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:	Matthew	**Student ID:**	764705436
Assessment No & Type:	Assessment 2 Project	**Cohort:**	
Due Date:	30/10/23	**Date Submitted:**	
Tutor's Name:	Jatindah 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**

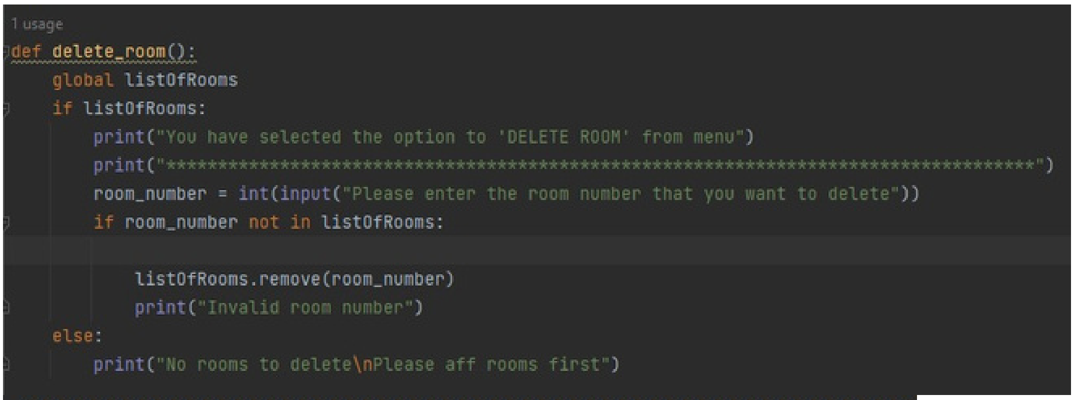
Table of Contents

Introduction:

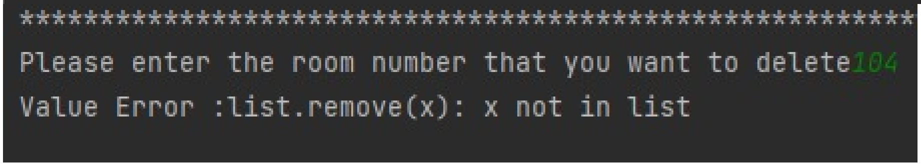
The purpose of this assessment is to re-implement its hotel management system using Python programming. The project, which aims to improve room allocation, deallocation, and status monitoring, is a pilot program with potential for future integration into a full-fledged database system. The Python-based application will be console-based, incorporating I/O operations and robust exception handling. The project represents a significant step towards streamlining operations, enhancing guest experience, and showcasing Python's adaptability in software development. The success of this project will have significant implications for LANGHAM Hotels' future hotel management software.

Task 2 (Errors & Exception Handling)

ValueError: Python code problem is that the remove() method is being called on an empty list. this causes a ValueError exception

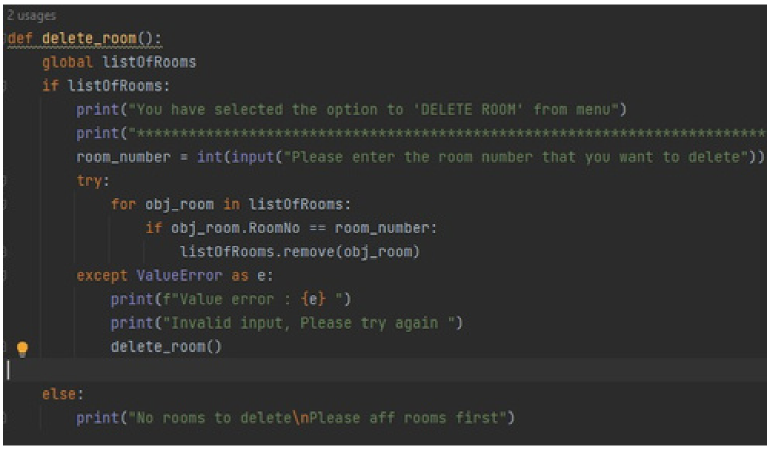


```
1 usage
def delete_room():
    global listOfRooms
    if listOfRooms:
        print("You have selected the option to 'DELETE ROOM' from menu")
        print("*************************************************************************************")
        room_number = int(input("Please enter the room number that you want to delete"))
        if room_number not in listOfRooms:

            listOfRooms.remove(room_number)
            print("Invalid room number")
    else:
        print("No rooms to delete\nPlease aff rooms first")
```



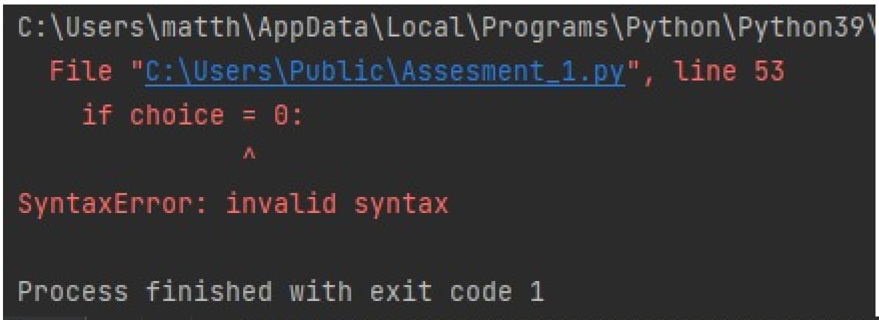
```
********************************************************
Please enter the room number that you want to delete104
Value Error :list.remove(x): x not in list
```

Solution: Try and except block has been implemented

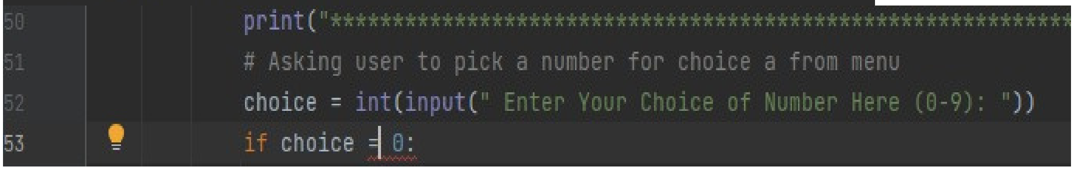


```
2 usages
def delete_room():
    global listOfRooms
    if listOfRooms:
        print("You have selected the option to 'DELETE ROOM' from menu")
        print("*********************************************************************
        room_number = int(input("Please enter the room number that you want to delete"))
        try:
            for obj_room in listOfRooms:
                if obj_room.RoomNo == room_number:
                    listOfRooms.remove(obj_room)
        except ValueError as e:
            print(f"Value error : {e} ")
            print("Invalid input, Please try again ")
            delete_room()

    else:
        print("No rooms to delete\nPlease aff rooms first")
```

SyntaxError:

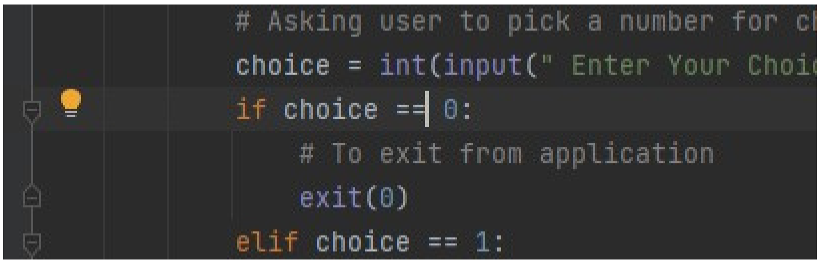


```
C:\Users\matth\AppData\Local\Programs\Python\Python39\
  File "C:\Users\Public\Assesment_1.py", line 53
    if choice = 0:
              ^
SyntaxError: invalid syntax

Process finished with exit code 1
```



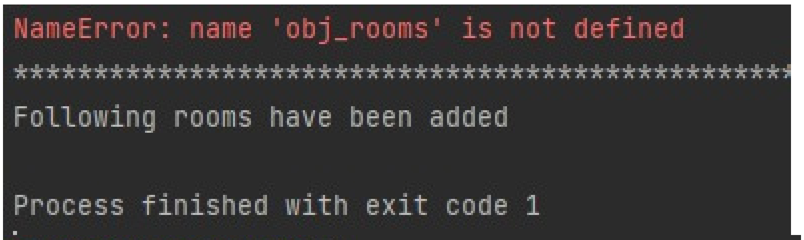
```
50            print("**************************************************************
51            # Asking user to pick a number for choice a from menu
52            choice = int(input(" Enter Your Choice of Number Here (0-9): "))
53            if choice = 0:
```

Solution: To compare the two values, you need to use the equality operator, which is why the code in the needs two equal signs

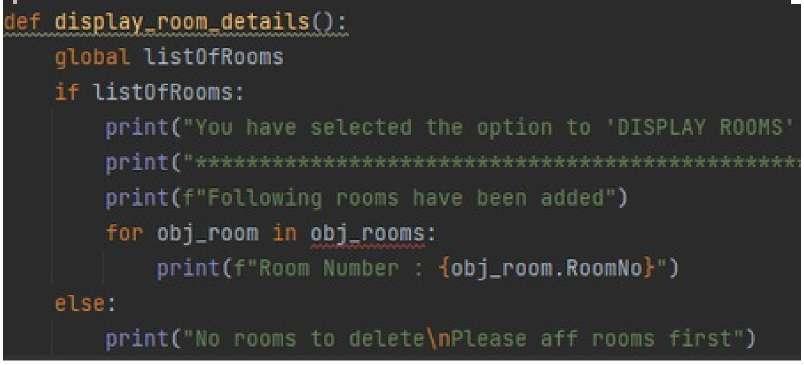


```
            # Asking user to pick a number for ch
            choice = int(input(" Enter Your Choic
            if choice == 0:
                # To exit from application
                exit(0)
            elif choice == 1:
```

NameError: the variable obj_rooms are not defined. This causes a NameError exception when the display_room_details() function tries to use it.

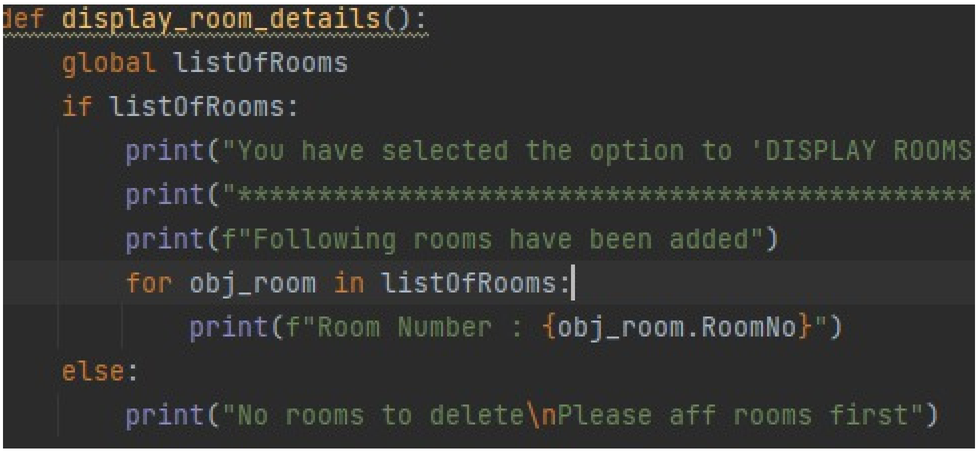


```
NameError: name 'obj_rooms' is not defined
*************************************************
Following rooms have been added

Process finished with exit code 1
```



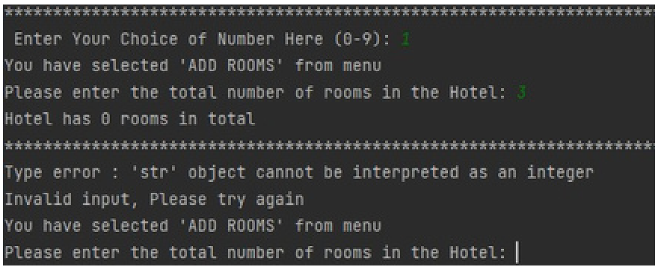
```
def display_room_details():
    global listOfRooms
    if listOfRooms:
        print("You have selected the option to 'DISPLAY ROOMS'
        print("**************************************************
        print(f"Following rooms have been added")
        for obj_room in obj_rooms:
            print(f"Room Number : {obj_room.RoomNo}")
    else:
        print("No rooms to delete\nPlease aff rooms first")
```

Solution: To fix the error, I needed to define the obj_rooms variable before calling the display_room_details() function.



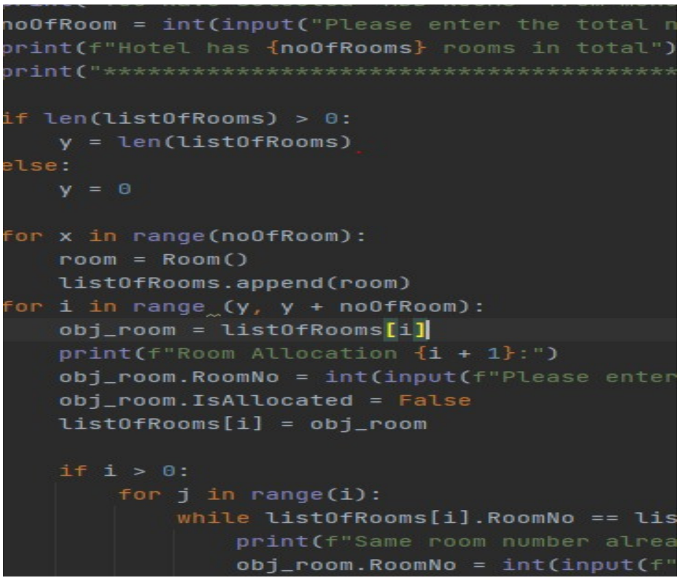
```
def display_room_details():
    global listOfRooms
    if listOfRooms:
        print("You have selected the option to 'DISPLAY ROOMS
        print("**********************************************
        print(f"Following rooms have been added")
        for obj_room in listOfRooms:
            print(f"Room Number : {obj_room.RoomNo}")
    else:
        print("No rooms to delete\nPlease aff rooms first")
```

TypeError: cannot perform arithmetic or comparison operations on strings, so this raises a TypeError exception.



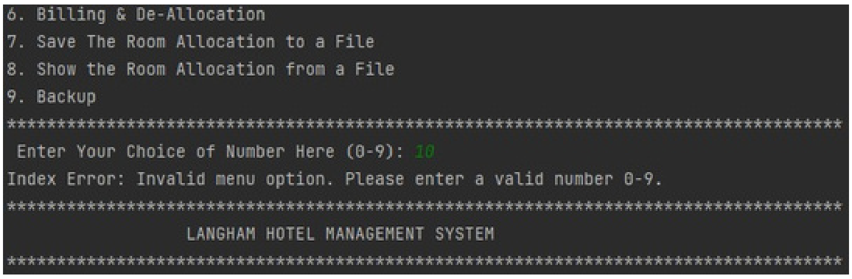
```
**************************************************************************
 Enter Your Choice of Number Here (0-9): 1
You have selected 'ADD ROOMS' from menu
Please enter the total number of rooms in the Hotel: 3
Hotel has 0 rooms in total
**************************************************************************
Type error : 'str' object cannot be interpreted as an integer
Invalid input, Please try again
You have selected 'ADD ROOMS' from menu
Please enter the total number of rooms in the Hotel: 
```

Solution: To fix the error, I needed to convert the string noOfRooms to an integer before using it in these operations. You can do this using the int() function.



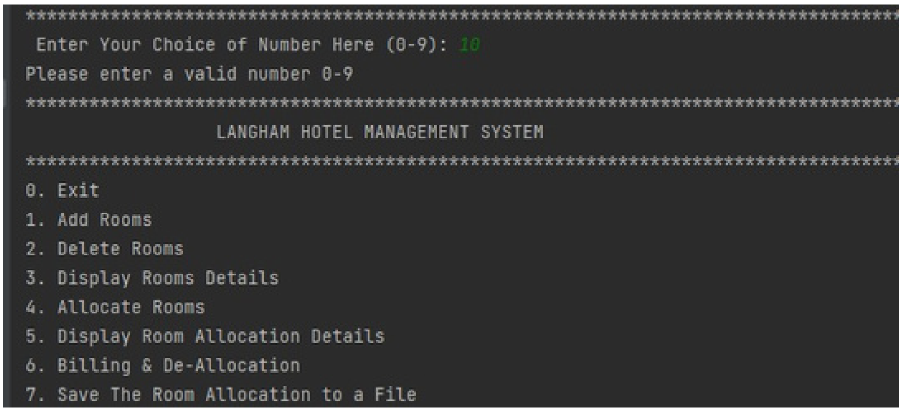
```
noOfRoom = int(input("Please enter the total n
print(f"Hotel has {noOfRooms} rooms in total")
print("****************************************

if len(listOfRooms) > 0:
    y = len(listOfRooms)
else:
    y = 0

for x in range(noOfRoom):
    room = Room()
    listOfRooms.append(room)
for i in range (y, y + noOfRoom):
    obj_room = listOfRooms[i]
    print(f"Room Allocation {i + 1}:")
    obj_room.RoomNo = int(input(f"Please enter
    obj_room.IsAllocated = False
    listOfRooms[i] = obj_room

    if i > 0:
        for j in range(i):
            while listOfRooms[i].RoomNo == lis
                print(f"Same room number alrea
                obj_room.RoomNo = int(input(f"
```

IndexError: User has selected a number out of range 0-9 of menu option.



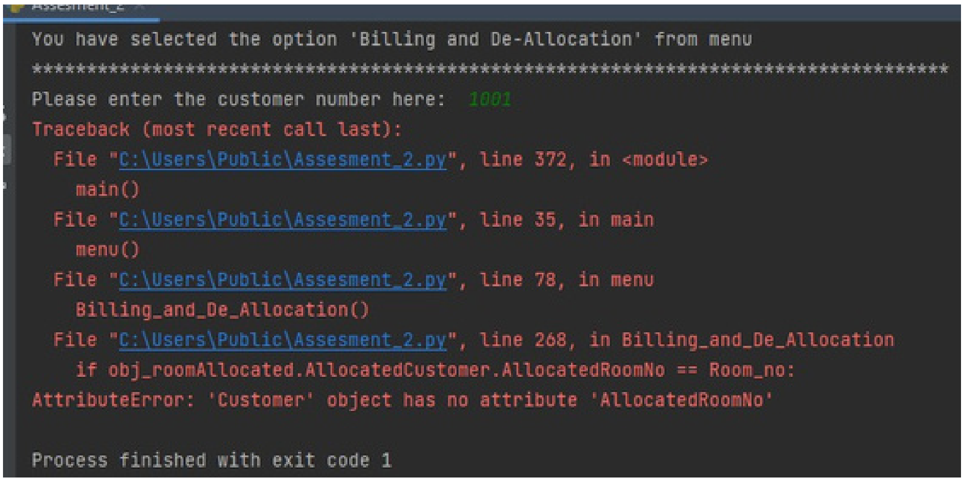
```
6. Billing & De-Allocation
7. Save The Room Allocation to a File
8. Show the Room Allocation from a File
9. Backup
*************************************************************************************
 Enter Your Choice of Number Here (0-9): 10
Index Error: Invalid menu option. Please enter a valid number 0-9.
*************************************************************************************
                  LANGHAM HOTEL MANAGEMENT SYSTEM
*************************************************************************************
```

Solution: Call back to menu option, warning customer to keep within range of menu options.



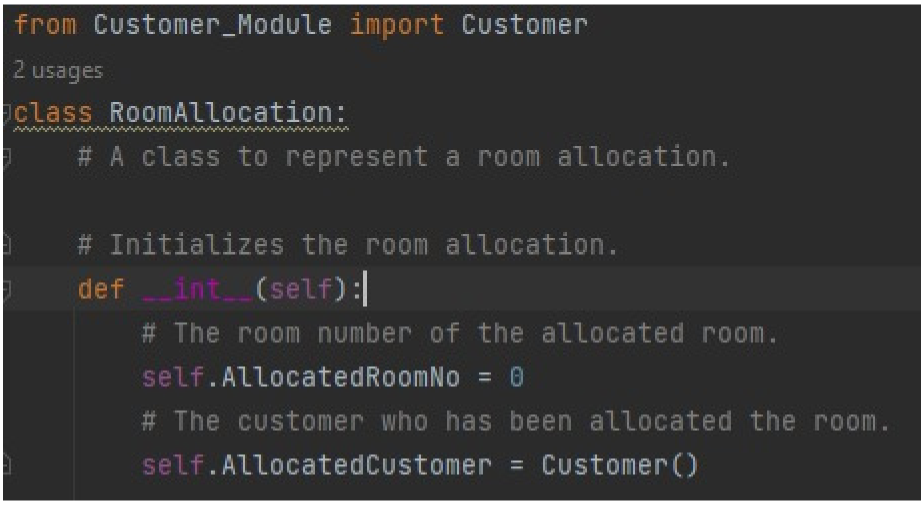
```
*************************************************************************************
 Enter Your Choice of Number Here (0-9): 10
Please enter a valid number 0-9
*************************************************************************************
                  LANGHAM HOTEL MANAGEMENT SYSTEM
*************************************************************************************
0. Exit
1. Add Rooms
2. Delete Rooms
3. Display Rooms Details
4. Allocate Rooms
5. Display Room Allocation Details
6. Billing & De-Allocation
7. Save The Room Allocation to a File
```

AttributeError:



```
You have selected the option 'Billing and De-Allocation' from menu
*************************************************************************************
Please enter the customer number here:  1001
Traceback (most recent call last):
  File "C:\Users\Public\Assesment_2.py", line 372, in <module>
    main()
  File "C:\Users\Public\Assesment_2.py", line 35, in main
    menu()
  File "C:\Users\Public\Assesment_2.py", line 78, in menu
    Billing_and_De_Allocation()
  File "C:\Users\Public\Assesment_2.py", line 268, in Billing_and_De_Allocation
    if obj_roomAllocated.AllocatedCustomer.AllocatedRoomNo == Room_no:
AttributeError: 'Customer' object has no attribute 'AllocatedRoomNo'

Process finished with exit code 1
```

Solution: The code (AllocatedRoomNo) is properly imported and attributed



```
from Customer_Module import Customer
2 usages
class RoomAllocation:
    # A class to represent a room allocation.

    # Initializes the room allocation.
    def __int__(self):
        # The room number of the allocated room.
        self.AllocatedRoomNo = 0
        # The customer who has been allocated the room.
        self.AllocatedCustomer = Customer()
```

Task 3:

Pseudocode:

```
FUNCTION menu()
        BEGIN choice := -1
              WHILE choice != 0 DISPLAY menu options READ choice IF choice == 0 THEN EXIT
               ELSE IF choice == 1
               THEN CALL add_room()
               ELSE IF choice == 2
               THEN CALL delete_room()
               ELSE IF choice == 3
               THEN CALL display_room_details()
               ELSE IF choice == 4
               THEN CALL allocate_rooms()
               ELSE IF choice == 5
               THEN CALL display_room_allocations_details()
               ELSE IF choice == 6
               THEN CALL Billing_and_De_Allocation()
               ELSE IF choice == 7
               THEN CALL save_room_allocations()
               ELSE IF choice == 8
               THEN CALL show_room_allocations()
               ELSE IF choice == 9
               THEN CALL backup()
      ELSE DISPLAY "Please enter a valid number 0-9"
      END
```

```
1.  DISPLAY "You have selected 'ADD ROOMS' from menu"
2.  READ noOfRooms
3.  IF noOfRooms is not an integer
            DISPLAY "Invalid input, Please try again"
            CALL add_room()
4.  ELSE
            DISPLAY "Hotel has {noOfRooms} rooms in total"
            IF len(listOfRooms) > 0
                    SET y = len(listOfRooms)
            ELSE
```

```
                SET y = 0
            FOR x in range(noOfRooms)
                CREATE a new room object
                ADD the room object to listOfRooms
            ENDFOR
            FOR i in range(y, y + noOfRooms)
                SET obj_room = listOfRooms[i]
                DISPLAY "Room Allocation {i + 1}:"
                READ obj_room.RoomNo
                SET obj_room.IsAllocated = False
                UPDATE listOfRooms[i] with obj_room
                IF i > 0
                    FOR j in range(i)
                        WHILE listOfRooms[i].RoomNo == listOfRooms[j].RoomNo
                            DISPLAY "Same room number already exist"
                            READ obj_room.RoomNo
                            SET obj_room.IsAllocated = False
                            UPDATE listOfRooms[i] with obj_room
                    ENDWHILE
                ENDFOR
          ENDIFENDFOR
END IF
HANDLING EXCEPTIONS:

    IF ValueError is raised
        DISPLAY "Value error : {e}"
        DISPLAY "Invalid input, Please try again"
        CALL add_room()
    END IF


IF listOfRooms is not empty
DISPLAY "You have selected the option to 'DELETE ROOM' from menu"
        READ room_number
        FOR obj_room in listOfRooms
            IF obj_room.RoomNo == room_number
                REMOVE obj_room from listOfRooms
                DISPLAY f"Room with Room Number : {room_number} has been deleted
                successfully"
                BREAK from the FOR loop

        ENDFOR
    ELSE
```

```
        DISPLAY "No rooms to delete\nPlease add rooms first"

    ENDIF

HANDLING EXCEPTIONS:

    IF ValueError is raised

        DISPLAY f"Value error : {e}"

        DISPLAY "Invalid input, Please try again"

        CALL delete_room()

    END IF

IF listOfRooms is not empty

        DISPLAY "You have selected the option to 'DISPLAY ROOMS' from menu"

        DISPLAY "Following rooms have been added"

        FOR obj_room in listOfRooms

            DISPLAY f"Room Number : {obj_room.RoomNo}"

        ENDFOR

    ELSE

        DISPLAY "No rooms to delete\nPlease add rooms first"

    ENDIF

 1. DISPLAY "You have selected the option to 'ALLOCATE ROOMS' from menu"
 2. READ allocate_room
 3. WHILE allocate_room > len(listOfRooms)

        DISPLAY "You cannot allocate more rooms than the total number of rooms in the Hotel\n"

        DISPLAY "Please enter a number between 1--{len(listOfRooms)}: "

        READ allocate_room

 4. ENDWHILE
 5. DISPLAY f"You are allocating {allocate_room} room(s)"
 6. FOR i in range(allocate_room)

        CREATE a new RoomAllocation object

        CREATE a new Customer object

        READ searchRoom

        FOR j in range(len(listOfRooms))

             IF searchRoom == listOfRooms[j].RoomNo

                IF not listOfRooms[j].IsAllocated

                    DISPLAY f"Room {listOfRooms[j].RoomNo} is empty "
                    READ customer.CustomerNo
                    READ customer.CustomerName
                    SET listOfRooms[j].IsAllocated = True
                    DISPLAY "Allocation has been done"
                    UPDATE the RoomAllocation object with the allocated room number and
                    customer
                    APPEND the RoomAllocation object to the list of room allocations
                    INCREMENT i
                    BREAK the loop

                ELSE
                    DISPLAY f"Room {listOfRooms[j].RoomNo} is already occupied\n"
                    DISPLAY "Please enter another room to allocate"
                    DECREMENT i
                    BREAK the loop
            ELSE
                WHILE j == len(listOfRooms) - 1
                    DISPLAY f"Could not find matching room number to allocate\n"
                    DISPLAY f"Please enter correct room number or add room first"
                    DECREMENT i
                    BREAK the loop
  7.  ENDFOR
HANDLING EXCEPTIONS:
    IF ValueError is raised
        DISPLAY "Invalid input. Please try again."
        CALL allocate_rooms()
    END IF
  1.  IF listOfRoomAllocations is not empty
        DISPLAY "You have selected 'DISPLAY ROOMS ALLOCATION DETAILS' from menu."
        DISPLAY "Following rooms have been allocated:"
        FOR obj_room in listOfRoomAllocations
            DISPLAY f"Allocated Room Number: {obj_room.AllocatedRoomNo}"
            DISPLAY f"Customer Number: {obj_room.AllocatedCustomer.CustomerNo}"
            DISPLAY f"Customer Name: {obj_room.AllocatedCustomer.CustomerName}"
```

```
2.  ELSE
            DISPLAY "No allocated rooms to display\nPlease allocate rooms first"
3.  ENDIF
```

Pseudocode for Billing_and_De_Allocation() function

INPUT: None

OUTPUT: None

GLOBAL VARIABLES:

```
    listOfRoomAllocations
```

STEPS:

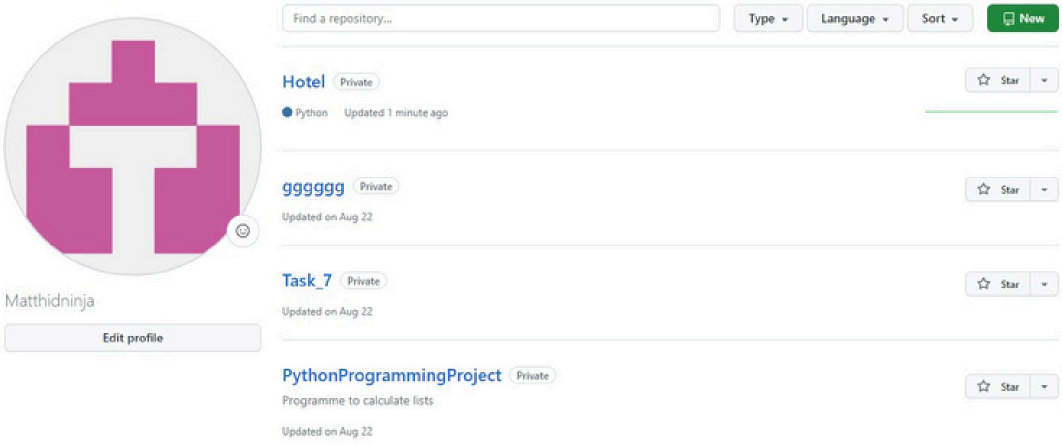
```
1.  DISPLAY "You have selected the option 'Billing and De-Allocation' from menu"
2.  READ Room_no
3.  FOR obj_roomAllocated in listOfRoomAllocations
            IF obj_roomAllocated.AllocatedRoomNo == Room_no
                    READ Days
                    CALCULATE Billing_Fee = Days * Fee
                    DISPLAY f"Customer detail {Room_no}: "
                    DISPLAY f"Days stayed: {Days}"
                    DISPLAY f"Daily fee: {Fee}"
                    DISPLAY f"Total room fee due: {Billing_Fee}"
                    REMOVE obj_roomAllocated from listOfRoomAllocations
                    DISPLAY f"Room has been De-Allocated for customer {Room_no}"
                    BREAK the loop

            ELSE
                    DISPLAY f"Allocated room has no matching room number {Room_no}"
4.  ENDFOR
5.  HANDLE EXCEPTIONS
```

```
1.  DISPLAY "You have selected 'SAVE THE ROOM ALLOCATIONS TO A FILE' from menu."
    OPEN the data file for writing.
    GET the current date and time.
    FOR obj_roomAllocated in listOfRoomAllocations
            CREATE a string to add to the file.
            WRITE the room allocation to the file.

    CLOSE the data file.

    DISPLAY a message informing the user that the room allocations have been saved successfully.
```

```
1.  DISPLAY "You have selected 'SHOW THE ROOM ALLOCATIONS FROM THE FILE' from menu."
2.  OPEN the data file for reading.
3.  READ the first line of the file.
4.  WHILE lines is not empty

            FOR obj_roomAllocated in listOfRoomAllocations

                    DISPLAY f"Room Number: {obj_roomAllocated.AllocatedRoomNo}"

                    DISPLAY f"Customer Number: {obj_roomAllocated.AllocatedCustomer.CustomerNo}"

                    DISPLAY f"Customer Name: {obj_roomAllocated.AllocatedCustomer.CustomerName}"

            READ the next line of the file.

5.  CLOSE the data file.
```

```
1.  DISPLAY "You have selected the option 'BACK UP' from menu"
2.  IF filePathBackUp exists
        o   DISPLAY "'lhms_backup.txt' file already exist \nExisting file will be deleted"
        o   DELETE filePathBackUp
3.  RENAME filepath to filePathBackUp
4.  DISPLAY a success message informing the user that the file has been backed up successfully.
```

GitHub repository evidence:



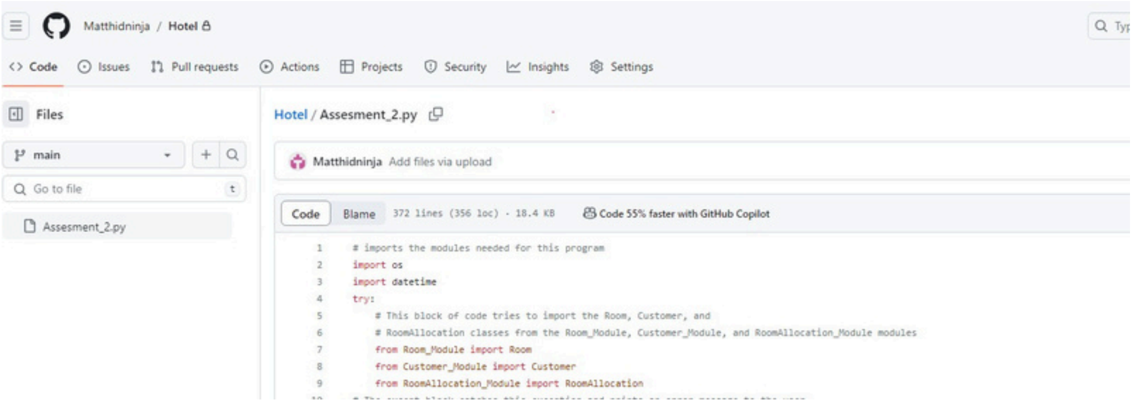
Matthidninja

Hotel Private
Python Updated 1 minute ago

gggggg Private
Updated on Aug 22

Task_7 Private
Updated on Aug 22

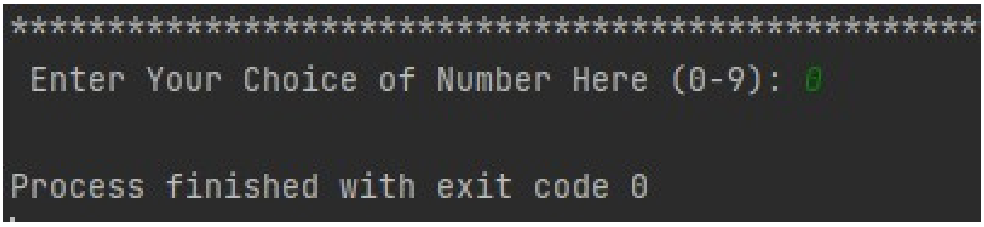
PythonProgrammingProject Private
Programme to calculate lists
Updated on Aug 22



```python
# imports the modules needed for this program
import os
import datetime
try:
    # This block of code tries to import the Room, Customer, and
    # RoomAllocation classes from the Room_Module, Customer_Module, and RoomAllocation_Module modules
    from Room_Module import Room
    from Customer_Module import Customer
    from RoomAllocation_Module import RoomAllocation
# The except block catches this exception and prints an error message to the user
```

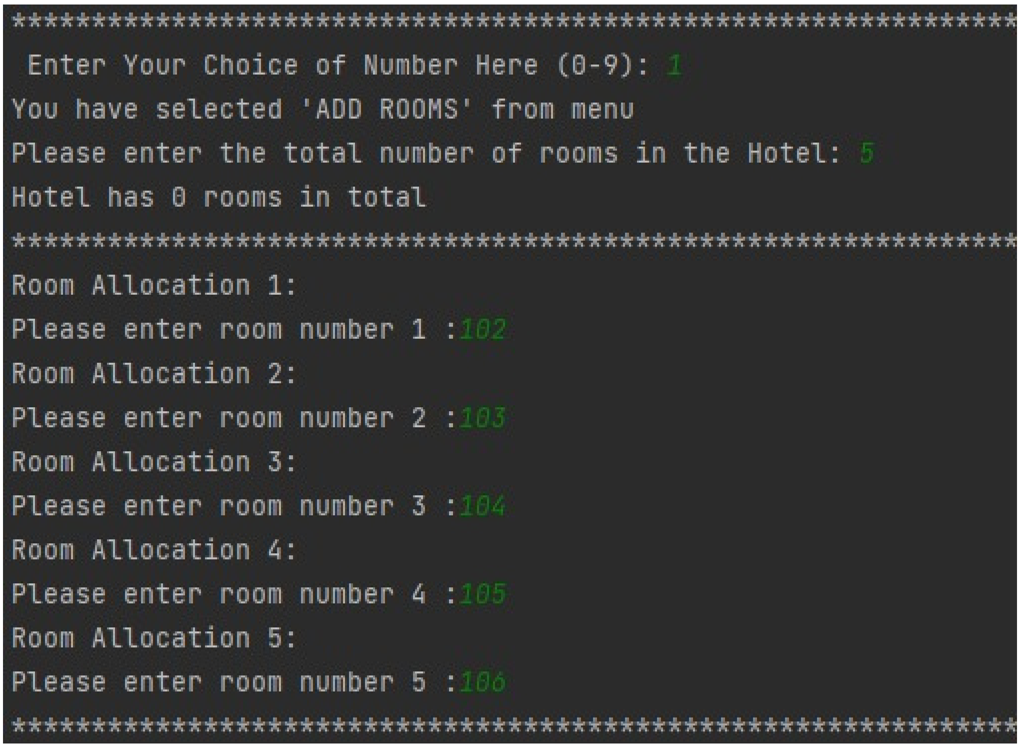
User guide:

Exit Application - Closes the Application



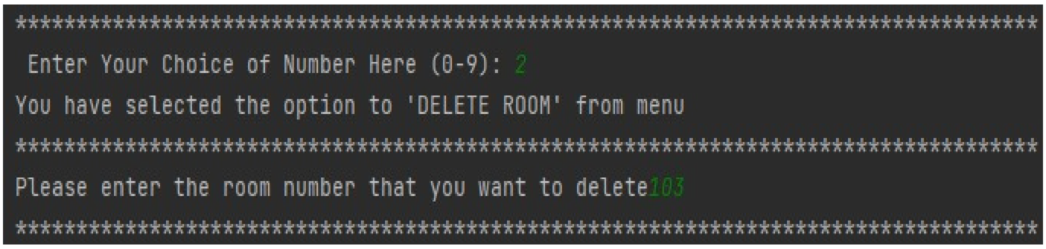
```
************************************************
 Enter Your Choice of Number Here (0-9): 0


Process finished with exit code 0
```

Add Room - add new room/s to the list.



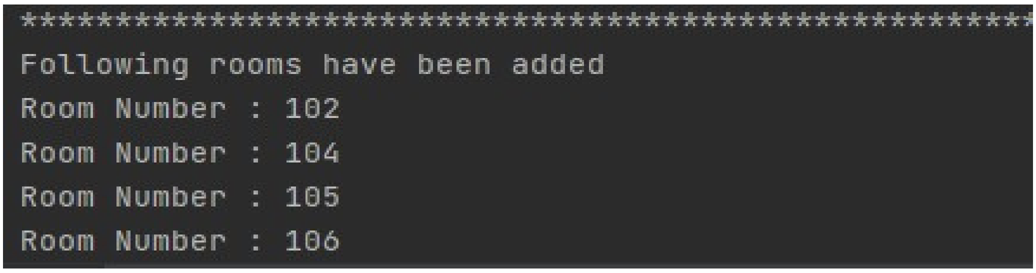
```
********************************************************
 Enter Your Choice of Number Here (0-9): 1
You have selected 'ADD ROOMS' from menu
Please enter the total number of rooms in the Hotel: 5
Hotel has 0 rooms in total
********************************************************
Room Allocation 1:
Please enter room number 1 :102
Room Allocation 2:
Please enter room number 2 :103
Room Allocation 3:
Please enter room number 3 :104
Room Allocation 4:
Please enter room number 4 :105
Room Allocation 5:
Please enter room number 5 :106
********************************************************
```

Delete Room - delete room/s from the list.



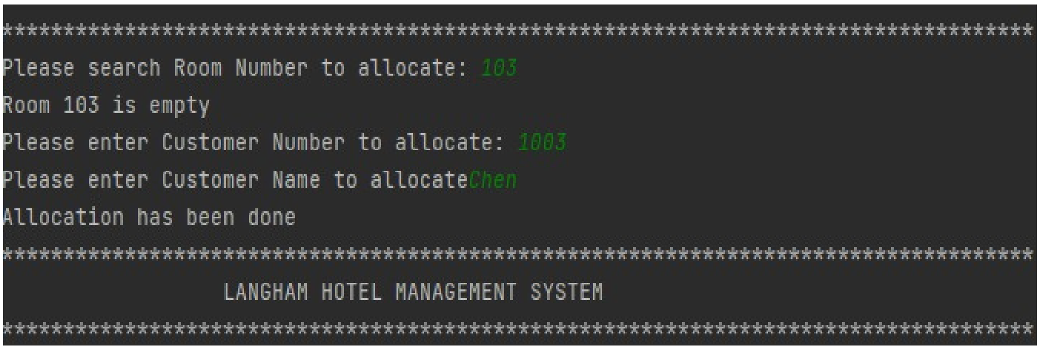
```
************************************************************************************
 Enter Your Choice of Number Here (0-9): 2
You have selected the option to 'DELETE ROOM' from menu
************************************************************************************
Please enter the room number that you want to delete103
************************************************************************************
```

Display Rooms Details - display room features before booking it for a customer.



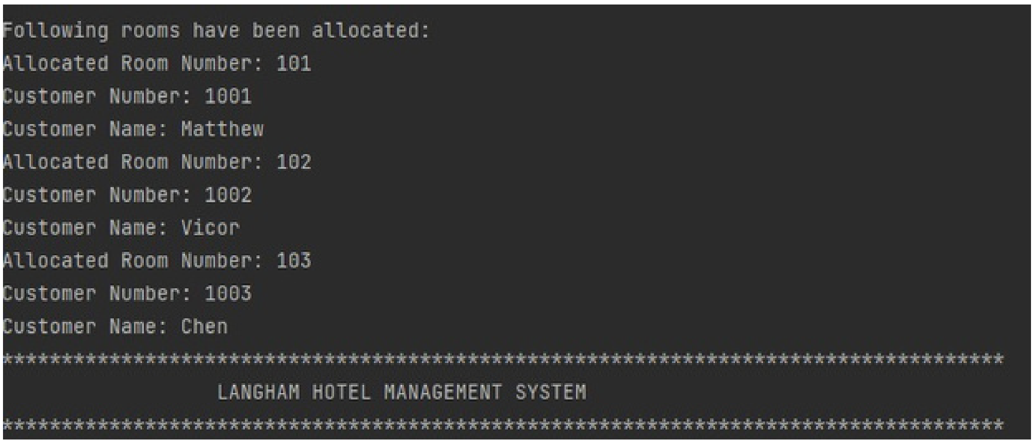
```
****************************************
Following rooms have been added
Room Number : 102
Room Number : 104
Room Number : 105
Room Number : 106
```

Allocate Rooms - book rooms for the customer.



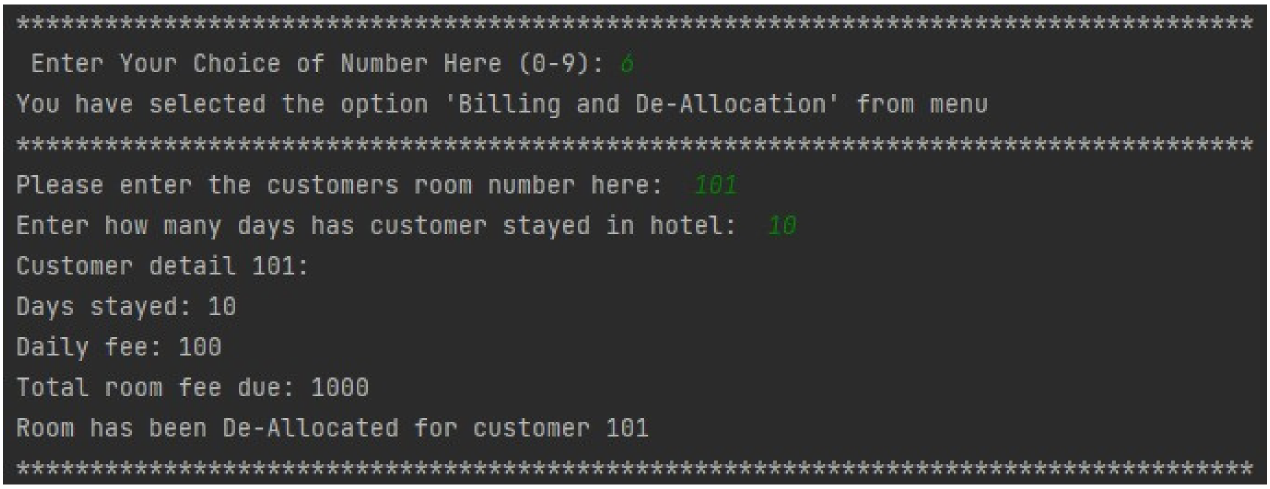
```
************************************************************************************
Please search Room Number to allocate: 103
Room 103 is empty
Please enter Customer Number to allocate: 1003
Please enter Customer Name to allocateChen
Allocation has been done
************************************************************************************
                  LANGHAM HOTEL MANAGEMENT SYSTEM
************************************************************************************
```

Display Room Allocation Details - display room allocation status.



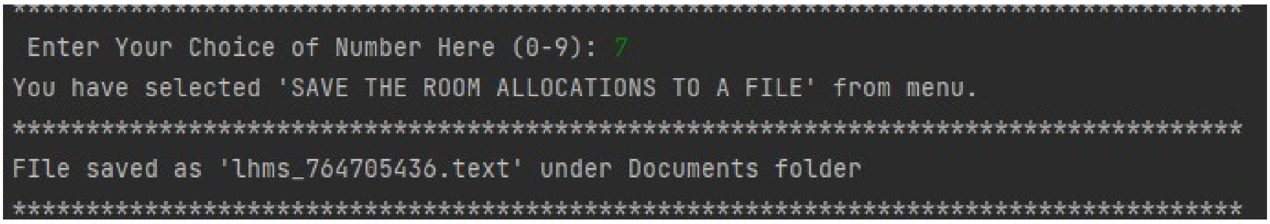
```
Following rooms have been allocated:
Allocated Room Number: 101
Customer Number: 1001
Customer Name: Matthew
Allocated Room Number: 102
Customer Number: 1002
Customer Name: Vicor
Allocated Room Number: 103
Customer Number: 1003
Customer Name: Chen
*********************************************************************************
                  LANGHAM HOTEL MANAGEMENT SYSTEM
*********************************************************************************
```

Billing & De-Allocation of customers rooms.



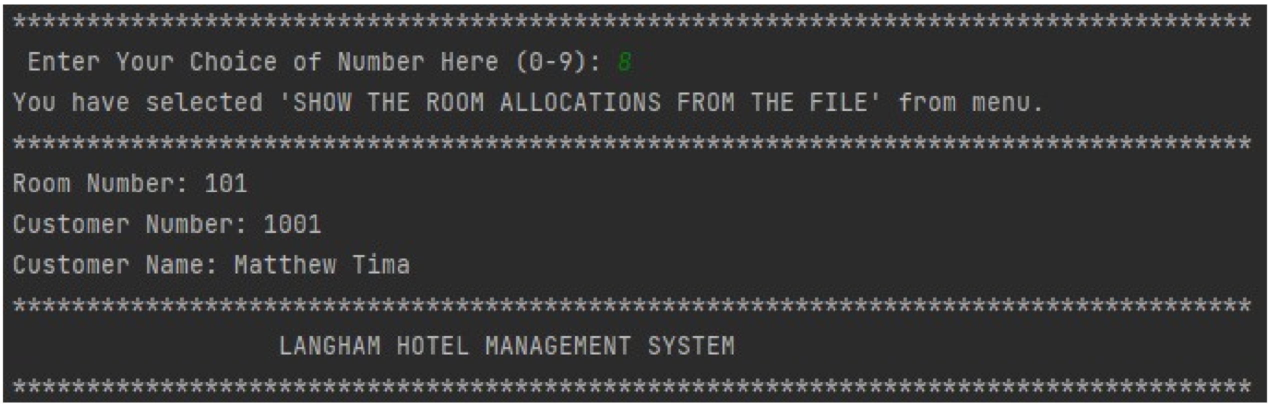
```
*********************************************************************************
 Enter Your Choice of Number Here (0-9): 6
You have selected the option 'Billing and De-Allocation' from menu
*********************************************************************************
Please enter the customers room number here:  101
Enter how many days has customer stayed in hotel:  10
Customer detail 101:
Days stayed: 10
Daily fee: 100
Total room fee due: 1000
Room has been De-Allocated for customer 101
*********************************************************************************
```

Save The Room Allocation to a file with my student Id number.

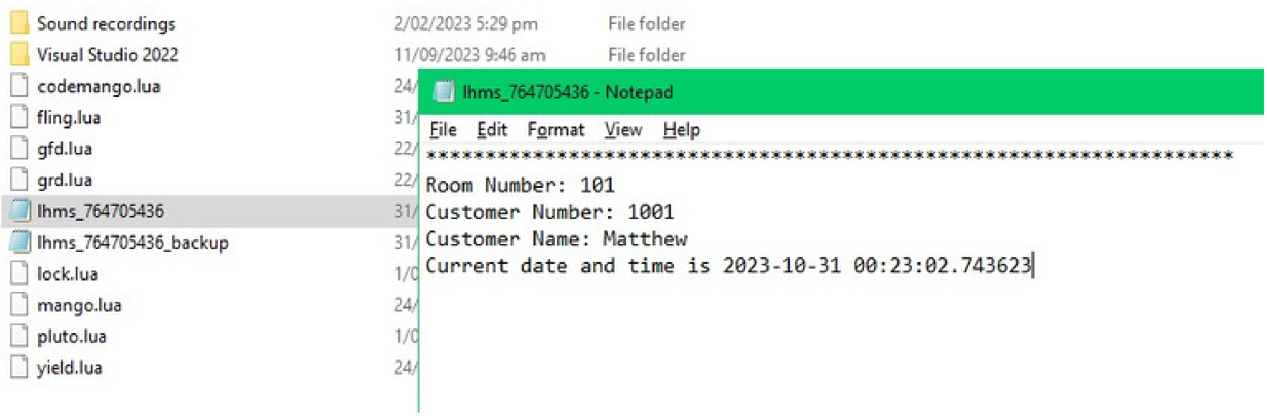


```
*********************************************************************************
 Enter Your Choice of Number Here (0-9): 7
You have selected 'SAVE THE ROOM ALLOCATIONS TO A FILE' from menu.
*********************************************************************************
FIle saved as 'lhms_764705436.text' under Documents folder
*********************************************************************************
```

Show the Room Allocation details from the file.



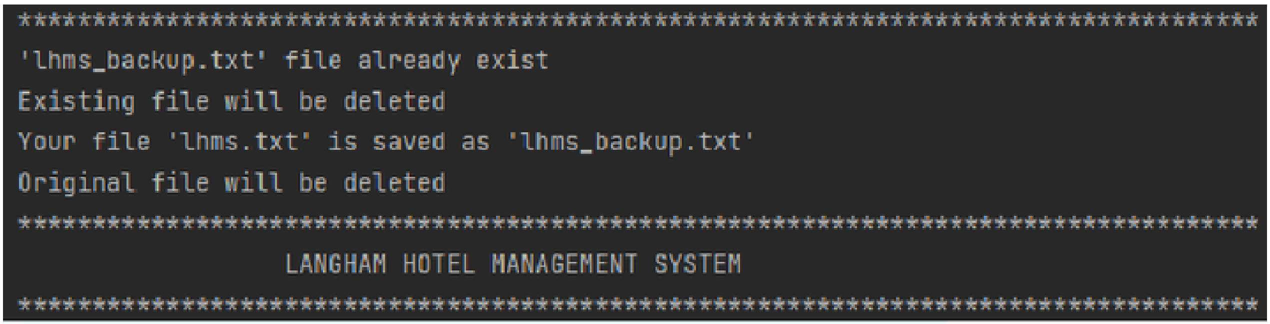
```
*********************************************************************************
 Enter Your Choice of Number Here (0-9): 8
You have selected 'SHOW THE ROOM ALLOCATIONS FROM THE FILE' from menu.
*********************************************************************************
Room Number: 101
Customer Number: 1001
Customer Name: Matthew Tima
*********************************************************************************
                  LANGHAM HOTEL MANAGEMENT SYSTEM
*********************************************************************************
```



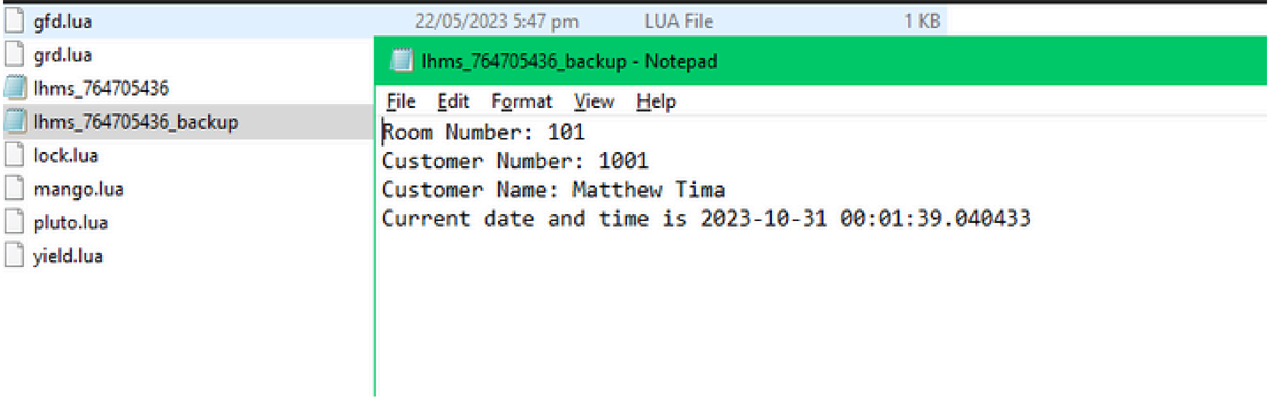
```
Sound recordings          2/02/2023 5:29 pm     File folder
Visual Studio 2022        11/09/2023 9:46 am    File folder
codemango.lua             24/
fling.lua                 31/
gfd.lua                   22/
grd.lua                   22/
lhms_764705436            31/
lhms_764705436_backup     31/
lock.lua                  1/0
mango.lua                 24/
pluto.lua                 1/0
yield.lua                 24/
```

```
*********************************************************************
Room Number: 101
Customer Number: 1001
Customer Name: Matthew
Current date and time is 2023-10-31 00:23:02.743623
```

Backup recent file of allocated room details.



```
************************************************************************************
'lhms_backup.txt' file already exist
Existing file will be deleted
Your file 'lhms.txt' is saved as 'lhms_backup.txt'
Original file will be deleted
************************************************************************************
                  LANGHAM HOTEL MANAGEMENT SYSTEM
************************************************************************************
```



```
gfd.lua                    22/05/2023 5:47 pm    LUA File    1 KB
grd.lua
lhms_764705436
lhms_764705436_backup
lock.lua
mango.lua
pluto.lua
yield.lua
```

```
Room Number: 101
Customer Number: 1001
Customer Name: Matthew Tima
Current date and time is 2023-10-31 00:01:39.040433
```

Self-reflection:

It has been a good learning experience for me to identify and fix coding issues. handling all kinds of error types, particularly value errors, which was fixed by adding numerous try and except blocks to the code. Creating a pseudocode from the main code was another challenge, specially It can be challenging to find a balance between giving just enough information to convey the idea of the algorithm and simplifying it enough, so it is easily understood. I will definitely benefit from having the ability to reflect on and develop insights from these experiences in future coding assessments.