user receives appropriate feedback regarding the cancellation attempt.

```
START
FUNCTION cancelBooking(BookingList BL)
    // Prompt user to enter booking number to cancel
    OUTPUT "Enter the booking number you want to cancel: "
    INPUT bookingNumberToCancel

    // Validate input for booking number
    WHILE input is invalid
        OUTPUT "Error: Please enter a valid booking number"
        OUTPUT "Booking Number : "
        INPUT bookingNumberToCancel
    END WHILE

    // Initialize variable to check if booking is found
    bookingFound = False

    // Print current bookings for debugging
    OUTPUT "Current bookings before cancellation:"
    FOR i = 0; i < BL.count; ++i
        OUTPUT "Booking Number: " + BL.bookings[i].bookingNumber
        + ", Room No: " + BL.bookings[i].roomNo
        + ", Quantity: " + BL.bookings[i].quantity
    END FOR

    // Search for the booking to cancel
    FOR i = 0; i < BL.count; ++i
        IF BL.bookings[i].bookingNumber == bookingNumberToCancel
            // Remove the booking
            FOR int j = i; j < BL.count - 1; ++
                BL.bookings[j] = BL.bookings[j + 1] // Shift elements to the left
            ENDFOR
            BL.count = BL.count - 1 // Decrement booking count
            OUTPUT "Booking number " + bookingNumberToCancel + " has been cancelled."
            bookingFound = True
            Break // Exit loop as booking has been removed
        ENDIF
    ENDFOR

    // If booking was not found, inform user
    IF bookingFound is False
        Output "Booking number " + bookingNumberToCancel + " not found."
    ENDIF
ENDFUNCTION
END
```

### 16. Staff login

#### (a) Setup Variables

- The process begins with setting NUM\_STAFFS to 3, indicating that the system is designed to handle three staff members. Arrays correctStaffIDs and correctPasswords are defined to store valid staff IDs and their corresponding passwords. This setup is crucial as it provides the necessary data for verifying user credentials during the authentication process.

#### (b) Infinite Loop

- The function then enters an infinite loop that will continue running until the user chooses to exit. This loop is fundamental for repeatedly handling multiple authentication attempts, allowing users to try logging in as many times as needed before successfully authenticating or deciding to quit.

#### (c) Initialize Authentication Status

- At the start of the loop, isAuthenticated is set to false. This flag keeps track of whether the user has successfully logged into the system. It ensures that the loop continues to prompt the user for credentials until successful authentication or until the user opts to exit.

#### (d) Request User Input

- Within the loop, the system prompts the user to enter their Staff ID and Password. These prompts are displayed to the user, and their responses are read and stored. This step is essential for collecting the necessary information that will be validated against the stored credentials.

#### (e) Verify Credentials

- **Check Staff ID:** The entered Staff ID is checked against the correctStaffIDs array. If a match is found, a flag idExists is set to true, indicating that the Staff ID is valid. This step is critical for ensuring that only recognized IDs are considered during the login process.
- **Validate Password:** Once the Staff ID is validated, the system proceeds to verify the Password by comparing it with the corresponding entry in the correctPasswords array. If both the ID and Password are correct, a success message is displayed, isAuthenticated is set to true, and the function returns true to signal successful login. If the Password is incorrect, or the Staff ID is not found, appropriate error messages are shown, and the user is prompted to either retry or exit.
- **Handle Errors:** Error handling involves printing messages if the provided ID does not exist or if the Password is incorrect. The system then breaks out of the current attempt loop, prompting the user to decide whether to try logging in again or to exit the process.

#### (f) User Decision

- If authentication fails, the system prompts the user to decide whether they want to retry entering their credentials or exit the program. The user's choice (staffchoice) is read and validated to ensure it is either 'Y' (yes) or 'N' (no). If the user opts to exit, a message is printed to indicate the termination of the process, and the function returns false, ending the authentication attempt.

#### (g) Post-Authentication Choices

- Upon successful authentication, the system asks the user if they wish to continue using the staff interface. This prompt allows the user to choose whether to continue working within the system or to exit. The user's choice is read and validated to ensure it is either 'Y' (yes) or 'N' (no). If the user chooses not to continue, a message is printed to confirm their exit, and the function returns false, concluding the session.

## Problem Solving and Programming

```
START
Function stafflogin() -> Boolean:
    Define constant NUM_STAFFS as 3
    Define array correctStaffIDs with values ["staff1", "staff2", "staff3"]
    Define array correctPasswords with values ["password1", "password2", "password3"]

    Initialize inputStaffID to empty string
    Initialize inputPassword to empty string
    Initialize staffchoice to empty character

    Loop forever:
        Initialize isAuthenticated to false

        // Loop until a correct Staff ID and Password are entered
        While not isAuthenticated:
            Print "Login"
            Print "-----"

            // Prompt and get Staff ID
            Print "Staff ID: "
            Read inputStaffID

            // Prompt and get Password
            Print "Password: "
            Read inputPassword

            Initialize idExists to false

            // Check if the entered Staff ID and Password are correct
            For i from 0 to NUM_STAFFS - 1:
                If inputStaffID matches correctStaffIDs[i]:
                    Set idExists to true
                If inputPassword matches correctPasswords[i]:
                    Print "Login successful!"
                    Set isAuthenticated to true
                    Return true
                Else:
                    Print "Incorrect Password. Please try again."
                    Break the loop to ask if the user wants to try again

            If not idExists:
                Print "Incorrect Staff ID. Please try again."

            // If authentication was not successful, ask if user wants to try again
            If not isAuthenticated:
                Print "Do you want to try again? (Y/N): "
                Read staffchoice

                While staffchoice is not 'Y', 'y', 'N', or 'n':
                    Print "Invalid input. Please enter 'Y' for Yes or 'N' for No: "
                    Read staffchoice
```

```
If staffchoice is not 'Y' or 'y':
    Print "Exiting staff interface..."
    Return false // Indicate failed authentication

// If authenticated, ask if the user wants to continue in the staff interface
Print "Do you want to continue in the staff interface? (Y/N): "
Read staffchoice

While staffchoice is not 'Y', 'y', 'N', or 'n':
    Print "Invalid input. Please enter 'Y' for Yes or 'N' for No: "
    Read staffchoice

If staffchoice is not 'Y' or 'y':
    Print "Exiting staff interface..."
    Return false // Indicate successful login

End Function
END
```

## 17. Voucher

### (a) Setup Arrays

- The process begins by initializing two key arrays: vouchers and discounts. The vouchers array contains a collection of valid voucher codes that can be used for discounts. Each entry in this array represents a different voucher code that users might input. Correspondingly, the discounts array holds the discount rates associated with each voucher code. These discount rates are percentages or amounts by which the original price (rebatePrice) will be reduced when a valid voucher code is applied.

### (b) Variables

- Voucher Input and Calculation Variables:** Several important variables are set up to manage the discount calculation:
- Vouchername:** This variable is designated to store the voucher code entered by the user. It serves as the key input for the voucher validation process.
- Finalprice:** Initially set to the value of rebatePrice, this variable will eventually hold the final price after any applicable discount has been applied. By starting with the original rebate price, finalprice ensures that the discount calculation can correctly adjust the price.
- Rebate:** This variable is used to keep track of the amount of discount calculated based on the voucher code. It represents the financial reduction applied to the original rebate price.
- VoucherFound:** A boolean flag used to indicate whether a valid voucher code has been found. This flag helps control the flow of the validation process by breaking out of the loop once a valid code is confirmed.

### (c) Voucher Code Input and Validation

- User Input Loop:** The function enters a loop that continuously prompts the user to input a voucher code until a valid code is provided. This loop is essential for ensuring that only correct and valid voucher codes are processed:
- Prompting for Input:** The user is prompted to enter a voucher code, which is then captured and stored in the vouchername variable. This input represents the code the user wants to apply for a discount.
- Setting Up Final Price:** Before checking the validity of the voucher code, finalprice is reset to the original rebatePrice. This setup is crucial because it ensures that the discount calculation starts with the correct base price.

### (d) Check Voucher Code

- Voucher Matching:** For each valid voucher code in the array, the function compares it with the vouchername entered by the user. If a match is found:
- Calculate Rebate:** The discount rate corresponding to the matched voucher code is retrieved from the discounts array. The rebate amount is then calculated using this discount rate, and voucherFound is set to true to indicate that a valid voucher code has been found.
- Exit Loop:** Setting voucherFound to true breaks the loop, concluding the validation process and proceeding to the next steps.
- No Match Found:** If the voucher code does not match any entry in the vouchers array, an error message is displayed to the user. The user is then prompted to try entering a voucher code again. This ensures that only valid codes are accepted and processed.

### (e) Display Rebate and Return Object

- Print Rebate:** The rebate amount is printed, formatted to two decimal places. This formatting provides clarity and precision, ensuring that the discount amount is presented in a user-friendly manner.
- Create and Return RebateValue Object:** A RebateValue object is instantiated with the calculated rebate and the vouchername. This object encapsulates the details of the applied discount and is returned by the function. The returned object can be used for further processing or display, providing a structured way to handle discount information.

```
START
Function Voucher(rebatePrice: Double) -> RebateValue:
    Define array vouchers with values ["CNY8888", "HR4444", "DP6666", "MC1225", "V2141314",
    "WELCOME123"]
    Define array discounts with values [0.08, 0.10, 0.12, 0.15, 0.18, 0.20]

    Initialize vouchernumber to empty string
    Initialize finalprice to rebatePrice
    Initialize rebate to 0.0

    Initialize voucherFound to false

    Loop while not voucherFound:
        Print "Enter your voucher code: "
        Read vouchernumber

        Set finalprice to rebatePrice

        // Check if vouchernumber is valid
        For i from 0 to 5:
            If vouchernumber matches vouchers[i]:
                Set rebate to rebatePrice * discounts[i]
                Set voucherFound to true
                Break the loop
            End If
        End For

        If not voucherFound:
            Print "Invalid voucher code. Please try again."

        // Output the final rebate and voucher information
        Print rebate with 2 decimal places
        Create RebateValue x with rebate and vouchernumber
        Return x

End Function
END
```

## 18. Member Login

### (a) User Membership Choice

- The function allows the user to decide between logging in as a current member or registering as a new member. This choice is stored in the variable choice\_membership, and the function ensures that the user can only proceed with valid input.

### (b) Input Validation

- To ensure correct user interaction, the function uses a loop to continuously prompt the user for input until a valid choice is made. It checks whether the input is a number and within the valid range (1 for current member, 2 for new member). If the input is invalid, the buffer is cleared, and the user is asked again, ensuring robust error handling.

### (c) Login or Registration Process

- If the user selects option 1 (current member), the function calls handleMemberSignUp() to process the login for an existing member.
- If the user selects option 2 (new member), it triggers handleNewMemberRegistration() to process the new member's registration.

### (d) Voucher and Rebate Calculation

- After successfully handling either login or registration, the function calls the Voucher function with the parameter rebatePrice to compute the rebate value. This rebate is then stored in the variable result along with any associated voucher code.

### (e) Return Value

- Once the voucher is processed, the function concludes by returning the result, which contains the calculated rebate value and the voucher code. This ensures that the user's membership status is correctly linked to any available discounts or rebates.

```

START
Function login(rebatePrice):
    Define choice_membership
    Define valid_input as false
    Create an instance of Membership named membership
    Define result as RebateValue

    While not valid_input:
        Print "Are you a member?"
        Print choices (1 for Yes, 2 for No)
        Prompt for choice_membership and read input

        If input is invalid:
            Clear input buffer
            Print "Invalid input. Please enter a number (1 or 2)."

        Else if choice_membership is 1:
            Call membership.handleMemberSignUp()
            Call Voucher function with rebatePrice
            Set result to the return value of Voucher function
            Set valid_input to true

        Else if choice_membership is 2:
            Call membership.handleNewMemberRegistration()
            Call Voucher function with rebatePrice
            Set result to the return value of Voucher function
            Set valid_input to true

        Else:
            Print "Invalid choice. Please enter 1 or 2."

    Return result
End Function
END

```

### 19. Membership Structure

#### (a) Constants and Member Variables

- **NUM\_MEMBERS:** This constant represents the total number of valid member IDs and corresponding passwords that the system can handle. It is used to limit the size of the arrays that store the member IDs and passwords.
- **correctMemberIDs & correctPasswords:** These arrays store valid member IDs and their corresponding passwords. The system uses these arrays during the login process to authenticate existing members by checking the entered credentials against the stored ones.
- **name, phoneNumber, email:** These are variables used during the registration process for new members. They store the information provided by the user, such as their full name, phone number, and email address.

#### (b) Validation Methods

- **isValidName(name):** This function checks if the entered name is valid. It first ensures that the name is not empty. Then, it iterates through each character in the name to verify that all characters are either letters or spaces. If any character is found that does not meet these criteria, the function returns false. If the entire name is valid, it returns true.
- **isValidPhoneNumber(phoneNumber):** This function validates the format of the entered phone number. It checks whether the length of the phone number is between 10 and 11 digits. Then, it iterates through each character to ensure that all characters are digits. If any character is not a digit, the function returns false. If all conditions are satisfied, it returns true, indicating the phone number is valid.
- **isValidEmail(email):** This function checks whether the entered email address is in a valid format. It uses a regular expression (regex) pattern to match the standard format of an email address, which includes characters followed by an "@" symbol and a domain. If the email matches the pattern, the function returns true; otherwise, it returns false.

#### (c) Member Management Methods

- **handleMemberSignUp():** This method handles the login process for existing members. It repeatedly prompts the user to enter their member ID and password until a valid combination is provided. The method checks if the entered member ID exists in the correctMemberIDs array. If it does, the corresponding password is verified against the correctPasswords array. If both the ID and password match, the system prints a success message, including any promotional details, and sets the isAuthenticated flag to true, indicating a successful login. If the password is incorrect, the system displays an error message and prompts the user to try again. If the member ID does not exist, an error message is shown, and the user is prompted to re-enter their credentials.
- **handleNewMemberRegistration():** This method manages the registration process for new members. It prompts the user to enter their personal details, including name, phone number, and email address. For each piece of information, the corresponding validation function (e.g., isValidName, isValidPhoneNumber, isValidEmail) is called to ensure the input is in the correct format. If the validation fails, the system continues prompting the user until valid data is entered for all fields. Once all details are validated, the system prints the member's information along with a new member code, confirming the successful registration. This code can be used by the new member for future login or promotional purposes.

```

Struct Membership:
// Constants
NUM_MEMBERS = 3
correctMemberIDs = ["member1", "member2", "member3"]
correctPasswords = ["password1", "password2", "password3"]

// Member details
name
phoneNumber
email

// Method to validate name
Function is_valid_name(name):
    If name is empty then return false
    For each character in name:
        If character is not a letter and not a space then return false
    Return true

// Method to validate phone number
Function is_valid_phone_number(phoneNumber):
    If phoneNumber length is less than 10 or greater than 11 then return false
    For each character in phoneNumber:
        If character is not a digit then return false
    Return true

// Method to validate email
Function is_valid_email(email):
    Define regex pattern for valid email format
    Return true if email matches the pattern, otherwise false

// Handle member sign up process
Function handle_member_sign_up():
    Set isAuthenticated to false
    While not isAuthenticated:
        Print "Member login"
        Print separator line

        Prompt for Member ID and read input into inputMemberID
        Prompt for Password and read input into inputPassword

        Set idExists to false
        For each index from 0 to NUM_MEMBERS - 1:
            If inputMemberID matches correctMemberIDs[i]:
                Set idExists to true
            If inputPassword matches correctPasswords[i]:
                Print "Login successful!"
                Print promotional details
                Set isAuthenticated to true
                Break loop
        Else:

```

```
Print "Incorrect Password. Please try again."  
Break loop  
If idExists is false:  
    Print "Incorrect Member ID. Please try again."  
  
// Handle new member registration process  
Function handleNewMemberRegistration():  
    Print registration prompt  
    While true:  
        Prompt for Name and read input into name  
        If name is valid according to isValidName:  
            Break loop  
        Else:  
            Print "Invalid name. Please enter a name containing only letters and spaces."  
    While true:  
        Prompt for Phone number and read input into phoneNumber  
        If phoneNumber is valid according to isValidPhoneNumber:  
            Break loop  
        Else:  
            Print "Invalid phone number. It should be numeric and between 10 to 11 digits  
long."  
  
    While true:  
        Prompt for Email and read input into email  
        If email is valid according to isValidEmail:  
            Break loop  
        Else:  
            Print "Invalid email format. Please enter a valid email address."  
  
    Print member details and new member code
```

## 20. Display summary report

- (a) **Print Header:** The function starts by printing the location name followed by the word "Summary" to indicate which location's summary is being displayed.
- (b) **Print Decorative Line:** A line of = characters is printed for header decoration. The width is set to 43 characters to align with the later content formatting.
- (c) **Print Column Headers:** The column headers "Total Units" and "Total Amount" are printed inside a formatted row. The columns are defined with specific widths (25 characters for "Total Units" and 15 characters for "Total Amount").
- (d) **Print Separator Line:** A line of - characters is used as a separator between the headers and the actual data, maintaining alignment with the previous decorative line.
- (e) **Print Data Values:** The totalUnits and totalAmount values are printed in a formatted row, similar to the headers. The values are aligned within their respective columns (25 characters wide for totalUnits and 15 characters wide for totalAmount).
- (f) **Print Footer Line:** Finally, a line of = characters is printed to conclude the summary section, mirroring the initial decorative line.
- (g) **Reset Fill Character:** The fill character is reset to space to ensure that any subsequent output is formatted correctly.

```

START
Function displaySummary(totalUnits, totalAmount, location):
    // Print the location header
    Print location followed by " Summary"

    // Print a line of equal signs for header decoration
    Set fill character to '='
    Print a line of 43 characters width with '='

    // Print column headers for total units and total amount
    Set fill character to space
    Print a row with the following format:
        - A vertical bar
        - Left-aligned "Total Units" (25 characters wide)
        - A vertical bar
        - Right-aligned "Total Amount" (15 characters wide)
        - A vertical bar

    // Print a line of dashes as a separator
    Set fill character to '-'
    Print a line of 43 characters width with '-'

    // Print the values for total units and total amount
    Set fill character to space
    Print a row with the following format:
        - A vertical bar
        - Left-aligned totalUnits (25 characters wide)
        - A vertical bar
        - Right-aligned totalAmount (15 characters wide)
        - A vertical bar

    // Print a line of equal signs for footer decoration
    Set fill character to '='
    Print a line of 43 characters width with '='

    // Reset fill character to space
    Set fill character to space
End Function
END

```

### 21. Report

#### (a) Initialization

- Begin by initializing the totalunit variable to 0. This variable is crucial as it will be used to aggregate the total number of units from all the bookings made. By setting it to 0 at the start, you ensure that any calculations performed subsequently are based on a clean slate, avoiding any erroneous accumulation from previous operations.

#### (b) Initialize Totals for Each Location

- Set up predefined totals for each location, including Kuala Lumpur, Melaka, Penang, Selangor, Sabah, and Sarawak. This involves initializing arrays or variables to store the number of units booked and the total monetary amount for each location. These predefined totals act as placeholders for accumulating and storing booking data, which will be updated as new bookings are processed.

#### (c) Calculate Total Units from Bookings

- Iterate through the list of bookings, known as BookingList (BL). For each booking entry, extract the quantity of units booked and add this quantity to the totalunit variable. This step involves looping through each booking record to ensure that all booked units are included in the total. Accumulating these quantities helps in calculating the overall booking metrics for the current session.

#### (d) Update Location-Based Totals

- Use a switch statement to determine which location the current booking pertains to, based on the locationNo. Update the respective location's totals by adding the computed totalunit and grandtotal (the total amount from the bookings) to the totals for that location. This update ensures that each location's booking data is accurately reflected and maintained.

#### (e) Display Menu and Handle User Choice

- Initialize staffChoice to 0 and enter a loop to handle user input. Display a menu that offers options for generating reports for different locations and includes an option to exit the program. Read the user's choice and validate the input to ensure it falls within the acceptable range of options. Handle invalid inputs by clearing the input buffer and prompting the user to enter a valid choice. Based on the user's selection, use a switch statement to call the displaySummary function with the appropriate totals and location name, or print a message for exiting or invalid options. This step ensures that user interaction is handled smoothly and efficiently.

#### (f) Reset Booking Count

- After the loop ends (when the user chooses to exit), reset the BL.count to 0. This action clears the count of bookings, preparing the system for a fresh set of data in the next reporting session. Resetting the count is crucial for maintaining accurate and up-to-date booking records and preventing any carry-over effects from previous operations.

```
START
Function Report(locationNo, grandtotal, BL):
    Initialize totalunit to 0

    // Initialize totals for each location
    Set kualaLumpurTotalUnits to 78
    Set kualaLumpurTotalAmount to 41000
    Set melakaTotalUnits to 65
    Set melakaTotalAmount to 20400
    Set penangTotalUnits to 99
    Set penangTotalAmount to 50880
    Set selangorTotalUnits to 56
    Set selangorTotalAmount to 20580
    Set sabahTotalUnits to 101
    Set sabahTotalAmount to 33840
    Set sarawakTotalUnits to 77
    Set sarawakTotalAmount to 35600

    // Calculate the total units from bookings
    For each booking in BL.bookings:
        Add booking.quantity to totalunit

    // Update totals based on the location number
    Switch(locationNo):
        Case 1:
            Add totalunit to kualaLumpurTotalUnits
            Add grandtotal to kualaLumpurTotalAmount
        Case 2:
            Add totalunit to melakaTotalUnits
            Add grandtotal to melakaTotalAmount
        Case 3:
            Add totalunit to penangTotalUnits
            Add grandtotal to penangTotalAmount
        Case 4:
            Add totalunit to selangorTotalUnits
            Add grandtotal to selangorTotalAmount
        Case 5:
            Add totalunit to sabahTotalUnits
            Add grandtotal to sabahTotalAmount
        Case 6:
            Add totalunit to sarawakTotalUnits
            Add grandtotal to sarawakTotalAmount
        Default:

    Initialize staffChoice to 0
    Do:
        // Display the report menu
        Print "Hotel Report System"
        Print "1. Kuala Lumpur"
        Print "2. Melaka"
        Print "3. Penang"
```

```
Print "4. Selangor"
Print "5. Sabah"
Print "6. Sarawak"
Print "7. Exit"
Print "Enter your choice (1-7): "
Read staffChoice

// Validate input
If input is invalid or staffChoice is not between 1 and 7:
    Clear input buffer
    Ignore invalid input
    Print "Invalid choice. Please enter a number between 1 and 7."
    Continue to re-prompt the user

// Process choice
Switch(staffChoice):
    Case 1:
        Call displaySummary with kualaLumpurTotalUnits, kualaLumpurTotalAmount, "Kuala Lumpur"
    Case 2:
        Call displaySummary with melakaTotalUnits, melakaTotalAmount, "Melaka"
    Case 3:
        Call displaySummary with penangTotalUnits, penangTotalAmount, "Penang"
    Case 4:
        Call displaySummary with selangorTotalUnits, selangorTotalAmount, "Selangor"
    Case 5:
        Call displaySummary with sabahTotalUnits, sabahTotalAmount, "Sabah"
    Case 6:
        Call displaySummary with sarawakTotalUnits, sarawakTotalAmount, "Sarawak"
    Case 7:
        Print "Exiting program."

While staffChoice is not 7 // Continue until user selects 7 (Exit)

Set BL.count to 0 // Reset the booking count to ensure a clean slate for the next call
End Function
END
```

## 22. Feedback

- (a) **Initialization:** Define constants, maximum number of feedback entries that can be stored to limit the number of feedback entries to 5. Initialize arrays to store feedback information.
- (b) **Print Feedback:** Display existing feedback in a table.
- (c) **Rating Input Validation and Check Feedback Capacity:** Prompt user if they want to leave a rating. If yes, make sure the feedback length does not reach the maximum feedback capacity. Ensure the user enters a valid rating between 1 and 5 stars. If the input is invalid, it prompt the user to try again.
- (d) **Name and Feedback Collection:** After user entered a valid rating, then will collects the user's name and feedback.
- (e) **Store feedback and Display Updated Feedback:** Add new feedback to arrays and increments the feedback count. Reprints the existing feedback, with updated feedback.
- (f) **Exit:** Displays a message that thanks the user for their visit and encourage them to provide additional comments or suggestions later, then end the function.

```

START
FUNCTION displayFeedback
    DEFINE MAX_FEEDBACK AS 5
    INITIALIZE ratings[MAX_FEEDBACK] TO 0
    INITIALIZE names[MAX_FEEDBACK] AS EMPTY STRINGS
    INITIALIZE comments[MAX_FEEDBACK] AS EMPTY STRINGS
    INITIALIZE feedbackCount TO 0

    PRINT "Customer Feedback:"

    Display feedback table header

    Display preloaded feedback table

    REPEAT
        PRINT "Would you like to leave a rating? (y/n): "
        READ decision

        IF decision IS 'y' THEN
            IF feedbackCount IS GREATER THAN OR EQUAL TO MAX_FEEDBACK THEN
                PRINT "Sorry, feedback capacity has been reached. Unable to add more feedback."
                BREAK
            END IF

            INITIALIZE rating TO 0
            REPEAT
                PRINT "Please enter your rating from 1 - 5 stars: "
                READ rating

                IF rating IS NOT BETWEEN 1 AND 5 THEN
                    PRINT "Invalid input. Rating must be between 1 and 5 stars."
                    CLEAR INPUT
                ELSE
                    BREAK
                END IF
            END REPEAT
        END IF
    END REPEAT
END FUNCTION

```

```
UNTIL FALSE

PRINT "Please enter your name: "

CLEAR NEWLINE
READ name FROM USER
names[feedbackCount] = name

PRINT "Please enter your feedback: "
READ feedback FROM USER
comments[feedbackCount] = feedback

ratings[feedbackCount] = rating
feedbackCount = feedbackCount + 1

PRINT "Thank you for your feedback! Your feedback has been recorded."

// Display all feedback (including new ones)
PRINT "Current Feedback:"

    Display preloaded feedback table

// Display new feedback
FOR i FROM 0 TO feedbackCount - 1
DO
    PRINT "| " + FORMAT_TEXT(RATING_STRING(ratings[i]), 10) + " | "
        FORMAT_TEXT(names[i], 20) + " | " + FORMAT_TEXT(comments[i], 80) + " | "
END FOR

ELSE IF decision IS NOT 'n' THEN
    PRINT "Invalid input. Please enter 'y' for yes or 'n' for no."
END IF

UNTIL decision IS 'n'

PRINT "Thank you for visiting. If you have any comments or suggestions later, please don't
hesitate to share!"
END FUNCTION
END
```

## Additional functionalities

## **Check-In & Check-Out**

Before staff can proceed with the check-in and check-out processes, they must first log into the system to ensure secure access to booking information and management tools.

**Check-In:** During check-in, staff collect the booking number, customer name, and identification card (IC) from the guest. This step is essential for verifying the reservation and confirming that the provided details match the records in the system. The system update at this stage reflects the commencement of the booking, enabling staff to efficiently manage the customer's arrival. This process not only ensures that the guest's reservation is accurately recorded but also prepares the room and facilitates a smooth start to the guest's stay.

# Problem Solving and Programming

**Check-Out:** The check-out process completes the booking cycle by finalizing the customer's stay. Staff review the reservation details and manage any potential late check-out charges. If the guest departs later than the scheduled time, delay fees are calculated and applied as necessary. This procedure ensures that all financial aspects are properly addressed, from handling any additional charges to settling the final bill. By meticulously managing these details, the system ensures that the booking is properly concluded, maintaining accurate records and preparing the room for future guests.

## Feedback

The feedback function allows guests to leave reviews, providing hotels with valuable insights into their performance. When guests share their experiences, it helps hotels understand what they are doing well and where they might need to make improvements. Positive reviews can attract more guests and enhance the hotel's reputation by showcasing the high quality of service and experiences offered.

On the other hand, addressing negative feedback demonstrates the hotel's commitment to resolving issues and improving guest satisfaction. By responding to complaints and making necessary changes, hotels show that they care about their guests and are dedicated to providing a better experience. This responsiveness can turn a negative situation into a positive one and build stronger relationships with guests.

Furthermore, thoughtful responses to reviews can highlight what sets a hotel apart from its competitors, showcasing unique features or exceptional service. By paying attention to and acting on guest feedback, hotels can better manage their bookings, enhance their services, and keep guests happy. This continuous improvement helps the hotel stand out in a competitive market, leading to sustained success and growth.

Welcome to Elegant Hideaway, select an option to proceed

---

Menu List

---

1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit

Your option: 5

You selected Feedback

Customer Feedback:

Rating	Name	Feedback
**	JOHN TERRY	The property's location is conveniently close to the Sky Arena Sports Complex. However, there are no nearby options for food or convenience stores within walking distance. The cleanliness left much to be desired.
***	KLAY NICK	Overall ok. Need to improve on room cleanliness. Aircond sejuk. Got toiletries, towel and mini pantry. Nearby a lot of shopping malls also.
****	CHAN	Staff helpful and friendly. Location ok but stay on 1st floor bit noisy earlier, but ok after midnight.
*	SAM	Very bad experience and not recommended.
*****	KANE	Very nice, will come again next time.
**	AUNE	Sound proof not good. Closed door can hear it clearly and window curtain bit dirty. Overall is ok. Have to give extra attention on room cleanliness.

Would you like to leave a rating? (y/n): y  
 Please enter your rating from 1 - 5 stars: 5  
 Please enter your name: pretty~  
 Please enter your feedback: Friendly staff, nice environment, worth my money ;)

Thank you for your feedback! Your feedback has been recorded.

Current Feedback:

Rating	Name	Feedback
**	JOHN TERRY	The property's location is conveniently close to the Sky Arena Sports Complex. However, there are no nearby options for food or convenience stores within walking distance. The cleanliness left much to be desired.
***	KLAY NICK	Overall ok. Need to improve on room cleanliness. Aircond sejuk. Got toiletries, towel and mini pantry. Nearby a lot of shopping malls also.
****	CHAN	Staff helpful and friendly. Location ok but stay on 1st floor bit noisy earlier, but ok after midnight.
*	SAM	Very bad experience and not recommended.
*****	KANE	Very nice, will come again next time.
**	AUNE	Sound proof not good. Closed door can hear it clearly and window curtain bit dirty. Overall is ok. Have to give extra attention on room cleanliness.
*****	pretty~	Friendly staff, nice environment, worth my money ;)

Would you like to leave a rating? (y/n): n  
 Thank you for visiting. If you have any comments or suggestions later, please don't hesitate to share!

## Cancellation

The cancellation feature enables customers to cancel bookings or service requests, with the system managing cancellations by updating booking statuses and processing any applicable refunds.

A flexible cancellation policy allows guests to change their plans without losing money, which makes them more willing to book a room. Knowing they won't face financial penalties if their plans change gives guests peace of mind, making them more likely to commit to a reservation.

For hotels, this policy has several advantages. First, it helps increase the number of bookings because guests are more confident in making a reservation when they know they have the option to cancel if needed. If a guest does cancel, the hotel can quickly fill the newly available room with another booking, which helps maintain steady revenue and reduces the risk of empty rooms.

Moreover, a flexible cancellation policy helps build trust between the hotel and its guests. When guests feel that their needs and concerns are being taken into account, they are more likely to have a positive experience. This can lead to repeat visits and positive reviews, which are valuable for attracting future guests. Overall, a flexible cancellation policy not only supports financial stability but also fosters stronger relationships with customers.

## Program Testing & Output

**Scenario 1:** Three customers select from a menu list to order a room, without using any vouchers or add-on services (including validation checks). If a customer makes an invalid selection, the system should repeat the prompt and ask the user to choose again.

® = Repeat and ask the user again if the input is invalid.

### 1<sup>st</sup> Order

User Inputs	
Location Area	a
Location Area ®	1
Number of Pax Adult	2
Number of Pax Children	p
Number of Pax Children ®	1
Number of Pax Infants	1
Start Month	2
End Month	2
Start Year	2024
End Year	2024
Check-In Date	5
Check-Out Date	12
Room Number	5
Quantity	2
Name	Yu8888
Name ®	Yu
Phone Number	017-88838888
Phone Number ®	01788838888

Expected Result / Outputs	
Validation Check ( Location Area)	Error: Please enter a number between 1 and 6.
Location Area	You have selected Kuala Lumpur
Validation Check (Number Of Pax Children)	Error: Please enter a valid number.
Total Number of Pax	$2+1+1=4$ <b>Output:</b> The total number of pax is 4
<b>Display The February Calendar in 2024</b>	
Number Of Days	Number of days = (End Date – Start Date) + 1 Number of days = (12–5) +1= 8 <b>Output:</b> Number of booking day is 8
Check-In Date	5/2/2024
Check-Out Date	12/2/2024
<b>Display the Kuala Lumpur Table</b>	
Validation Check (Name)	Invalid name : Please try again.
Validation Check (Phone Number)	Error: Please enter a valid phone number consisting only of digits.
<b>Display Overview of Bookings with auto generated booking number 1002</b>	
Total Price	Total Price = Price Per Room * Quantity Total Price = 450.00 * 2 Total Price = 900.00
Total before Tax	Total before Tax = Total Price * Number Of Days Total before Tax = 900.00 * 8 Total before Tax = 7200.00

## Problem Solving and Programming

---

<b>User Inputs</b>	
Add On Service (Y/N)	1
Add On Service (Y/N) ®	n
Voucher Code (Y/N)	n
Payment Method	6
Payment Method ®	1
Card Number	22222222
PIN	77
PIN ®	778855
Continue Order (Y/N)	N

<b>Expected Result / Outputs</b>	
<b>Validation Check ( Add On Service)</b>	Invalid input. Please enter 'Y' for Yes or 'N' for No
<b>Display Invoice</b>	
<b>SST</b>	$\text{SST} = \text{Total before Tax} * \text{SSTRATE (0.06)}$ $\text{SST} = 7200 * 0.06$ $\text{SST} = 432.00$
<b>Add On Service</b>	NO EXTRA SERVICE / RM 0.00
<b>Voucher Code</b>	NO VOUCHER / RM 0.00
<b>Grand Total</b>	$\text{Grand Total} = \text{Total before Tax} + \text{SST} + \text{Service Price} - \text{Discount}$ $\text{Grand Total} = 7200 + 432 + 0 - 0$ $\text{Grand Total} = 7632.00$
<b>Validation Check (Payment Method)</b>	Error: Please enter a number between 1 and 5.
<b>Validation Check (PIN)</b>	Error: Please enter a valid PIN number with 4 to 6 digits.
<b>Display the Booking Confirmation and Payment Summary</b>	

# Problem Solving and Programming

## **1<sup>st</sup> Order Actual Output**

# Problem Solving and Programming

---

```

Please enter your information below
-----
Name : Yu8888
Invalid name : Please try again.
Name : Yu
Phone Number : 017-88838888
Error: Please enter a valid phone number consisting only of digits.
Phone Number : 01788838888

=====
|===== Overview of Bookings =====|
|=====|
|Name :Yu Check-In Date :5 /2 /2024
|Phone Number :01788838888 Check-Out Date :12/2 /2024
|Total number of pax :4 Number Of Days :8
|=====|
|BookingNo Room Number Hotel Quantity Price Per Room Total Price
|=====|
| 1002 5 Panorama Hotel 2 RM 450.00 RM 900.00
|Services: Wifi, Breakfast & Pool
|=====|
|Total before taxes RM 7200.00
|=====|
Do you need any add on service (Y/N): 1
Invalid input. Please enter 'Y' for Yes or 'N' for No
Do you need any add on service (Y/N): n
Do you want to apply voucher code (Y/N) :n
NO VOUCHER
=====
|===== Invoice =====|
|=====|
|Name :Yu Check-In Date :5 /2 /2024
|Phone Number :01788838888 Check-Out Date :12/2 /2024
|Total number of pax :4 Number Of Days :8
|=====|
|BookingNo Room Number Hotel Quantity Price Per Room Total Price
|=====|
| 1002 5 Panorama Hotel 2 RM 450.00 RM 900.00
|Services: Wifi, Breakfast & Pool
|=====|
|Total before taxes RM 7200.00
|SST RM 432.00
|NO EXTRA SERVICE RM 0.00
|NO VOUCHER -RM 0.00
|=====|
|Grand Total RM 7632.00
|=====|
=====
| Payment Method |
|-----|
| 1. Credit/Debit Cards |
| 2. Bank Transfer |
| 3. Touch N Go |
| 4. Apple Pay |
| 5. Cash Payment |
|-----|
Please select payment method (1-5) :6
Error: Please enter a number between 1 and 5.
Please select payment method (1-5) :1
Card Number :22222222
PIN :77
Error: Please enter a valid PIN number with 4 to 6 digits.
PIN :778855
=====
|===== Elegant Hideaway Airbnb =====|
|eleganthideaway@gmail.com +6018-889 8888 Elegant Kuala Lumpur www.eleganthideaway.com
|=====|
|Booking Confirmation
|=====|
|Dear,Yu
|Your hotel booking is confirmed.
|Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit
|Your booking details are as follow :
|=====|
|Name :Yu Check-In Date :5 /2 /2024
|Phone Number :01788838888 Check-Out Date :12/2 /2024
|Total number of pax :4 Number Of Days :8
|=====|
|BookingNo Room Number Hotel Quantity Price Per Room Total Price
|=====|
| 1002 5 Panorama Hotel 2 RM 450.00 RM 900.00
|Services: Wifi, Breakfast & Pool
|=====|
|Payment Summary : |
|=====|
|Total before taxes RM 7200.00
|SST RM 432.00
|NO EXTRA SERVICE RM 0.00
|NO VOUCHER -RM 0.00
|=====|
|Grand Total RM 7632.00
|=====|
|Payment Method : Credit/Debit Cards
|Payment Amount : RM 7632.00
|Change : RM 0.00
|=====|
|Additional Information : Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.
|If you have any questions, please don't hesitate to contact us.
|We hope you enjoy your stay with us !!!
|=====|
Do you want to continue order? (Y/N): N
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |

```

## 2<sup>nd</sup> Order

User Inputs	
Location Area	2
Number of Pax Adult	-
Number of Pax Adult ®	1
Number of Pax Children	1
Number of Pax Infants	11
Number of Pax Infants®	0
Start Month	15
Start Month ®	4
End Month	k
End Month ®	5
Start Year	2024
End Year	2024
Check-In Date	31
Check-In Date ®	30
Check-Out Date	8
Room Number	4
Quantity	100
Quantity ®	2
Name	Lisa Wang
Phone Number	01266688990

Expected Result / Outputs	
Location Area	You have selected Melaka
Validation Check (Number of Pax Adult)	Error: Please enter a valid number.
Validation Check (Number of Pax Infants)	Error: Please enter a valid number.
Total Number of Pax	$1+1+0=2$ <b>Output:</b> The total number of pax is 2
Validation Check (Start Month)	Invalid month. Please enter a number between 1 - 12
Validation Check (End Month)	Invalid month. Please enter a number between 1 - 12
<b>Display the April Calendar in 2024 for Check In Date</b>	
<b>Display the May Calendar in 2024 Check Out Date</b>	
Validation Check (Check-In Date)	Not available date. Please enter a number between 1 - 30
Number Of Days	Number of Days = (Day Number of End Date) – (Day Number of Start Date) + 1 Number of Days = $(128 - 120) + 1 = 9$ <b>Output :</b> Number of your booking day is 9.
Check-In Date	30/4/2024
Check-Out Date	8/5/2024
<b>Display the Melaka Table</b>	
Validation Check (Quantity)	Sorry! The maximum number of rooms that can be booked is 10, and the minimum is 1.
<b>Display Overview of Bookings with auto generated booking number 1003</b>	
Total Price	Total Price = Price Per Room * Quantity Total Price = $500.00 * 2$ Total Price = 1000.00
Total before Tax	Total before Tax = Total Price * Number Of Days Total before Tax = $1000.00 * 9$ Total before Tax = 9000.00

## Problem Solving and Programming

---

<b>User Inputs</b>	
Add On Service (Y/N)	N
Voucher Code (Y/N)	a
Voucher Code (Y/N) ®	n
Payment Method	5
Payment Amount	8000
Payment Amount ®	9600
Continue Order (Y/N)	j
Continue Order (Y/N) ®	n

<b>Expected Result / Outputs</b>	
<b>Validation Check (Voucher Code)</b>	Invalid input. Please enter 'Y' for Yes or 'N' for No
<b>Display Invoice</b>	
<b>SST</b>	$SST = \text{Total before Tax} * \text{SSTRATE (0.06)}$ $SST = 9000 * 0.06$ $SST = 540.00$
<b>Add On Service</b>	NO EXTRA SERVICE / RM 0.00
<b>Voucher Code</b>	NO VOUCHER / RM 0.00
<b>Grand Total</b>	$\text{Grand Total} = \text{Total before Tax} + \text{SST} + \text{Service Price} - \text{Discount}$ $\text{Grand Total} = 9000 + 540 + 0 - 0$ $\text{Grand Total} = 9540.00$
<b>Validation Check (Payment Amount)</b>	Insufficient amount. Please make payment again.
<b>Change</b>	$\text{Change} = \text{Payment Amount} - \text{Grand Total}$ $\text{Change} = 9600 - 9540$ $\text{Change} = 60.00$
<b>Display the Booking Confirmation and Payment Summary</b>	
<b>Validation Check (Continue Order)</b>	Invalid input. Please enter 'Y' for Yes or 'N' for No:

# Problem Solving and Programming

---

## 2<sup>nd</sup> Order Actual Output

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 1
You selected Order Room
Welcome to Elegant Hideaway Airbnb
Please let us know the location you wish to travel to at your convenience
-----
1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
Please enter the number of the location: 2
You have selected Melaka.
-----
Please enter number of pax adult (13 years and older) :-
Error: Please enter a valid number.
Please enter number of pax adult (13 years and older) :1
Please enter number of pax children (2-12 years old) :1
Please enter number of pax infants (under 2 years old) :1
Error: Please enter a valid number.
Please enter number of pax infants (under 2 years old) :0
-----
The total number of pax is 2
Select your Check-In & Check-Out Date
-----
| Month | Year |
=====
| 1 2 3 4 5 6 | 2024 |
| 7 8 9 10 11 12 | 2025 |
=====
Start Month: 15
Invalid month. Please enter a number between 1 - 12
Start Month: 4
End Month: k
Invalid month. Please enter a number between 1 - 12
End Month: 5
Start Year: 2024
End Year: 2024
Select your Check-In Date, __/__/2024
-----
| Date |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 |
=====
Check-In Date: 31
Not available date. Please enter a number between 1 - 30
Check-In Date: 30
30/4/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/__/2024
-----
| Date |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 31 |
=====
Check-Out Date: 8
8/5/2024, Check-Out Date selected successfully.

Number of your booking day is 9.
Check-In Date 30/4/2024, Check-Out Date 8/5/2024 selected successfully.

Melaka
-----
| No | Name | Location | Price | Bed Type | Room Type | Room Size | Service |
=====
| 1 | Heritage Haven | 45, Jalan Hang Jebat, 75200 | RM 300.00 | Queen | Superior | 25-35 sqm | WiFi Breakfast |
| | Melaka | | | | | | |
| 2 | Melaka River Inn | 10, Jalan Laksamana, 75000 | RM 250.00 | King | Deluxe | 20-30 sqm | WiFi Breakfast Shuttle Service |
| | Melaka | | | | | | |
| 3 | Coastal Retreat | 8, Jalan Kampung, 45500 | RM 350.00 | Double | Suite | 30-40 sqm | WiFi Breakfast Gym Spa |
| | Melaka | | | | | | |
| 4 | Grand Melaka Resort | 100, Jalan Melaka Raya, 75000 | RM 500.00 | Queen | Executive | 35-45 sqm | WiFi Breakfast Lounge Business Center |
| | Melaka | | | | | | |
| 5 | Historical Lodge | 23, Jalan Merdeka, 75000 | RM 200.00 | Single | Standard | 20-30 sqm | WiFi Breakfast Laundry Service |
| | Melaka | | | | | | |
=====
```

# Problem Solving and Programming

```

Room Number (1-5) :4
Quantity :100
Sorry! The maximum number of rooms that can be booked is 10, and the minimum is 1.
Quantity :2
Please enter your information below
-----
Name : Lisa Wang
Phone Number : 01266688990
|=====
|===== Overview of Bookings =====|
|=====|
|Name :Lisa Wang Check-In Date :30/4 /2024
|Phone Number :01266688990 Check-Out Date :8 /5 /2024
|Total number of pax :2 Number Of Days :9
|=====|
|BookingNo Room Number Hotel Quantity Price Per Room Total Price
|=====|
| 1003 4 Grand Melaka Resort 2 RM 500.00 RM 1000.00
|Services: Wifi, Breakfast, Lounge & Business Center
|=====|
|Total before taxes RM 9000.00
|=====|
Do you need any add on service (Y/N): N
Do you want to apply voucher code (Y/N) :a
Invalid input. Please enter 'Y' for Yes or 'N' for No
Do you want to apply voucher code (Y/N) :n
NO VOUCHER
|=====
|===== Invoice =====|
|=====|
|Name :Lisa Wang Check-In Date :30/4 /2024
|Phone Number :01266688990 Check-Out Date :8 /5 /2024
|Total number of pax :2 Number Of Days :9
|=====|
|BookingNo Room Number Hotel Quantity Price Per Room Total Price
|=====|
| 1003 4 Grand Melaka Resort 2 RM 500.00 RM 1000.00
|Services: Wifi, Breakfast, Lounge & Business Center
|=====|
|Total before taxes RM 9000.00
|SST RM 540.00
|NO EXTRA SERVICE RM 0.00
|NO VOUCHER -RM 0.00
|=====|
|Grand Total RM 9540.00
|=====|
|===== Payment Method =====|
|=====|
| 1. Credit/Debit Cards
| 2. Bank Transfer
| 3. Touch N Go
| 4. Apple Pay
| 5. Cash Payment
|=====|
Please select payment method (1-5) :5
Payment Amount :8000
Insufficient amount. Please make payment again.
Payment Amount :9600
|=====
|===== Elegant Hideaway Airbnb =====|
|eleganthideaway@gamil.com +6018-889 8888 Elegant Kuala Lumpur www.eleganthideaway.com
|=====|
|Booking Confirmation =====|
|Dear,Lisa Wang
|Your hotel booking is confirmed.
|Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit
|Your booking details are as follow :
|=====
|Name :Lisa Wang Check-In Date :30/4 /2024
|Phone Number :01266688990 Check-Out Date :8 /5 /2024
|Total number of pax :2 Number Of Days :9
|=====|
|BookingNo Room Number Hotel Quantity Price Per Room Total Price
|=====|
| 1003 4 Grand Melaka Resort 2 RM 500.00 RM 1000.00
|Services: Wifi, Breakfast, Lounge & Business Center
|=====|
|Payment Summary :
|=====|
|Total before taxes RM 9000.00
|SST RM 540.00
|NO EXTRA SERVICE RM 0.00
|NO VOUCHER -RM 0.00
|=====|
|Grand Total RM 9540.00
|=====|
|Payment Method : Cash Payment
|Payment Amount : RM 9600.00
|Change : RM 60.00
|=====|
|Additional Information :
|Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.
|If you have any questions, please don't hesitate to contact us.
|We hope you enjoy your stay with us !!!
|=====|
Do you want to continue order? (Y/N): j
Invalid input. Please enter 'Y' for Yes or 'N' for No: n
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |

```

**3<sup>rd</sup> Order**

<b>User Inputs</b>	
<b>Location Area</b>	3
<b>Number of Pax Adult</b>	2
<b>Number of Pax Children</b>	3
<b>Number of Pax Infants</b>	0
<b>Start Month</b>	6
<b>End Month</b>	4
<b>Start Year</b>	2024
<b>End Year</b>	2024
<b>Check-In Date</b>	6
<b>Check-Out Date</b>	4
<b>Start Month</b>	4
<b>End Month</b>	6
<b>Start Year</b>	2024
<b>End Year</b>	2024
<b>Check-In Date</b>	4
<b>Check-Out Date</b>	6
<b>Room Number</b>	0
<b>Room Number ®</b>	1
<b>Quantity</b>	1
<b>Name</b>	Fish Yu
<b>Phone Number</b>	0178993858

<b>Expected Result / Outputs</b>	
<b>Location Area</b>	You have selected Penang
<b>Total Number of Pax</b>	$2+3+0=5$ <b>Output:</b> The total number of pax is 5
<b>Display the June Calendar in 2024 for Check In Date</b>	
<b>Display the April Calendar in 2024 Check Out Date</b>	
<b>Validation Check (Number of Days)</b>	Number of Days = (Day Number of End Date) – (Day Number of Start Date) + 1 Number of Days = $(95-158) + 1 = -62$ Number of your booking day is -62. Invalid date range. Please select your Check-In & Check-Out Date again
<b>Display the April Calendar in 2024 for Check In Date</b>	
<b>Display the June Calendar in 2024 Check Out Date</b>	
<b>Number Of Days</b>	Number of Days = (Day Number of End Date) – (Day Number of Start Date) + 1 Number of Days = $(158-95) + 1 = 64$ <b>Output:</b> Number of your booking day is 64.
<b>Check-In Date</b>	4/4/2024
<b>Check-Out Date</b>	6/6/2024
<b>Display the Penang Table</b>	
<b>Validation Check (Room Number)</b>	Invalid number. Please select a valid room number (1-5) :
<b>Display Overview of Bookings with auto generated booking number 1004</b>	
<b>Total Price</b>	Total Price = Price Per Room * Quantity Total Price = $300.00 * 1$ Total Price = 300.00
<b>Total before Tax</b>	Total before Tax = Total Price * Number Of Days Total before Tax = $300.00 * 64$ Total before Tax = 19200.00

## Problem Solving and Programming

---

<b>User Inputs</b>	
Add On Service (Y/N)	n
Voucher Code (Y/N)	n
Payment Method	3
PIN	889955
Continue Order (Y/N)	n

<b>Expected Result / Outputs</b>	
<b>Validation Check (Voucher Code)</b>	Invalid input. Please enter 'Y' for Yes or 'N' for No
<b>Display Invoice</b>	
<b>SST</b>	$SST = \text{Total before Tax} * \text{SSTRATE (0.06)}$ $SST = 19200 * 0.06$ $SST = 1152.00$
<b>Add On Service</b>	NO EXTRA SERVICE / RM 0.00
<b>Voucher Code</b>	NO VOUCHER / RM 0.00
<b>Grand Total</b>	$\text{Grand Total} = \text{Total before Tax} + SST + \text{Service Price} - \text{Discount}$ $\text{Grand Total} = 19200 + 1152 + 0 - 0$ $\text{Grand Total} = 20352.00$
<b>Display the Booking Confirmation and Payment Summary</b>	

## **3rd Order Actual Output**

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 1
You selected Order Room
Welcome to Elegant Hideaway Airbnb
Please let us know the location you wish to travel to at your convenience
-----
1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
Please enter the number of the location: 3
You have selected Penang.
-----
Please enter number of pax adult (13 years and older) :2
Please enter number of pax children (2-12 years old) :3
Please enter number of pax infants (under 2 years old) :0
-----
The total number of pax is 5
-----
Select your Check-In & Check-Out Date
=====
|       Month      |   Year   |
=====
| 1 2 3 4 5 6 | 2024   |
| 7 8 9 10 11 12 | 2025   |
=====
Start Month: 6
End Month: 4
Start Year: 2024
End Year: 2024
Select your Check-In Date, __/6/2024
=====
|           Date        |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 31 |
=====
Check-In Date: 6
6/6/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/4/2024
=====
|           Date        |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 |
=====
Check-Out Date: 4
4/4/2024, Check-Out Date selected successfully.
Number of your booking day is -62.
Invalid date range. Please select your Check-In & Check-Out Date again
Select your Check-In & Check-Out Date
=====
|       Month      |   Year   |
=====
| 1 2 3 4 5 6 | 2024   |
| 7 8 9 10 11 12 | 2025   |
=====
Start Month: 4
End Month: 6
Start Year: 2024
End Year: 2024
Select your Check-In Date, __/4/2024
=====
|           Date        |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 |
=====
Check-In Date: 4
4/4/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/6/2024
=====
|           Date        |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 |
=====
Check-Out Date: 6
6/6/2024, Check-Out Date selected successfully.
```

# Problem Solving and Programming

Number of your booking day is 64.  
Check-In Date 4/4/2024, Check-Out Date 6/6/2024 selected successfully.

Penang

No	Name	Location	Price	Bed Type	Room Type	Room Size	Service
1	Bayview Grand Hotel   Melaka	45, Jalan Hang Jebat, 75200     Penang	RM 300.00	Queen	Superior	25-35 sqm	WiFi Breakfast
2	Georgetown Inn	456, Jalan Beach, 10200     Penang	RM 250.00	Queen	Standard	25-35 sqm	WiFi Breakfast Shuttle Service
3	Seaside Resort	789, Jalan Seaside, 10300     Penang	RM 400.00	Double	Suite	35-45 sqm	WiFi Breakfast Gym Spa
4	Penang Heritage Lodge	101, Jalan Heritage, 10400     Penang	RM 300.00	Queen	Superior	20-30 sqm	WiFi Breakfast Lounge
5	Bayview Penang	202, Jalan Bayview, 10500     Melaka	RM 500.00	King	Executive	40-50 sqm	WiFi Breakfast Business Center

Room Number (1-5) : 0  
Invalid number. Please select a valid room number (1-5) : 1  
Quantity : 1  
Please enter your information below

Name : Fish Yu  
Phone Number : 0178993858

Overview of Bookings						
Name : Fish Yu	Check-In Date : 4 / 4 / 2024					
Phone Number : 0178993858	Check-Out Date : 6 / 6 / 2024					
Total number of pax : 5	Number Of Days : 64					
BookingNo Room Number Hotel Quantity Price Per Room Total Price						
1004 1 Bayview Grand Hotel 1 RM 300.00 RM 300.00						
Services: WiFi & Breakfast						
Total before taxes RM 19200.00						

Do you need any add on service (Y/N): n  
Do you want to apply voucher code (Y/N) : n  
NO VOUCHER

Invoice						
Name : Fish Yu	Check-In Date : 4 / 4 / 2024					
Phone Number : 0178993858	Check-Out Date : 6 / 6 / 2024					
Total number of pax : 5	Number Of Days : 64					
BookingNo Room Number Hotel Quantity Price Per Room Total Price						
1004 1 Bayview Grand Hotel 1 RM 300.00 RM 300.00						
Services: WiFi & Breakfast						
Total before taxes RM 19200.00						
SST RM 1152.00						
NO EXTRA SERVICE RM 0.00						
NO VOUCHER -RM 0.00						
Grand Total RM 20352.00						

Payment Method	
1. Credit/Debit Cards	
2. Bank Transfer	
3. Touch N Go	
4. Apple Pay	
5. Cash Payment	

Please select payment method (1-5) : 3  
PIN : 889955

Elegant Hideaway Airbnb			
eleghanthideaway@gmail.com	+6018-889 8888	Elegant Kuala Lumpur	www.eleganthideaway.com
Booking Confirmation			
Dear, Fish Yu			
Your hotel booking is confirmed.			
Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit			
Your booking details are as follow :			
Name : Fish Yu Check-In Date : 4 / 4 / 2024			
Phone Number : 0178993858 Check-Out Date : 6 / 6 / 2024			
Total number of pax : 5 Number Of Days : 64			
BookingNo Room Number Hotel Quantity Price Per Room Total Price			
1004 1 Bayview Grand Hotel 1 RM 300.00 RM 300.00			
Services: WiFi & Breakfast			
Payment Summary :			
Total before taxes RM 19200.00			
SST RM 1152.00			
NO EXTRA SERVICE RM 0.00			
NO VOUCHER -RM 0.00			
Grand Total RM 20352.00			
Payment Method : Touch N Go			
Payment Amount : RM 20352.00			
Change : RM 0.00			
Additional Information : Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.			
If you have any questions, please don't hesitate to contact us.			
We hope you enjoy your stay with us !!!			

Do you want to continue order? (Y/N): n

## Problem Solving and Programming

---

```
Welcome to Elegant Hideaway, select an option to proceed
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

**Scenario 2:** Two customers select from a menu list to order a room, with options to use vouchers as a member, as a new registered member, or without any add-on services (including validation checks). If a customer makes an invalid selection, the system should repeat the prompt and ask the user to choose again.

® = Repeat and ask the user again if the input is invalid.

### 1<sup>st</sup> Order (New Registered Member)

User Inputs	
Location Area	4
Number of Pax Adult	5
Number of Pax Children	0
Number of Pax Infants	0
Start Month	9
End Month	9
Start Year	2026
Start Year ®	2025
End Year	2025
Check-In Date	10
Check-Out Date	16
Room Number	3
Quantity	5
Name	Asha
Phone Number	0168908585

Expected Result / Outputs	
Location Area	You have selected Selangor
Total Number of Pax	$5+0+0=5$ <b>Output:</b> The total number of pax is 5
Validation Check (Start Year)	Not available year. Please enter 2024 or 2025
<b>Display the September Calendar in 2025</b>	
Number Of Days	Number of Days = (Day Number of End Date) – (Day Number of Start Date) + 1 Number of Days = $(16-10) + 1 = 7$ <b>Output:</b> Number of your booking day is 7.
Check-In Date	10/9/2025
Check-Out Date	16/9/2025
<b>Display the Selangor Table</b>	
<b>Display Overview of Bookings with auto generated booking number 1005</b>	
Total Price	Total Price = Price Per Room * Quantity Total Price = $480.00 * 5$ Total Price = 2400.00
Total before Tax	Total before Tax = Total Price * Number Of Days Total before Tax = $2400.00 * 64$ Total before Tax = 16800.00

## Problem Solving and Programming

---

User Inputs	
Add On Service (Y/N)	n
Voucher Code (Y/N)	Y
Membership (1 or 2)	y
Membership (1 or 2) ®	2
Name	asha5588
Name ®	asha
Phone Number	0168908585
Email	asha5588
Email ®	asha5588@gmail.com
Voucher Code	welcome123
Voucher Code ®	WELCOME123
Payment Method	4
PIN	7070
Continue Order (Y/N)	n

Expected Result / Outputs	
<b>Validation Check (Membership)</b>	Invalid choice. Please enter 1 or 2.
<b>Validation Check (Name)</b>	Invalid name. Please enter a name containing only letters and spaces.
<b>Validation Check (Email)</b>	Invalid email format. Please enter a valid email address.
<b>Display member details and provide new member code WELCOME123</b>	
<b>Validation Check (Voucher Code)</b>	Invalid voucher code. Please try again.
<b>Display Invoice</b>	
<b>SST</b>	$SST = \text{Total before Tax} * SSTRATE (0.06)$ $SST = 16800.00 * 0.06$ $SST = 1008.00$
<b>Add On Service</b>	NO EXTRA SERVICE / RM 0.00
<b>Voucher Code</b>	WELCOME123 (20% discount) $\text{Discount} = 16800 * 0.20$ $\text{Discount} = 3360.00$
<b>Grand Total</b>	$\text{Grand Total} = \text{Total before Tax} + SST + \text{Service Price} - \text{Discount}$ $\text{Grand Total} = 16800.00 + 1008.00 + 0 - 3360.00$ $\text{Grand Total} = 14448.00$
<b>Display the Booking Confirmation and Payment Summary</b>	

# Problem Solving and Programming

---

## **1<sup>st</sup> Order Actual Output (New Registered Member)**

```
Welcome to Elegant Hideaway, select an option to proceed
-----  

Menu List  

-----  

1. Order Room  

2. Check-In & Check-Out (Staff)  

3. Cancellation  

4. Report (Staff)  

5. Feedback  

6. Exit  

-----  

Your option: 1  

You selected Order Room  

Welcome to Elegant Hideaway Airbnb  

Please let us know the location you wish to travel to at your convenience  

-----  

1. Kuala Lumpur  

2. Melaka  

3. Penang  

4. Selangor  

5. Sabah  

6. Sarawak  

Please enter the number of the location: 4  

You have selected Selangor.  

-----  

Please enter number of pax adult (13 years and older) :5  

Please enter number of pax children (2-12 years old) :0  

Please enter number of pax infants (under 2 years old) :0  

-----  

The total number of pax is 5  

Select your Check-In & Check-Out Date  

=====  

| Month | Year |  

=====  

| 1 2 3 4 5 6 | 2024 |  

| 7 8 9 10 11 12 | 2025 |  

=====  

Start Month: 9  

End Month: 9  

Start Year: 2026  

Not available year. Please enter 2024 or 2025  

Start Year: 2025  

End Year: 2025  

Select your Check-In Date, ___/9/2025  

=====  

| Date |  

=====  

| 1 2 3 4 5 6 7 |  

| 8 9 10 11 12 13 14 |  

| 15 16 17 18 19 20 21 |  

| 22 23 24 25 26 27 28 |  

| 29 30 |  

=====  

Check-In Date: 10  

10/9/2025, Check-In Date selected successfully.  

Select your Check-Out Date, ___/9/2025  

=====  

| Date |  

=====  

| 1 2 3 4 5 6 7 |  

| 8 9 10 11 12 13 14 |  

| 15 16 17 18 19 20 21 |  

| 22 23 24 25 26 27 28 |  

| 29 30 |  

=====  

Check-Out Date: 16  

16/9/2025, Check-Out Date selected successfully.  

Number of your booking day is 7.  

Check-In Date 10/9/2025, Check-Out Date 16/9/2025 selected successfully.  

Selangor  

=====  

| No | Name | Location | Price | Bed Type | Room Type | Room Size | Service |  

|---|---|---|---|---|---|---|---|  

| 1 | Metro Plaza Hotel | 45, Jalan Metro, 40000 Shah Alam, Selangor | RM 420.00 | King | Executive | 40-50 sqm | WiFi Gym Spa Lounge |  

|---|---|---|---|---|---|---|---|  

| 2 | City View Resort | 18, Jalan Sunset, 40100 Petaling Jaya, Selangor | RM 550.00 | Queen | Deluxe | 35-45 sqm | WiFi Dinner Free Parking Airport Shuttle |  

|---|---|---|---|---|---|---|---|  

| 3 | Elite Comfort Hotel | 15, Jalan Elite, 40200 Subang Jaya, Selangor | RM 480.00 | Double | Suite | 50-60 sqm | WiFi Breakfast Conference Room Laundry Service |  

|---|---|---|---|---|---|---|---|  

| 4 | Urban Haven Hotel | 20, Jalan Urban, 40300 Klang, Selangor | RM 390.00 | Queen | Premier | 30-40 sqm | WiFi Breakfast Afternoon Tea Business Center |  

|---|---|---|---|---|---|---|---|  

| 5 | Selangor Grand | 25, Jalan Grand, 40400 Shah Alam, Selangor | RM 350.00 | Single | Superior | 30-35 sqm | WiFi Breakfast 24Hour Front Desk Lounge |  

=====  

Room Number (1-5) :3  

Quantity :5  

Please enter your information below  

-----  

Name : Asha  

Phone Number : 0168908585
```

# Problem Solving and Programming

```

=====
=====Overview of Bookings=====
=====
Name :Asha
Phone Number :0168908585
Total number of pax :5
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1005 3 Elite Comfort Hotel 5 RM 480.00 RM 2400.00
Services: Wifi, Breakfast, Conference Room & Laundry Service
=====
Total before taxes RM 16800.00
=====

Do you need any add on service (Y/N): n
Do you want to apply voucher code (Y/N) :Y
Are you a member?
-----
1. Yes
2. No
Please enter your choice (1 or 2): y
Invalid choice. Please enter 1 or 2.
Are you a member?
-----
1. Yes
2. No
Please enter your choice (1 or 2): 2
Please fill up the information below to become our member!
Registration
-----
Name: asha5588
Invalid name. Please enter a name containing only letters and spaces.
Name: asha
Phone number: 0168908585
Email: asha5588
Invalid email format. Please enter a valid email address.
Email: asha5588@gmail.com

Thank you for becoming our member. Below are the member details.
Name : asha
Phone number: 0168908585
Email : asha5588@gmail.com
This is your new member code. Make a booking now to enjoy our benefits!
Member Code : WELCOME123
Enter your voucher code: welcome123
Invalid voucher code. Please try again.
Enter your voucher code: WELCOME123
=====
=====Invoice=====
=====
Name :Asha
Phone Number :0168908585
Total number of pax :5
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1005 3 Elite Comfort Hotel 5 RM 480.00 RM 2400.00
Services: Wifi, Breakfast, Conference Room & Laundry Service
=====
Total before taxes RM 16800.00
|SST RM 1008.00
|NO EXTRA SERVICE RM 0.00
|WELCOME123 -RM 3360.00
=====
Grand Total RM 14448.00
=====

=====
Payment Method
-----
1. Credit/Debit Cards
2. Bank Transfer
3. Touch N Go
4. Apple Pay
5. Cash Payment
-----
Please select payment method (1-5) :4
PIN :7970
=====
=====
Elegant Hideaway Airbnb
eleganhideaway@gmail.com +6018-889 8888 Elegant Kuala Lumpur www.eleganhideaway.com
=====
Booking Confirmation
-----
Dear,Asha
Your hotel booking is confirmed.
Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit
Your booking details are as follow :
=====
Name :Asha
Phone Number :0168908585
Total number of pax :5
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1005 3 Elite Comfort Hotel 5 RM 480.00 RM 2400.00
Services: Wifi, Breakfast, Conference Room & Laundry Service
=====
=====
Payment Summary :
=====
Total before taxes RM 16800.00
|SST RM 1008.00
|NO EXTRA SERVICE RM 0.00
|WELCOME123 -RM 3360.00
=====
Grand Total RM 14448.00
=====
Payment Method :
Payment Amount :
Change :
RM 14448.00
RM 0.00
=====
Additional Information :
Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.
|
If you have any questions, please don't hesitate to contact us.
We hope you enjoy your stay with us !!!
=====
Do you want to continue order? (Y/N): n
=====
```

## Problem Solving and Programming

---

```
Welcome to Elegant Hideaway, select an option to proceed
-----
      Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

**2<sup>nd</sup> Order (Member)**

<b>User Inputs</b>	
<b>Location Area</b>	5
<b>Number of Pax Adult</b>	2
<b>Number of Pax Children</b>	1
<b>Number of Pax Infants</b>	0
<b>Start Month</b>	11
<b>End Month</b>	11
<b>Start Year</b>	2024
<b>End Year</b>	2023
<b>End Year ®</b>	2024
<b>Check-In Date</b>	9
<b>Check-Out Date</b>	12
<b>Room Number</b>	4
<b>Quantity</b>	1
<b>Name</b>	Jack***
<b>Name ®</b>	Jack
<b>Phone Number</b>	0152229898

<b>Expected Result / Outputs</b>	
<b>Location Area</b>	You have selected Sabah
<b>Total Number of Pax</b>	$2+1+0=3$ <b>Output:</b> The total number of pax is 3
<b>Validation Check (End Year)</b>	Not available year. Please enter 2024 or 2025
<b>Display the November Calendar in 2025</b>	
<b>Number Of Days</b>	Number of Days = (Day Number of End Date) – (Day Number of Start Date) + 1 Number of Days = $(12-9) + 1 = 4$ <b>Output:</b> Number of your booking day is 4.
<b>Check-In Date</b>	9/11/2024
<b>Check-Out Date</b>	12/11/2024
<b>Display the Sabah Table</b>	
<b>Validation Check (Name)</b>	Invalid name : Please try again.
<b>Display Overview of Bookings with auto generated booking number 1006</b>	
<b>Total Price</b>	Total Price = Price Per Room * Quantity Total Price = $290.00 * 1$ Total Price = 290.00
<b>Total before Tax</b>	Total before Tax = Total Price * Number Of Days Total before Tax = $290.00 * 4$ Total before Tax = 1160.00

## Problem Solving and Programming

---

<b>User Inputs</b>	
Add On Service (Y/N)	n
Voucher Code (Y/N)	y
Membership (1 or 2)	1
Member ID	Member111
Password	password1
Member ID ®	member1
Password ®	password1
Voucher Code	8888
Voucher Code ®	MC1225
Payment Method	2
Account Number	26565655
PIN	885599
Continue Order (Y/N)	n

<b>Expected Result / Outputs</b>	
<b>Validation Check (Member ID)</b>	Incorrect Member ID. Please try again.
<b>Display the different Seasonal Voucher Code in Table</b>	
<b>Validation Check (Voucher Code)</b>	Invalid voucher code. Please try again.
<b>Display Invoice</b>	
<b>SST</b>	$SST = \text{Total before Tax} * SSTRATE (0.06)$ $SST = 1160.00 * 0.06$ $SST = 69.60$
<b>Add On Service</b>	NO EXTRA SERVICE / RM 0.00
<b>Voucher Code</b>	MC1225 (15% discount) $\text{Discount} = 1160 * 0.15$ $\text{Discount} = 174.00$
<b>Grand Total</b>	$\text{Grand Total} = \text{Total before Tax} + SST + \text{Service Price} - \text{Discount}$ $\text{Grand Total} = 1160.00 + 69.60 + 0 - 174.00$ $\text{Grand Total} = 1055.60$
<b>Display the Booking Confirmation and Payment Summary</b>	

# Problem Solving and Programming

## 2<sup>nd</sup> Order Actual Output (Member)

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 1
You selected Order Room
Welcome to Elegant Hideaway Airbnb
Please let us know the location you wish to travel to at your convenience
-----
1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
Please enter the number of the location: 5
You have selected Sabah.
-----
Please enter number of pax adult (13 years and older) :2
Please enter number of pax children (2-12 years old) :1
Please enter number of pax infants (under 2 years old) :0
-----
The total number of pax is 3
Select your Check-In & Check-Out Date
-----
|----- Month -----|----- Year -----|
|----- 1 2 3 4 5 6 |----- 2024 |
|----- 7 8 9 10 11 12 |----- 2025 |
|-----|
Start Month: 11
End Month: 11
Start Year: 2024
End Year: 2023
Not available year. Please enter 2024 or 2025
End Year: 2024
Select your Check-In Date, __/11/2024
-----
|----- Date -----|
|----- 1 2 3 4 5 6 7 |
|----- 8 9 10 11 12 13 14 |
|----- 15 16 17 18 19 20 21 |
|----- 22 23 24 25 26 27 28 |
|----- 29 30 |
|-----|
Check-In Date: 9
9/11/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/11/2024
-----
|----- Date -----|
|----- 1 2 3 4 5 6 7 |
|----- 8 9 10 11 12 13 14 |
|----- 15 16 17 18 19 20 21 |
|----- 22 23 24 25 26 27 28 |
|----- 29 30 |
|-----|
Check-Out Date: 12
12/11/2024, Check-Out Date selected successfully.
Number of your booking day is 4.
Check-In Date 9/11/2024, Check-Out Date 12/11/2024 selected successfully.
Sabah
-----
| No | Name | Location | Price | Bed Type | Room Type | Room Size | Service |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | Borneo Bliss Hotel | 123, Jalan Pantai, 88000  
Kota Kinabalu | RM 320.00 | King | Deluxe | 38-40 sqm | WiFi  
Breakfast  
Pool  
Parking |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 2 | Kinabalu View | 102, Jalan Central Centre,  
88100 Kota Kinabalu, Sabah | RM 270.00 | Queen | Standard | 25-35 sqm | WiFi  
Breakfast  
Shuttle Service  
Airport Transfer |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 3 | Sunset Coastal Resort | 1-76, Jalan Seaside Tanjung  
Aru, 88300 Kota Kinabalu | RM 380.00 | Double | Suite | 35-45 sqm | WiFi  
Breakfast  
Restaurant  
Spa |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 4 | Sabah Serenity Lodge | 8, Jalan Heritage Old Town,  
88400 Kota Kinabalu, Sabah | RM 290.00 | Queen | Superior | 20-30 sqm | WiFi  
Breakfast  
Concierge |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 5 | Horizon Hotel | 55, Jalan View Waterfront,  
88500 Kota Kinabalu, Sabah | RM 450.00 | King | Executive | 40-50 sqm | WiFi  
Breakfast  
Business Center  
Conference Rooms |
|-----|-----|-----|-----|-----|-----|-----|-----|
Room Number (1-5) :4
Quantity :1
Please enter your information below
Name : Jack**
Invalid name : Please try again.
Name : Jack
Phone Number : 0152229898
```

# Problem Solving and Programming

```

=====
=====Overview of Bookings=====
=====
Name :Jack
Phone Number :0152229898
Total number of pax :3
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1006 4 Sabah Serenity Lodge 1 RM 290.00 RM 290.00
Services: Wifi, Breakfast, Concierge
=====
Total before taxes RM 1160.00
=====

Do you need any add on service (Y/N): n
Do you want to apply voucher code (Y/N) :y
Are you a member?
-----
1. Yes
2. No
Please enter your choice (1 or 2): 1
Member login
-----
Member ID: Member111
Password: password1
Incorrect Member ID. Please try again.
Member login
-----
Member ID: member1
Password: password1
Login successful!
Enjoy our benefits now!
=====
| No | Seasonal Promotion | Promo Code |
|---|---|---|
| 1 | Chinese New year | CNY8888 |
| 2 | Hari Raya | HR4444 |
| 3 | Deepavali | DP6666 |
| 4 | Merry Christmas | MC1225 |
| 5 | Valentine | V2141314 |
=====
Enter your voucher code: MC1225
=====
=====Invoice=====
=====
Name :Jack
Phone Number :0152229898
Total number of pax :3
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1006 4 Sabah Serenity Lodge 1 RM 290.00 RM 290.00
Services: Wifi, Breakfast, Concierge
=====
Total before taxes RM 1160.00
SST RM 69.60
NO EXTRA SERVICE RM 0.00
MC1225 -RM 174.00
=====
Grand Total RM 1055.60
=====

=====
| Payment Method |
|---|
| 1. Credit/Debit Cards |
| 2. Bank Transfer |
| 3. Touch N Go |
| 4. Apple Pay |
| 5. Cash Payment |
=====

Please select payment method (1-5) :2
Account Number :26565655
PIN :885599
=====
=====
Elegant Hideaway Airbnb
eleganthideaway@gmail.com +6018-889 8888 Elegant Kuala Lumpur www.eleganthideaway.com
=====
Booking Confirmation
=====
Dear, Jack
Your hotel booking is confirmed.
Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit
Your booking details are as follow :
=====
Name :Jack
Phone Number :0152229898
Total number of pax :3
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1006 4 Sabah Serenity Lodge 1 RM 290.00 RM 290.00
Services: Wifi, Breakfast, Concierge
=====
Payment Summary :
=====
Total before taxes RM 1160.00
SST RM 69.60
NO EXTRA SERVICE RM 0.00
MC1225 -RM 174.00
=====
Grand Total RM 1055.60
=====
Payment Method : Bank Transfer
Payment Amount : RM 1055.60
Change : RM 0.00
=====
Additional Information :
Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.
=====
If you have any questions, please don't hesitate to contact us.
We hope you enjoy your stay with us !!!
=====
Do you want to continue order? (Y/N): n
=====
```

## Problem Solving and Programming

---

```
Welcome to Elegant Hideaway, select an option to proceed
```

```
-----  
Menu List  
-----
```

- ```
1. Order Room  
2. Check-In & Check-Out (Staff)  
3. Cancellation  
4. Report (Staff)  
5. Feedback  
6. Exit
```

```
-----  
Your option: |
```

**Scenario 3:** Two customers select from a menu list to order a room, with options to use vouchers as a member, as a new registered member, or to use any add-on services (including validation checks). If a customer makes an invalid selection, the system should repeat the prompt and ask the user to choose again.

® = Repeat and ask the user again if the input is invalid.

### 1<sup>st</sup> Order (With Add on Service / No membership)

| User Inputs            |            |
|------------------------|------------|
| Location Area          | 6          |
| Number of Pax Adult    | 10         |
| Number of Pax Children | 2          |
| Number of Pax Infants  | 1          |
| Start Month            | 10         |
| End Month              | 10         |
| Start Year             | 2024       |
| End Year               | 2024       |
| Check-In Date          | 2          |
| Check-Out Date         | 15         |
| Room Number            | 4          |
| Quantity               | 10         |
| Name                   | Emma       |
| Phone Number           | 0158887454 |

| Expected Result / Outputs                                                   |                                                                                                                                                                         |
|-----------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Location Area                                                               | You have selected Sarawak                                                                                                                                               |
| Total Number of Pax                                                         | $10+2+1=13$<br><b>Output:</b> The total number of pax is 13                                                                                                             |
| <b>Display the October Calendar in 2024</b>                                 |                                                                                                                                                                         |
| Number Of Days                                                              | Number of Days =<br>(Day Number of End Date) – (Day Number of Start Date) + 1<br>Number of Days = $(15-2) + 1 = 14$<br><b>Output:</b> Number of your booking day is 14. |
| Check-In Date                                                               | 2/10/2024                                                                                                                                                               |
| Check-Out Date                                                              | 15/10/2024<br>selected successfully.                                                                                                                                    |
| <b>Display the Sarawak Table</b>                                            |                                                                                                                                                                         |
| <b>Display Overview of Bookings with auto generated booking number 1007</b> |                                                                                                                                                                         |
| Total Price                                                                 | Total Price = Price Per Room * Quantity<br>Total Price = $460.00 * 10$<br>Total Price = 4600.00                                                                         |
| Total before Tax                                                            | Total before Tax = Total Price * Number Of Days<br>Total before Tax = $4600.00 * 14$<br>Total before Tax = 64400.00                                                     |

## Problem Solving and Programming

---

| User Inputs          |        |
|----------------------|--------|
| Add On Service (Y/N) | y      |
| Service              | k      |
| Service ®            | 1      |
| Voucher Code (Y/N)   | n      |
| Payment Method       | 3      |
| PIN                  | 556688 |
| Continue Order (Y/N) | n      |

| Expected Result / Outputs                                   |                                                                                                                                                                                            |
|-------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Display the Add on Service Table with Price</b>          |                                                                                                                                                                                            |
| <b>Validation Check (Service)</b>                           | Error: Please enter a number between 1 and 5.                                                                                                                                              |
| <b>Display Invoice</b>                                      |                                                                                                                                                                                            |
| <b>SST</b>                                                  | $SST = \text{Total before Tax} * \text{SSTRATE (0.06)}$<br>$SST = 64400.00 * 0.06$<br>$SST = 3864.00$                                                                                      |
| <b>Add On Service</b>                                       | High Speed Wi - Fi, RM 30.00 per day<br>Service Price = $30.00 * 14$<br>Service Price = 420.00                                                                                             |
| <b>Voucher Code</b>                                         | NO VOUCHER / RM 0.00                                                                                                                                                                       |
| <b>Grand Total</b>                                          | $\text{Grand Total} = \text{Total before Tax} + SST + \text{Service Price} - \text{Discount}$<br>$\text{Grand Total} = 64400.00 + 3864.00 + 420.00 - 0$<br>$\text{Grand Total} = 68684.00$ |
| <b>Display the Booking Confirmation and Payment Summary</b> |                                                                                                                                                                                            |

# Problem Solving and Programming

## 1<sup>st</sup> Order (With Add on Service / No membership)

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 1
You selected Order Room
Welcome to Elegant Hideaway Airbnb
Please let us know the location you wish to travel to at your convenience
-----
1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
Please enter the number of the location: 6
You have selected Sarawak.
-----
Please enter number of pax adult (13 years and older) :10
Please enter number of pax children (2-12 years old) :2
Please enter number of pax infants (under 2 years old) :1
-----
The total number of pax is 13
Select your Check-In & Check-Out Date
-----
|     Month      | Year   |
=====
| 1 2 3 4 5 6 | 2024  |
| 7 8 9 10 11 12 | 2025  |
=====
Start Month: 10
End Month: 10
Start Year: 2024
End Year: 2024
Select your Check-In Date, __/10/2024
-----
|     Date      |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 31 |
=====
Check-In Date: 2
2/10/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/10/2024
-----
|     Date      |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 31 |
=====
Check-Out Date: 15
15/10/2024, Check-Out Date selected successfully.
Number of your booking day is 14.
Check-In Date 2/10/2024, Check-Out Date 15/10/2024 selected successfully.
Sarawak
-----
| No | Name | Location | Price | Bed Type | Room Type | Room Size | Service |
=====
1	Riverfront Resort	12, Jalan River, 93000 RM 350.00	Queen	Deluxe	35-45 sqm	WiFi Dinner Spa Free Parking
2	Heritage City Hotel	34, Jalan Heritage, 93100 RM 580.00	King	Executive	45-50 sqm	WiFi Breakfast Pool Laundry
3	Borneo Vista Lodge	56, Jalan Vista, 93200 RM 400.00	Double	Suite	40-50 sqm	WiFi Breakfast Gym Spa
4	Tropical Haven Hotel	90, Jalan Tropical, 93400 RM 460.00	King	Premier	20-30 sqm	WiFi Breakfast Afternoon Tea
5	Kuching Grand Inn	78, Jalan Grand, 93300 RM 310.00	Single	Superior	30-40 sqm	WiFi Breakfast Business Center Lounge
=====
Room Number (1-5) :4
Quantity :10
Please enter your information below
Name : Emma
Phone Number : 0158887454
```

# Problem Solving and Programming

```

=====
=====Overview of Bookings=====
=====
Name :Emma
Phone Number :0158887454
Total number of pax :13
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
| 1007 | 4 | Tropical Haven Hotel | 10 | RM | 460.00 | RM | 4600.00 |
| Services: Wifi, Breakfast, Afternoon Tea |
=====
Total before taxes RM 64400.00
=====

Do you need any add on service (Y/N): y
=====
=====Add On Service=====
=====
1. High Speed Wi-Fi RM 30.00 per day
2. Breakfast RM 50.00 per day
3. Airport Shuttle RM 150.00 per day
4. Room Upgrade RM 300.00 per day
5. Mid-Stay Cleaning RM 100.00 per day
=====
Please select the service you needed(1-5) :k
Error: Please enter a number between 1 and 5.
Please select the service you needed(1-5) :1
Do you want to apply voucher code (Y/N) :n
NO VOUCHER
=====
=====Invoice=====
=====
Name :Emma
Phone Number :0158887454
Total number of pax :13
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
| 1007 | 4 | Tropical Haven Hotel | 10 | RM | 460.00 | RM | 4600.00 |
| Services: Wifi, Breakfast, Afternoon Tea |
=====
Total before taxes RM 64400.00
|SST RM 3864.00
|High Speed Wi - Fi, RM 30.00 per day RM 420.00
|NO VOUCHER -RM 0.00
=====
Grand Total RM 68684.00
=====

=====Payment Method=====
=====
1. Credit/Debit Cards
2. Bank Transfer
3. Touch N Go
4. Apple Pay
5. Cash Payment
=====
Please select payment method (1-5) :3
PIN :556688
=====
=====Elegant Hideaway Airbnb=====
eleganthideaway@gmail.com +6018-889 8888 Elegant Kuala Lumpur www.eleganthideaway.com
=====
Booking Confirmation
=====
Dear Emma
Your hotel booking is confirmed.
Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit
Your booking details are as follow :
=====
Name :Emma
Phone Number :0158887454
Total number of pax :13
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
| 1007 | 4 | Tropical Haven Hotel | 10 | RM | 460.00 | RM | 4600.00 |
| Services: Wifi, Breakfast, Afternoon Tea |
=====
Payment Summary :
=====
Total before taxes RM 64400.00
|SST RM 3864.00
|High Speed Wi - Fi, RM 30.00 per day RM 420.00
|NO VOUCHER -RM 0.00
=====
Grand Total RM 68684.00
=====
Payment Method : Touch N Go
Payment Amount : RM 68684.00
Change : RM 0.00
=====
Additional Information :
Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.
=====
If you have any questions, please don't hesitate to contact us.
We hope you enjoy your stay with us !!!
=====
Do you want to continue order? (Y/N): n
Welcome to Elegant Hideaway, select an option to proceed
=====
Menu List
=====
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
=====
Your option: |
=====
```

**2<sup>nd</sup> Order (With Add on Service / Member Voucher)**

| User Inputs            |            |
|------------------------|------------|
| Location Area          | 6          |
| Number of Pax Adult    | 2          |
| Number of Pax Children | 0          |
| Number of Pax Infants  | 0          |
| Start Month            | 12         |
| End Month              | 12         |
| Start Year             | 2024       |
| End Year               | 2024       |
| Check-In Date          | 3          |
| Check-Out Date         | 6          |
| Room Number            | 2          |
| Quantity               | 1          |
| Name                   | Isabella   |
| Phone Number           | 0128889595 |

| Expected Result / Outputs                                            |                                                                                                                                                                      |
|----------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Location Area                                                        | You have selected Sarawak                                                                                                                                            |
| Total Number of Pax                                                  | $2+0+0=13$<br><b>Output:</b> The total number of pax is 13                                                                                                           |
| Display the December Calendar in 2024                                |                                                                                                                                                                      |
| Number Of Days                                                       | Number of Days =<br>(Day Number of End Date) – (Day Number of Start Date) + 1<br>Number of Days = $(6-3) + 1 = 4$<br><b>Output:</b> Number of your booking day is 4. |
| Check-In Date                                                        | 3/12/2024                                                                                                                                                            |
| Check-Out Date                                                       | 6/12/2024                                                                                                                                                            |
| selected successfully.                                               |                                                                                                                                                                      |
| Display the Sarawak Table                                            |                                                                                                                                                                      |
| Display Overview of Bookings with auto generated booking number 1008 |                                                                                                                                                                      |
| Total Price                                                          | Total Price = Price Per Room * Quantity<br>Total Price = $580.00 * 1$<br>Total Price = 580.00                                                                        |
| Total before Tax                                                     | Total before Tax = Total Price * Number Of Days<br>Total before Tax = $580.00 * 4$<br>Total before Tax = 2320.00                                                     |

## Problem Solving and Programming

---

| User Inputs          |           |
|----------------------|-----------|
| Add On Service (Y/N) | y         |
| Service              | 5         |
| Voucher Code (Y/N)   | y         |
| Membership (1 or 2)  | 1         |
| Member ID            | member2   |
| Password             | password1 |
| Member ID ®          | member2   |
| Password ®           | password2 |
| Voucher Code         | V2141314  |
| Payment Method       | 4         |
| PIN                  | 228899    |
| Continue Order (Y/N) | n         |

| Expected Result / Outputs                                   |                                                                                                                                                                                           |
|-------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Display the Add on Service Table with Price</b>          |                                                                                                                                                                                           |
| Validation Check (Password)                                 | Incorrect Password. Please try again.                                                                                                                                                     |
| <b>Display the different Seasonal Voucher Code in Table</b> |                                                                                                                                                                                           |
| <b>Display Invoice</b>                                      |                                                                                                                                                                                           |
| SST                                                         | $SST = \text{Total before Tax} * SSTRATE (0.06)$<br>$SST = 2320.00 * 0.06$<br>$SST = 139.20$                                                                                              |
| Add On Service                                              | Mid-Stay Cleaning, RM 100.00 per day<br>Service Price = $100.00 * 4$<br>Service Price = 400.00                                                                                            |
| Voucher Code                                                | V2141314(18% discount)<br>Discount = $2320.00 * 0.18$<br>Discount = 417.60                                                                                                                |
| Grand Total                                                 | $\text{Grand Total} = \text{Total before Tax} + SST + \text{Service Price} - \text{Discount}$<br>$\text{Grand Total} = 2320.00 + 139.20 + 400 - 417.60$<br>$\text{Grand Total} = 2441.60$ |
| <b>Display the Booking Confirmation and Payment Summary</b> |                                                                                                                                                                                           |

# Problem Solving and Programming

## 2<sup>nd</sup> Order (With Add on Service / Member Voucher)

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 1
You selected Order Room
Welcome to Elegant Hideaway Airbnb
Please let us know the location you wish to travel to at your convenience
-----
1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
Please enter the number of the location: 6
You have selected Sarawak.
-----
Please enter number of pax adult (13 years and older) :2
Please enter number of pax children (2-12 years old) :0
Please enter number of pax infants (under 2 years old) :0
-----
The total number of pax is 2
Select your Check-In & Check-Out Date
=====
| Month | Year |
|-----|
| 1 2 3 4 5 6 | 2024 |
| 7 8 9 10 11 12 | 2025 |
|-----|
Start Month: 12
End Month: 12
Start Year: 2024
End Year: 2024
Select your Check-In Date, __/12/2024
=====
Date
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
-----
Check-In Date: 3
3/12/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/12/2024
=====
Date
-----
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
-----
Check-Out Date: 6
6/12/2024, Check-Out Date selected successfully.
Number of your booking day is 4.
Check-In Date 3/12/2024, Check-Out Date 6/12/2024 selected successfully.
Sarawak
=====
No
-----
1
-----
2
-----
3
-----
4
-----
5
-----
Room Number (1-5) :2
Quantity :1
Please enter your information below
-----
Name : Isabella
Phone Number : 0128889595
```

# Problem Solving and Programming

---

```

=====
          Overview of Bookings
=====
Name      :Isabella
Phone Number :0128889595
Total number of pax :2
Check-In Date   :3 /12/2024
Check-Out Date :6 /12/2024
Number Of Days :4
=====
BookingNo Room Number     Hotel       Quantity    Price Per Room    Total Price
=====
1008      2             Heritage City Hotel    1           RM        580.00  RM        580.00
Services: Wifi, Breakfast, Pool & Laundry
=====
Total before taxes   RM        2320.00
=====

Do you need any add on service (Y/N): y
=====
          Add On Service
=====
1. High Speed Wi-Fi   RM 30.00 per day
2. Breakfast          RM 50.00 per day
3. Airport Shuttle    RM 150.00 per day
4. Room Upgrade       RM 300.00 per day
5. Mid-Stay Cleaning  RM 100.00 per day
=====
Please select the service you needed(1-5) :5
Do you want to apply voucher code (Y/N) :y
Are you a member?
-----
1. Yes
2. No
Please enter your choice (1 or 2): 1
Member login
-----
Member ID: member2
Password: password1
Incorrect Password. Please try again.
Member login
-----
Member ID: member2
Password: password2
Login successful!
Enjoy our benefits now!
=====
          Seasonal Promotion
=====
No	Seasonal Promotion	Promo Code
1	Chinese New year	CNY8888
2	Hari Raya	HRU4444
3	Deepavali	DP6666
4	Merry Christmas	MC1225
5	Valentine	V2141314
=====
Enter your voucher code: V2141314
=====
          Invoice
=====
Name      :Isabella
Phone Number :0128889595
Total number of pax :2
Check-In Date   :3 /12/2024
Check-Out Date :6 /12/2024
Number Of Days :4
=====
BookingNo Room Number     Hotel       Quantity    Price Per Room    Total Price
=====
1008      2             Heritage City Hotel    1           RM        580.00  RM        580.00
Services: Wifi, Breakfast, Pool & Laundry
=====
Total before taxes   RM        2320.00
SST   RM        139.20
Mid-Stay Cleaning, RM 100.00 per day                  RM        400.00
V2141314   -RM      417.60
=====
Grand Total   RM        2441.60
=====

          Payment Method
=====
1. Credit/Debit Cards
2. Bank Transfer
3. Touch N Go
4. Apple Pay
5. Cash Payment
=====
Please select payment method (1-5) :4
PIN :228899

```

## Problem Solving and Programming

```
=====
Elegant Hideaway Airbnb
eleganthideaway@gmail.com +6018-889 8888 Elegant Kuala Lumpur www.eleganthideaway.com
=====
Booking Confirmation
=====
Dear, Isabella
Your hotel booking is confirmed.
Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit
Your booking details are as follow :
=====
Name :Isabella Check-In Date :3 /12/2024
Phone Number :0128889595 Check-Out Date :6 /12/2024
Total number of pax :2 Number Of Days :4
=====
BookingNo Room Number Hotel Quantity Price Per Room Total Price
=====
1008 2 Heritage City Hotel 1 RM 580.00 RM 580.00
Services: Wifi, Breakfast, Pool & Laundry
=====
Payment Summary :
=====
Total before taxes RM 2320.00
SST RM 139.20
Mid-Stay Cleaning, RM 100.00 per day RM 400.00
V2141314 -RM 417.60
=====
Grand Total RM 2441.60
=====
Payment Method : Apple Pay
Payment Amount : RM 2441.60
Change : RM 0.00
=====
Additional Information :
Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.
=====
If you have any questions, please don't hesitate to contact us.
We hope you enjoy your stay with us !!!
=====
Do you want to continue order? (Y/N): n
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

**Scenario 4:** One customer wants to continue ordering two rooms at the same location. If the customer makes an invalid selection during the ordering process, the system should repeat the prompt and ask the customer to choose again.

® = Repeat and ask the user again if the input is invalid.

| User Inputs            |            |
|------------------------|------------|
| Location Area          | 2          |
| Number of Pax Adult    | 2          |
| Number of Pax Children | 1          |
| Number of Pax Infants  | 1          |
| Start Month            | 10         |
| End Month              | 10         |
| Start Year             | 2024       |
| End Year               | 2024       |
| Check-In Date          | 2          |
| Check-Out Date         | 10         |
| Room Number            | 5          |
| Quantity               | 2          |
| Name                   | Steven     |
| Phone Number           | 0189984545 |
| Add On Service (Y/N)   | n          |
| Voucher Code (Y/N)     | n          |
| Payment Method         | 1          |
| Card Number            | 8889988    |
| PIN                    | 778855     |
| Continue Order (Y/N)   | Y          |

| Expected Result / Outputs                                                   |                                                                                                                                                                |
|-----------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Location Area                                                               | You have selected Melaka                                                                                                                                       |
| Total Number of Pax                                                         | $2+1+1 = 4$<br><b>Output:</b> The total number of pax is 4                                                                                                     |
| Display the October Calendar in 2024                                        |                                                                                                                                                                |
| Number Of Days                                                              | Number of Days =<br>(Day Number of End Date) – (Day Number of Start Date) + 1<br>Number of Days = $(10-2) + 1 = 9$<br>Output: Number of your booking day is 9. |
| Check-In Date                                                               | 2/10/2024                                                                                                                                                      |
| Check-Out Date                                                              | 10/10/2024<br>selected successfully.                                                                                                                           |
| Display the Melaka Table                                                    |                                                                                                                                                                |
| Display Overview of Bookings with auto generated booking number <b>1009</b> |                                                                                                                                                                |
| Total Price                                                                 | Total Price = Price Per Room * Quantity<br>Total Price = $200.00 * 2$<br>Total Price = 400.00                                                                  |
| Total before Tax                                                            | Total before Tax = Total Price * Number Of Days<br>Total before Tax = $400.00 * 9$<br>Total before Tax = 3600.00                                               |
| Display Invoice                                                             |                                                                                                                                                                |
| SST                                                                         | $SST = Total\ before\ Tax * SSTRATE\ (0.06)$<br>$SST = 3600 * 0.06$<br>$SST = 216.00$                                                                          |
| Add On Service                                                              | NO EXTRA SERVICE / RM 0.00                                                                                                                                     |
| Voucher Code                                                                | NO VOUCHER / RM 0.00                                                                                                                                           |
| Grand Total                                                                 | Grand Total = Total before Tax + SST + Service Price – Discount<br>Grand Total = $3600 + 216 + 0 - 0$<br>Grand Total = 3816.00                                 |
| Display the Booking Confirmation and Payment Summary                        |                                                                                                                                                                |

## Problem Solving and Programming

---

| User Inputs            |            |
|------------------------|------------|
| Number of Pax Adult    | 2          |
| Number of Pax Children | 1          |
| Number of Pax Infants  | 1          |
| Start Month            | 10         |
| End Month              | 10         |
| Start Year             | 2025       |
| End Year               | 2025       |
| Check-In Date          | 2          |
| Check-Out Date         | 10         |
| Room Number            | 5          |
| Quantity               | 2          |
| Name                   | Steven     |
| Phone Number           | 0189984545 |
| Add On Service (Y/N)   | n          |
| Voucher Code (Y/N)     | n          |
| Payment Method         | 1          |
| Card Number            | 8889988    |
| PIN                    | fhjh       |
| PIN ®                  | 778855     |
| Continue Order (Y/N)   | N          |

| Expected Result / Outputs                                                   |                                                                                                                                                                |
|-----------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Total Number of Pax                                                         | $2+1+1 = 4$<br><b>Output:</b> The total number of pax is 4                                                                                                     |
| <b>Display the October Calendar in 2025</b>                                 |                                                                                                                                                                |
| Number Of Days                                                              | Number of Days =<br>(Day Number of End Date) – (Day Number of Start Date) + 1<br>Number of Days = $(10-2) + 1 = 9$<br>Output: Number of your booking day is 9. |
| Check-In Date                                                               | 2/10/2025                                                                                                                                                      |
| Check-Out Date                                                              | 10/10/2025                                                                                                                                                     |
| <b>Display the Melaka Table</b>                                             |                                                                                                                                                                |
| <b>Display Overview of Bookings with auto generated booking number 1010</b> |                                                                                                                                                                |
| Total Price                                                                 | Total Price = Price Per Room * Quantity<br>Total Price = $200.00 * 2$<br>Total Price = 400.00                                                                  |
| Total before Tax                                                            | Total before Tax = Total Price * Number Of Days<br>Total before Tax = $400.00 * 9$<br>Total before Tax = 3600.00                                               |
| <b>Display Invoice</b>                                                      |                                                                                                                                                                |
| SST                                                                         | SST = Total before Tax * SSTRATE (0.06)<br>SST = $3600 * 0.06$<br>SST = 216.00                                                                                 |
| Add On Service                                                              | NO EXTRA SERVICE / RM 0.00                                                                                                                                     |
| Voucher Code                                                                | NO VOUCHER / RM 0.00                                                                                                                                           |
| Grand Total                                                                 | Grand Total = Total before Tax + SST + Service Price – Discount<br>Grand Total = $3600 + 216 + 0 - 0$<br>Grand Total = 3816.00                                 |
| Validation Check (PIN)                                                      | Error: Please enter a valid PIN number with 4 to 6 digits.                                                                                                     |
| <b>Display the Booking Confirmation and Payment Summary</b>                 |                                                                                                                                                                |

## **Scenario 4 Actual Output**

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 1
You selected Order Room
Welcome to Elegant Hideaway Airbnb
Please let us know the location you wish to travel to at your convenience
-----
1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
Please enter the number of the location: 2
You have selected Melaka.
-----
Please enter number of pax adult (13 years and older) :2
Please enter number of pax children (2-12 years old) :1
Please enter number of pax infants (under 2 years old) :1
-----
The total number of pax is 4
-----
Select your Check-In & Check-Out Date
-----
| Month | Year |
=====
| 1 2 3 4 5 6 | 2024 |
| 7 8 9 10 11 12 | 2025 |
=====
Start Month: 10
End Month: 10
Start Year: 2024
End Year: 2024
Select your Check-In Date, __/10/2024
-----
| Date |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 31 |
=====
Check-In Date: 2
2/10/2024, Check-In Date selected successfully.
Select your Check-Out Date, __/10/2024
-----
| Date |
=====
| 1 2 3 4 5 6 7 |
| 8 9 10 11 12 13 14 |
| 15 16 17 18 19 20 21 |
| 22 23 24 25 26 27 28 |
| 29 30 31 |
=====
Check-Out Date: 10
10/10/2024, Check-Out Date selected successfully.
Number of your booking day is 9.
Check-In Date 2/10/2024, Check-Out Date 10/10/2024 selected successfully.
Melaka
-----
| No | Name | Location | Price | Bed Type | Room Type | Room Size | Service |
-----
1	Heritage Haven	45, Jalan Hang Jebat, 75200 RM	300.00	Queen	Superior	25-35 sqm	WiFi Breakfast
2	Melaka River Inn	18, Jalan Laksamana, 75000 RM	250.00	King	Deluxe	20-30 sqm	WiFi Breakfast Shuttle Service
3	Coastal Retreat	8, Jalan Kampung, 45500 RM	350.00	Double	Suite	30-40 sqm	WiFi Breakfast Gym Spa
4	Grand Melaka Resort	100, Jalan Melaka Raya, 75000 Melaka RM	500.00	Queen	Executive	35-45 sqm	WiFi Breakfast Lounge Business Center
5	Historical Lodge	23, Jalan Merdeka, 75000 Melaka RM	200.00	Single	Standard	20-30 sqm	WiFi Breakfast Laundry Service
-----
Room Number (1-5) :5
Quantity :2
Please enter your information below
-----
Name : Steven
Phone Number : 0189984545
```

# Problem Solving and Programming

| Overview of Bookings                                                                                       |                          |                            |                         |                |             |
|------------------------------------------------------------------------------------------------------------|--------------------------|----------------------------|-------------------------|----------------|-------------|
| Name :Steven                                                                                               | Phone Number :0189984545 | Check-In Date :2 /10/2024  |                         |                |             |
| Total number of pax :4                                                                                     |                          | Check-Out Date :10/10/2024 |                         |                |             |
| BookingNo                                                                                                  | Room Number              | Hotel                      | Quantity                | Price Per Room | Total Price |
| 1009                                                                                                       | 5                        | Historical Lodge           | 2                       | RM 200.00      | RM 400.00   |
| Services: Wifi, Breakfast & Laundry Service                                                                |                          |                            |                         |                |             |
| Total before taxes RM 3600.00                                                                              |                          |                            |                         |                |             |
| Do you need any add on service (Y/N): n                                                                    |                          |                            |                         |                |             |
| Do you want to apply voucher code (Y/N) :n                                                                 |                          |                            |                         |                |             |
| NO VOUCHER                                                                                                 |                          |                            |                         |                |             |
| Invoice                                                                                                    |                          |                            |                         |                |             |
| Name :Steven                                                                                               | Phone Number :0189984545 | Check-In Date :2 /10/2024  |                         |                |             |
| Total number of pax :4                                                                                     |                          | Check-Out Date :10/10/2024 |                         |                |             |
| BookingNo                                                                                                  | Room Number              | Hotel                      | Quantity                | Price Per Room | Total Price |
| 1009                                                                                                       | 5                        | Historical Lodge           | 2                       | RM 200.00      | RM 400.00   |
| Services: Wifi, Breakfast & Laundry Service                                                                |                          |                            |                         |                |             |
| Total before taxes RM 3600.00                                                                              |                          |                            |                         |                |             |
| SST                                                                                                        |                          | RM 216.00                  |                         |                |             |
| NO EXTRA SERVICE                                                                                           |                          | RM 0.00                    |                         |                |             |
| NO VOUCHER                                                                                                 |                          | -RM 0.00                   |                         |                |             |
| Grand Total RM 3816.00                                                                                     |                          |                            |                         |                |             |
| Payment Method                                                                                             |                          |                            |                         |                |             |
| 1. Credit/Debit Cards                                                                                      |                          |                            |                         |                |             |
| 2. Bank Transfer                                                                                           |                          |                            |                         |                |             |
| 3. Touch N Go                                                                                              |                          |                            |                         |                |             |
| 4. Apple Pay                                                                                               |                          |                            |                         |                |             |
| 5. Cash Payment                                                                                            |                          |                            |                         |                |             |
| Please select payment method (1-5)                                                                         |                          |                            |                         |                |             |
| Card Number                                                                                                |                          |                            |                         |                | :1 8889988  |
| PIN                                                                                                        |                          |                            |                         |                | :778855     |
| Elegant Hideaway Airbnb                                                                                    |                          |                            |                         |                |             |
| eleganthideaway@gmail.com                                                                                  | +6018-889 8888           | Elegant Kuala Lumpur       | www.eleganthideaway.com |                |             |
| Booking Confirmation                                                                                       |                          |                            |                         |                |             |
| Dear, Steven                                                                                               |                          |                            |                         |                |             |
| Your hotel booking is confirmed.                                                                           |                          |                            |                         |                |             |
| Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit         |                          |                            |                         |                |             |
| Your booking details are as follow :                                                                       |                          |                            |                         |                |             |
| Name :Steven                                                                                               | Phone Number :0189984545 | Check-In Date :2 /10/2024  |                         |                |             |
| Total number of pax :4                                                                                     |                          | Check-Out Date :10/10/2024 |                         |                |             |
| BookingNo                                                                                                  | Room Number              | Hotel                      | Quantity                | Price Per Room | Total Price |
| 1009                                                                                                       | 5                        | Historical Lodge           | 2                       | RM 200.00      | RM 400.00   |
| Services: Wifi, Breakfast & Laundry Service                                                                |                          |                            |                         |                |             |
| Payment Summary :                                                                                          |                          |                            |                         |                |             |
| Total before taxes RM 3600.00                                                                              |                          |                            |                         |                |             |
| SST                                                                                                        |                          | RM 216.00                  |                         |                |             |
| NO EXTRA SERVICE                                                                                           |                          | RM 0.00                    |                         |                |             |
| NO VOUCHER                                                                                                 |                          | -RM 0.00                   |                         |                |             |
| Grand Total RM 3816.00                                                                                     |                          |                            |                         |                |             |
| Payment Method : Credit/Debit Cards                                                                        |                          |                            |                         |                |             |
| Payment Amount :                                                                                           |                          |                            |                         |                | RM 3816.00  |
| Change :                                                                                                   |                          |                            |                         |                | RM 0.00     |
| Additional Information :                                                                                   |                          |                            |                         |                |             |
| Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour. |                          |                            |                         |                |             |
| If you have any questions, please don't hesitate to contact us.                                            |                          |                            |                         |                |             |
| We hope you enjoy your stay with us !!!                                                                    |                          |                            |                         |                |             |
| Do you want to continue order? (Y/N): Y                                                                    |                          |                            |                         |                |             |
| Please enter number of pax adult (13 years and older) :2                                                   |                          |                            |                         |                |             |
| Please enter number of pax children (2-12 years old) :1                                                    |                          |                            |                         |                |             |
| Please enter number of pax infants (under 2 years old) :1                                                  |                          |                            |                         |                |             |
| The total number of pax is 4                                                                               |                          |                            |                         |                |             |
| Select your Check-In & Check-Out Date                                                                      |                          |                            |                         |                |             |
| Month Year                                                                                                 |                          |                            |                         |                |             |
| 1 2 3 4 5 6   2024                                                                                         |                          |                            |                         |                |             |
| 7 8 9 10 11 12   2025                                                                                      |                          |                            |                         |                |             |
| Start Month: 10<br>End Month: 10<br>Start Year: 2025<br>End Year: 2025                                     |                          |                            |                         |                |             |
| Select your Check-In Date, __/10/2025                                                                      |                          |                            |                         |                |             |
| Date                                                                                                       |                          |                            |                         |                |             |
| 1 2 3 4 5 6 7<br>8 9 10 11 12 13 14<br>15 16 17 18 19 20 21<br>22 23 24 25 26 27 28<br>29 30 31            |                          |                            |                         |                |             |
| Check-In Date: 2                                                                                           |                          |                            |                         |                |             |
| 2/10/2025, Check-In Date selected successfully.                                                            |                          |                            |                         |                |             |

# Problem Solving and Programming

Select your Check-Out Date, \_\_/10/2025

| Date |    |    |    |    |    |    |
|------|----|----|----|----|----|----|
| 1    | 2  | 3  | 4  | 5  | 6  | 7  |
| 8    | 9  | 10 | 11 | 12 | 13 | 14 |
| 15   | 16 | 17 | 18 | 19 | 20 | 21 |
| 22   | 23 | 24 | 25 | 26 | 27 | 28 |
| 29   | 30 | 31 |    |    |    |    |

Check-Out Date: 10  
10/10/2025, Check-Out Date selected successfully.

Number of your booking day is 9.  
Check-In Date 2/10/2025, Check-Out Date 10/10/2025 selected successfully.

Melaka

| No | Name                | Location                               | Price  | Bed Type | Room Type | Room Size | Service                               |
|----|---------------------|----------------------------------------|--------|----------|-----------|-----------|---------------------------------------|
| 1  | Heritage Haven      | 145, Jalan Hang Jebat, 75200 RM Melaka | 300.00 | Queen    | Superior  | 25-35 sqm | WiFi Breakfast                        |
| 2  | Melaka River Inn    | 10, Jalan Laksamana, 75000 Melaka      | 250.00 | King     | Deluxe    | 20-30 sqm | WiFi Breakfast Shuttle Service        |
| 3  | Coastal Retreat     | 8, Jalan Kampung, 45500 Melaka         | 350.00 | Double   | Suite     | 30-40 sqm | WiFi Breakfast Gym Spa                |
| 4  | Grand Melaka Resort | 100, Jalan Melaka Raya, 75000 Melaka   | 500.00 | Queen    | Executive | 35-45 sqm | WiFi Breakfast Lounge Business Center |
| 5  | Historical Lodge    | 23, Jalan Merdeka, 75000 Melaka        | 200.00 | Single   | Standard  | 20-30 sqm | WiFi Breakfast Laundry Service        |

Room Number (1-5) :5  
Quantity :2

Please enter your information below

Name : Steven  
Phone Number : 0189984545

| Overview of Bookings                        |                            |                  |          |                |             |  |
|---------------------------------------------|----------------------------|------------------|----------|----------------|-------------|--|
| Name :Steven                                | Check-In Date :2 /10/2025  |                  |          |                |             |  |
| Phone Number :0189984545                    | Check-Out Date :10/10/2025 |                  |          |                |             |  |
| Total number of pax :4                      | Number Of Days :9          |                  |          |                |             |  |
| BookingNo                                   | Room Number                | Hotel            | Quantity | Price Per Room | Total Price |  |
| 1010                                        | 5                          | Historical Lodge | 2        | RM 200.00      | RM 400.00   |  |
| Services: Wifi, Breakfast & Laundry Service |                            |                  |          |                |             |  |
| Total before taxes RM 3600.00               |                            |                  |          |                |             |  |

Do you need any add on service (Y/N): n  
Do you want to apply voucher code (Y/N) :n  
NO VOUCHER

| Invoice                                     |                            |                  |          |                |             |  |
|---------------------------------------------|----------------------------|------------------|----------|----------------|-------------|--|
| Name :Steven                                | Check-In Date :2 /10/2025  |                  |          |                |             |  |
| Phone Number :0189984545                    | Check-Out Date :10/10/2025 |                  |          |                |             |  |
| Total number of pax :4                      | Number Of Days :9          |                  |          |                |             |  |
| BookingNo                                   | Room Number                | Hotel            | Quantity | Price Per Room | Total Price |  |
| 1010                                        | 5                          | Historical Lodge | 2        | RM 200.00      | RM 400.00   |  |
| Services: Wifi, Breakfast & Laundry Service |                            |                  |          |                |             |  |
| Total before taxes RM 3600.00               |                            |                  |          |                |             |  |
| SST RM 216.00                               |                            |                  |          |                |             |  |
| NO EXTRA SERVICE RM 0.00                    |                            |                  |          |                |             |  |
| NO VOUCHER -RM 0.00                         |                            |                  |          |                |             |  |
| Grand Total RM 3816.00                      |                            |                  |          |                |             |  |

| Payment Method        |  |
|-----------------------|--|
| 1. Credit/Debit Cards |  |
| 2. Bank Transfer      |  |
| 3. Touch N Go         |  |
| 4. Apple Pay          |  |
| 5. Cash Payment       |  |

Please select payment method (1-5) :1  
Card Number :8889988  
PIN :fhjgh  
Error: Please enter a valid PIN number with 4 to 6 digits.  
PIN :778855

## Problem Solving and Programming

```
=====
|eleganthideaway@gmail.com          +6018-889 8888          Elegant Kuala Lumpur      www.eleganthideaway.com|
=====
|Booking Confirmation|
=====
|Dear Steven|
|Your hotel booking is confirmed.|
|Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit|
|Your booking details are as follow :|
=====
|Name           :Steven                         Check-In Date   :2 /10/2025|
|Phone Number  :0189984545                     Check-Out Date :10/10/2025|
|Total number of pax :4                         Number Of Days :9|
=====
|BookingNo    Room Number     Hotel            Quantity       Price Per Room    Total Price|
|=====|
| 1010        5             Historical Lodge      2            RM              200.00 RM          400.00|
|Services: Wifi, Breakfast & Laundry Service|
=====
|Payment Summary :|
=====
|Total before taxes                           RM      3600.00|
|SST   RM      216.00|
|NO EXTRA SERVICE                           RM      0.00|
|NO VOUCHER                                  -RM     0.00|
=====
|Grand Total                                 RM      3816.00|
=====
|Payment Method :                           Credit/Debit Cards|
|Payment Amount   :                         RM      3816.00|
|Change   :                                    RM      0.00|
=====
|Additional Information :|
|Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.|
|If you have any questions, please don't hesitate to contact us.|
|We hope you enjoy your stay with us !!!|
=====
|Do you want to continue order? (Y/N): N|
Welcome to Elegant Hideaway, select an option to proceed
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
|Your option: |
```

**Scenario 5:** The customer is able to cancel an order if they make a mistake or select the wrong date by choosing option 3. If the customer makes an invalid selection during the cancellation process, the system should repeat the prompt and ask the customer to choose again.

® = Repeat and ask the user again if the input is invalid.

### 1<sup>st</sup> Cancellation

| User Inputs             |      | Expected Result / Outputs                                                              |
|-------------------------|------|----------------------------------------------------------------------------------------|
| Booking Number Cancel   | abc  |                                                                                        |
| Booking Number Cancel ® | 1005 | Validation Check<br>(Booking Number Cancel) Error: Please enter a valid booking number |
| Continue Cancel (Y/N)   | y    | Display Current Bookings before Cancel                                                 |
| Booking Number Cancel   | 1005 | Founded the booking Number                                                             |
| Continue Cancel (Y/N)   | N    | Output: Booking number 1005 has been cancelled.                                        |
|                         |      | Display Current Bookings before Cancel                                                 |
|                         |      | Output: Booking number 1005 not found.                                                 |

### 2<sup>nd</sup> Cancellation

| User Inputs                           |      | Expected Result / Outputs                                |
|---------------------------------------|------|----------------------------------------------------------|
| Booking Number Cancel                 | 1002 | Display Current Bookings before Cancel                   |
| Continue Cancel (Y/N)                 | j    | Founded the booking Number                               |
| Continue Cancel (Y/N) ®               | n    | Output: Booking number 1002 has been cancelled.          |
| Validation Check<br>(Continue Cancel) |      | Invalid input. Please enter 'Y' for Yes or 'N' for No: n |

## **1<sup>st</sup> Cancellation Actual Output**

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 3

You selected Cancellation
Enter the booking number you want to cancel: abc
Error: Please enter a valid booking number
Booking Number : 1005

Current bookings before cancellation:
Booking Number: 1001, Room No: 1, Quantity: 0
Booking Number: 1002, Room No: 5, Quantity: 2
Booking Number: 1003, Room No: 4, Quantity: 2
Booking Number: 1004, Room No: 5, Quantity: 1
Booking Number: 1005, Room No: 3, Quantity: 5
Booking Number: 1006, Room No: 4, Quantity: 1
Booking Number: 1007, Room No: 4, Quantity: 10
Booking Number: 1008, Room No: 2, Quantity: 1
Booking Number: 1009, Room No: 5, Quantity: 2
Booking Number: 1010, Room No: 5, Quantity: 2

Booking number 1005 has been cancelled.
Do you want to continue cancellation? (Y/N): y
Enter the booking number you want to cancel: 1005

Current bookings before cancellation:
Booking Number: 1001, Room No: 1, Quantity: 0
Booking Number: 1002, Room No: 5, Quantity: 2
Booking Number: 1003, Room No: 4, Quantity: 2
Booking Number: 1004, Room No: 5, Quantity: 1
Booking Number: 1006, Room No: 4, Quantity: 1
Booking Number: 1007, Room No: 4, Quantity: 10
Booking Number: 1008, Room No: 2, Quantity: 1
Booking Number: 1009, Room No: 5, Quantity: 2
Booking Number: 1010, Room No: 5, Quantity: 2

Booking number 1005 not found.
Do you want to continue cancellation? (Y/N): N

Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

## **2<sup>nd</sup> Cancellation Actual Output**

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 3

You selected Cancellation
Enter the booking number you want to cancel: 1002

Current bookings before cancellation:
Booking Number: 1001, Room No: 1, Quantity: 0
Booking Number: 1002, Room No: 5, Quantity: 2
Booking Number: 1003, Room No: 4, Quantity: 2
Booking Number: 1004, Room No: 5, Quantity: 1
Booking Number: 1006, Room No: 4, Quantity: 1
Booking Number: 1007, Room No: 4, Quantity: 10
Booking Number: 1008, Room No: 2, Quantity: 1
Booking Number: 1009, Room No: 5, Quantity: 2
Booking Number: 1010, Room No: 5, Quantity: 2

Booking number 1002 has been cancelled.
Do you want to continue cancellation? (Y/N): j
Invalid input. Please enter 'Y' for Yes or 'N' for No: n

Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

**Scenario 6:** Staff need to log in to access the check-in/check-out system by selecting option 2. If the end user makes an invalid selection during the process, the system should repeat the prompt and ask the end user to choose again.

® = Repeat and ask the user again if the input is invalid.

| User Inputs               |                | Expected Result / Outputs                                                                                       |
|---------------------------|----------------|-----------------------------------------------------------------------------------------------------------------|
| Staff ID                  | Sttaff1        | Validation Check (Staff ID)<br>Incorrect Staff ID. Please try again.                                            |
| Password                  | password1      | Validation Check (Try Again)<br>Invalid input. Please enter 'Y' for Yes or 'N' for No:                          |
| Try Again (Y/N)           | j              | Validation Check (Password)<br>Incorrect Password. Please try again.                                            |
| Try Again (Y/N) ®         | Y              | Display Staff Interface                                                                                         |
| Staff ID ®                | staff1         | Validation Check (Selection)<br>Error: Please enter a number between 1 - 3                                      |
| Password ®                | PASSWORD1      | Validation Check (Booking Number Check In)<br>Booking number 1011 not found.                                    |
| Try Again (Y/N)           | Y              | Booking number 1008 has been Checked In.                                                                        |
| Staff ID ®                | staff1         | Validation Check (Customer Name)<br>Invalid name: Please try again.                                             |
| Password ®                | password1      | Validation Check (Customer IC Number)<br>Invalid IC number: Please enter a valid IC number without any symbols. |
| Selection                 | 0              | Display Check in Successful and Display Staff Interface                                                         |
| Selection ®               | 1              | Validation Check (Delayed Time)<br>Error: Please enter a valid number for delayed time.                         |
| Booking Number Check In   | 1011           | Fines For Customer =<br><b>Delayed Time * FINESPERHOURS (20.00)</b><br>Fines For Customer = 2 * 20.00 = 40.00   |
| Booking Number Check In ® | 1008           | Display Check Out Completed and Display Staff Interface                                                         |
| Customer Name             | Isabella99     | Exit the Staff Interface                                                                                        |
| Customer Name ®           | Isabella       |                                                                                                                 |
| Customer IC Number        | 051109-10-1899 |                                                                                                                 |
| Customer IC Number ®      | 051109101899   |                                                                                                                 |
| Selection                 | 2              |                                                                                                                 |
| Booking Number Check Out  | 1008           |                                                                                                                 |
| Delayed Time              | k              |                                                                                                                 |
| Delayed Time ®            | 2              |                                                                                                                 |
| Payment Method            | 3              |                                                                                                                 |
| PIN                       | 223355         |                                                                                                                 |
| Selection                 | 3              |                                                                                                                 |

## **Scenario 6 Actual Output**

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 2
You selected Check-In & Check-Out
Login
-----
Staff ID: Sstaff1
Password: password1
Incorrect Staff ID. Please try again.
Do you want to try again? (Y/N): j
Invalid input. Please enter 'Y' for Yes or 'N' for No: Y
Login
-----
Staff ID: staff1
Password: PASSWORD1
Incorrect Password. Please try again.
Do you want to try again? (Y/N): Y
Login
-----
Staff ID: staff1
Password: password1
Login successful!
WELCOME TO STAFF INTERFACE
-----
1. CHECK IN
2. CHECK OUT
3. EXIT
-----
Please select 1, 2, or 3: 0
Error: Please enter a number between 1 - 3
Your selection: 1
Enter the booking number you want to Check In: 1011
Booking number 1011 not found.
Enter the booking number you want to Check In: 1008
Booking number 1008 has been Checked In.
Customer Name :Isabella99
Invalid name: Please try again.
Customer Name :Isabella
Customer IC number :051109-10-1899
Invalid IC number: Please enter a valid IC number without any symbols.
Customer IC number :051109101899
-----
Check-in successful!
WELCOME TO STAFF INTERFACE
-----
1. CHECK IN
2. CHECK OUT
3. EXIT
-----
Please select 1, 2, or 3: 2
Enter the booking number you want to Check Out: 1008
Customer delayed time (in hours): k
Error: Please enter a valid number for delayed time.
Customer delayed time (in hours): 2
Fine for Customer: RM40.00
Please select the customer payment method
=====
| Payment Method |
=====
| 1. Credit/Debit Cards |
| 2. Bank Transfer |
| 3. Touch N Go |
| 4. Apple Pay |
| 5. Cash Payment |
=====
Please select payment method (1-5) :3
PIN :223355
-----
Check-out complete!
Booking number 1008 has been Checked Out.
WELCOME TO STAFF INTERFACE
-----
1. CHECK IN
2. CHECK OUT
3. EXIT
-----
Please select 1, 2, or 3: 3
Exiting the system...
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

**Scenario 7:** Once an order is cancelled, it will no longer appear in the check-in/check-out system. Staff need to log in to access the check-in/check-out system by selecting option 2.

| User Inputs    |           | Expected Result / Outputs              |
|----------------|-----------|----------------------------------------|
| Staff ID       | staff2    | Display Staff Interface                |
| Password       | password2 | Display Booking Number Not Found       |
| Booking Number | 1005      | Output: Booking number 1005 not found. |

Welcome to Elegant Hideaway, select an option to proceed

-----  
Menu List  
-----

1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit

-----  
Your option: 2

You selected Check-In & Check-Out

-----  
Login

-----  
Staff ID: staff2  
Password: password2  
Login successful!

-----  
WELCOME TO STAFF INTERFACE

1. CHECK IN
2. CHECK OUT
3. EXIT

-----  
Please select 1, 2, or 3: 1  
Enter the booking number you want to Check In: 1005  
Booking number 1005 not found.

**Scenario 8:** When a customer accidentally clicks on the staff login page (option 2 or 4) and enters the wrong staff ID or password, the system will prompt the user to indicate whether they want to try again. The customer can enter 'N' for no to decline and exit the login attempt.

| User Inputs     |       | Expected Result / Outputs             |
|-----------------|-------|---------------------------------------|
| Staff ID        | sdjsd | Validation Check (Staff ID)           |
| Password        | 55655 | Incorrect Staff ID. Please try again. |
| Try Again (Y/N) | n     | Exit the staff login interface        |

```
Welcome to Elegant Hideaway, select an option to proceed
```

```
-----  
Menu List  
-----
```

1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit

```
-----  
Your option: 2
```

```
You selected Check-In & Check-Out
```

```
-----  
Login  
-----
```

```
Staff ID: sdjsd  
Password: 55655  
Incorrect Staff ID. Please try again.  
Do you want to try again? (Y/N): n  
Exiting staff interface...
```

```
Welcome to Elegant Hideaway, select an option to proceed
```

```
-----  
Menu List  
-----
```

1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit

```
-----  
Your option: |
```

**Scenario 9:** The customer is able to view feedback and reviews from previous customers, as well as provide their own review based on their experiences with our Airbnb. This feature allows the customer to make informed decisions based on others' feedback. If the end user makes an invalid selection during the review process, the system should repeat the prompt and ask the end user to choose again.

® = Repeat and ask the user again if the input is invalid.

#### The customer does not provide a review

| User Inputs        |   |
|--------------------|---|
| Leave Review (Y/N) | h |
| Leave Review (Y/N) | n |

| Expected Result / Outputs              |                                                        |
|----------------------------------------|--------------------------------------------------------|
| <b>Display Feedback Table</b>          |                                                        |
| <b>Validation Check (Leave Review)</b> | Invalid input. Please enter 'y' for yes or 'n' for no. |
| <b>Exit the Feedback</b>               |                                                        |

#### The customer provides a review

| User Inputs        |                                                          |
|--------------------|----------------------------------------------------------|
| Leave Review (Y/N) | y                                                        |
| Stars              | 6                                                        |
| Stars ®            | 5                                                        |
| Name               | Mickey                                                   |
| Feedback           | I had a good experience with this Airbnb booking system. |
| Leave Review (Y/N) | N                                                        |
| Leave Review (Y/N) | n                                                        |

| Expected Result / Outputs                             |                                                        |
|-------------------------------------------------------|--------------------------------------------------------|
| <b>Display Feedback Table</b>                         |                                                        |
| <b>Validation Check (Stars)</b>                       | Invalid input. Rating must be between 1 and 5 stars.   |
| <b>Display Feedback Table with The Current Review</b> |                                                        |
| <b>Validation Check (Leave Review)</b>                | Invalid input. Please enter 'y' for yes or 'n' for no. |
| <b>Exit the Feedback</b>                              |                                                        |

# Problem Solving and Programming

## The customer does not provide a review

```
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 5
You selected Feedback
Customer Feedback:
+=====+
| Rating | Name | Feedback |
+=====+
**	JOHN TERRY	The property's location is conveniently close to the Sky Arena Sports Complex. However, there are no nearby options for food or convenience stores within walking distance. The cleanliness left much to be desired.
***	KLAY NICK	Overall ok. Need to improve on room cleanliness. Aircond sejuk. Got toiletries, towel and mini pantry. Nearby a lot of shopping malls also.
****	CHAN	Staff helpful and friendly. Location ok but stay on 1st floor bit noisy earlier, but ok after midnight.
*	SAM	Very bad experience and not recommended.
*****	KANE	Very nice, will come again next time.
**	AUNE	Sound proof not good. Closed door can hear it clearly and window curtain bit dirty. Overall is ok. Have to give extra attention on room cleanliness.
+=====+
Would you like to leave a rating? (y/n): h
Invalid input. Please enter 'y' for yes or 'n' for no.
Would you like to leave a rating? (y/n): n
Thank you for visiting. If you have any comments or suggestions later, please don't hesitate to share!
Welcome to Elegant Hideaway, select an option to proceed
-----
Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

# Problem Solving and Programming

## The customer provides a review

```
Welcome to Elegant Hideaway, select an option to proceed
-----
      Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: 5

You selected Feedback

Customer Feedback:
+=====+
| Rating | Name          | Feedback
+=====+
| **    | JOHN TERRY    | The property's location is conveniently close to the Sky Arena Sports Complex. However, there are no nearby options for food or convenience stores within walking distance.The cleanliness left much to be desired.
+=====+
| ***   | KLAY NICK     | Overall ok. Need to improve on room cleanliness. Aircond sejuk. Got toiletries, towel and mini pantry.Nearby a lot of shopping malls also.
+=====+
| ****  | CHAN          | Staff helpful and friendly. Location ok but stay on 1st floor bit noisy earlier, but ok after midnight.
+=====+
| *     | SAM            | Very bad experience and not recommended.
+=====+
| *****| KANE          | Very nice, will come again next time.
+=====+
| **    | AUNE           | Sound proof not good. Closed door can hear it clearly and window curtain bit dirty. Overall is ok. Have to give extra attention on room cleanliness.
+=====+

Would you like to leave a rating? (y/n): y
Please enter your rating from 1 - 5 stars: 6
Invalid input. Rating must be between 1 and 5 stars.
Please enter your rating from 1 - 5 stars: 5
Please enter your name: Mickey
Please enter your feedback: I had a good experience with this Airbnb booking system.

Thank you for your feedback! Your feedback has been recorded.

Current Feedback:
+=====+
| Rating | Name          | Feedback
+=====+
| **    | JOHN TERRY    | The property's location is conveniently close to the Sky Arena Sports Complex. However, there are no nearby options for food or convenience stores within walking distance.The cleanliness left much to be desired.
+=====+
| ***   | KLAY NICK     | Overall ok. Need to improve on room cleanliness. Aircond sejuk. Got toiletries, towel and mini pantry.Nearby a lot of shopping malls also.
+=====+
| ****  | CHAN          | Staff helpful and friendly. Location ok but stay on 1st floor bit noisy earlier, but ok after midnight.
+=====+
| *     | SAM            | Very bad experience and not recommended.
+=====+
| *****| KANE          | Very nice, will come again next time.
+=====+
| **    | AUNE           | Sound proof not good. Closed door can hear it clearly and window curtain bit dirty. Overall is ok. Have to give extra attention on room cleanliness.
+=====+
| ****  | Mickey         | I had a good experience with this Airbnb booking system.
+=====+

Would you like to leave a rating? (y/n): N
Invalid input. Please enter 'y' for yes or 'n' for no.

Would you like to leave a rating? (y/n): n

Thank you for visiting. If you have any comments or suggestions later, please don't hesitate to share!

Welcome to Elegant Hideaway, select an option to proceed
-----
      Menu List
-----
1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit
-----
Your option: |
```

**Scenario 10:** Staff members are required to log in initially to access the report by selecting option 4. If an incorrect staff ID or password is entered, the system will prompt the user to attempt the login process again. If a customer selects a hotel located in the Kuala Lumpur area, the sales amount will be recorded exclusively under the Kuala Lumpur area. Conversely, if the customer selects a hotel in the Sabah area, the sales amount will be recorded solely under the Sabah area.

® = Repeat and ask the user again if the input is invalid.

### 1<sup>st</sup> Report

| <b>User Inputs</b> |           |
|--------------------|-----------|
| Staff ID           | ==        |
| Password           | password1 |
| Try Again (Y/N)    | ~         |
| Try Again (Y/N) ®  | Y         |
| Staff ID ®         | staff1    |
| Password ®         | 123       |
| Try Again (Y/N)    | Y         |
| Staff ID ®         | staff1    |
| Password ®         | password1 |
| Selection          | 1         |
| Selection          | 2         |
| Selection          | 7         |
| Selection          | n         |

| <b>Expected Result / Outputs</b>                                                                                                                                                                           |                                                        |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| Validation Check (Staff ID)                                                                                                                                                                                | Incorrect Staff ID. Please try again.                  |
| Validation Check (Try Again)                                                                                                                                                                               | Invalid input. Please enter 'Y' for Yes or 'N' for No: |
| Validation Check (Password)                                                                                                                                                                                | Incorrect Password. Please try again.                  |
| <b>Staff logged in successfully. Generating report...</b>                                                                                                                                                  |                                                        |
| <b>Display Current Hotel Report</b>                                                                                                                                                                        |                                                        |
| Kuala Lumpur Summary<br>Total Units : Old record + new record<br>Total Amount : Old record + new record<br>Total Units : 78 + 2<br>Total Amount : 41000 + 7632<br>Total Units : 80<br>Total Amount : 48632 |                                                        |
| <b>Display Current Hotel Report</b>                                                                                                                                                                        |                                                        |
| Melaka Summary<br>Total Units : 65<br>Total Amount : 20400                                                                                                                                                 |                                                        |
| <b>Display Existing program</b>                                                                                                                                                                            | Do you want to continue Report?                        |
| <b>Exit Report (Staff)</b>                                                                                                                                                                                 |                                                        |

## 2<sup>nd</sup> Report

| <b>User Inputs</b>       |           |
|--------------------------|-----------|
| <b>Staff ID</b>          | ==        |
| <b>Password</b>          | password1 |
| <b>Try Again (Y/N)</b>   | ~         |
| <b>Try Again (Y/N) ®</b> | Y         |
| <b>Staff ID ®</b>        | staff1    |
| <b>Password ®</b>        | 123       |
| <b>Try Again (Y/N)</b>   | Y         |
| <b>Staff ID ®</b>        | staff1    |
| <b>Password ®</b>        | password1 |
| <b>Selection</b>         | 8         |
| <b>Selection ®</b>       | 5         |
| <b>Selection</b>         | 2         |
| <b>Selection</b>         | 7         |
| <b>Selection</b>         | n         |

| <b>Expected Result / Outputs</b>                                                                                                                                                                            |                                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| <b>Validation Check (Staff ID)</b>                                                                                                                                                                          | Incorrect Staff ID. Please try again.                  |
| <b>Validation Check (Try Again)</b>                                                                                                                                                                         | Invalid input. Please enter 'Y' for Yes or 'N' for No: |
| <b>Validation Check (Password)</b>                                                                                                                                                                          | Incorrect Password. Please try again.                  |
| <b>Validation Check (Selection)</b>                                                                                                                                                                         | Invalid choice. Please enter a number between 1 and 7. |
| <b>Staff logged in successfully. Generating report...</b>                                                                                                                                                   |                                                        |
| <b>Display Current Hotel Report</b>                                                                                                                                                                         |                                                        |
| Sabah Summary<br>Total Units : Old record + new record<br>Total Amount : Old record + new record<br>Total Units : 101 + 1<br>Total Amount : 33840 + 1055.60<br>Total Units : 102<br>Total Amount : 34895.60 |                                                        |
| <b>Display Current Hotel Report</b>                                                                                                                                                                         |                                                        |
| Melaka Summary<br>Total Units : 65<br>Total Amount : 20400                                                                                                                                                  |                                                        |
| <b>Display Existing program</b>                                                                                                                                                                             | Do you want to continue Report?                        |
| <b>Exit Report (Staff)</b>                                                                                                                                                                                  |                                                        |

## 1<sup>st</sup> Report

```
Welcome to Elegant Hideaway, select an option to proceed
```

```
-----  
Menu List  
-----
```

1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit

```
Your option: 4
```

```
You selected Report (Staff)
```

```
Login
```

```
-----  
Staff ID: ==  
Password: password1  
Incorrect Staff ID. Please try again.  
Do you want to try again? (Y/N): Y
```

```
Login
```

```
-----  
Staff ID: staff1  
Password: 123  
Incorrect Password. Please try again.  
Do you want to try again? (Y/N): Y
```

```
Login
```

```
-----  
Staff ID: staff1  
Password: password1  
Login successful!
```

```
Staff logged in successfully. Generating report...
```

```
Hotel Report System
```

1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
7. Exit

```
Enter your choice (1-7): 1
```

```
Kuala Lumpur Summary
```

```
=====|Total Units |Total Amount |  
|80 |48632.00 |  
=====
```

```
Hotel Report System
```

1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
7. Exit

```
Enter your choice (1-7): 2
```

```
Melaka Summary
```

```
=====|Total Units |Total Amount |  
|65 |20400.00 |  
=====
```

```
Hotel Report System
```

1. Kuala Lumpur
2. Melaka
3. Penang
4. Selangor
5. Sabah
6. Sarawak
7. Exit

```
Enter your choice (1-7): 7
```

```
Exiting program.
```

```
Do you want to continue Report? (Y/N): n
```

```
Welcome to Elegant Hideaway, select an option to proceed
```

```
-----  
Menu List  
-----
```

1. Order Room
2. Check-In & Check-Out (Staff)
3. Cancellation
4. Report (Staff)
5. Feedback
6. Exit

```
Your option: |
```

# Problem Solving and Programming

## 2nd Report

```
C:\Users\Administrator\Down < - + >
Welcome to Elegant Hideaway, select an option to proceed
-----  
Menu List  
-----  
1. Order Room  
2. Check-In & Check-Out (Staff)  
3. Cancellation  
4. Report (Staff)  
5. Feedback  
6. Exit  
-----  
Your option: 4  
-----  
You selected Report (Staff)  
-----  
Login  
-----  
Staff ID: ==  
Password: password1  
Incorrect Staff ID. Please try again.  
Do you want to try again? (Y/N): ~  
Invalid input. Please enter 'Y' for Yes or 'N' for No: Y  
-----  
Login  
-----  
Staff ID: staff1  
Password: 123  
Incorrect Password. Please try again.  
Do you want to try again? (Y/N): Y  
-----  
Login  
-----  
Staff ID: staff1  
-----  
Password: password1  
Login successful!  
-----  
Staff logged in successfully. Generating report...  
-----  
Hotel Report System  
1. Kuala Lumpur  
2. Melaka  
3. Penang  
4. Selangor  
5. Sabah  
6. Sarawak  
7. Exit  
Enter your choice (1-7): 8  
Invalid choice. Please enter a number between 1 and 7.  
-----  
Hotel Report System  
1. Kuala Lumpur  
2. Melaka  
3. Penang  
4. Selangor  
5. Sabah  
6. Sarawak  
7. Exit  
Enter your choice (1-7): 5  
Sabah Summary  
=====|Total Units |Total Amount |  
|102 |34895.60 |  
=====  
-----  
Hotel Report System  
1. Kuala Lumpur  
2. Melaka  
3. Penang  
4. Selangor  
5. Sabah  
6. Sarawak  
7. Exit  
Enter your choice (1-7): 2  
Melaka Summary  
=====|Total Units |Total Amount |  
|65 |20400.00 |  
=====  
-----  
Hotel Report System  
1. Kuala Lumpur  
2. Melaka  
3. Penang  
4. Selangor  
5. Sabah  
6. Sarawak  
7. Exit  
Enter your choice (1-7): 7  
Exiting program.  
Do you want to continue Report? (Y/N): n  
-----  
Welcome to Elegant Hideaway, select an option to proceed  
-----  
Menu List  
-----  
1. Order Room
```

# Program Listing

## 1. Header Files

### 1.1 BookingSystem

```
1  /*#ifndef BOOKING_SYSTEM_H
2  #define BOOKING_SYSTEM_H
3
4  #include <string>
5  using namespace std;
6
7  // Constants for the maximum number of bookings and rooms
8  const int MAX_BOOKINGS = 100;
9
10 // Struct to store hotel information
11 struct BookingRoom {
12     int roomNo; // Room Number
13     string hotelName; // Name of hotel depend on locationNumber
14     double price;
15     string service;
16 };
17
18 // store booking detail
19 struct Booking {
20     int bookingNumber;
21     int roomNo;
22     int quantity;
23 };
24
25 struct BookingList {
26     Booking bookings[MAX_BOOKINGS]; // Array of bookings
27     int count; // currentcount for bookings
28 };
29
30 struct RebateValue {
31     double rebate;
32     string selectedvoucher;
33 };
34
35 #endif // BOOKING_SYSTEM_H
```

### 1.2 getvalid

```
1  /*#ifndef FUNCTIONS_H
2  #define FUNCTIONS_H
3
4  #include <string>
5
6  // Function to get a validated integer input from the user
7  int getValidatedInput(const std::string& prompt);
8
9 #endif // FUNCTIONS_H
```



## 2.5 Add on service Table

```

1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  void ADDONSEVICE() {
6      cout << setfill(' ') << setfill('=') << setw(50) << "=" << endl;
7      cout << setfill(' ') << "=" << setw(17) << " " << "Add On Service" << setw(17) << " " << "|" << endl;
8      cout << setfill(' ') << setw(50) << "=" << endl;
9      cout << setfill(' ') << '|' << setw(2) << " " << "1. High Speed Wi-Fi" << setw(6) << " " << "RM 30.00 per day" << setw(4) << " " << "|" << endl;
10     cout << setfill(' ') << '|' << setw(2) << " " << "2. Breakfast" << setw(13) << " " << "RM 50.00 per day" << setw(5) << "|" << endl;
11     cout << setfill(' ') << '|' << setw(2) << " " << "3. Airport Shuttle" << setw(7) << " " << "RM 150.00 per day" << setw(4) << " " << "|" << endl;
12     cout << setfill(' ') << '|' << setw(2) << " " << "4. Room Upgrade" << setw(10) << " " << "RM 300.00 per day" << setw(4) << " " << "|" << endl;
13     cout << setfill(' ') << '|' << setw(2) << " " << "5. Mid-Stay Cleaning" << setw(5) << " " << "RM 100.00 per day" << setw(4) << " " << "|" << endl;
14     cout << setfill(' ') << setw(50) << "=" << endl;
15 }

```

## 2.6 CallRoomTable

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4
5  using namespace std;
6
7  //Declaration Function Table
8  void KualaLumpurArea();
9  void SelangorArea();
10 void MelakaArea();
11 void PenangArea();
12 void SabahArea();
13 void SarawakArea();
14
15 void CallRoomTable(int locationNo){
16     // table to let customer select
17     switch (locationNo) {
18         case 1:
19             KualaLumpurArea();
20             break;
21         case 2:
22             MelakaArea();
23             break;
24         case 3:
25             PenangArea();
26             break;
27         case 4:
28             SelangorArea();
29             break;
30         case 5:
31             SabahArea();
32             break;
33         case 6:
34             SarawakArea();
35             break;
36     }
37 }

```

## 2.7 CancellationBooking

```

1  #include <iostream>
2  #include <iomanip>
3  #include "BookingSystem.h"
4
5  using namespace std;
6
7  void cancelBooking(BookingList& BL) {
8      int bookingNumberToCancel;
9      cout << "Enter the booking number you want to cancel: ";
10     cin >> bookingNumberToCancel;
11
12     bool bookingFound = false;
13
14     // Print current bookings for debugging
15     cout << "\nCurrent bookings before cancellation:" << endl;
16     for (int i = 0; i < BL.count; ++i) {
17         cout << "Booking Number: " << BL.bookings[i].bookingNumber
18             << ", Room No: " << BL.bookings[i].roomNo
19             << ", Quantity: " << BL.bookings[i].quantity << endl;
20     }
21
22     // Iterate through the bookings to find the one to cancel
23     for (int i = 0; i < BL.count; ++i) {
24         if (BL.bookings[i].bookingNumber == bookingNumberToCancel) {
25             // Found the booking, now remove it
26             for (int j = i; j < BL.count - 1; ++j) {
27                 BL.bookings[j] = BL.bookings[j + 1]; // Shift elements to the left
28             }
29             BL.count--; // Decrement booking count
30             cout << "Booking number " << bookingNumberToCancel << " has been cancelled." << endl;
31             bookingFound = true;
32             break;
33         }
34     }
35
36     if (!bookingFound) {
37         cout << "Booking number " << bookingNumberToCancel << " not found." << endl;
38     }
39 }

```

## 2.8 Check In & Check Out

```

1  //#include <iostream>
2  #include <iomanip>
3  #include <string>
4  #include <limits> // for numeric_limits
5  #include "BookingSystem.h"
6
7  using namespace std;
8
9  // Declare Validation Function
10 bool validateName(const string& name);
11 int getValidatedInput(const string& prompt);
12 bool validateICNumber(const string& ic_num);
13 int pinValidation(const string& prompt);
14
15 // Declare Function Table
16 void paymentMethodTable();
17
18 void checkInAndCheckout(BookingList &BL) {
19     int selection;
20
21     do {
22         // Display the menu
23         cout << "\nWELCOME TO STAFF INTERFACE" << endl;
24         cout << "-----" << endl;
25         cout << "1. CHECK IN" << endl;
26         cout << "2. CHECK OUT" << endl;
27         cout << "3. EXIT" << endl; // Option to exit the program
28         cout << "-----" << endl;
29         cout << "Please select 1, 2, or 3: ";
30         cin >> selection;
31
32         // Input validation for selection
33         while (cin.fail() || selection < 1 || selection > 3) {
34             cin.clear(); // Clear error flag
35             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
36             cout << "Error: Please enter a number between 1 - 3" << endl;
37             cout << "Your selection: ";
38             cin >> selection;
39         }
40
41         if (selection == 1) { // CHECK IN
42             int bookingNumberToCheckIn;
43             cout << "Enter the booking number you want to Check In: ";
44             cin >> bookingNumberToCheckIn;
45
46             while (cin.fail()) {
47                 cin.clear(); // Clear error flag
48                 cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
49                 cout << "Error: Please enter a valid booking number" << endl;
50                 cout << "Booking Number : ";
51                 cin >> bookingNumberToCheckIn;
52             }
53
54             bool bookingFoundCheckIN = false;
55
56             while (!bookingFoundCheckIN) {
57                 // Iterate through the bookings to find the one to CheckIn
58                 for (int i = 0; i < BL.count(); ++i) {
59                     if (BL.bookings[i].bookingNumber == bookingNumberToCheckIn) {
60                         cout << "Booking number " << bookingNumberToCheckIn << " has been Checked In." << endl;
61                         bookingFoundCheckIN = true;
62                         break;
63                     }
64                 }
65
66                 if (!bookingFoundCheckIN) {
67                     cout << "Booking number " << bookingNumberToCheckIn << " not found." << endl;
68                     cout << "Enter the booking number you want to Check In: ";
69                     cin >> bookingNumberToCheckIn;
70                 }
71
72             // Enter the customer Name
73             string name;
74             cin.ignore();
75             do {
76                 cout << setw(19) << setfill(' ') << left << "Customer Name" << ":";
77                 getline(cin, name);
78                 if (!validateName(name)) {
79                     cout << "Invalid name: Please try again." << endl;
80                 }
81             } while (!validateName(name));
82
83             // Enter the IC number
84             string ic_num;
85             bool icValid = false;
86             while (!icValid) {
87                 cout << "Customer IC number : ";
88                 cin >> ic_num;
89
90                 // Validate input type and IC number
91                 if (!validateICNumber(ic_num)) {
92                     cout << "Invalid IC number: Please enter a valid IC number without any symbols." << endl;
93                 }
94                 else {
95                     icValid = true; // Input is valid
96                 }
97             }
98
99             cout << "-----" << endl;
100            cout << "Check-in successful!" << endl; // Feedback for successful check-in
101            cout << "-----" << endl;
102        }
}

```

## Problem Solving and Programming

---

```
103     ↓
104     ↓
105     ↓
106     ↓
107     ↓
108     ↓
109     ↓
110     ↓
111     ↓
112     ↓
113     ↓
114     ↓
115     ↓
116     ↓
117     ↓
118     ↓
119     ↓
120     ↓
121     ↓
122     ↓
123     ↓
124     ↓
125     ↓
126     ↓
127     ↓
128     ↓
129     ↓
130     ↓
131     ↓
132     ↓
133     ↓
134     ↓
135     ↓
136     ↓
137     ↓
138     ↓
139     ↓
140     ↓
141     ↓
142     ↓
143     ↓
144     ↓
145     ↓
146     ↓
147     ↓
148     ↓
149     ↓
150     ↓
151     ↓
152     ↓
153     ↓
154     ↓
155     ↓
156     ↓
157     ↓
158     ↓
159     ↓
160     ↓
161     ↓
162     ↓
163     ↓
164     ↓
165     ↓
166     ↓
167     ↓
168     ↓
169     ↓
170     ↓
171     ↓
172     ↓
173     ↓
174     ↓
175     ↓
176     ↓
177     ↓
178     ↓
179     ↓
180     ↓
181     ↓
182     ↓
183     ↓
184     ↓
185     ↓
186     ↓
187     ↓
188     ↓
189     ↓
190     ↓
191     ↓
192     ↓
193     ↓
194     ↓
195     ↓
196     ↓
197     ↓
198     ↓
199     ↓
200     ↓
201     ↓
202     ↓
203     ↓
204     ↓
```

```
else if (selection == 2) { // CHECK OUT
    int bookingNumberToCheckOut;
    cout << "Enter the booking number you want to Check Out: ";
    cin >> bookingNumberToCheckOut;

    while (cin.fail()) {
        cin.clear(); // Clear error flag
        cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
        cout << "Error: Please enter a valid booking number" << endl;
        cout << "Booking Number : ";
        cin >> bookingNumberToCheckOut;
    }

    bool bookingFoundCheckOUT = false;
    const double FINESPERHOURS = 20.00;
    int delayed_time;
    double fines = 0.00;
    int paymentMethod, accountNo, cardNumber, PIN;
    string paymentMethodName;
    double cashAmount, change = 0.00;

    while (!bookingFoundCheckOUT) {
        // Iterate through the bookings to find the one to CheckOut
        for (int i = 0; i < BL.count(); ++i) {
            if (BL.bookings[i].bookingNumber == bookingNumberToCheckOut) {
                cout << "Customer delayed time (in hours): ";
                cin >> delayed_time;

                // Input validation for delayed_time
                while (cin.fail() || delayed_time < 0) {
                    cin.clear(); // Clear the error
                    cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Ignore unwanted characters for input
                    cout << "Error: Please enter a valid number for delayed time." << endl;
                    cout << "Customer delayed time (in hours): ";
                    cin >> delayed_time;
                }

                if (delayed_time > 0) {
                    fines = delayed_time * FINESPERHOURS;
                    cout << "Fine for Customer: RM" << fines << endl;
                    cout << "\nPlease select the customer payment method" ;
                    paymentMethodTable();
                    cout << setw(56) << left << "Please select payment method (1-5)" << ":" ;
                    cin >> paymentMethod;

                    // Validation for paymentMethod
                    while (cin.fail() || paymentMethod < 1 || paymentMethod > 5) {
                        cin.clear(); // Clear the error
                        cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Ignore unwanted characters for input
                        cout << "Error: Please enter a number between 1 and 5." << endl;
                        cout << setw(56) << left << "Please select payment method (1-5)" << ":" ;
                        cin >> paymentMethod;
                    }

                    switch (paymentMethod) {
                        case 1:
                            paymentMethodName = "Credit/Debit Cards";
                            cardNumber = getValidatedInput("Card Number");
                            PIN = pinValidation("PIN");
                            cashAmount = fines;
                            break;
                        case 2:
                            paymentMethodName = "Bank Transfer";
                            accountNo = getValidatedInput("Account Number");
                            PIN = pinValidation("PIN");
                            cashAmount = fines;
                            break;
                        case 3:
                            paymentMethodName = "Touch N Go";
                            PIN = pinValidation("PIN");
                            cashAmount = fines;
                            break;
                        case 4:
                            paymentMethodName = "Apple Pay";
                            PIN = pinValidation("PIN");
                            cashAmount = fines;
                            break;
                        case 5:
                            do {
                                paymentMethodName = "Cash Payment";
                                cashAmount = getValidatedInput("Payment Amount");
                                if (cashAmount >= fines) {
                                    change = cashAmount - fines;
                                }
                                else {
                                    cout << "Insufficient amount. Please make payment again." << endl;
                                }
                            } while (cashAmount < fines);
                            break;
                    }
                }
                else {
                    cout << "-----" << endl;
                    cout << "No fines incurred." << endl;
                }
            }
            cout << "-----" << endl;
            cout << "Check-out complete!" << endl; // Feedback for successful check-out
            cout << "Booking number " << bookingNumberToCheckOut << " has been Checked Out." << endl;
            bookingFoundCheckOUT = true;
            break;
        }
    }
}
```

```

205     if (!bookingFoundCheckOUT) {
206         cout << "Booking number " << bookingNumberToCheckOut << " not found." << endl;
207         cout << "Enter the booking number you want to Check Out: ";
208         cin >> bookingNumberToCheckOut;
209     }
210 }
211 }
212 }
213 else if (selection == 3) { // EXIT
214     cout << "Exiting the system..." << endl;
215 }
216 else {
217     cout << "Invalid selection. Please choose 1, 2, or 3." << endl;
218 }
219 }
220 } while (selection != 3); // Continue until user selects exit
221 }
```

## 2.9 DisplaySummaryReport

```

1 #include <iostream>
2 #include <iomanip>
3 #include <string>
4
5 using namespace std;
6
7 // Definition for the summary display function
8 void displaySummary(int totalUnits, double totalAmount, const string& location) {
9     cout << location << " Summary" << endl;
10    cout << setfill('=') << left << setw(43) << "=" << endl;
11    cout << setfill(' ') << "|" << setw(25) << left << "Total Units" << "|" <<
12        setw(15) << "Total Amount" << "|" << endl;
13    cout << setfill('-') << left << setw(43) << "-" << endl;
14    cout << setfill(' ');
15
16    cout << "|" << setw(25) << left << totalUnits << "|"
17        << setw(15) << totalAmount << "|" << endl;
18    cout << setfill('=') << left << setw(43) << "=" << endl;
19    cout << setfill(' ');
20 }
```

## 2.10 Feedback

```

1  #include <iostream> // Include this for cout
2  #include <iomanip>
3  #include <limits>
4  #include <string>
5
6  using namespace std;
7
8  void displayFeedback() {
9      const int MAX_FEEDBACK = 5; // Define maximum feedback entries
10     int ratings[MAX_FEEDBACK] = { 0 };
11     string names[MAX_FEEDBACK];
12     string comments[MAX_FEEDBACK];
13     int feedbackCount = 0;
14
15     cout << "\nCustomer Feedback:\n";
16
17     // Print table header
18     cout << "+-----+-----+-----+-----+-----+\n";
19     cout << setfill(' ') << " " << setw(10) << left << "Rating" << " " << setw(20) << left << "Name" << " " << setw(80) << left << "Feedback" << " " >
20         << endl;
21     cout << "+-----+-----+-----+-----+-----+\n";
22
23     // Print preloaded feedback again (if needed, or you might choose to skip this)
24     cout << setfill(' ') << " " << setw(10) << left << "***" << " " << setw(20) << left << "JOHN TERRY" << " " << setw(80) << left << "The property's >
25         location is conveniently close to the Sky Arena Sports Complex." << " " << endl;
26     cout << setfill(' ') << " " << setw(10) << " " << setw(20) << " " << setw(80) << left << "However, there are no nearby options for >
27         food or convenience stores within " << endl;
28     cout << setfill(' ') << " " << setw(10) << " " << setw(20) << " " << setw(80) << left << "Walking distance. The cleanliness left >
29         much to be desired." << endl;
30     cout << "+-----+\n";
31
32     cout << setfill(' ') << " " << setw(10) << left << "***" << " " << setw(20) << left << "KLAY NICK" << " " << setw(80) << left << "Overall ok. >
33         Need to improve on room cleanliness. Aircond sejuk. Got toiletries." << " " << endl;
34     cout << setfill(' ') << " " << setw(10) << " " << setw(20) << " " << setw(80) << left << "towel and mini pantry. Nearby a lot of >
35         +-----+\n";
36
37     cout << setfill(' ') << " " << setw(10) << left << "***" << " " << setw(20) << left << "CHAN" << " " << setw(80) << left << "Staff helpful and >
38         friendly. Location ok but stay on 1st floor bit noisy earlier." << " " << endl;
39     cout << setfill(' ') << " " << setw(10) << " " << setw(20) << " " << setw(80) << left << "but ok after midnight." << " " << endl;
40     cout << "+-----+\n";
41
42     cout << setfill(' ') << " " << setw(10) << left << "***" << " " << setw(20) << left << "SAM" << " " << setw(80) << left << "Very bad experience and >
43         not recommended." << " " << endl;
44     cout << "+-----+\n";
45
46     char decision;
47     do {
48         // Adding new feedback
49         cout << "\nWould you like to leave a rating? (y/n): ";
50         cin >> decision;
51
52         if (decision == 'y') {
53             if (feedbackCount >= MAX_FEEDBACK) {
54                 cout << "Sorry, feedback capacity has been reached. Unable to add more feedback." << endl;
55                 break;
56             }
57
58             int rating;
59             do {
60                 cout << "Please enter your rating from 1 - 5 stars: ";
61                 cin >> rating;
62
63                 if (cin.fail() || rating < 1 || rating > 5) {
64                     cout << "Invalid input. Rating must be between 1 and 5 stars." << endl;
65                     cin.clear();
66                     cin.ignore(numeric_limits<streamsize>::max(), '\n');
67                 }
68                 else {
69                     break;
70                 }
71             } while (true);
72
73             cout << "Please enter your name: ";
74             cin.ignore(); // To clear the newline character from the buffer
75             getline(cin, names[feedbackCount]);
76
77             cout << "Please enter your feedback: ";
78             getline(cin, comments[feedbackCount]);
79
80             ratings[feedbackCount] = rating;
81             feedbackCount++;
82
83             cout << "\nThank you for your feedback! Your feedback has been recorded." << endl;
84
85             // Display all feedback (including new ones)
86             cout << "\nCurrent Feedback:\n";
87             cout << "+-----+-----+-----+-----+-----+\n";
88             cout << setfill(' ') << " " << setw(10) << left << "Rating" << " " << setw(20) << left << "Name" << " " << setw(80) << left << "Feedback" >
89                 << " " << endl;
90             cout << "+-----+-----+-----+-----+-----+\n";
91
92             // Print preloaded feedback again (if needed, or you might choose to skip this)

```

## Problem Solving and Programming

```
92 cout << setfill(' ') << "| " << setw(10) << left << "***" << "| " << setw(20) << left << "JOHN TERRY" << "| " << setw(80) << left << "The      "
93 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "However, there are no nearby      "
94 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "walking distance. The cleanliness      "
95 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "left much to be desired." << "| " << endl;
96 cout << "#-----+\n";
97 cout << setfill(' ') << "| " << setw(10) << left << "***" << "| " << setw(20) << left << "KLAY NICK" << "| " << setw(80) << left << "Overall      "
98 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "ok. Need to improve on room cleanliness. Aircond seuk. Got toiletries," << "| " << endl;
99 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "towel and mini pantry. Nearby a      "
100 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "+-----+\n";
101 cout << setfill(' ') << "| " << setw(10) << left << "****" << "| " << setw(20) << left << "CHAN" << "| " << setw(80) << left << "Staff      "
102 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "helpful and friendly. Location ok but stay on 1st floor bit noisy earlier," << "| " << endl;
103 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "but ok after midnight." << "| " << endl;
104 cout << "#-----+\n";
105 cout << setfill(' ') << "| " << setw(10) << left << "*" << "| " << setw(20) << left << "SAM" << "| " << setw(80) << left << "Very bad      "
106 cout << setfill(' ') << "| " << setw(10) << left << "experience and not recommended." << "| " << endl;
107 cout << "#-----+\n";
108 cout << setfill(' ') << "| " << setw(10) << left << "*****" << "| " << setw(20) << left << "KANE" << "| " << setw(80) << left << "Very nice,      "
109 cout << setfill(' ') << "| " << setw(10) << left << "will come again next time." << "| " << endl;
110 cout << "#-----+\n";
111 cout << setfill(' ') << "| " << setw(10) << left << "***" << "| " << setw(20) << left << "AUNE" << "| " << setw(80) << left << "Sound proof      "
112 cout << setfill(' ') << "| " << setw(10) << left << "not good. Closed door can hear it clearly and window curtain" << "| " << endl;
113 cout << setfill(' ') << "| " << setw(10) << " " << "| " << setw(20) << " " << "| " << setw(80) << left << "bit dirty. Overall is ok. Have to      "
114 cout << setfill(' ') << "| " << setw(10) << left << "give extra attention on room cleanliness." << "| " << endl;
115 cout << "#-----+\n";
116 // Display new feedback
117 for (int i = 0; i < feedbackCount; i++) {
118     cout << "| " << setw(10) << left << string(ratings[i], '*') << "| " << setw(20) << left << names[i] << "| " << setw(80) << left <<      "
119     comments[i] << "| " << endl;
120     cout << "#-----+\n";
121 }
122
123 } while (decision != 'n');
124
125 cout << "\nThank you for visiting. If you have any comments or suggestions later, please don't hesitate to share!" << endl;
126
127 }
```

## 2.11 Invoice

```

1  #include <iostream>
2  #include <iomanip>
3  #include "BookingSystem.h"
4
5  using namespace std;
6
7  // Define the Table
8  void ADDONSERVICE();
9  void paymentMethodTable();
10
11 // Validation Check Function ;
12 int getValidatedInput(const string& prompt);
13 int pinValidation(const string& prompt);
14
15 double displayInvoice(int locationNo, int totalPax, string name, string phoneNumber, int startMonth, int endMonth, int startYear, int endYear, int
16 checkInDay, int checkOutDay, int numberOfDays, double totalPriceAlldays, BookingList& BL) {
17
18     // Define and initialize hotel rooms for Kuala Lumpur
19     BookingRoom roomsKualaLumpur[] = {
20         {1, "Cubic Botanical", 650.00, "Wifi & Breakfast"}, 
21         {2, "Mont Kiara", 200.00, "Wifi & Breakfast"}, 
22         {3, "Urban Retreat", 450.00, "Wifi, Breakfast & Gym"}, 
23         {4, "Luxe Residence", 450.00, "Wifi, Breakfast & Lounge"}, 
24         {5, "Panorama Hotel", 450.00, "Wifi, Breakfast & Pool"} 
25     };
26     const int numRoomsKualaLumpur = sizeof(roomsKualaLumpur) / sizeof(roomsKualaLumpur[0]);
27
28     // Define and initialize hotel rooms for Melaka
29     BookingRoom roomsMelaka[] = {
30         {1, "Heritage Haven", 300.00, "Wifi & Breakfast"}, 
31         {2, "Melaka River Inn", 250.00, "Wifi, Breakfast & Shuttle Service"}, 
32         {3, "Coastal Retreat", 350.00, "Wifi, Breakfast, Gym & Spa"}, 
33         {4, "Grand Melaka Resort", 500.00, "Wifi, Breakfast, Lounge & Business Center"}, 
34         {5, "Historical Lodge", 200.00, "Wifi, Breakfast & Laundry Service"} 
35     };
36     const int numRoomsMelaka = sizeof(roomsMelaka) / sizeof(roomsMelaka[0]);
37
38     // Define and initialize hotel rooms for Penang
39     BookingRoom roomsPenang[] = {
40         {1, "Bayview Grand Hotel", 300.00, "Wifi & Breakfast"}, 
41         {2, "Georgetown Inn", 250.00, "Wifi, Breakfast & Shuttle Service"}, 
42         {3, "Seaside Resort", 400.00, "Wifi, Breakfast, Gym & Spa"}, 
43         {4, "Penang Heritage Lodge", 300.00, "Wifi, Breakfast & Lounge"}, 
44         {5, "Bayview Penang", 500.00, "Wifi, Breakfast & Business Center"} 
45     };
46     const int numRoomsPenang = sizeof(roomsPenang) / sizeof(roomsPenang[0]);
47
48     // Define and initialize hotel rooms for Selangor
49     BookingRoom roomsSelangor[] = {
50         {1, "Metro Plaza Hotel", 420.00, "Wifi, Gym, Spa & Lounge"}, 
51         {2, "City View Resort", 550.00, "Wifi, Dinner, Free Parking & Airport Shuttle"}, 
52         {3, "Elite Comfort Hotel", 480.00, "Wifi, Breakfast, Conference Room & Laundry Service"}, 
53         {4, "Urban Haven Hotel", 390.00, "Wifi, Breakfast, Afternoon Tea & Business Center"}, 
54         {5, "Selangor Grand", 350.00, "Wifi, Breakfast, 24 Hours Front Desk & Lounge"} 
55     };
56     const int numRoomsSelangor = sizeof(roomsSelangor) / sizeof(roomsSelangor[0]);
57
58     // Define and initialize hotel rooms for Sabah
59     BookingRoom roomsSabah[] = {
60         {1, "Borneo Bliss Hotel", 320.00, "Wifi, Breakfast, Pool & Parking"}, 
61         {2, "Kinabalu View", 270.00, "Wifi, Breakfast, Shuttle Service & Airport Transfer"}, 
62         {3, "Sunset Coastal Resort", 380.00, "Wifi, Breakfast, Restaurant & Spa"}, 
63         {4, "Sabah Serenity Lodge", 290.00, "Wifi, Breakfast, Concierge"}, 
64         {5, "Horizon Hotel", 450.00, "Wifi, Breakfast, Business Center & Conference Rooms"} 
65     };
66     const int numRoomsSabah = sizeof(roomsSabah) / sizeof(roomsSabah[0]);
67
68     // Define and initialize hotel rooms for Sarawak
69     BookingRoom roomsSarawak[] = {
70         {1, "Riverfront Resort", 350.00, "Wifi, Dinner, Spa & Free Parking"}, 
71         {2, "Heritage City Hotel", 580.00, "Wifi, Breakfast, Pool & Laundry"}, 
72         {3, "Borneo Vista Lodge", 400.00, "Wifi, Breakfast, Gym & Spa"}, 
73         {4, "Tropical Haven Hotel", 460.00, "Wifi, Breakfast, Afternoon Tea"}, 
74         {5, "Kuching Grand Inn", 310.00, "Wifi, Breakfast, Business Center & Lounge"} 
75     };
76     const int numRoomsSarawak = sizeof(roomsSarawak) / sizeof(roomsSarawak[0]);
77
78     // Choose the room array based on locationNo
79     const BookingRoom* rooms;
80     int totalRooms;
81
82     if (locationNo == 1) {
83         rooms = roomsKualaLumpur;
84         totalRooms = numRoomsKualaLumpur;
85     }
86     else if (locationNo == 2) {
87         rooms = roomsMelaka;
88         totalRooms = numRoomsMelaka;
89     }
90     else if (locationNo == 3) {
91         rooms = roomsPenang;
92         totalRooms = numRoomsPenang;
93     }
94     else if (locationNo == 4) {
95         rooms = roomsSelangor;
96         totalRooms = numRoomsSelangor;
97     }
98     else if (locationNo == 5) {
99         rooms = roomsSabah;
100        totalRooms = numRoomsSabah;
101    }
102    else if (locationNo == 6) {
103        rooms = roomsSarawak;
104        totalRooms = numRoomsSarawak;
105    }

```

# Problem Solving and Programming

```
103
104 }  
105 else {  
106     cout << "Invalid location number." << endl;  
107     return 1;  
108 }  
  
// Add on service  
109 char service;  
110 int serviceNo;  
111 double servicePrice = 0.00;  
112 string serviceName;  
113  
114 // Validation for service  
115 do {  
116     cout << "\nDo you need any add on service (Y/N): ";  
117     cin >> service;  
118     if (service == 'Y' || service == 'y') {  
119         ADDONSERVICE();  
120  
121         cout << "Please select the service you needed(1-5) :";  
122         cin >> serviceNo;  
123  
124         // Validation for serviceNo  
125         while (cin.fail() || serviceNo < 1 || serviceNo > 5) {  
126             cin.clear(); // clear the error  
127             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input  
128             cout << "Error: Please enter a number between 1 and 5." << endl;  
129             cout << "Please select the service you needed(1-5) :";  
130             cin >> serviceNo;  
131         }  
132  
133         switch (serviceNo) {  
134             case 1:  
135                 serviceName = "High Speed Wi - Fi, RM 30.00 per day";  
136                 servicePrice = 30.00 * numberOfDays;  
137                 break;  
138             case 2:  
139                 serviceName = "Breakfast, RM 50.00 per day";  
140                 servicePrice = 50.00 * numberOfDays;  
141                 break;  
142             case 3:  
143                 serviceName = "Airport Shuttle, RM 150.00 per day";  
144                 servicePrice = 150.00 * numberOfDays;  
145                 break;  
146             case 4:  
147                 serviceName = "Room Upgrade, RM 300.00 per day";  
148                 servicePrice = 300.00 * numberOfDays;  
149                 break;  
150             case 5:  
151                 serviceName = "Mid-Stay Cleaning, RM 100.00 per day";  
152                 servicePrice = 100.00 * numberOfDays;  
153                 break;  
154         }  
155     }  
156     else if (service == 'N' || service == 'n') {  
157         serviceName = "NO EXTRA SERVICE";  
158     }  
159     else {  
160         cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No" << endl;  
161     }  
162 }  
163  
164 } while (service != 'Y' && service != 'y' && service != 'N' && service != 'n');  
  
165  
166 // Enter voucher code  
167 char gotvoucher;  
168 RebateValue memberlogin(double rebatPrice);  
169 RebateValue finalresult;  
170 do {  
171     cout << "Do you want to apply voucher code (Y/N) :";  
172     cin >> gotvoucher;  
173  
174     if (gotvoucher == 'Y' || gotvoucher == 'y') {  
175         finalresult = memberlogin(totalPriceAlldays);  
176     }  
177     else if (gotvoucher == 'N' || gotvoucher == 'n') {  
178         cout << "NO VOUCHER";  
179         finalresult.selectedvoucher = "NO VOUCHER";  
180         finalresult.rebate = 0.00;  
181     }  
182     else {  
183         cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No" << endl;  
184     }  
185 } while (gotvoucher != 'Y' && gotvoucher != 'y' && gotvoucher != 'N' && gotvoucher != 'n');  
  
186  
187  
188 // Invoice  
189 double totalPrice = 0;  
190 cout << setfill(' ') << "\n|=====|=====|=====|=====|=====|=====|=====|=====|=====|=====|";  
191 cout << " |" << "Invoice" << " |" << endl;  
192 cout << " |" << endl;  
193 cout << " |" << endl;  
194 cout << " |" << endl;  
195 cout << " |" << setw(20) << left << "Name" << ":" << setw(20) << left << name << setw(43) << " " << "Check-In Date" << setw(4) << right << ":" << setw(2) << left << showpoint << checkInDay << "/" << setw(2) << left << startMonth << "/" << startYear << " |" << endl;  
196 cout << " |" << setw(20) << left << "Phone Number" << ":" << setw(20) << left << showpoint << phoneNumber << setw(43) << " " << "Check-Out Date" << setw(4) << right << ":" << setw(2) << showpoint << left << checkOutDay << "/" << setw(2) << left << endMonth << "/" << endYear << " |" << endl;  
197 cout << " |" << setw(20) << left << "Total number of pax" << ":" << setw(20) << left << totalPax << setw(43) << " " << "Number Of Days" << setw(4) << right << ":" << setw(12) << left << numberOfDays << "|" << endl;
```

# Problem Solving and Programming

```
199 cout << "=====|" << endl;
200 cout << "|" << setw(9) << left << "BookingNo" << setw(15) << right << "Room Number" << " " << setw(15) << right << "Hotel" << setw(10) << right << " " << setw(13) << right << "Quantity" << setw(5) << right << " " << setw(20) << right << "Price Per Room" << setw(5) << right << " " << setw(17) << right << "Total Price" << setw(5) << right << " " << endl;
201 cout << "=====|" << endl;
202
203 int latestBookingIndex = BL.count - 1;
204 int roomNo = BL.bookings[latestBookingIndex].roomNo;
205 int quantity = BL.bookings[latestBookingIndex].quantity;
206 int bookingNo = BL.bookings[latestBookingIndex].bookingNumber;
207
208 for (int j = 0; j < totalRooms; ++j) {
209     if (rooms[j].roomNo == roomNo) {
210         double roomPrice = rooms[j].price;
211         double bookingTotalPrice = roomPrice * quantity;
212         totalPrice += bookingTotalPrice;
213
214         cout << " " << setw(6) << right << bookingNo << setw(7) << " " << setw(5) << right << rooms[j].roomNo << setw(7) << " " << setw(25) << " " << left << rooms[j].hotelName << " " << setw(9) << right << quantity << setw(9) << right << " " << "RM" << setw(21) << right << fixed << " " << endl;
215         setprecision(2) << rooms[j].price << setw(4) << right << "RM" << setw(18) << right << fixed << setprecision(2) << bookingTotalPrice << " " << endl;
216         // Print services
217         cout << "|Services: " << setw(104) << left << rooms[j].service << " " << endl;
218         cout << " " << setw(115) << right << " " << endl;
219     }
220 }
221
222 // Calculation for SST, Discount and Grand Total
223 const double SSTRATE = 0.06;
224 double SST = totalPriceAllDays * SSTRATE;
225 double grandtotal = (totalPriceAllDays + SST + servicePrice) - finalresult.rebate;
226
227 cout << "=====|" << endl;
228 cout << " " << setw(94) << left << "Total before taxes" << "RM" << setw(18) << fixed << setprecision(2) << right << totalPriceAllDays << " " << endl;
229 cout << " " << setw(94) << left << "SST" << "RM" << setw(18) << fixed << setprecision(2) << right << SST << " " << endl;
230 cout << " " << setw(94) << left << "Service Name" << "RM" << setw(18) << fixed << setprecision(2) << right << servicePrice << " " << endl;
231 cout << " " << setw(93) << left << finalresult.selectedVoucher << "RM" << setw(18) << fixed << setprecision(2) << right << finalresult.rebate << " " << endl;
232 cout << " " << endl;
233 cout << " " << setw(94) << left << "Grand Total" << "RM" << setw(18) << fixed << setprecision(2) << right << grandtotal << " " << endl;
234 cout << "=====|" << endl;
235
236 // Select Payment Method
237 int paymentMethod, accountNo, cardNumber, PIN;
238 string paymentMethodName;
239 double cashAmount, change = 0.00;
240
241 // Call table
242 paymentMethodTable();
243
244 cout << setw(56) << left << "Please select payment method (1-5)" << ":" ;
245 cin >> paymentMethod;
246
247 // Validation for paymentMethod
248 while (cin.fail() || paymentMethod < 1 || paymentMethod > 5) {
249     cin.clear(); // clear the error
250     cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
251     cout << "Error: Please enter a number between 1 and 5." << endl;
252     cout << setw(56) << left << "Please select payment method (1-5)" << ":" ;
253     cin >> paymentMethod;
254 }
255
256 switch (paymentMethod) {
257 case 1:
258     paymentMethodName = "Credit/Debit Cards";
259     cardNumber = getValidatedInput("Card Number");
260     PIN = pinValidation("PIN");
261     cashAmount = grandtotal;
262     break;
263 case 2:
264     paymentMethodName = "Bank Transfer";
265     accountNo = getValidatedInput("Account Number");
266     PIN = pinValidation("PIN");
267     cashAmount = grandtotal;
268     break;
269 case 3:
270     paymentMethodName = "Touch N Go";
271     PIN = pinValidation("PIN");
272     cashAmount = grandtotal;
273     break;
274 case 4:
275     paymentMethodName = "Apple Pay";
276     PIN = pinValidation("PIN");
277     cashAmount = grandtotal;
278     break;
279 case 5:
280     do {
281         paymentMethodName = "Cash Payment";
282         cashAmount = getValidatedInput("Payment Amount");
283         if (cashAmount >= grandtotal) {
284             change = cashAmount - grandtotal;
285         }
286         else {
287             cout << "Insufficient amount. Please make payment again." << endl;
288         }
289     } while (cashAmount < grandtotal);
290 }
```

# Problem Solving and Programming

---

```

289     .... break;
290 }
291
292 // Booking Confirmation
293 cout << setfill(' ') << "\n" =====| " << endl;
294 cout << "  Elegant Hideaway Airbnb      |" << endl;
295 cout << "|eleganthideaway@gmail.com      +6018-889 8888      Elegant Kuala Lumpur      www.eleganthideaway.com|" << endl;
296 cout << "=====| " << endl;
297 cout << "|Booking Confirmation      |" << endl;
298 cout << "|=====| " << endl;
299 cout << "|Dear," << setw(10) << left << name << "|" << endl;
300 cout << "|Your hotel booking is confirmed.      |" << endl;
301 cout << "|Thank you for booking your stay with Elegant Hideaway Airbnb. We are looking forward to your visit      |" << endl;
302 cout << "|Your booking details are as follow :      |" << endl;
303 cout << "|=====| " << endl;
304 cout << "|" << setw(20) << left << "Name" << ":" << setw(20) << left << name << setw(43) << " " << "Check-In Date" << setw(4) << right << ":" << endl;
305 cout << "|" << setw(2) << left << showpoint << checkInDay << "/" << setw(2) << left << startMonth << "/" << startYear << " |" << endl;
306 cout << "|" << setw(20) << left << "Phone Number" << ":" << setw(20) << left << showpoint << phoneNumber << setw(43) << " " << "Check-Out Date" << endl;
307 cout << "|" << setw(4) << right << ":" << setw(2) << showpoint << left << checkOutDay << "/" << setw(2) << left << endMonth << "/" << endYear << " |" << endl;
308 cout << "|" << setw(20) << left << "Total number of pax" << ":" << setw(20) << left << totalPax << setw(43) << " " << "Number Of Days" << setw(4) << right << ":" << setw(12) << left << numberOfDays << " |" << endl;
309 cout << "|=====| " << endl;
310 cout << "|" << setw(9) << left << "BookingNo" << setw(15) << right << "Room Number" << ":" << setw(15) << right << "Hotel" << setw(10) << right << endl;
311 cout << "|" << setw(13) << right << "Quantity" << setw(5) << right << ":" << setw(20) << right << "Price Per Room" << setw(5) << right << ":" << setw(17) << endl;
312 cout << "|" << setw(13) << right << "Total Price" << setw(5) << right << " |" << endl;
313 cout << "|=====| " << endl;
314
315 for (int j = 0; j < totalRooms; ++j) {
316     if (rooms[j].roomNo == roomNo) {
317         double roomPrice = rooms[j].price;
318         double bookingTotalPrice = roomPrice * quantity;
319         totalPrice += bookingTotalPrice;
320
321         cout << "|" << setw(6) << right << bookingNo << setw(7) << " " << setw(5) << right << rooms[j].roomNo << setw(7) << " " << setw(25) << endl;
322         cout << "|" << setw(2) << left << rooms[j].hotelName << " " << setw(9) << right << quantity << setw(9) << right << " " << "RM" << setw(21) << right << fixed << endl;
323         cout << "|" << setw(2) << rooms[j].price << setw(4) << right << "RM" << setw(18) << right << fixed << setprecision(2) << bookingTotalPrice << endl;
324         cout << "|" << endl;
325         // Print services
326         cout << "|Services: " << setw(104) << left << rooms[j].service << " |" << endl;
327         cout << "|" << setw(115) << right << " |" << endl;
328     }
329 }
330
331 cout << "|=====| " << endl;
332 cout << "|Payment Summary :      |" << endl;
333 cout << "|=====| " << endl;
334 cout << "|" << setw(94) << left << "Total before taxes" << "RM" << setw(18) << fixed << setprecision(2) << right << totalPriceAlldays << " |" << endl;
335 cout << "|" << setw(94) << left << "SST" << "RM" << setw(18) << fixed << setprecision(2) << right << SST << " |" << endl;
336 cout << "|" << setw(94) << left << serviceName << "RM" << setw(18) << fixed << setprecision(2) << right << servicePrice << " |" << endl;
337 cout << "|" << setw(93) << left << finalresult.selectedvoucher << "RM" << setw(18) << fixed << setprecision(2) << right << finalresult.rebate << endl;
338 cout << "|" << endl;
339 cout << "|Grand Total" << "RM" << setw(18) << fixed << setprecision(2) << right << grandtotal << " |" << endl;
340 cout << "|" << setw(94) << left << "Payment Method :" << setw(20) << right << paymentMethodName << " |" << endl;
341 cout << "|" << setw(94) << left << "Payment Amount :" << "RM" << setw(18) << fixed << setprecision(2) << right << cashAmount << " |" << endl;
342 cout << "|" << setw(94) << left << "Change\t :" << "RM" << setw(18) << fixed << setprecision(2) << right << change << " |" << endl;
343 cout << "|Additional Information :      |" << endl;
344 cout << "|Please CHECK-IN from 3 PM and CHECK-OUT before 12 PM. If delayed check-out, we will charge RM 20 per hour.      |" << endl;
345 cout << "|" << endl;
346 cout << "|If you have any questions, please don't hesitate to contact us.      |" << endl;
347 cout << "|We hope you enjoy your stay with us !!!      |" << endl;
348
349 return grandtotal;
350 }

```

## 2.12 Kuala Lumpur Table

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5
6 void KualaLumpurArea() {
7     // Kuala Lumpur Area
8     cout << "\nKuala Lumpur" << endl;
9
10    // Title of the table
11    cout << setfill('=') << left << setw(153) << "=" << endl;
12    cout << setfill(' ') << "|" << setw(3) << right << "No" << setw(2) << "|" << setw(14) << "Name" << setw(11) << "|" << setw(18) << "Location" << setw(11) << endl;
13    cout << setw(10) << "Price" << setw(6) << "|" << setw(13) << "Bed Type" << setw(6) << "|";
14    cout << setw(14) << "Room Type" << setw(6) << "|" << setw(14) << "Room Size" << setw(6) << "|" << setw(12) << "Service" << setw(6) << "|" << endl;
15
16    // Information of Cubic Botanical
17    cout << setfill('=') << setw(154) << " " << endl;
18    cout << setfill(' ') << "|" << setw(3) << "1" << setw(2) << "|" << setw(24) << left << "Cubic Botanical" << "|" << setw(28) << left << "12, Jln Pantai Sentral 3, " << endl;
19    cout << setw(9) << left << "RM" << setw(6) << left << "650.00" << " " << setw(13) << right << " Single " << setw(6) << "|";
20    cout << setw(14) << "Deluxe" << setw(6) << "|" << setw(14) << "20-30 sqm" << setw(6) << "|" << setw(12) << " WiFi " << setw(6) << " " << endl;
21    cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << " " << setw(28) << left << "bangsar south, 59200 KL" << endl;
22    cout << setw(9) << left << " " << setw(6) << left << " " << setw(13) << right << " " << setw(6) << "|";
23    cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(13) << "Breakfast" << setw(5) << " " << endl;
24
25    // Information of Mont Kiara
26    cout << setfill('=') << setw(154) << " " << endl;
27    cout << setfill(' ') << "|" << setw(3) << "2" << setw(2) << "|" << setw(24) << left << "Mont Kiara" << "|" << setw(28) << left << "1, Jalan Kiara, Mont Kiara" << endl;
28    cout << setw(9) << left << "RM" << setw(6) << left << "200.00" << " " << setw(13) << right << " Queen " << setw(6) << "|";
29    cout << setw(14) << "Standard" << setw(6) << "|" << setw(14) << "20-30 sqm" << setw(6) << "|" << setw(12) << " WiFi " << setw(6) << " " << endl;
30    cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << " " << setw(28) << left << "50480 Kuala Lumpur" << endl;

```

# Problem Solving and Programming

```
31     cout << setw(9) << left << "" << setw(6) << left << "" << setw(13) << right << "" << setw(6) << "|";
32     cout << setw(14) << "" << setw(6) << "|" << setw(14) << "" << setw(6) << "|" << setw(13) << "Breakfast" << setw(5) << "" |" << endl;
33
34 // Information of Urban Retreat
35 cout << setfill('=') << setw(154) << " " << endl;
36 cout << setfill(' ') << "|" << setw(3) << "3" << setw(2) << "" |" << setw(24) << left << "Urban Retreat" << "" |" << setw(28) << left << "15, Jalan
    Pudu, 55100 " << "|";
37 cout << setw(9) << left << "RM" << setw(6) << left << "450.00" << "" |" << setw(13) << right << " Double " << setw(6) << "|";
38 cout << setw(14) << "Superior" << setw(6) << "" |" << setw(14) << "25-35 sqm" << setw(6) << "" |" << setw(12) << " WiFi " << setw(6) << "" |" << endl;
39 cout << setw(9) << left << "" << setw(6) << left << "" << setw(24) << left << "" << setw(28) << left << "Kuala Lumpur" << "|";
40 cout << setw(14) << "" << setw(6) << "" |" << setw(14) << "" << setw(6) << "" |" << setw(13) << "Breakfast" << setw(5) << "" |" << endl;
41 cout << setfill(' ') << "" |" << setw(3) << "2" << setw(2) << "" |" << setw(24) << left << "" << "" |" << setw(28) << left << "" << "" |" ;
42 cout << setw(9) << left << "" << setw(6) << left << "" << setw(13) << right << "" << setw(6) << "" |" ;
43 cout << setw(14) << "" << setw(6) << "" |" << setw(14) << "" << setw(6) << "" |" << setw(12) << "Gym" << setw(6) << "" |" << endl;
44
45 // Information of Luxe Residence
46 cout << setfill(' =') << setw(154) << " " << endl;
47 cout << setfill(' ') << "|" << setw(3) << "4" << setw(2) << "" |" << setw(24) << left << "Luxe Residence" << "" |" << setw(28) << left << "8, Jalan Tun
    Razak, 55000" << "|";
48 cout << setw(9) << left << "RM" << setw(6) << left << "950.00" << "" |" << setw(13) << right << " King " << setw(6) << "|";
49 cout << setw(14) << "Executive" << setw(6) << "" |" << setw(14) << "35-45 sqm" << setw(6) << "" |" << setw(12) << " WiFi " << setw(6) << "" |" << endl;
50 cout << setfill(' ') << "" |" << setw(3) << "3" << setw(2) << "" |" << setw(24) << left << "" << "" |" << setw(28) << left << "Kuala Lumpur" << "|";
51 cout << setw(9) << left << "" << setw(6) << left << "" << setw(13) << right << "" << setw(6) << "" |" ;
52 cout << setw(14) << "" << setw(6) << "" |" << setw(14) << "" << setw(6) << "" |" << setw(13) << "Breakfast" << setw(5) << "" |" << endl;
53 cout << setfill(' ') << "" |" << setw(3) << " " << setw(2) << "" |" << setw(24) << left << "" << "" |" << setw(28) << left << "" << "" |" ;
54 cout << setw(9) << left << "" << setw(6) << left << "" << setw(13) << right << "" << setw(6) << "" |" ;
55 cout << setw(14) << "" << setw(6) << "" |" << setw(14) << "" << setw(6) << "" |" << setw(12) << "Lounge" << setw(6) << "" |" << endl;
56
57 // Information of Panorama Hotel
58 cout << setfill(' =') << setw(154) << " " << endl;
59 cout << setfill(' ') << "|" << setw(3) << "5" << setw(2) << "" |" << setw(24) << left << "Panorama Hotel" << "" |" << setw(28) << left << "6, Jalan
    Changkat Thambi," << "|";
60 cout << setw(9) << left << "RM" << setw(6) << left << "850.00" << "" |" << setw(13) << right << " Queen " << setw(6) << "|";
61 cout << setw(14) << "Suite" << setw(6) << "" |" << setw(14) << "30-40 sqm" << setw(6) << "" |" << setw(12) << " WiFi " << setw(6) << "" |" << endl;
62 cout << setfill(' ') << "" |" << setw(3) << "2" << setw(2) << "" |" << setw(24) << left << "" << "" |" << setw(28) << left << "Kuala Lumpur" << "|";
63 cout << setw(9) << left << "" << setw(6) << left << "" << setw(13) << right << "" << setw(6) << "" |" ;
64 cout << setw(14) << "" << setw(6) << "" |" << setw(14) << "" << setw(6) << "" |" << setw(13) << "Breakfast" << setw(5) << "" |" << endl;
65 cout << setfill(' ') << "" |" << setw(3) << "2" << setw(2) << "" |" << setw(24) << left << "" << "" |" << setw(28) << left << "" << "" |" ;
66 cout << setw(9) << left << "" << setw(6) << left << "" << setw(13) << right << "" << setw(6) << "" |" ;
67 cout << setw(14) << "" << setw(6) << "" |" << setw(14) << "" << setw(6) << "" |" << setw(12) << "Pool" << setw(6) << "" |" << endl;
68
69 cout << setfill(' =') << setw(154) << " " << setfill(' ') << endl;
70 }
```

## 2.13 Location Number

```
1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4
5  using namespace std;
6
7  int LocationNumber() {
8
9      int locationNo = 0;
10
11     // Display welcome message and location options
12     cout << "\nWelcome to Elegant Hideaway Airbnb" << endl;
13     cout << "Please let us know the location you wish to travel to at your convenience" << endl;
14     cout << setfill(' -') << setw(75) << " " << setfill(' ') << endl;
15     cout << "1. Kuala Lumpur" << endl;
16     cout << "2. Melaka" << endl;
17     cout << "3. Penang" << endl;
18     cout << "4. Selangor" << endl;
19     cout << "5. Sabah" << endl;
20     cout << "6. Sarawak" << endl;
21     cout << "Please enter the number of the location: ";
22     cin >> locationNo;
23
24     // Validate location input
25     while (cin.fail() || locationNo < 1 || locationNo > 6) {
26         cin.clear(); // Clear error flag
27         cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
28         cout << "Error: Please enter a number between 1 and 6.\n" << endl;
29         cout << "Welcome to Elegant Hideaway Airbnb" << endl;
30         cout << "Please let us know the location you wish to travel to at your convenience" << endl;
31         cout << setfill(' -') << setw(75) << " " << setfill(' ') << endl;
32         cout << "1. Kuala Lumpur" << endl;
33         cout << "2. Melaka" << endl;
34         cout << "3. Penang" << endl;
35         cout << "4. Selangor" << endl;
36         cout << "5. Sabah" << endl;
37         cout << "6. Sarawak" << endl;
38         cout << "Please enter the number of the location: ";
39         cin >> locationNo;
40     }
41
42     // Confirm the selected location
43     switch (locationNo) {
44     case 1: cout << "You have selected Kuala Lumpur." << endl; break;
45     case 2: cout << "You have selected Melaka." << endl; break;
46     case 3: cout << "You have selected Penang." << endl; break;
47     case 4: cout << "You have selected Selangor." << endl; break;
48     case 5: cout << "You have selected Sabah." << endl; break;
49     case 6: cout << "You have selected Sarawak." << endl; break;
50     default: cout << "Invalid selection." << endl; break;
51
52
53     return locationNo;
54 }
```

## 2.14 Logo

## 2.15 Melaka Table

## 2.16 Member Login

```

1  // #include <iostream>
2  // #include <iomanip>
3  // #include <string>
4  // #include <cctype>
5  // #include <regex>
6  // #include <limits>
7  // #include "BookingSystem.h"
8
9  using namespace std;
10
11 struct Membership {
12     static const int NUM_MEMBERS = 3;
13     string correctMemberIDs[NUM_MEMBERS] = { "member1", "member2", "member3" };
14     string correctPasswords[NUM_MEMBERS] = { "password1", "password2", "password3" };
15
16     string name;
17     string phoneNumber;
18     string email;
19
20     static bool isValidName(const string& name) {
21         if (name.empty()) return false;
22         for (char ch : name) {
23             if (!isalpha(ch) && !isspace(ch)) return false;
24         }
25         return true;
26     }
27
28     static bool isValidPhoneNumber(const string& phoneNumber) {
29         if (phoneNumber.length() < 10 || phoneNumber.length() > 11) return false;
30         for (char c : phoneNumber) {
31             if (!isdigit(c)) return false;
32         }
33         return true;
34     }
35
36     static bool isValidEmail(const string& email) {
37         regex emailPattern(R"^(^[\w\.-]+@[^\w\.-]+\.[^\w\.-]+)$");
38         return regex_match(email, emailPattern);
39     }
40
41     void handleMemberSignUp() {
42         string inputMemberID;
43         string inputPassword;
44         bool isAuthenticated = false;
45
46         while (!isAuthenticated) {
47             cout << "Member login" << endl;
48             cout << "-----" << endl;
49
50             cout << "Member ID: ";
51             getline(cin, inputMemberID);
52
53             cout << "Password: ";
54             getline(cin, inputPassword);
55
56             bool idExists = false;
57             for (int i = 0; i < NUM_MEMBERS; ++i) {
58                 if (inputMemberID == correctMemberIDs[i]) {
59                     idExists = true;
60                 if (inputPassword == correctPasswords[i]) {
61                     cout << "Login successful!" << endl;
62                     cout << "\nEnjoy our benefits now!" << endl;
63                     cout << "+" << setfill(' ') << setw(48) << "+" << endl;
64                     cout << "+" << " No |" << setfill(' ') << setw(20) << " Seasonal Promotion " << setfill(' ') << setw(18) << " | Promo Code" << " |
65                     setfill(' ') << setw(6) << " |" << endl;
66                     cout << "+" << setfill(' ') << setw(48) << "+" << endl;
67                     cout << "+" << " 1 | Chinese New year" << setfill(' ') << setw(18) << " | CNV8888" << setfill(' ') << setw(8) << " |" << endl;
68                     cout << "+" << " 2 | Hari Raya" << setfill(' ') << setw(24) << " | HR4444" << setfill(' ') << setw(10) << " |" << endl;
69                     cout << "+" << " 3 | Deepavali" << setfill(' ') << setw(25) << " | DP6666" << setfill(' ') << setw(9) << " |" << endl;
70                     cout << "+" << " 4 | Merry Christmas" << setfill(' ') << setw(19) << " | MCL225" << setfill(' ') << setw(9) << " |" << endl;
71                     cout << "+" << " 5 | Valentine" << setfill(' ') << setw(27) << " | V2141314" << setfill(' ') << setw(7) << " |" << endl;
72                     cout << "+" << setfill(' ') << setw(48) << "+" << endl;
73                     isAuthenticated = true;
74                 } else {
75                     cout << "Incorrect Password. Please try again." << endl;
76                 }
77                 break;
78             }
79         }
80
81         if (!idExists) {
82             cout << "Incorrect Member ID. Please try again." << endl;
83         }
84     }
85
86     void handleNewMemberRegistration() {
87         cout << "\nPlease fill up the information below to become our member!" << endl;
88         cout << "Registration" << endl;
89         cout << "-----" << endl;
90
91         while (true) {
92             cout << "Name: ";
93             getline(cin >> ws, name);
94             if (isValidName(name)) {
95                 break;
96             } else {
97                 cout << "Invalid name. Please enter a name containing only letters and spaces." << endl;
98             }
99         }
100    }

```

## Problem Solving and Programming

---

```
102
103     while (true) {
104         cout << "Phone number: ";
105         getline(cin, phoneNumber);
106         if (isValidPhoneNumber(phoneNumber)) {
107             break;
108         }
109         else {
110             cout << "Invalid phone number. It should be numeric and between 10 to 11 digits long." << endl;
111         }
112     }
113
114     while (true) {
115         cout << "Email: ";
116         getline(cin, email);
117         if (isValidEmail(email)) {
118             break;
119         }
120         else {
121             cout << "Invalid email format. Please enter a valid email address." << endl;
122         }
123     }
124
125     cout << "\nThank you for becoming our member. Below are the member details." << endl;
126     cout << "Name : " << name << endl;
127     cout << "Phone number: " << phoneNumber << endl;
128     cout << "Email : " << email << endl;
129     cout << "This is your new member code. Make a booking now to enjoy our benefits!" << endl;
130     cout << "Member Code : WELCOME123" << endl;
131 }
132
133 RebateValue memberlogin(double rebatePrice) {
134     int choice_membership;
135     bool valid_input = false;
136     Membership membership; // Create an instance of Membership
137     RebateValue result;
138
139     while (!valid_input) {
140         cout << "Are you a member?" << endl;
141         cout << "-----" << endl;
142         cout << "1. Yes" << endl;
143         cout << "2. No" << endl;
144         cout << "Please enter your choice (1 or 2): ";
145
146         cin >> choice_membership;
147         cin.clear(); // Clear input buffer
148         cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
149
150         if (cin.fail()) {
151             cin.clear(); // Clear the error flag
152             cout << "Invalid input. Please enter a number (1 or 2)." << endl;
153         }
154
155         else if (choice_membership == 1) {
156             membership.handleMemberSignUp();
157             RebateValue Voucher(double rebatePrice);
158             result = Voucher(rebatePrice); // Calling Voucher() function
159             valid_input = true;
160         }
161
162         else if (choice_membership == 2) {
163             membership.handleNewMemberRegistration();
164             RebateValue Voucher(double rebatePrice);
165             result = Voucher(rebatePrice); // Calling Voucher() function
166             valid_input = true;
167         }
168         else {
169             cout << "Invalid choice. Please enter 1 or 2." << endl;
170         }
171     }
172
173     return result;
174 }
```

## 2.17 Menu List

```

1  #include <iostream>
2  #include <iomanip>
3  #include <limits> // for numeric_limits
4  #include "BookingSystem.h"
5
6  using namespace std;
7
8  // Function declarations
9  void displayFeedback();
10 BookingList initializePrePopulatedBookings();
11 BookingList returnStruct(BookingList& bookingList);
12 void cancelBooking(BookingList& BL);
13 double displayOverview(int locationNo, int totalPax, string name, string phoneNumber, int startMonth, int endMonth, int startYear, int endYear, int
14 checkInDay, int checkOutDay, int numberOfDays, BookingList& BL);
15 double displayInvoice(int locationNo, int totalPax, string name, string phoneNumber, int startMonth, int endMonth, int startYear, int endYear, int
16 checkInDay, int checkOutDay, int numberOfDays, double totalPriceAlldays, BookingList& BL);
17 void Report(int locationNo, double grandtotal, BookingList& BL);
18 int LocationNumber();
19 int NumberOfPax();
20 void CallRoomTable(int locationNo);
21 string NameInput();
22 string InputPhoneNumber();
23 bool staffLogin();
24 void checkInAndCheckout(BookingList &BL);
25
26 // Declare Function Table
27 void monthYear();
28 void days28();
29 void days29();
30 void days30();
31 void days31();
32 void logo();
33 void Menulisttable();
34
35 // Calculation Number of Days Function
36 int NoOfDays(int startMonth, int endMonth, int startYear, int endYear, int checkInDay, int checkOutDay);
37
38 int main() {
39     int option;
40     char continueOption = 'Y'; // Initialize to 'Y' to enter the loop
41     int locationNo = 0;
42     double grandtotal = 0;
43     bool loggedIn;
44
45     // Initialize pre-populated bookings
46     BookingList prePopulatedBookings = initializePrePopulatedBookings();
47     BookingList myBookings = prePopulatedBookings; // Use the initialized bookings for this session
48
49     logo();
50
51     do {
52         cout << "\nWelcome to Elegant Hideaway, select an option to proceed" << endl;
53         Menulisttable(); // Display menu
54         cout << "Your option: ";
55         cin >> option;
56
57         // Validate option input
58         while (cin.fail() || option < 1 || option > 6) {
59             cin.clear(); // Clear the error
60             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Ignore unwanted characters for input
61             cout << "\nInvalid option. Please enter a number between 1 - 6" << endl;
62             Menulisttable();
63             cout << "Your option: ";
64             cin >> option;
65         }
66
67         switch (option) {
68             case 1:
69                 cout << "\nYou selected Order Room";
70
71                 // Call Function LocationNumber
72                 locationNo = LocationNumber();
73
74                 do {
75
76                     // Call Function Number of Pax
77                     int totalPax = NumberOfPax();
78
79                     // Selection Date Month Year
80                     int startMonth = 0, endMonth = 0, startYear = 0, endYear = 0, checkInDay = 0, checkOutDay = 0, numberOfDays = 0;
81
82                     do {
83                         cout << "Select your Check-In & Check-Out Date\n";
84                         monthYear();
85
86                         cout << "\nStart Month: ";
87                         cin >> startMonth;
88
89                         while (cin.fail() || startMonth < 1 || startMonth > 12) {
90                             cin.clear(); // clear the error
91                             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
92                             cout << "Invalid month. Please enter a number between 1 - 12" << endl;
93                             cout << "\nStart Month: ";
94                             cin >> startMonth;
95                         }
96
97                         cout << "End Month: ";
98                         cin >> endMonth;
99
100                        while (cin.fail() || endMonth < 1 || endMonth > 12) {
101                            cin.clear(); // clear the error
102                            cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input

```

# Problem Solving and Programming

```
101 cout << "Invalid month. Please enter a number between 1 - 12" << endl;
102 cout << "\nEnd Month: ";
103 cin >> endMonth;
104 }
105
106 cout << "Start Year: ";
107 cin >> startYear;
108
109 while (cin.fail() || startYear < 2024 || startYear > 2025) {
110     cin.clear(); // clear the error
111     cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
112     cout << "Not available year. Please enter 2024 or 2025" << endl;
113     cout << "\nStart Year: ";
114     cin >> startYear;
115 }
116
117 cout << "End Year: ";
118 cin >> endYear;
119
120 while (cin.fail() || endYear < 2024 || endYear > 2025) {
121     cin.clear(); // clear the error
122     cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
123     cout << "Not available year. Please enter 2024 or 2025" << endl;
124     cout << "\nEnd Year: ";
125     cin >> endYear;
126 }
127
128 cout << "\nSelect your Check-In Date, _/_<< startMonth << "/" << startYear << "\n" << endl;
129
130 // for february in 2024
131 if (startYear == 2024 && startMonth == 2) {
132     days29();
133
134     cout << "\nCheck-In Date: ";
135     cin >> checkInDay;
136
137     while (cin.fail() || checkInDay < 1 || checkInDay > 29) {
138         cin.clear(); // clear the error
139         cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
140         cout << "Not available date. Please enter a number between 1 - 29" << endl;
141         cout << "\nCheck-In Date: ";
142         cin >> checkInDay;
143     }
144     cout << "\n" << checkInDay << "/" << startMonth << "/" << startYear << ", Check-In Date selected successfully.\n" << endl;
145 }
146 else {
147     // for february in 2025
148     if (startYear == 2025 && startMonth == 2) {
149         days28();
150
151         cout << "\nCheck-In Date: ";
152         cin >> checkInDay;
153
154         while (cin.fail() || checkInDay < 1 || checkInDay > 28) {
155             cin.clear(); // clear the error
156             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
157             cout << "Not available date. Please enter a number between 1 - 28" << endl;
158             cout << "\nCheck-In Date: ";
159             cin >> checkInDay;
160         }
161         cout << "\n" << checkInDay << "/" << startMonth << "/" << startYear << ", Check-In Date selected successfully.\n" << endl;
162     }
163 else {
164     // for months with 31 days
165     if (startMonth == 1 || startMonth == 3 || startMonth == 5 || startMonth == 6 || startMonth == 8 || startMonth == 10 || startMonth == 12) {
166         days31();
167
168         cout << "\nCheck-In Date: ";
169         cin >> checkInDay;
170
171         while (cin.fail() || checkInDay < 1 || checkInDay > 31) {
172             cin.clear(); // clear the error
173             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
174             cout << "Not available date. Please enter a number between 1 - 31" << endl;
175             cout << "\nCheck-In Date: ";
176             cin >> checkInDay;
177         }
178         cout << "\n" << checkInDay << "/" << startMonth << "/" << startYear << ", Check-In Date selected successfully.\n" << endl;
179     }
180 else {
181     if (startMonth == 4 || startMonth == 7 || startMonth == 9 || startMonth == 11) {
182         // for months with 30 days
183         days30();
184
185         cout << "\nCheck-In Date: ";
186         cin >> checkInDay;
187
188         while (cin.fail() || checkInDay < 1 || checkInDay > 30) {
189             cin.clear(); // clear the error
190             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // ignore unwanted characters for input
191             cout << "Not available date. Please enter a number between 1 - 30" << endl;
192             cout << "\nCheck-In Date: ";
193             cin >> checkInDay;
194         }
195         cout << "\n" << checkInDay << "/" << startMonth << "/" << startYear << ", Check-In Date selected successfully.\n" << endl;
196     }
197 }
198 }
```

# Problem Solving and Programming

# Problem Solving and Programming

```
297 // Call Function Invoice
298 grandtotal += displayInvoice(locationNo, totalPax, name, phoneNumber, startMonth, endMonth, startYear, endYear, checkInDay, checkOutDay, ▷
299     numberofDays, totalPriceAlldays, myBookings);
300
301 cout << "Do you want to continue order? (Y/N): ";
302 cin >> continueOption;
303 // Validate continueOrder input
304 while (continueOption != 'Y' && continueOption != 'N' && continueOption != 'n') {
305     cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No: ";
306     cin >> continueOption;
307 }
308 } while (continueOption == 'Y' || continueOption == 'y');
309 break;
310 case 2:
311 cout << "\nYou selected Check-In & Check-Out" << endl;
312 loggedIn = stafflogin();
313 if (loggedIn) {
314     checkInAndCheckOut(myBookings);
315 }
316 continueOption = 'N';
317 break;
318
319 case 3:
320 cout << "\nYou selected Cancellation" << endl;
321 do {
322     cancelBooking(myBookings); // Pass the bookings to cancel
323     cout << "Do you want to continue cancellation? (Y/N): ";
324     cin >> continueOption;
325     // Validate continueOrder input
326     while (continueOption != 'Y' && continueOption != 'N' && continueOption != 'n') {
327         cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No: ";
328         cin >> continueOption;
329     }
330 } while (continueOption == 'Y' || continueOption == 'y');
331 break;
332 case 4:
333 do {
334     cout << "\nYou selected Report (Staff)" << endl;
335     loggedIn = stafflogin();
336 if (loggedIn) {
337     cout << "Staff logged in successfully. Generating report..." << endl;
338     // Call the Report function with correct parameters
339     Report(locationNo, grandtotal, myBookings);
340 }
341 else {
342     cout << "Login failed. Cannot generate report." << endl;
343 }
344
345 cout << "Do you want to continue Report? (Y/N): ";
346 cin >> continueOption;
347
348 // Validate continueOption input
349 while (continueOption != 'Y' && continueOption != 'y' && continueOption != 'N' && continueOption != 'n') {
350     cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No: ";
351     cin >> continueOption;
352 }
353 } while (continueOption == 'Y' || continueOption == 'y');
354 break;
355 case 5:
356 cout << "\nYou selected Feedback" << endl;
357 displayFeedback(); // Collect and process feedback
358 continueOption = 'N';
359 break;
360
361 case 6:
362 cout << "\nThank you for visiting Elegant Hideaway, see you next time!" << endl;
363 continueOption = 'Y';
364 break;
365
366 default:
367 cout << "Invalid option. Please try again." << endl;
368 break;
369 }
370
371 } while (continueOption == 'N' || continueOption == 'n');
372
373 return 0;
374 }
```

## 2.18 Menulisttable

```
1 #include <iostream>
2 #include <iomanip>
3 using namespace std;
4
5 void Menulisttable() {
6     cout << setfill('-') << setw(40) << "=" << endl;
7     cout << setfill(' ') << setw(15) << " " << "Menu List" << endl;
8     cout << setfill('-') << setw(40) << "=" << endl;
9     cout << "1. Order Room" << endl;
10    cout << "2. Check-In & Check-Out" << endl;
11    cout << "3. Cancellation" << endl;
12    cout << "4. Report (Staff)" << endl;
13    cout << "5. Feedback" << endl;
14    cout << "6. Exit" << endl;
15    cout << setfill('-') << setw(40) << "=" << endl;
16 }
```

## 2.19 Month Year Table

```

1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  void monthYear() {
6      cout << setfill('=') << setw(40) << "=" << endl;
7      cout << setfill(' ') << "|" << setw(10) << " " << "Month" << setw(8) << " " << setw(2) << " " << "|" << setw(4) << " " << "Year" << setw(5) << right << " "
8          << " " << endl;
9      cout << setfill(' ') << setw(40) << "=" << endl;
10     cout << setfill(' ') << "1" << setw(3) << " " << "1" << setw(2) << " " << "2" << setw(2) << " " << "3" << setw(3) << " " << "4" << setw(3) << " " << "5"
11         << setw(3) << " " << "6" << setw(3) << " " << "7" << setw(2) << " " << "8" << setw(2) << " " << "9" << setw(2) << " " << "10" << setw(2) << " " << "11" << setw(2) << " "
12         << " " << "12" << setw(18) << setw(3) << " " << "|" << setw(4) << " " << "2024" << setw(5) << right << " " << endl;
13     cout << "|" << setw(3) << " " << "7" << setw(2) << " " << "8" << setw(2) << " " << "9" << setw(2) << " " << "10" << setw(2) << " " << "11" << setw(2) << " "
14         << " " << "12" << setw(18) << setw(3) << " " << "|" << setw(4) << " " << "2025" << setw(5) << right << " " << endl;
15     cout << setfill('=') << setw(40) << "=" << endl;
16 }
```

## 2.20 Name

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4
5  using namespace std;
6
7  // Declare Validation Name Function
8  bool validateName(const string& name);
9
10 string NameInput(){
11
12     //User input the information
13     string name;
14
15     cout << "Please enter your information below" << endl;
16     cout << "-----" << endl;
17
18     cin.ignore();
19     do {
20         cout << "Name\t\t: ";
21         getline(cin, name);
22         if (!validateName(name)) {
23             cout << "Invalid name : Please try again." << endl;
24         }
25     } while (!validateName(name));
26
27     return name;
28 }
```

## 2.21 Number of Days

```

1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  // function to check whether the year is leap year
6  bool isLeapYear(int year) {
7      return (year % 4 == 0 && (year % 100 != 0 || year % 400 == 0));
8
9
10 // function to get the number of days in a month
11 int daysInMonth(int month, int year) {
12     switch (month) {
13         case 1: case 3: case 5: case 6: case 8: case 10: case 12:
14             return 31;
15         case 4: case 7: case 9: case 11:
16             return 30;
17         case 2:
18             return isLeapYear(year) ? 29 : 28;
19     }
20 }
21
22 // function to calculate number of days from 1 Jan to a specific date
23 int daysFromStartOfYear(int day, int month, int year) {
24     int days = day; // start with the days in start/endMonth
25     for (int m = 1; m < month; ++m) { // loop through all the months before start/endMonth
26         days += daysInMonth(m, year); // add number of days in each those months
27     }
28     return days;
29 }
30
31 // function to calculate the number of days between two dates
32 int NoOfDays(int startMonth, int endMonth, int startYear, int endYear, int checkInDay, int checkOutDay) {
33     int numberofDays;
34
35     if (startYear == endYear) {
36         // same year calculation
37         // days from 1 Jan to check-out date - days from 1 Jan to check-in date + 1 to include both checkin & checkout days
38         numberofDays = daysFromStartOfYear(checkOutDay, endMonth, endYear) - daysFromStartOfYear(checkInDay, startMonth, startYear) + 1;
39     }
40     else {
41         // diff year calculation
42         // get total number of days in startYear(2024/2025) - days from 1 Jan to check-in date + 1 to include check-in day + days from 1 Jan to check-out +
43         // date
44         numberofDays = (daysFromStartOfYear(daysInMonth(startMonth, startYear), 12, startYear) - daysFromStartOfYear(checkInDay, startMonth, startYear) + 1) +
45             (daysFromStartOfYear(checkOutDay, endMonth, endYear));
46     }
47
48     return numberofDays;
49 }
```

## 2.22 Number of Pax

```

1 #include <iostream>
2 using namespace std;
3
4 // Definition Function Validate Integer Input
5 int getValidatedInput(const string& prompt);
6
7 int NumberOfPax() {
8
9     // Prompt user to enter number of adult, children and infants and get validated input
10    cout << "----->" << endl;
11    int adult = getValidatedInput("Please enter number of pax adult (13 years and older)");
12    int children = getValidatedInput("Please enter number of pax children (2-12 years old)");
13    int infants = getValidatedInput("Please enter number of pax infants (under 2 years old)");
14
15    // Calculate the total number of pax
16    int totalPax = adult + children + infants;
17
18    // Display the total number of pax
19    cout << "----->" << endl;
20    cout << "The total number of pax is " << totalPax << endl;
21    cout << "----->" << endl;
22
23    return totalPax;
24 }

```

## 2.23 overviewbooking

```

1 #include <iostream>
2 #include <iomanip>
3 #include "BookingSystem.h"
4
5 using namespace std;
6
7 double displayOverview(int locationNo, int totalPax, string name, string phoneNumber, int startMonth, int endMonth, int startYear, int endYear, int
8 checkInDay, int checkOutDay, int numOfDay, BookingList& BL) {
9
10    // Define and initialize hotel rooms for Kuala Lumpur
11    BookingRoom roomsKualaLumpur[] = {
12        {1, "Cubis Botanical", 650.00, "Wifi & Breakfast"}, 
13        {2, "Mont Kiara", 200.00, "Wifi & Breakfast"}, 
14        {3, "Urban Retreat", 450.00, "Wifi, Breakfast & Gym"}, 
15        {4, "Luxe Residence", 450.00, "Wifi, Breakfast & Lounge"}, 
16        {5, "Panorama Hotel", 450.00, "Wifi, Breakfast & Pool"}};
17    const int numRoomsKualaLumpur = sizeof(roomsKualaLumpur) / sizeof(roomsKualaLumpur[0]);
18
19    // Define and initialize hotel rooms for Melaka
20    BookingRoom roomsMelaka[] = {
21        {1, "Heritage Haven", 300.00, "Wifi & Breakfast"}, 
22        {2, "Melaka River Inn", 250.00, "Wifi, Breakfast & Shuttle Service"}, 
23        {3, "Coastal Retreat", 350.00, "Wifi, Breakfast, Gym & Spa"}, 
24        {4, "Grand Melaka Resort", 500.00, "Wifi, Breakfast, Lounge & Business Center"}, 
25        {5, "Historical Lodge", 200.00, "Wifi, Breakfast & Laundry Service"}};
26    const int numRoomsMelaka = sizeof(roomsMelaka) / sizeof(roomsMelaka[0]);
27
28    // Define and initialize hotel rooms for Penang
29    BookingRoom roomsPenang[] = {
30        {1, "Bayview Grand Hotel", 300.00, "Wifi & Breakfast"}, 
31        {2, "Georgetown Inn", 250.00, "Wifi, Breakfast & Shuttle Service"}, 
32        {3, "Seaside Resort", 400.00, "Wifi, Breakfast, Gym & Spa"}, 
33        {4, "Penang Heritage Lodge", 300.00, "Wifi, Breakfast & Lounge"}, 
34        {5, "Bayview Penang", 500.00, "Wifi, Breakfast & Business Center"}};
35    const int numRoomsPenang = sizeof(roomsPenang) / sizeof(roomsPenang[0]);
36
37    // Define and initialize hotel rooms for Selangor
38    BookingRoom roomsSelangor[] = {
39        {1, "Metro Plaza Hotel", 420.00, "Wifi, Gym, Spa & Lounge"}, 
40        {2, "City View Resort", 550.00, "Wifi, Dinner, Free Parking & Airport Shuttle"}, 
41        {3, "Elite Comfort Hotel", 480.00, "Wifi, Breakfast, Conference Room & Laundry Service"}, 
42        {4, "Urban Haven Hotel", 390.00, "Wifi, Breakfast, Afternoon Tea & Business Center"}, 
43        {5, "Selangor Grand", 350.00, "Wifi, Breakfast, 24 Hours Front Desk & Lounge"}};
44    const int numRoomsSelangor = sizeof(roomsSelangor) / sizeof(roomsSelangor[0]);
45
46    // Define and initialize hotel rooms for Sabah
47    BookingRoom roomsSabah[] = {
48        {1, "Borneo Bliss Hotel", 320.00, "Wifi, Breakfast, Pool & Parking"}, 
49        {2, "Kinabalu View", 270.00, "Wifi, Breakfast, Shuttle Service & Airport Transfer"}, 
50        {3, "Sunset Coastal Resort", 380.00, "Wifi, Breakfast, Restaurant & Spa"}, 
51        {4, "Sabah Serenity Lodge", 290.00, "Wifi, Breakfast, Concierge"}, 
52        {5, "Horizon Hotel", 450.00, "Wifi, Breakfast, Business Center & Conference Rooms"}};
53    const int numRoomsSabah = sizeof(roomsSabah) / sizeof(roomsSabah[0]);
54
55    // Define and initialize hotel rooms for Sarawak
56    BookingRoom roomsSarawak[] = {
57        {1, "Riverfront Resort", 350.00, "Wifi, Dinner, Spa & Free Parking"}, 
58        {2, "Heritage City Hotel", 580.00, "Wifi, Breakfast, Pool & Laundry"}, 
59        {3, "Borneo Vista Lodge", 400.00, "Wifi, Breakfast, Gym & Spa"}, 
60        {4, "Tropical Haven Hotel", 460.00, "Wifi, Breakfast, Afternoon Tea"}, 
61        {5, "Kuching Grand Inn", 310.00, "Wifi, Breakfast, Business Center & Lounge"}};
62    const int numRoomsSarawak = sizeof(roomsSarawak) / sizeof(roomsSarawak[0]);
63
64
65
66
67

```

# Problem Solving and Programming

```
68 // Choose the room array based on locationNo
69 const BookingRoom* rooms;
70 int totalRooms;
71 if (locationNo == 1) {
72     rooms = roomsKualaLumpur;
73     totalRooms = numRoomsKualaLumpur;
74 }
75 else if (locationNo == 2) {
76     rooms = roomsMelaka;
77     totalRooms = numRoomsMelaka;
78 }
79 else if (locationNo == 3) {
80     rooms = roomsPenang;
81     totalRooms = numRoomsPenang;
82 }
83 else if (locationNo == 4) {
84     rooms = roomsSelangor;
85     totalRooms = numRoomsSelangor;
86 }
87 else if (locationNo == 5) {
88     rooms = roomsSabah;
89     totalRooms = numRoomsSabah;
90 }
91 else if (locationNo == 6) {
92     rooms = roomsSarawak;
93     totalRooms = numRoomsSarawak;
94 }
95 else {
96     cout << "Invalid location number." << endl;
97     return 1;
98 }
99
100 // Overview of total bookings
101 cout << setfill(' ') << "\n" =====| " << endl;
102 cout << "=====
103 =====| " << endl;
104 cout << "=====
105 =====| " << endl;
106 cout << "|" << setw(20) << left << "Name" << ":" << setw(20) << left << name << setw(43) << " " << "Check-In Date" << setw(4) << right << ":" << setw(2)
107 (2) << left << showpoint << checkInDay << "/" << setw(2) << left << startMonth << "/" << startY << " |" << endl;
108 cout << "|" << setw(20) << left << "Phone Number" << ":" << setw(20) << left << showpoint << phoneNumber << setw(43) << " " << "Check-Out Date" << setw(4)
109 << right << ":" << setw(2) << showpoint << left << checkOutDay << "/" << setw(2) << left << endMonth << "/" << endYear << " |" << endl;
110 cout << "|" << setw(20) << left << "Total number of Pax" << ":" << setw(20) << left << totalPax << setw(43) << " " << "Number Of Days" << setw(4) << right
111 << ":" << setw(12) << left << numberOfDays << "|" << endl;
112 cout << "|" << setw(9) << left << "BookingNo" << setw(15) << right << "Room Number" << ":" << setw(15) << right << "Hotel" << setw(10) << right << " "
113 << setw(13) << right << "Quantity" << setw(5) << right << " " << setw(20) << right << "Price Per Room" << setw(5) << right << " " << setw(17) << right
114 << "Total Price" << setw(5) << right << "|" << endl;
115 cout << "|" << endl;
116 // Declaration totalPrice
117 double totalPrice = 0;
118
119 int latestBookingIndex = BL.count - 1;
120 int roomNo = BL.bookings[latestBookingIndex].roomNo;
121 int quantity = BL.bookings[latestBookingIndex].quantity;
122 int bookingNo = BL.bookings[latestBookingIndex].bookingNumber;
123
124 for (int j = 0; j < totalRooms; ++j) {
125     if (rooms[j].roomNo == roomNo) {
126         double roomPrice = rooms[j].price;
127         double bookingTotalPrice = roomPrice * quantity;
128         totalPrice += bookingTotalPrice;
129
130         cout << "Services: " << setw(104) << left << rooms[j].service << " |" << endl;
131         cout << "|" << setw(115) << right << " |" << endl;
132     }
133 }
134
135 double totalPriceAllDays = totalPrice * numberOfDays;
136 cout << "|" << endl;
137 cout << "|" << "Total before taxes" << setw(78) << "RM" << setw(18) << fixed << setprecision(2) << right << totalPriceAllDays << " |" << endl;
138 cout << "|" << endl;
139
140 return totalPriceAllDays;
141 }
```

## 2.24 Payment Method Table

```
1 #include <iostream>
2 #include <iomanip>
3 using namespace std;
4
5 void paymentMethodTable() {
6     cout << endl;
7     cout << setfill('=') << setw(36) << "=" << endl;
8     cout << setfill(' ') << "|" << setw(10) << " " << "Payment Method" << setw(10) << " " << " |" << endl;
9     cout << setfill(' ') << setw(36) << "=" << endl;
10    cout << setfill(' ') << '!' << setw(3) << " " << "1. Credit/Debit Cards" << setw(10) << " " << " |" << endl;
11    cout << setfill(' ') << '!' << setw(3) << " " << "2. Bank Transfer" << setw(15) << " " << " |" << endl;
12    cout << setfill(' ') << '!' << setw(3) << " " << "3. Touch N Go" << setw(18) << " " << " |" << endl;
13    cout << setfill(' ') << '!' << setw(3) << " " << "4. Apple Pay" << setw(19) << " " << " |" << endl;
14    cout << setfill(' ') << '!' << setw(3) << " " << "5. Cash Payment" << setw(16) << " " << " |" << endl;
15    cout << setfill(' ') << setw(36) << "=" << setfill(' ') << endl;
16 }
```

## 2.25 Penang Table

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5
6  void PenangArea() {
7      // Penang Area
8      cout << "\nPenang" << endl;
9
10     // Title of the table
11     cout << setfill('=') << left << setw(153) << "=" << endl;
12     cout << setfill(' ') << "|" << setw(3) << right << "No" << setw(2) << "|" << setw(14) << "Name" << setw(11) << "|" << setw(18) << "Location" << setw(7)
13     (11) << "|";
14     cout << setw(18) << "Price" << setw(6) << "|" << setw(13) << "Bed Type" << setw(6) << "|";
15     cout << setw(14) << "Room Type" << setw(6) << "|" << setw(14) << "Room Size" << setw(6) << "|" << setw(12) << "Service" << setw(6) << "|" << endl;
16
17     // Information of Bayview Grand Hotel
18     cout << setfill('=') << setw(154) << " " << endl;
19     cout << setfill(' ') << "|" << setw(3) << "I" << setw(2) << "|" << setw(24) << left << "Bayview Grand Hotel" << "|" << setw(28) << left << "45, Jalan
20     Hang Jebat, 75200" << "|";
21     cout << setw(9) << left << "RM" << setw(6) << left << "300.00" << "|" << setw(13) << right << " Queen " << setw(6) << "|";
22     cout << setw(14) << " Superior " << setw(6) << "|" << setw(14) << "25-35 sqm" << setw(6) << "|" << setw(12) << " WiFi " << setw(6) << "|"
23     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << "Melaka" << "|";
24     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
25     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(13) << "Breakfast" << setw(5) << "|"
26     cout << setfill(' ') << setw(154) << " " << endl;
27     cout << setfill(' ') << "|" << setw(3) << "2" << setw(2) << "|" << setw(24) << left << "Georgetown Inn" << "|" << setw(28) << left << "456, Jalan
28     Beach, 10200" << "|";
29     cout << setw(9) << left << "RM" << setw(6) << left << "250.00" << "|" << setw(13) << right << " Queen " << setw(6) << "|";
30     cout << setw(12) << "Standard" << setw(6) << "|" << setw(14) << "25-35 sqm" << setw(6) << "|" << setw(12) << " WiFi " << setw(6) << "|"
31     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << "Penang" << "|";
32     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
33     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(13) << "Breakfast" << setw(5) << "|"
34     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << " " << "|";
35     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
36     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(12) << " Shuttle Service" << setw(2) << "|"
37     cout << setfill(' ') << setw(154) << " " << endl;
38     // Information of Seaside Resort
39     cout << setfill(' ') << setw(154) << " " << endl;
40     cout << setfill(' ') << "|" << setw(3) << "3" << setw(2) << "|" << setw(24) << left << "Seaside Resort" << "|" << setw(28) << left << "789, Jalan
41     Seaside, 10300" << "|";
42     cout << setw(9) << left << "RM" << setw(6) << left << "400.00" << "|" << setw(13) << right << " Double " << setw(6) << "|";
43     cout << setw(14) << "Suite" << setw(6) << "|" << setw(14) << "35-45 sqm" << setw(6) << "|" << setw(12) << " WiFi " << setw(6) << "|"
44     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << "Penang" << "|";
45     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
46     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(13) << "Breakfast" << setw(5) << "|"
47     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << " " << "|";
48     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
49     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(12) << " Spa " << setw(6) << "|"
50     cout << setfill(' ') << setw(154) << " " << endl;
51
52     // Information of Penang Heritage Lodge
53     cout << setfill(' ') << setw(154) << " " << endl;
54     cout << setfill(' ') << "|" << setw(3) << "4" << setw(2) << "|" << setw(24) << left << "Penang Heritage Lodge" << "|" << setw(28) << left << "101,
55     Jalan Heritage, 10400" << "|";
56     cout << setw(9) << left << "RM" << setw(6) << left << "300.00" << "|" << setw(13) << right << " Queen " << setw(6) << "|";
57     cout << setw(14) << "Superior" << setw(6) << "|" << setw(14) << "20-30 sqm" << setw(6) << "|" << setw(12) << " WiFi " << setw(6) << "|"
58     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << "Penang" << "|";
59     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
60     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(13) << "Breakfast" << setw(5) << "|"
61     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << " " << "|";
62     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
63     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(12) << " Lounge " << setw(6) << "|"
64
65     // Information of Bayview Penang
66     cout << setfill(' ') << setw(154) << " " << endl;
67     cout << setfill(' ') << "|" << setw(3) << "5" << setw(2) << "|" << setw(24) << left << "Bayview Penang" << "|" << setw(28) << left << "202, Jalan
68     Bayview, 10500" << "|";
69     cout << setw(9) << left << "RM" << setw(6) << left << "500.00" << "|" << setw(13) << right << " King " << setw(6) << "|";
70     cout << setw(14) << "Executive" << setw(7) << " | " << setw(14) << "40-50 sqm" << setw(6) << " | " << setw(12) << " WiFi " << setw(6) << " | "
71     cout << setfill(' ') << "|" << setw(3) << " " << setw(2) << "|" << setw(24) << left << " " << "|" << setw(28) << left << "Melaka" << "|";
72     cout << setw(9) << left << " " << setw(6) << left << " " << "|" << setw(13) << right << " " << setw(6) << "|";
73     cout << setw(14) << " " << setw(6) << "|" << setw(14) << " " << setw(6) << "|" << setw(12) << " Business Center" << setw(2) << "|"
74
75     cout << setfill(' ') << setw(154) << " " << setfill(' ') << endl;
76
77 }

```

## 2.26 Phone Number

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4
5  using namespace std;
6
7  // Declare validatePhoneNumber
8  bool validatePhoneNumber(const string& phoneNumber);
9
10 string InputPhoneNumber() {
11     string phoneNumber;
12
13     do {
14         cout << "Phone Number\t: ";
15         getline(cin, phoneNumber); // Use getline to handle the input as a string
16
17         if (!validatePhoneNumber(phoneNumber)) {
18             cout << "Error: Please enter a valid phone number consisting only of digits." << endl;
19         }
20     } while (!validatePhoneNumber(phoneNumber)); // Repeat until a valid phone number is entered
21
22     return phoneNumber; // Return the valid phone number
23 }
```

## 2.27 Pin Validation

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  #include <limits>
5
6  using namespace std;
7
8  int pinValidation(const string& prompt) {
9      int value;
10     cout << setw(56) << left << prompt << ":";
11
12     while (true) {
13         // Check if input is valid and within the range 4 to 6 digits
14         if (cin >> value && value >= 1000 && value <= 999999) {
15             int length = to_string(value).length();
16             if (length >= 4 && length <= 6) {
17                 break; // Input is valid
18             }
19         }
20
21         // Clear the error state and ignore the invalid input
22         cin.clear();
23         cin.ignore(numeric_limits<streamsize>::max(), '\n');
24
25         // Prompt the user again
26         cout << "Error: Please enter a valid PIN number with 4 to 6 digits." << endl;
27         cout << setw(56) << left << prompt << ":";
28     }
29
30     return value;
31 }
```

## 2.28 Report

```

1  // #include <iostream>
2  // #include <iomanip>
3  // #include <string>
4  // #include <limits>
5  #include "BookingSystem.h"
6
7  using namespace std;
8
9  // Function prototypes
10 void displaySummary(int totalUnits, double totalAmount, const string& location);
11
12 // Report function to display the summarized information
13 void Report(int locationNo, double grandtotal, BookingList& BL) {
14     int totalunit = 0;
15
16     // variables to store totals
17     int kualalumpurTotalUnits = 78;
18     double kualalumpurTotalAmount = 41000;
19
20     int melakaTotalUnits = 65;
21     double melakaTotalAmount = 20400;
22
23     int penangTotalUnits = 99;
24     double penangTotalAmount = 50880;
25
26     int selangorTotalUnits = 56;
27     double selangorTotalAmount = 20580;
28
29     int sabahTotalUnits = 101;
30     double sabahTotalAmount = 33840;
31
32     int sarawakTotalUnits = 77;
33     double sarawakTotalAmount = 35600;
34
35     // Calculate total units from bookings
36     for (int i = 0; i < BL.count; ++i) {
37         totalunit += BL.bookings[i].quantity;
38     }
39
40     // Update totals based on locationNo
41     switch (locationNo) {
42         case 1: kualalumpurTotalUnits += totalunit; kualalumpurTotalAmount += grandtotal; break;
43         case 2: melakaTotalUnits += totalunit; melakaTotalAmount += grandtotal; break;
44         case 3: penangTotalUnits += totalunit; penangTotalAmount += grandtotal; break;
45         case 4: selangorTotalUnits += totalunit; selangorTotalAmount += grandtotal; break;
46         case 5: sabahTotalUnits += totalunit; sabahTotalAmount += grandtotal; break;
47         case 6: sarawakTotalUnits += totalunit; sarawakTotalAmount += grandtotal; break;
48         default: cout << "Invalid location number." << endl;
49     }
50
51     int staffChoice;
52     do {
53         // Display menu
54         cout << "\nHotel Report System\n";
55         cout << "1. Kuala Lumpur\n";
56         cout << "2. Melaka\n";
57         cout << "3. Penang\n";
58         cout << "4. Selangor\n";
59         cout << "5. Sabah\n";
60         cout << "6. Sarawak\n";
61         cout << "7. Exit\n";
62         cout << "Enter your choice (1-7): ";
63         cin >> staffChoice;
64
65         // Validate input
66         if (cin.fail() || staffChoice < 1 || staffChoice > 7) {
67             cin.clear(); // Clear input buffer
68             cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
69             cout << "Invalid choice. Please enter a number between 1 and 7." << endl;
70             continue; // Re-prompt the user
71         }
72
73         // Call appropriate function based on user choice
74         switch (staffChoice) {
75             case 1: displaySummary(kualalumpurTotalUnits, kualalumpurTotalAmount, "Kuala Lumpur"); break;
76             case 2: displaySummary(melakaTotalUnits, melakaTotalAmount, "Melaka"); break;
77             case 3: displaySummary(penangTotalUnits, penangTotalAmount, "Penang"); break;
78             case 4: displaySummary(selangorTotalUnits, selangorTotalAmount, "Selangor"); break;
79             case 5: displaySummary(sabahTotalUnits, sabahTotalAmount, "Sabah"); break;
80             case 6: displaySummary(sarawakTotalUnits, sarawakTotalAmount, "Sarawak"); break;
81             case 7: cout << "Exiting program." << endl; break;
82             default: cout << "Invalid choice." << endl; break;
83         }
84
85     } while (staffChoice != 7); // Continue until the user selects 7 (Exit)
86     BL.count = 0; // This ensure that the next call to Report start with a clean slate
87 }
```

## 2.29 return struct BookingList

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  #include <limits> // For numeric_limits
5  #include "BookingSystem.h"
6
7  using namespace std;
8
9  int nextBookingNumber = 1001;
10 int getValidatedInput(const string& prompt);
11
12 BookingList initializePrePopulatedBookings() {
13     BookingList bookingList;
14     bookingList.count = 0; // Initialize booking count to 0
15
16     // Add pre-populated bookings if needed (for testing or initial setup)
17     if (bookingList.count < MAX_BOOKINGS) {
18         bookingList.bookings[bookingList.count].bookingNumber = nextBookingNumber++;
19         bookingList.bookings[bookingList.count].roomNo = 1;
20         bookingList.bookings[bookingList.count].quantity = 0;
21         bookingList.count++;
22     }
23
24     return bookingList;
25 }
26
27 BookingList returnStruct(BookingList& bookingList) {
28
29     // Initialize the roomNo and quantity
30     int roomNo = 0, quantity = 0;
31
32     cout << setw(56) << left << "Room Number (1-5)" << ":";
33     cin >> roomNo;
34
35     // Validate input for "roomNo"
36     while (cin.fail() || roomNo < 1 || roomNo > 5) {
37         cin.clear(); // Clear error flag
38         cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Discard invalid input
39         cout << setw(56) << left << "Invalid number. Please select a valid room number (1-5)" << ":";
40         cin >> roomNo;
41     }
42
43     quantity = getValidatedInput("Quantity");
44
45     while (quantity <= 0 || quantity > 10) {
46         cout << "Sorry! The maximum number of rooms that can be booked is 10." << endl;
47         quantity = getValidatedInput("Quantity");
48     }
49
50     // Store the room number, quantity, and auto-generated booking number
51     if (bookingList.count < MAX_BOOKINGS) {
52         bookingList.bookings[bookingList.count].bookingNumber = nextBookingNumber++;
53         bookingList.bookings[bookingList.count].roomNo = roomNo;
54         bookingList.bookings[bookingList.count].quantity = quantity;
55         bookingList.count++;
56     }
57     else {
58         cout << "Booking limit reached. Cannot add more rooms." << endl;
59     }
60
61     // Return struct
62     return bookingList;
63 }

```

## 2.30 Sabah Table

## 2.31 Sarawak Table

## 2.32 Selangor Table

## 2.33 StaffLogin

```

1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  bool stafflogin() {
7      const int NUM_STAFFS = 3;
8      const string correctStaffIDs[NUM_STAFFS] = { "staff1", "staff2", "staff3" };
9      const string correctPasswords[NUM_STAFFS] = { "password1", "password2", "password3" };
10
11     string inputStaffID;
12     string inputPassword;
13     char staffchoice;
14
15     while (true) {
16         bool isAuthenticated = false; // Flag to check if user is authenticated
17
18         // Loop until the correct Staff ID and Password are entered
19         while (!isAuthenticated) {
20             cout << "\nLogin" << endl;
21             cout << "-----" << endl;
22
23             // Prompt and get Staff ID
24             cout << "Staff ID: ";
25             cin >> inputStaffID;
26
27             // Prompt and get Password
28             cout << "Password: ";
29             cin >> inputPassword;
30
31             // Check if the entered Staff ID and Password are correct
32             bool idExists = false;
33             for (int i = 0; i < NUM_STAFFS; ++i) {
34                 if (inputStaffID == correctStaffIDs[i]) {
35                     idExists = true;
36                     if (inputPassword == correctPasswords[i]) {
37                         cout << "Login successful!" << endl;
38                         isAuthenticated = true; // Exit the loop
39                         return true;
40                         break;
41                     }
42                 } else {
43                     cout << "Incorrect Password. Please try again." << endl;
44                     break; // Exit to ask if user wants to continue
45                 }
46             }
47
48             if (!idExists) {
49                 cout << "Incorrect Staff ID. Please try again." << endl;
50             }
51
52             // If authentication was not successful, ask if user wants to try again
53             if (!isAuthenticated) {
54                 cout << "Do you want to try again? (Y/N): ";
55                 cin >> staffchoice;
56                 while (staffchoice != 'Y' && staffchoice != 'y' && staffchoice != 'N' && staffchoice != 'n') {
57                     cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No: ";
58                     cin >> staffchoice;
59                 }
60
61                 // If user does not want to continue, exit the loop and return false
62                 if (staffchoice != 'Y' && staffchoice != 'y') {
63                     cout << "Exiting staff interface..." << endl;
64                     return false; // Indicate failed authentication
65                 }
66             }
67         }
68
69         // If authenticated, ask if the user wants to continue in the staff interface
70         cout << "Do you want to continue in the staff interface? (Y/N): ";
71         cin >> staffchoice;
72         while (staffchoice != 'Y' && staffchoice != 'y' && staffchoice != 'N' && staffchoice != 'n') {
73             cout << "Invalid input. Please enter 'Y' for Yes or 'N' for No: ";
74             cin >> staffchoice;
75         }
76
77         // Exit if the user does not want to continue
78         if (staffchoice != 'Y' && staffchoice != 'y') {
79             cout << "Exiting staff interface..." << endl;
80             return false; // Indicate successful login
81         }
82     }
83 }
84

```

## 2.34 Validate Integer Input

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5
6  int getValidatedInput(const string& prompt) {
7      int value;
8      cout << setw(56) << left << prompt << ":";
9
10     while (!(cin >> value) || value < 0) {
11         cin.clear(); // Clear the error state
12         cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Ignore incorrect input
13
14         // Prompt the user again
15         cout << "Error: Please enter a valid number." << endl;
16         cout << setw(56) << left << prompt << ":";
17     }
18
19     return value;
20 }

```

## 2.35 Validation IC

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4
5  using namespace std;
6
7  // Validate IC number (example: 12-digit number as a string)
8  bool validateICNumber(const string& ic_num) {
9      if (ic_num.length() != 12) return false; // Check if IC number has exactly 12 digits
10
11     for (char c : ic_num) {
12         if (!isdigit(c)) return false; // Ensure all characters are digits
13     }
14
15     return true;
16 }
```

## 2.36 Validation Phone Number

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  bool validatePhoneNumber(const string& phoneNumber) {
6      // Check if the phone number is non-empty and consists only of digits
7      if (phoneNumber.empty()) {
8          return false;
9      }
10
11     // Check if all characters are digits
12     if (phoneNumber.find_first_not_of("0123456789") != std::string::npos) {
13         return false;
14     }
15
16     // Check if the length is at least 10 digits
17     if (phoneNumber.length() < 10) {
18         return false;
19     }
20
21     return true;
22 }
```

## 2.37 Validation String Input

```

1  #include <iostream>
2  #include <string>
3  #include <cctype> // for character checking & conversion
4  using namespace std;
5
6  // Function to validate string input
7  bool validateName(const string& name) {
8      if (name.empty() || name.size() > 100) {
9          return false; // Check for empty or too long names
10     }
11
12     for (char ch : name) {
13         // Check if the character is an alphabetic letter, space, hyphen, or apostrophe
14         if (!isalpha(ch) && ch != ' ' && ch != '-' && ch != '\'') {
15             return false; // Invalid character found
16         }
17     }
18
19     return true;
20 }
```

## 2.38 Voucher

```

1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  #include "BookingSystem.h"
5
6  using namespace std;
7
8 RebateValue Voucher(double rebatePrice) {
9     // Define the voucher codes and discounts
10    string vouchers[6] = { "CNY8888", "HR4444", "DP6666", "MC1225", "V2141314", "WELCOME123" };
11    double discounts[6] = { 0.08, 0.10, 0.12, 0.15, 0.18, 0.20 };
12
13    string vouchername;
14    double finalprice;
15    double rebate;
16
17    bool voucherFound = false;
18
19    while (!voucherFound) {
20        cout << "Enter your voucher code: ";
21        cin >> vouchername;
22
23        finalprice = rebatePrice;
24
25        for (int i = 0; i < 6; ++i) {
26            if (vouchername == vouchers[i]) {
27                rebate = rebatePrice * discounts[i];
28                voucherFound = true;
29                break;
30            }
31        }
32
33        if (!voucherFound) {
34            cout << "Invalid voucher code. Please try again." << endl;
35        }
36    }
37
38    cout << fixed << setprecision(2);
39
40    RebateValue x = { rebate, vouchername };
41
42    return x;
43}

```