NZSE

School of IT & Business Technologies Bachelor of Business Information Management Cover Sheet and Student Declaration

This sheet must be signed by the student and attached to the submitted assessment.

Course Title:	**BBIM502 Introduction to Programming**	**Course code:**	BBIM502
Student Name:		**Student ID:**	764706455
Assessment No & Type:	Assessment 2 Project	**Cohort:**	BBIM7123C
Due Date:	31/10/2023	**Date Submitted:**	31/10/2023
Tutor's Name:	Jatinder Singh		
Assessment Weighting	40%		
Total Marks	100		

Student Declaration:

I declare that:

- I have read the New Zealand School of Education Ltd policies and regulations on assessments and understand what plagiarism is.
- I am aware of the penalties for cheating and plagiarism as laid down by the New Zealand School of Education Ltd.
- This is an original assessment and is entirely my own work.
- Where I have quoted or made use of the ideas of other writers, I have acknowledged the source.
- This assessment has been prepared exclusively for this course and has not been or will not be submitted as assessed work in any other course.

- It has been explained to me that this assessment may be used by NZSE Ltd, for internal and/or external moderation.
- If I am late in handing in this assessment without prior approval (see student regulations in handbook), marks will be deducted, to a maximum of 50%.

Student signature:

Date:

Tutor only to complete		
Assessment result:	**Mark /100**	**Grade**

Contents

Introduction

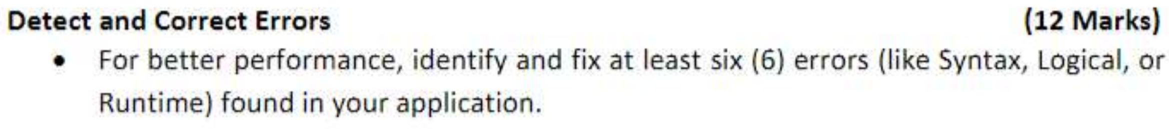
This report covers the development of a console-based application for the Langham Hotels chain, a high-end lodging option in major New Zealand cities. The Langham Hotels are a subsidiary of the Elite Hotels Company Limited, with its headquarters located on Queen Street Auckland. The development of a console-based program that performs input/output (I/O) tasks, including exceptional handling, is the main objective of this research project. It acts as a pilot project; if it proves successful, it will eventually be merged with the database system. To display the program's capabilities, such as room allocation, deallocation, room status display, and other functionality, we will be utilizing our Python programming skills. The report will be supported by concise code documentation, a thorough user guide, error types, and self-reflection.

Pseudocode

```
READ options.

Step 1: WHILE choice is not equal to 0:

DISPLAY the menu options.

    Step 2 DISPLAY message “Enter Your Choice Number Here (0-9):”

    Step 3 READ choice.

    Step 4 IF choice is 0.

            Exit the program.

    Step 5 ELIF choice is 1.

            Add Room module will run.

            Step 5.1 DISPLAY Add Room choice.

            Step 5.2 DISPLAY “Please enter the total number of rooms in the Hotel:”

            Step 5.3 IF length of list of rooms is greater than 0.

                    Variable y is equal to length of list of rooms.

            Step 5.4 ELSE

                    Variable y is equal to 0.

           Step 5.5 FOR in range of number of rooms.

                    room is equal to Room.

                    list of rooms.APPEND room.

           Step 5.6 FOR in range of y, y plus number of rooms

                    object room is equal to list of rooms[i]

                Step 5.7 DISPLAY “Room Allocation {i + 1}:”
            Step 5.8 DISPLAY “Please enter room number {i + 1}:”
            Step 5.9 READ object room.Room Number
                object room.is allocated is equal to FALSE.
                list of rooms[i] is equal to object room.
            Step 5.10 IF i > 0:
              Step 5.11 FOR j in range of i:
                Step 5.12 WHILE list of rooms[i].Room Number is equal to the previous input of list of rooms[j].Room Number:
                Step 5.13 DISPLAY “Same room number already exists”.
                Step 5.14 DISPLAY “Please enter a new room number {i + 1}:”
                Step 5.15 READ object room.Room Number
                room.is allocated is equal to FALSE.
                list of rooms[i] is equal to object room.
Step 6 ELIF choice is 2.
            Delete Room module will run.
            Step 6.1 IF in list of rooms
                Step 6.1 DISPLAY Delete Room choice.
                Step 6.2        DISPLAY “Please enter the room number that you want to delete”.
                Step 6.3 READ room number.
                    Step 6.4 FOR object room in list of rooms.
                        Step 6.5 IF object room is equal to room number.
                            List of rooms.remove object room.
                Step 6.6 DISPLAY “Following rooms have been deleted”.
            Step 6.7 ELSE
                Step 6.8 DISPLAY “No rooms to delete. Please add rooms first”
Step 7 ELIF choice is 3.
            Display Room Details module will run.
        Step 7.1 IF in list of rooms.
                Step 7.2 DISPLAY Display Room choice.
                Step 7.3 FOR object room in list of rooms.
                Step 7.4 DISPLAY “Room Number:{object room.Room Number}”
            Step 7.5 ELSE
                Step 7.6 DISPLAY “No rooms to display. Please add rooms first”.
Step 8 ELIF choice is 4.
            Allocate Rooms module will run.
    Step 8.1 DISPLAY “Allocate Rooms choice.”
    Step 8.2 DISPLAY “How many rooms would you like to allocate?”
    Step 8.3 READ allocate room.
    Step 8.4 WHILE allocate room > length of list of rooms:
        Step 8.5 DISPLAY “You cannot allocate more rooms than total number of rooms in the Hotel. Please enter number between 1- {length of the list of rooms)}:”
        Step 8.6 DISPLAY “How many rooms would you like to allocate?”
        Step 8.7 READ allocate room.
    Step 8.8 DISPLAY “You are allocating {allocate room} room(s)”
            i = 0
    Step 8.9 WHILE I < allocate room:
                room allocation is equal to Room Allocation.
                customer is equal to Customer.
    Step 8.10 DISPLAY “Room Allocation {i + 1}:”
    Step 8.11 DISPLAY “Please search Room Number to allocate:”
    Step 8.12 READ search room.
    Step 8.13 FOR j in range of length of list of rooms:
        Step 8.14 IF search room is matching with list of rooms[j].Room Number
            Step 8.15 DISPLAY “Found matching room number to allocate.”
            Step 8.16 IF list of rooms[j].is allocated is NOT allocated:
            Step 8.17 DISPLAY “Room {list of rooms[j].Room Number} is empty”
            Step 8.18 DISPLAY “Please enter Customer Number to allocate:”
            Step 8.19 READ customer.Customer Number
            Step 8.20 DISPLAY “Please enter Customer Name to allocate:”
```

```
                    Step 8.21 READ customer.Customer Name
                            list of rooms[j].is allocated is equal to TRUE.
                    Step 8.22 DISPLAY “Allocation has been done”.
                            room allocation.Allocated Room Number is equal to search room.
                            room allocation.Allocated Customer is equal to customer.
                            list of room allocations.APPEND room allocation
                            i += 1
                    Step 8.23 BREAK from the loop
            Step 8.24 ELSE:
                    Step 8.25 DISPLAY “Room {list of rooms[j].RoomNo} is already occupied. Please enter another room to allocate".
                            i -= 1
                    Step 8.26 BREAK from the (WHILE) loop
  Step 8.27 ELSE:
            Step 8.28 WHILE j is matching the length of list of rooms – 1:
                    Step 8.29 DISPLAY “Could not find matching room number to allocate. Please enter correct room number or add room first”.
                            i -= 1
                    Step 8.30 BREAK
i += 1
 Step 9 ELIF choice is 5.
                    Display Room Allocation Details module will run.
        Step 9.1 IF in list of room allocations.
                Step 9.2 DISPLAY Display Room Allocation Details choice.
                Step 9.3 FOR object room in list of room allocations.
                                Step 9.4 DISPLAY “Allocated Room Number: {object room.Allocated Room Number}”
                                Step 9.5 DISPLAY “Customer Number: {object room.Allocated Customer.Customer Number}”
                                Step 9.6 DISPLAY “Customer Name: {object room.Allocated Customer.Customer Name}”
        Step 9.7 ELSE:
                Step 9.8 DISPLAY “No allocated rooms to display. Please allocate rooms first.”
Step 10 ELIF choice is 6.
         Billing & Deallocation module will run.
    Step 10.1 DISPLAY De-allocate rooms choice.
    Step 10.2 DISPLAY “Please enter Room Number for billing and de-allocation:”
    Step 10.3 READ room number.
    Step 10.4 FOR object room number IN list of room allocations:
       Step 10.5 IF object room number. Allocated Room Number matches room number.
          Step 10.6 DISPLAY “Please enter the total number of days stayed:”
          Step 10.7 READ days stayed.
              rate is equal to 150.
              room charge is equal to days stayed * rate.
          Step 10.8 DISPLAY “Room has been Billed for a total of $, room charge”.
              list of room allocations.remove object room number.
          Step 10.9 DISPLAY “Room has been successfully de-allocated {room number}”
          Step 10.10 BREAK
     Step 10.11 ELSE:
          Step 10.12 DISPLAY “Invalid input. There is no Allocated Room”.
Step 11 ELIF choice is 7.
         Save the Room Allocation to a File module will run.
        Step 11.1 DISPLAY Save the Room Allocations to a File
        Step 11.2 WITH opening file path
                Assigning variable {now} with date and time
            Step 11.3 FOR object room allocated in list of room allocations.
                Step 11.4 DISPLAY “Allocated Room Number: {object room Allocated.Allocated Room Number}”
                Step 11.5 DISPLAY “Customer Number: {object room Allocated.Allocated Customer.Customer Number}”
                Step 11.6 DISPLAY “Customer Name: {object room Allocated.Allocated Customer.Customer Name}
                Step 11.7 DISPLAY “Current Date and Time is {now}”
                Step 11.8 WRITE the file.
            Step 11.9 CLOSE the file.
            Step 11.10 DISPLAY “File saved as ‘LHMS_764706455.txt’ under Document folder”.
Step 12 ELIF choice is 8.
        Show the Room Allocation from a File module will run.
        Step 12.1 DISPLAY Show the Room Allocations from a File
        Step 12.2 WITH opening file path
            Step 12.3 READ lines in file.
            Step 12.4 FOR line in lines
                Step 12.5 DISPLAY without whitespace
Step 13 ELIF choice is 9.
        Backup module will run.
        Step 13.1 DISPLAY Backup choice.
        Step 13.2 WHILE file path back up exists.
            Step 13.3 DISPLAY “'LHMS_backup.txt' file already exist. Existing file will be deleted”.
                remove file path back up.
            File path is renamed as file path back up.
Step 14 ELSE:
    Step 15 DISPLAY “Please choose (0-9)”
Step 16 END
```

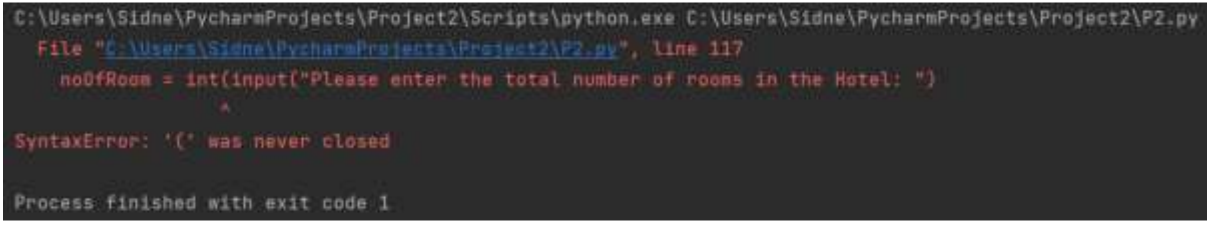
Task 2 (Errors & Exception Handling)



Detect and Correct Errors **(12 Marks)**

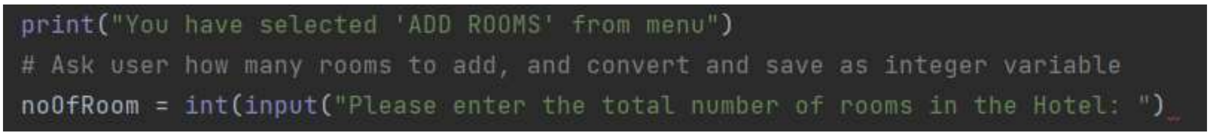
- For better performance, identify and fix at least six (6) errors (like Syntax, Logical, or Runtime) found in your application.

Syntax Error



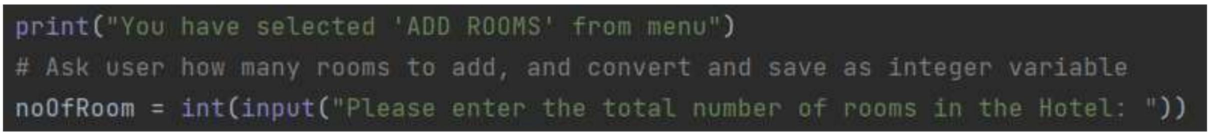
```
C:\Users\Sidne\PycharmProjects\Project2\Scripts\python.exe C:\Users\Sidne\PycharmProjects\Project2\P2.py
  File "C:\Users\Sidne\PycharmProjects\Project2\P2.py", line 117
    noOfRoom = int(input("Please enter the total number of rooms in the Hotel: ")
                  ^
SyntaxError: '(' was never closed

Process finished with exit code 1
```

Error missing a parenthesis.



```
print("You have selected 'ADD ROOMS' from menu")
# Ask user how many rooms to add, and convert and save as integer variable
noOfRoom = int(input("Please enter the total number of rooms in the Hotel: ")
```

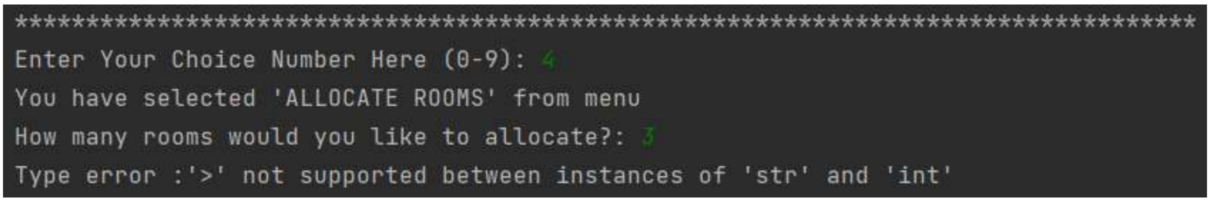
Fixing error.



```
print("You have selected 'ADD ROOMS' from menu")
# Ask user how many rooms to add, and convert and save as integer variable
noOfRoom = int(input("Please enter the total number of rooms in the Hotel: "))
```

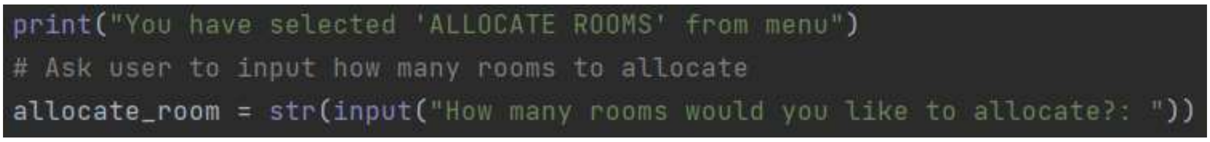
A Syntax Error occurs when there is an error in the syntax (structure) of the python code. To resolve this error, simply review the code to find underlined errors and input the correct syntax.

Type Error



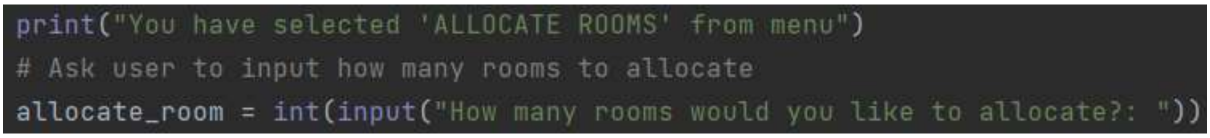
```
***********************************************************************************
Enter Your Choice Number Here (0-9): 4
You have selected 'ALLOCATE ROOMS' from menu
How many rooms would you like to allocate?: 3
Type error :'>' not supported between instances of 'str' and 'int'
```

Type Error of string instead of integer.



```
print("You have selected 'ALLOCATE ROOMS' from menu")
# Ask user to input how many rooms to allocate
allocate_room = str(input("How many rooms would you like to allocate?: "))
```

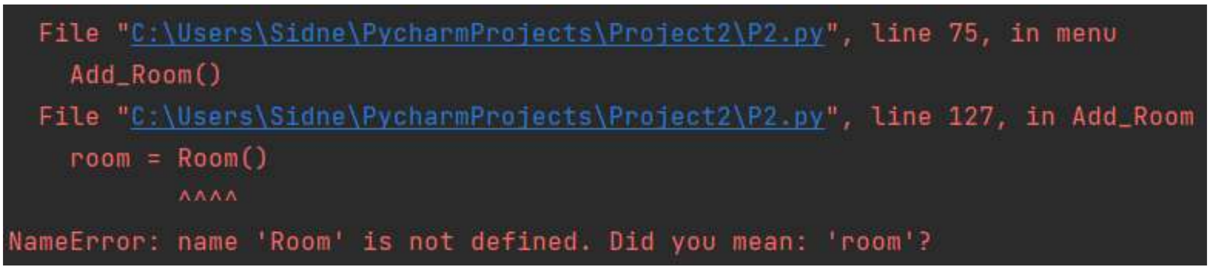
Fixing error.



```
print("You have selected 'ALLOCATE ROOMS' from menu")
# Ask user to input how many rooms to allocate
allocate_room = int(input("How many rooms would you like to allocate?: "))
```

A Type Error is when an operator is supplied with a value of incorrect datatype, such as a string instead of a integer.

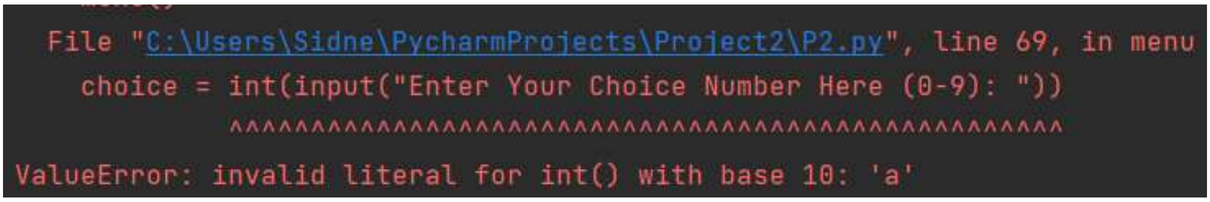
Name Error



```
  File "C:\Users\Sidne\PycharmProjects\Project2\P2.py", line 75, in menu
    Add_Room()
  File "C:\Users\Sidne\PycharmProjects\Project2\P2.py", line 127, in Add_Room
    room = Room()
           ^^^^
NameError: name 'Room' is not defined. Did you mean: 'room'?
```

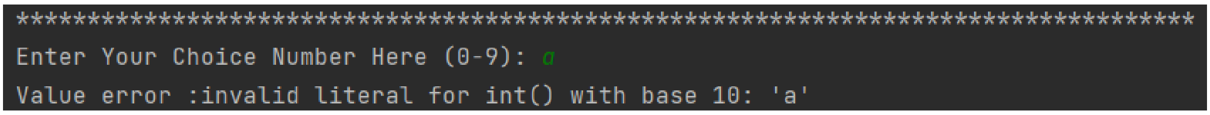
A Name Error is when a local or global variable name is not defined.

Value Error



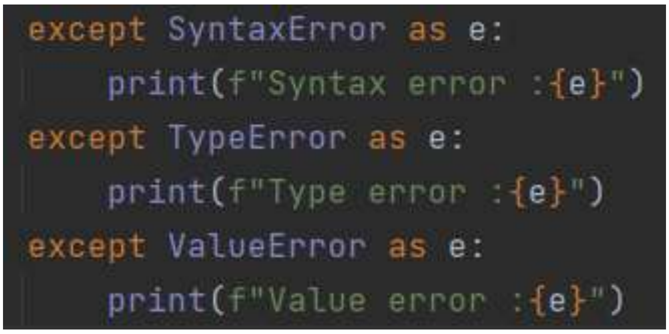
```
  File "C:\Users\Sidne\PycharmProjects\Project2\P2.py", line 69, in menu
    choice = int(input("Enter Your Choice Number Here (0-9): "))
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
ValueError: invalid literal for int() with base 10: 'a'
```

Encountering error for wrong input of value, string instead of integer.



```
*********************************************************************************
Enter Your Choice Number Here (0-9): a
Value error :invalid literal for int() with base 10: 'a'
```

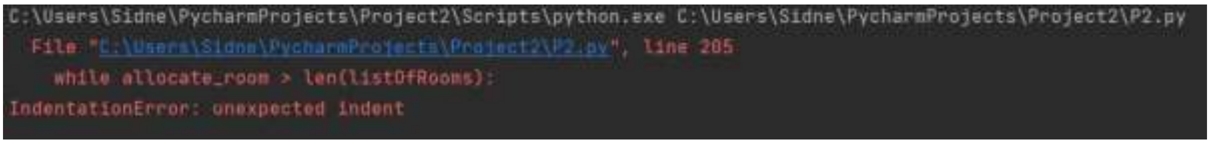
A Value Error is when a built-in method or operation receives an argument that has the right data type but mismatched or inappropriate values.



```
except SyntaxError as e:
    print(f"Syntax error :{e}")
except TypeError as e:
    print(f"Type error :{e}")
except ValueError as e:
    print(f"Value error :{e}")
```

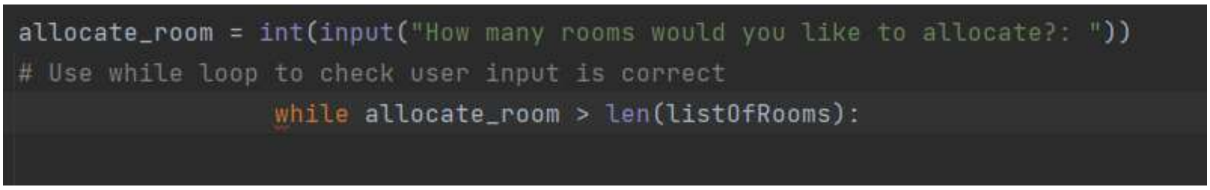
To resolve the Value, Type and Syntax Errors, Try and Except blocks have been utilised to address these errors.

Indentation Error



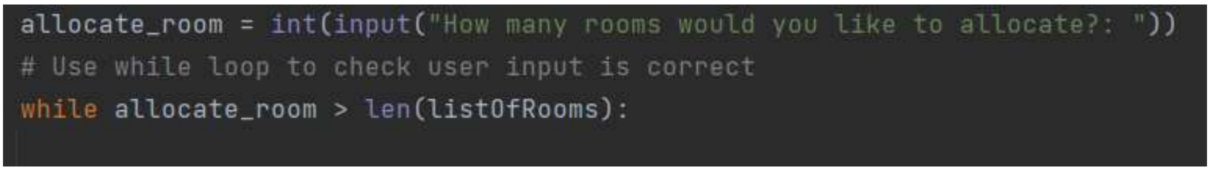
```
C:\Users\Sidne\PycharmProjects\Project2\Scripts\python.exe C:\Users\Sidne\PycharmProjects\Project2\P2.py
  File "C:\Users\Sidne\PycharmProjects\Project2\P2.py", line 205
    while allocate_room > len(listOfRooms):
IndentationError: unexpected indent
```

Multiple spaces were placed ahead.



```
allocate_room = int(input("How many rooms would you like to allocate?: "))
# Use while loop to check user input is correct
                while allocate_room > len(listOfRooms):
```

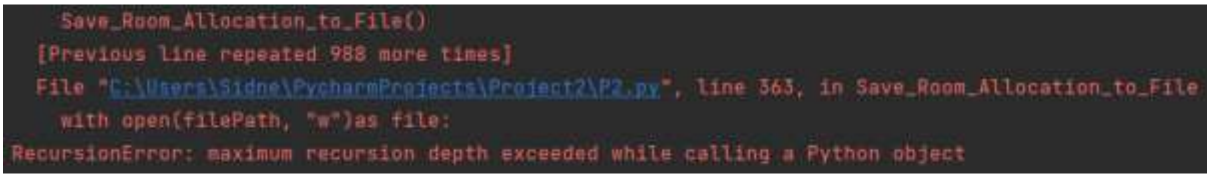
Fixing the code.



```
allocate_room = int(input("How many rooms would you like to allocate?: "))
# Use while loop to check user input is correct
while allocate_room > len(listOfRooms):
```

An indentation error occurs when the spaces or tabs are not placed properly.

Recursion Error



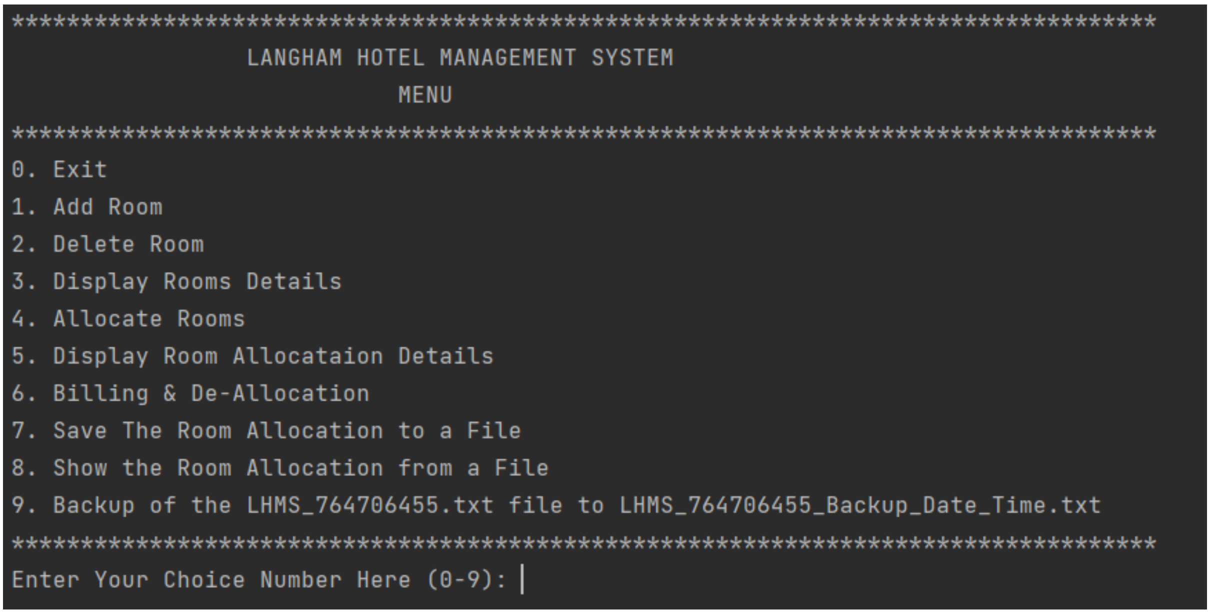
```
    Save_Room_Allocation_to_File()
  [Previous line repeated 988 more times]
  File "C:\Users\Sidne\PycharmProjects\Project2\P2.py", line 363, in Save_Room_Allocation_to_File
    with open(filePath, "w")as file:
RecursionError: maximum recursion depth exceeded while calling a Python object
```

A Recursion Error is caused by a function calling itself repeatedly. Python has a limit on the number of times a function can call itself recursively. This is to make sure that the function does not execute continuously. If this limit is gone beyond by a recursive function, a Recursion Error is raised. This can be resolved by using try and catch blocks to catch the recursion error.

TASK 3 (Documentation)

4. User guide - Output Screenshots – Check the Application by running it with sample data using all the menu options **(2 Marks)**

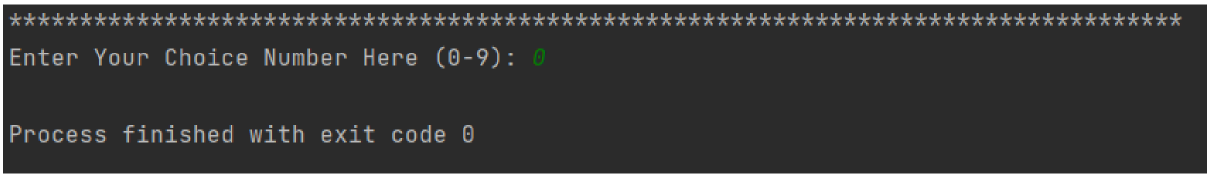
User guide with screenshots and documentation:



```
**********************************************************************************
                 LANGHAM HOTEL MANAGEMENT SYSTEM
                            MENU
**********************************************************************************
0. Exit
1. Add Room
2. Delete Room
3. Display Rooms Details
4. Allocate Rooms
5. Display Room Allocataion Details
6. Billing & De-Allocation
7. Save The Room Allocation to a File
8. Show the Room Allocation from a File
9. Backup of the LHMS_764706455.txt file to LHMS_764706455_Backup_Date_Time.txt
**********************************************************************************
Enter Your Choice Number Here (0-9): 
```

This is the Langham Hotel Management System. Enter a number through numbers (0-9) to select an option of your choice.

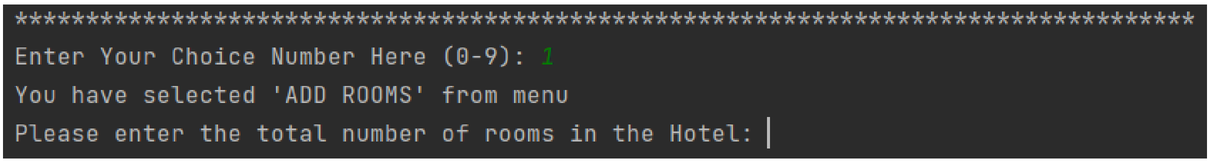
If 0 is selected:



```
*******************************************************************************
Enter Your Choice Number Here (0-9): 0

Process finished with exit code 0
```

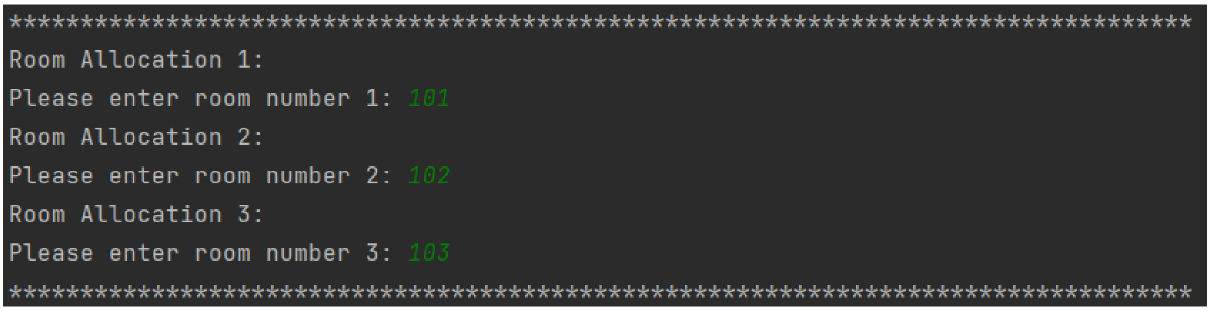
This will Exit and finish the program.

If 1 is selected:



```
*******************************************************************************
Enter Your Choice Number Here (0-9): 1
You have selected 'ADD ROOMS' from menu
Please enter the total number of rooms in the Hotel: |
```

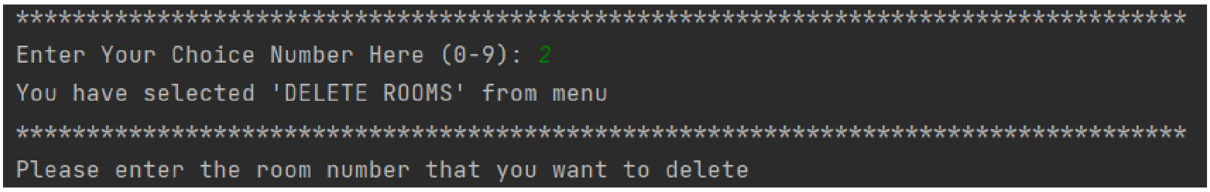
This is to add a room, where the user can enter the total number of rooms they wish to add.



```
*******************************************************************************
Room Allocation 1:
Please enter room number 1: 101
Room Allocation 2:
Please enter room number 2: 102
Room Allocation 3:
Please enter room number 3: 103
*******************************************************************************
```

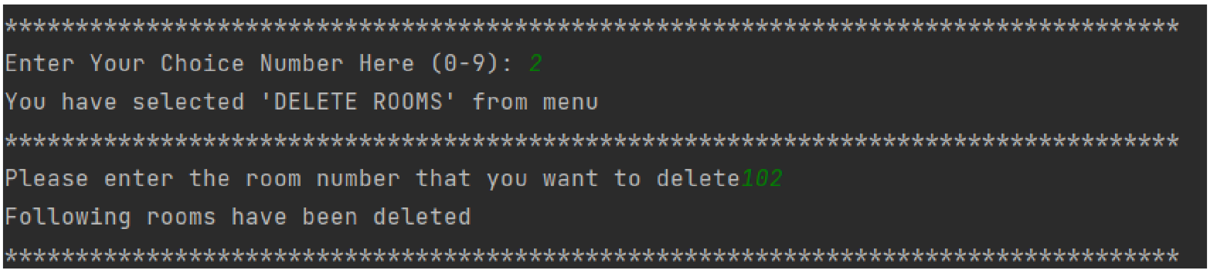
This is where the user will be asked to assign a room number to each given room.

If 2 is selected:



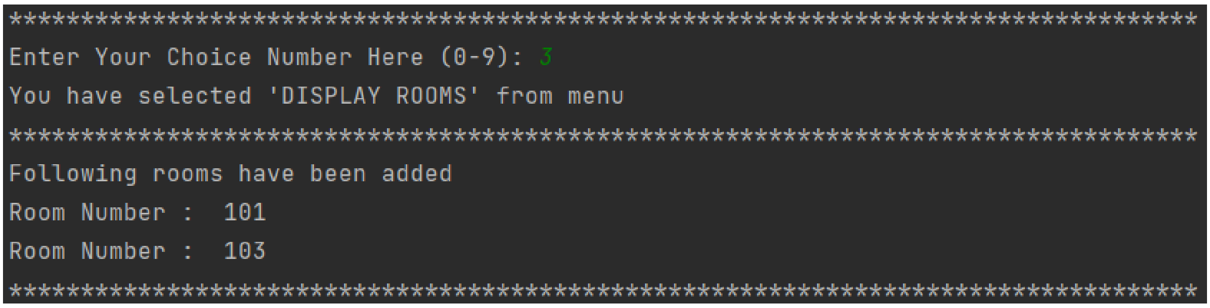
```
*******************************************************************************
Enter Your Choice Number Here (0-9): 2
You have selected 'DELETE ROOMS' from menu
*******************************************************************************
Please enter the room number that you want to delete
```

This is for deleting a room, where the user will enter the room number they wish to delete.



```
*******************************************************************************
Enter Your Choice Number Here (0-9): 2
You have selected 'DELETE ROOMS' from menu
*******************************************************************************
Please enter the room number that you want to delete102
Following rooms have been deleted
*******************************************************************************
```

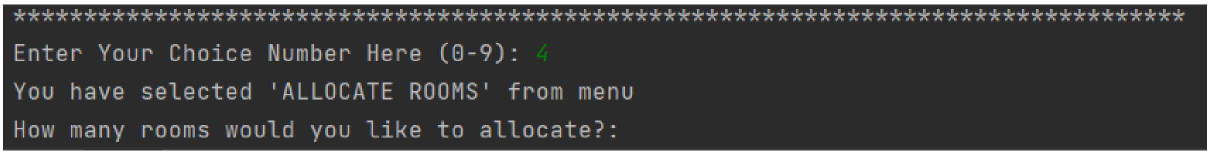
If 3 is selected:



```
*********************************************************************************
Enter Your Choice Number Here (0-9): 3
You have selected 'DISPLAY ROOMS' from menu
*********************************************************************************
Following rooms have been added
Room Number :  101
Room Number :  103
*********************************************************************************
```

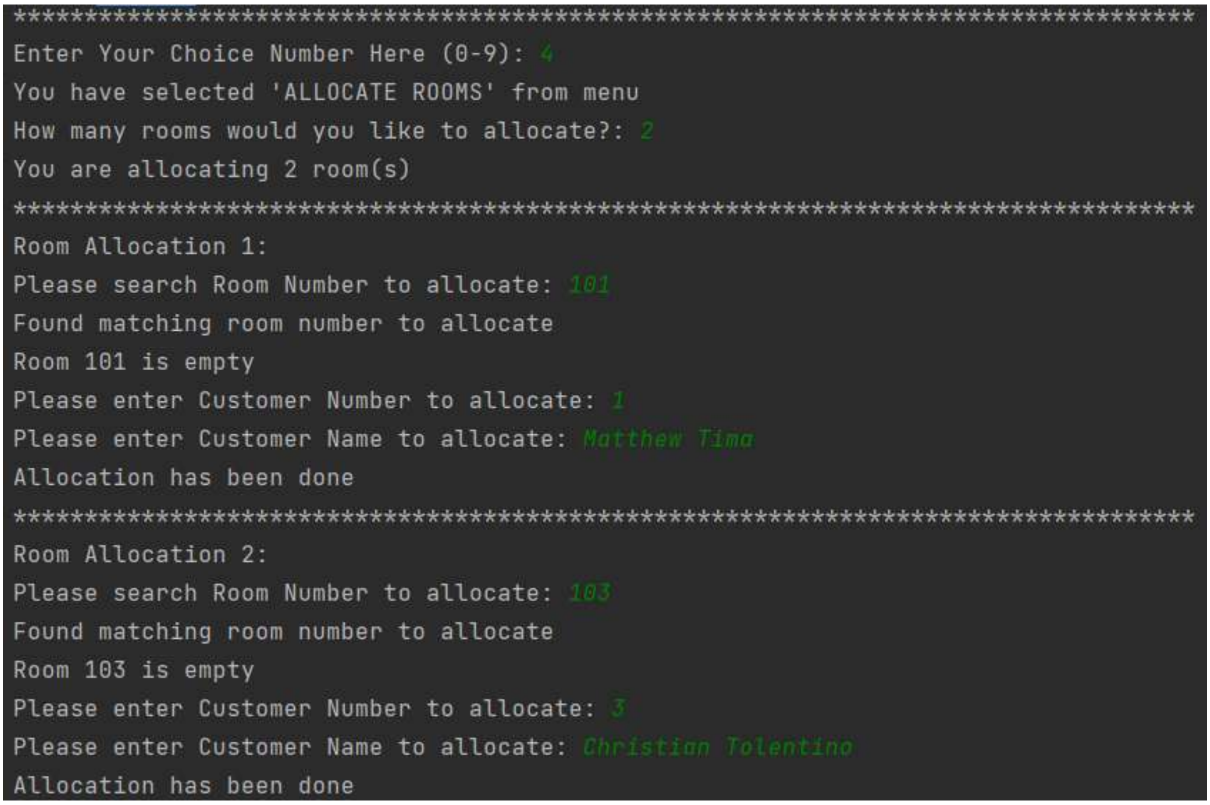
This is to display the number of rooms added from option 1, in the beginning. As shown in the example of selection 2, deleting a room, room number 102 has been deleted.

If 4 is selected:



```
*********************************************************************************
Enter Your Choice Number Here (0-9): 4
You have selected 'ALLOCATE ROOMS' from menu
How many rooms would you like to allocate?:
```

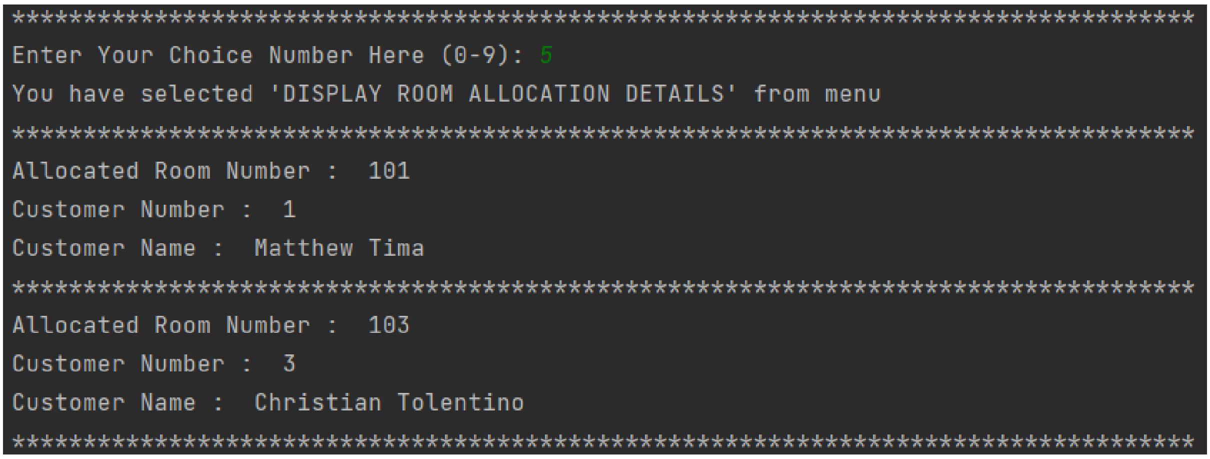
This is for allocating a room to a customer.



```
*********************************************************************************
Enter Your Choice Number Here (0-9): 4
You have selected 'ALLOCATE ROOMS' from menu
How many rooms would you like to allocate?: 2
You are allocating 2 room(s)
*********************************************************************************
Room Allocation 1:
Please search Room Number to allocate: 101
Found matching room number to allocate
Room 101 is empty
Please enter Customer Number to allocate: 1
Please enter Customer Name to allocate: Matthew Tima
Allocation has been done
*********************************************************************************
Room Allocation 2:
Please search Room Number to allocate: 103
Found matching room number to allocate
Room 103 is empty
Please enter Customer Number to allocate: 3
Please enter Customer Name to allocate: Christian Tolentino
Allocation has been done
```

After selecting how many rooms the user would like to allocate, the program will search to see if the room is empty or not. If the room is empty, the user will be asked to assign a customer number and customer name.

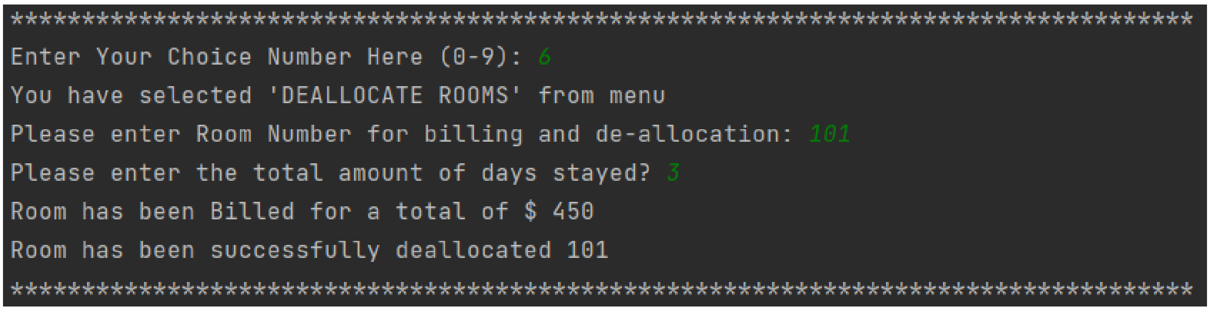
If 5 is selected:



```
***********************************************************************************
Enter Your Choice Number Here (0-9): 5
You have selected 'DISPLAY ROOM ALLOCATION DETAILS' from menu
***********************************************************************************
Allocated Room Number :  101
Customer Number :  1
Customer Name :  Matthew Tima
***********************************************************************************
Allocated Room Number :  103
Customer Number :  3
Customer Name :  Christian Tolentino
***********************************************************************************
```

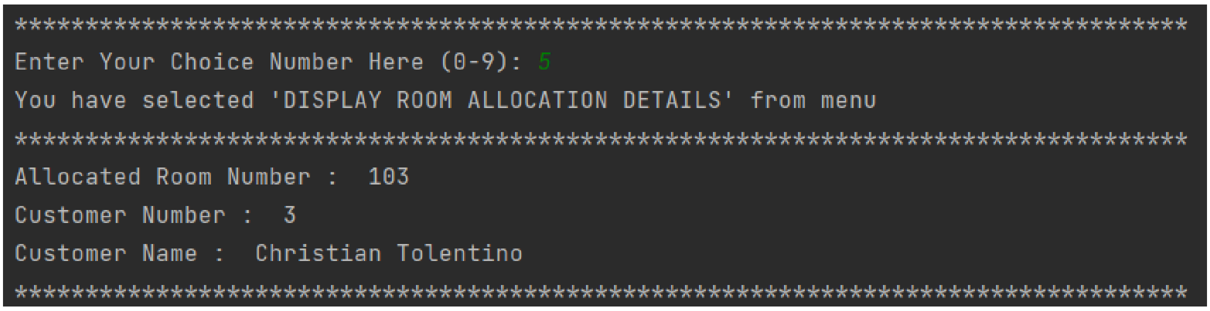
This is used to display the room allocation details, displaying the allocated room number, customer number, and customer name.

If 6 is selected:



```
***********************************************************************************
Enter Your Choice Number Here (0-9): 6
You have selected 'DEALLOCATE ROOMS' from menu
Please enter Room Number for billing and de-allocation: 101
Please enter the total amount of days stayed? 3
Room has been Billed for a total of $ 450
Room has been successfully deallocated 101
***********************************************************************************
```

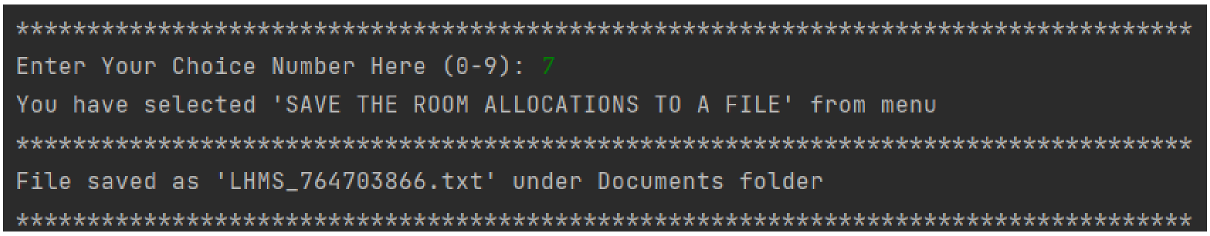
This is to deallocate and bill a room, charging the user per night.



```
***********************************************************************************
Enter Your Choice Number Here (0-9): 5
You have selected 'DISPLAY ROOM ALLOCATION DETAILS' from menu
***********************************************************************************
Allocated Room Number :  103
Customer Number :  3
Customer Name :  Christian Tolentino
***********************************************************************************
```

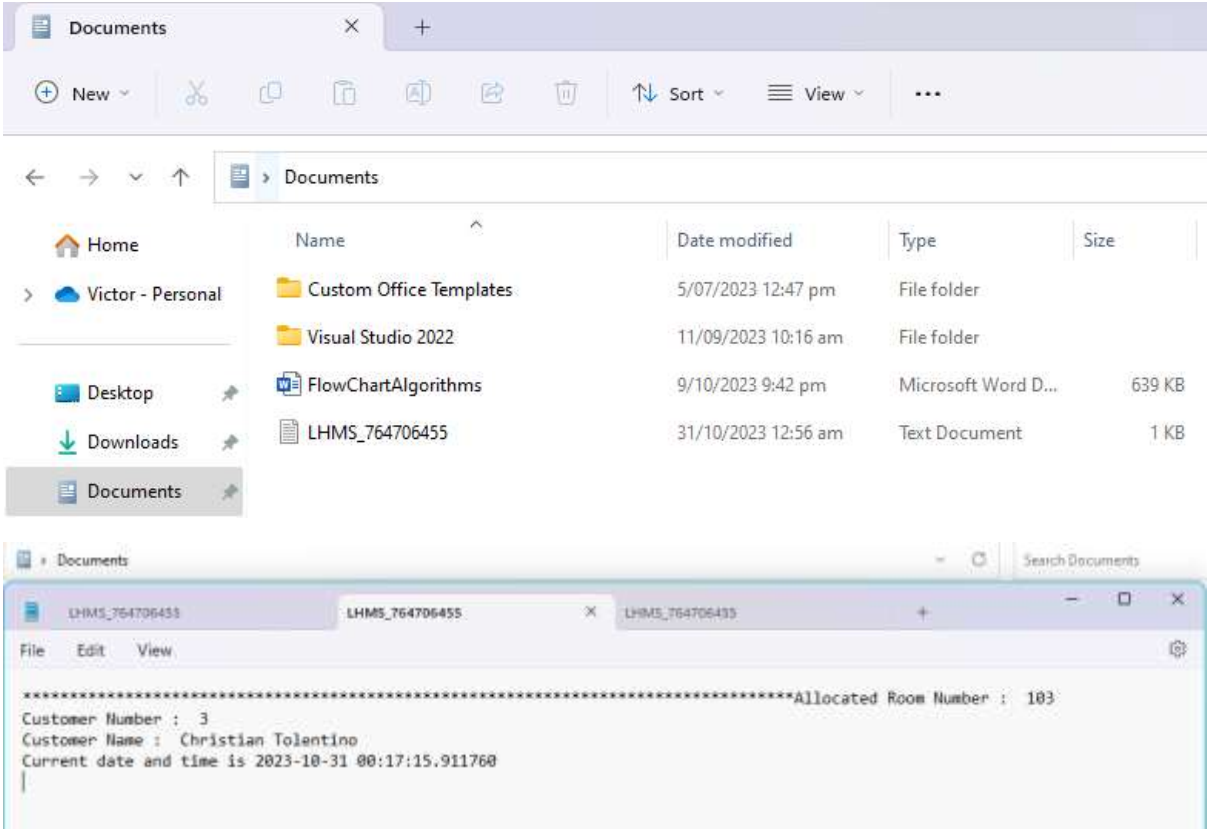
The room has been deallocated.

If 7 is selected:



```
***********************************************************************************
Enter Your Choice Number Here (0-9): 7
You have selected 'SAVE THE ROOM ALLOCATIONS TO A FILE' from menu
***********************************************************************************
File saved as 'LHMS_764703866.txt' under Documents folder
***********************************************************************************
```

This is to save the room allocations to a file that will be found under the Documents folder.

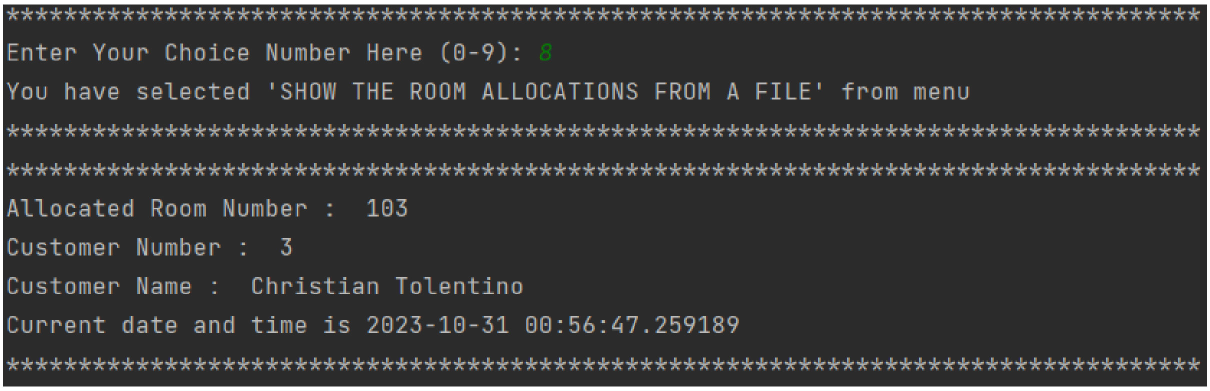


Name	Date modified	Type	Size
Custom Office Templates	5/07/2023 12:47 pm	File folder	
Visual Studio 2022	11/09/2023 10:16 am	File folder	
FlowChartAlgorithms	9/10/2023 9:42 pm	Microsoft Word D...	639 KB
LHMS_764706455	31/10/2023 12:56 am	Text Document	1 KB

```
*******************************************************************************************Allocated Room Number :  103
Customer Number :  3
Customer Name :  Christian Tolentino
Current date and time is 2023-10-31 00:17:15.911760
```

The file containing the customer information.

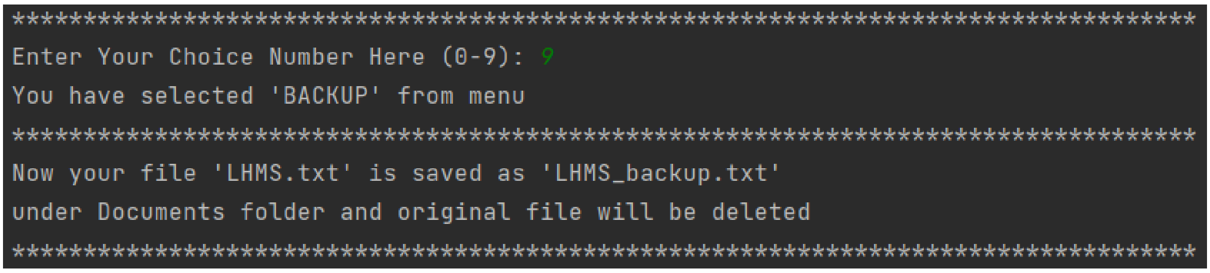
If 8 is selected:



```
************************************************************************************
Enter Your Choice Number Here (0-9): 8
You have selected 'SHOW THE ROOM ALLOCATIONS FROM A FILE' from menu
************************************************************************************
************************************************************************************
Allocated Room Number :  103
Customer Number :  3
Customer Name :  Christian Tolentino
Current date and time is 2023-10-31 00:56:47.259189
************************************************************************************
```

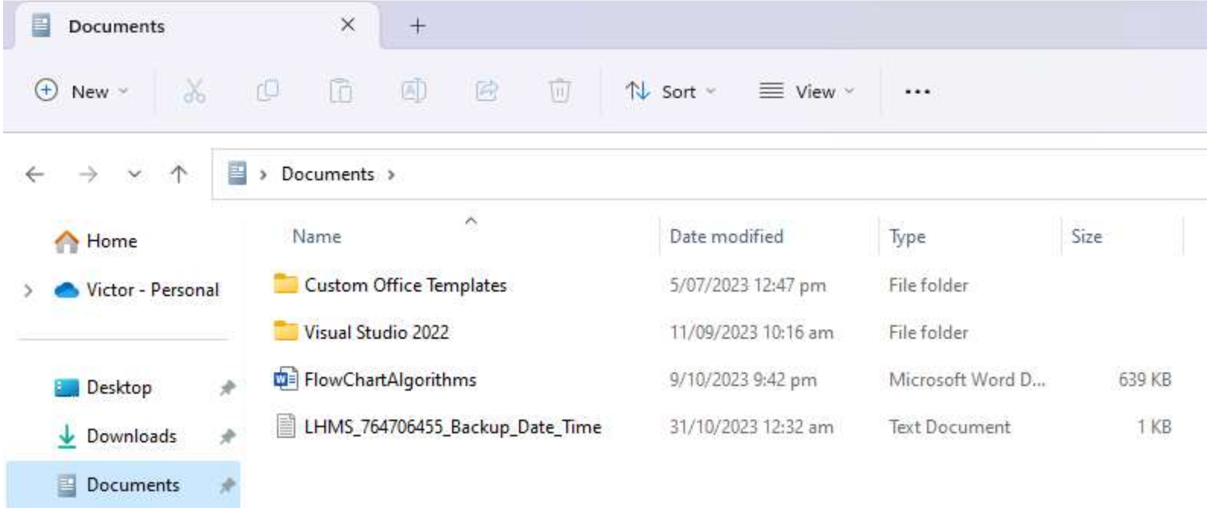
This is to show the room allocation from a file, within the program, displaying the customer information.

If 9 is selected:



```
***********************************************************************************
Enter Your Choice Number Here (0-9): 9
You have selected 'BACKUP' from menu
***********************************************************************************
Now your file 'LHMS.txt' is saved as 'LHMS_backup.txt'
under Documents folder and original file will be deleted
***********************************************************************************
```

This is where the file be backed up under documents, and the original file will be deleted.



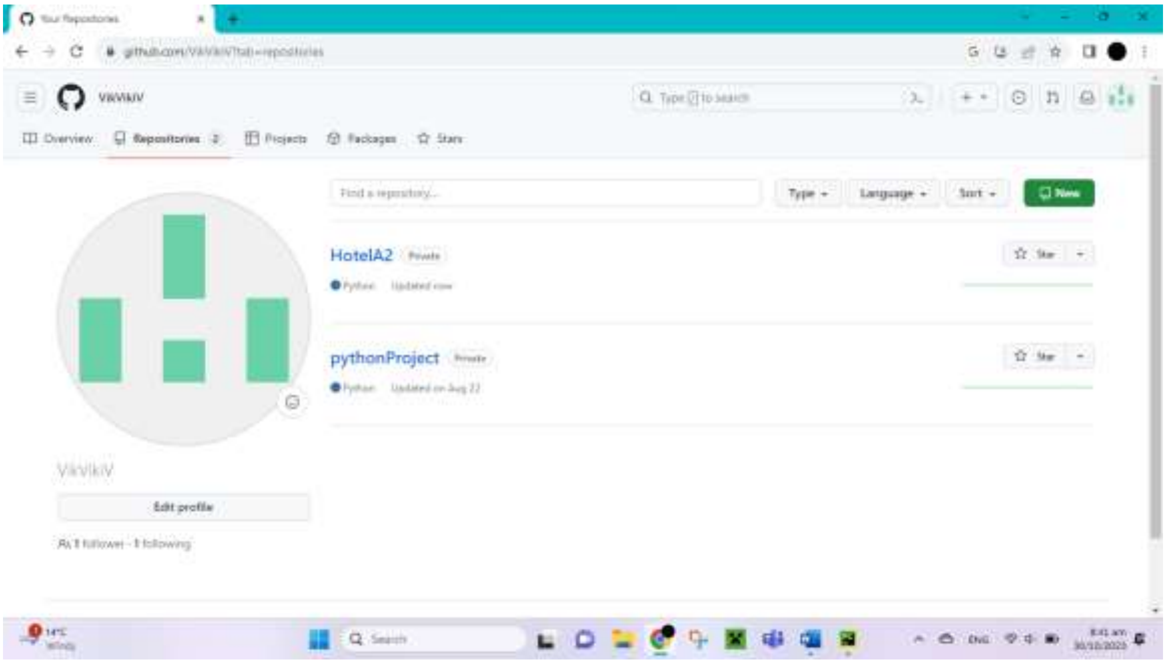
Name	Date modified	Type	Size
Custom Office Templates	5/07/2023 12:47 pm	File folder	
Visual Studio 2022	11/09/2023 10:16 am	File folder	
FlowChartAlgorithms	9/10/2023 9:42 pm	Microsoft Word D...	639 KB
LHMS_764706455_Backup_Date_Time	31/10/2023 12:32 am	Text Document	1 KB

The backed-up file found under documents.

This will conclude the user guide for the Langham Hotel Management System.

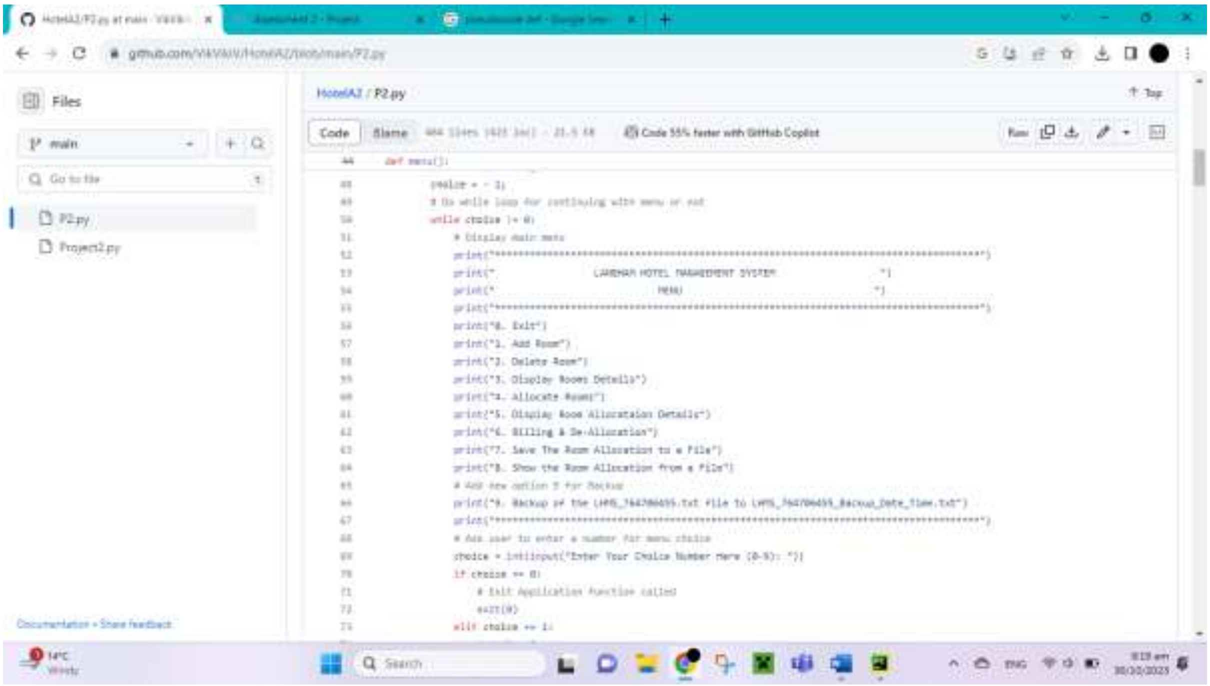
GitHub Repository Evidence

e. Version control: GitHub repository evidence **(2 Marks)**



HotelA2 Private
Python Updated now

pythonProject Private
Python Updated on Aug 22


github.com/VikVikVi/HotelA2/blob/main/P2.py
Files
main
Go to file
P2.py
Project2.py
Documentation • Share feedback
HotelA2 / P2.py
Code
Blame
Code 55% faster with GitHub Copilot
Raw
Top
Search

```
    def menu():
        choice = - 1;
        # Do while loop for continuing with menu or exit
        while choice != 0:
            # Display main menu
            print("********************************************************************************")
            print("                    LANGHAM HOTEL MANAGEMENT SYSTEM                    ")
            print("                                 MENU                                  ")
            print("********************************************************************************")
            print("0. Exit")
            print("1. Add Room")
            print("2. Delete Room")
            print("3. Display Rooms Details")
            print("4. Allocate Rooms")
            print("5. Display Room Allocation Details")
            print("6. Billing & De-Allocation")
            print("7. Save The Room Allocation to a File")
            print("8. Show the Room Allocation from a File")
            # Add new option 9 for Backup
            print("9. Backup of the LHMS_764700455.txt file to LHMS_764700455_Backup_Date_Time.txt")
            print("********************************************************************************")
            # Ask user to enter a number for menu choice
            choice = int(input("Enter Your Choice Number Here (0-9): "))
            if choice == 0:
                # Exit Application function called
                exit(0)
            elif choice == 1:
```

Self-Reflection

5. Self-reflection - Building the Quality Application using besting coding practices and standards: What changes were made to the program and why? **(4 Marks)**

In the process of constructing or converting the code from C# to Python, certain changes were made to make sense of how the code would function from a C# into a program written in Python code. In the given code of C#, the billing part of the program was missing, so it was needed to be implemented in the Python code. With the transfiguration of the code, most of the code was relatively easy to understand on how to change it into Python, but there were a few that were difficult to interpret, thus lending assistance from Sir Jatinder. Certain Errors were encountered and as a counter measure, Try and Except blocks were used to capture specific errors and solve them or provide an alternative solution. Overall, I learned to see how the code was broken into various phases for an easier understanding, breakdown, and implementation for a complete and functional Hotel Management program.