



HOTEL MANAGEMENT SYSTEM

Overview

Hello,

This is a project, named “Hotel Management System”. This project is created by using :

- Python Programming Language, and
- SQL(Structured Query Language)

In this project there are 1000+ lines of python code and 70+ lines of SQL Script, this program is created for managing the Hotel tasks like:-

- Adding customers
- Adding Rooms
- Modifying Details of Rooms
- Modifying details of Customer
- Room Booking
- Bill Generation
- Searching rooms
- Searching Customers
- Searching Bookings
- Searching Bills
- Getting report of Rooms
- Getting report of Bookings
- Getting report of collections, etc



Requirements

Software Requirements :

- Python IDLE or any other software
- MySQL Command line or MySQL Workbench or any other software

Libraries Required :

- MySQL Connector (to be installed if your PC doesn't have)
- Datetime

Skills Required :

- You should have knowledge of Python Programming.
- You should have knowledge of SQL.
- You should have knowledge of : How to connect a SQL database (or script) with your SQL server.



Installing MySQL Python Connector

Prerequisites for installing MySQL Connector Python

Before you install MySQL Connector Python, you must ensure that:

- i. Python is installed on your computer and you are connected to the Internet.
- ii. Python is on system's PATH. You often check the boxes to include Python in system's PATH variable. If it is not, then on Windows, you can see the properties of Computer icon on desktop (right click and then click *Properties*).

Then from *Advance system settings* → *Advanced* → *Environment variables*. Edit the PATH variable and without deleting anything add; `<pythonpath>`; to it. e.g., if python's path on your computer is “`C:\Python37-32\python.exe`” then add; `C:\Python37-32\python.exe`; to the PATH variable.

- iii. You need root or administrator privileges to perform the installation process.

Installing MySQL connector, using pip

Open your CMD in Windows, and type the following command-

`pip install mysql-connector-python`



About the Python Source Code

In the Python Source code, there are 1000+ lines of code in which I used the following for the working of the program:-

1. Libraries:

- MySQL Connector
- Datetime

2. Functions or Methods:

- format()
- cursor()
- execute()
- fetchone()
- fetchmany()
- fetchall()
- date.today()
- strftime()
- str()
- int()
- \n
- \t
- commit()
- close()
- range()

3. Statements:

- If, elif & else statements
- Break statement
- Pass

4. Loops : for & while loops

5. User-defined function (def & lambda)

Follow & Connect with me on :





6. Comments
7. Lists, Tuples, Dictionary

List of user-defined functions are as follows-

- a. hotelDetails()
- b. change_hotelDetails()
- c. room_exist(room_no)
- d. customer_exist(cust_no)
- e. add_room()
- f. modify_room()
- g. add_customer()
- h. modify_customer()
- i. room_booking()
- j. bill_generation()
- k. search_rooms()
- l. search_customers()
- m. search_booking()
- n. search_bills()
- o. search_menu()
- p. report_room_status()
- q. report_booking_status()
- r. report_menu()
- s. reportCollection()
- t. main_menu()
- u. clear()
- v. greet()

and so on.....



About SQL Script

In the SQL Script, there are 70+ lines of script, in which I used to create only 5 tables, in which “Hotel_Details” only have its data and the remaining will get there data while running or using the program.

The tables which are in our script (or database) are-

- rooms
- customers
- bookings
- hotel_details
- bills

these are only 5 tables which I used to create.

In this script I used the following-

a) Data Types:

- Int
- Float
- Date
- Varchar
- Bigint

b) Constraints:

- Unique
- Primary key
- Foreign key (or references)

c) Data Manipulation Language Commands:

- Insert into
- Update

d) Data Definition Commands:

- Alter table
- Drop



e) Functions: sum & auto_increment

How to Run

Import the SQL Script first to your server (or database), as the python program runs according to the SQL Script.

I am sharing with you two methods for importing database.

Change the password from every python file with your root password.

Method 1: (Using a software : MySQL Command Line)

- To import the SQL Script by using any software, in my case it is MySQL Command Line, and use the following steps:
 - 1) Create a database named “The_Taj_Hotel”, by using the command :

Create database The_Taj_Hotel;

- 2) Now, use this created database, by using the command:

Use The_taj_hotel;

- 3) Now, copy the path of “The Taj Hotel.sql”, and paste this path in the following command:

Source [paste the copied path] ;

- Now, you can use the python program, named “Hotel_Management_System by Sagar yadav.py”.

A picture of doing this task on ‘MySQL Command Line Client’ is given below or on the next page.

Method 2: Using MySQL Workbench

To do the same task with MySQL Workbench follow the steps of file “Importing Database by MySQL Workbench.zip”.



```
MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.28 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> DROP DATABASE IF EXISTS the_taj_hotel;
Query OK, 0 rows affected (0.11 sec)

mysql> CREATE DATABASE the_TAJ_Hotel;
Query OK, 1 row affected (0.56 sec)

mysql> USE the_TAJ_Hotel;
Database changed
mysql> SOURCE C:\Users\Sagar Yadav\Desktop\Hotel Management System Project\The_Taj_Hotel.sql
Query OK, 0 rows affected (0.07 sec)

Query OK, 1 row affected (0.09 sec)

Database changed
Query OK, 0 rows affected (1.00 sec)

Query OK, 0 rows affected (0.56 sec)

Query OK, 0 rows affected (0.32 sec)

Query OK, 0 rows affected (0.39 sec)

Query OK, 6 rows affected (0.27 sec)
Records: 6  Duplicates: 0  Warnings: 0

Query OK, 0 rows affected (0.42 sec)

Query OK, 0 rows affected (0.00 sec)

mysql>
```

Fig 1. Importing database by using ‘MySQL Command Line Client’

Method 3: (Using Python file)

In this method we are using another Python file for doing the same task. We are using the file “Creating Tables for running HMS file.py” Just run it once and your database is created....

This another python file is available to you, open “Creating Tables for running HMS file.py” file and run it once.

This files source code is also available in this document so don’t worry if you didn’t get the file and you can get them from my github page.

NOTE: If you are running it again and again your stored data will lost.



How to Run at glance

Make sure that you have installed "mysql.connector" library or module.
Also make sure you have imported the sql script, named "The_Taj_Hotel.sql", or
Run the "Creating Table for running HMS file.py" file first
then use this file "Hotel_Management_System by Sagar yadav.py"
Change the password from every 'conn' variable and enter your own password.
Change the user if you have imported 'The_Taj_Hotel.sql' file in another user, other than root user.
Change the host if you have another host.

And other instructions are given in files itself.

Thanks



SQL Script(The Taj Hotel.sql)

```
# The_Taj_Hotel SQL Script
# Created by: SAGAR YADAV
/*
Execute this script in your SQL database server
first.
*/
# Creating Database
drop database if exists The_Taj_Hotel;
CREATE database The_Taj_Hotel;
-- CREATED BY: SAGAR YADAV
# Using the database "The_Taj_Hotel"
use The_Taj_Hotel;

# Creating Tables with thier Structures - Table
Rooms
create table Rooms (
    -- RoomID int primary key AUTO_INCREMENT,
    RoomNo int PRIMARY key,
    Room_Type varchar(25) default 'AC',
    Rent float not null default '2500.00',
    Bed_Type varchar(15),
    Room_Status varchar(15),
    CustomerID int default null)
ENGINE=InnoDB AUTO_INCREMENT = 101;
-- CREATED BY: SAGAR YADAV
# Creating Tables with thier Structures - Table
Customers
create table Customers (
    CustomerID int primary key AUTO_INCREMENT,
    CustomerName varchar(50) Not Null,
    Address varchar(70) Not null,
    Contact_No bigint default '1234567890',
```



```
Email varchar(50) default 'Not Available',
ID_Proof varchar(50) not null default 'Aadhar
Card',
Id_Number varchar(20) not null unique,
Males int default '1',
Females int default '0',
Children int default '0')
AUTO_INCREMENT = 101 ENGINE=InnoDB;
```

-- CREATED BY: SAGAR

YADAV

```
# Creating Tables with thier Structures - Table:
Bookings
CREATE TABLE Bookings (
    Book_id int primary key AUTO_INCREMENT,
    RoomNo int references Rooms(RoomNo),
    DateOfBooking date Not null,
    DateOfLeaving date default null,
    Advance int DEFAULT '00',
    CustomerID varchar(50) Not null references
customers.CustomerID,
    CustomerName varchar(50) Not Null)
AUTO_INCREMENT = 101 ENGINE=InnoDB;
```

-- CREATED BY:

SAGAR YADAV

```
# Creating Tables with thier Structures - Table
Hotel_Details
CREATE TABLE Hotel_Details (
    ID int primary key,
    Field_name varchar(30) Not NULL,
    Entry varchar(100) not NULL)
ENGINE=InnoDB;
```

-- CREATED BY: SAGAR

YADAV

```
# Dumping data for table `Hotel_Details`
INSERT INTO Hotel_Details (ID, Field_name, Entry)
VALUES
```



```
(1, 'Hotel_Name', "The Taj Hotel"),
(2, 'Address', 'ABC Sector, Delhi -22'),
(3, 'Phone', '011-8000111,12,13'),
(4, 'Email', 'thetajhotel_delhi@gmail.com'),
(5, 'GST', '42'),
(6, 'ST', '16');

-- CREATED BY: SAGAR YADAV
# Table structure for table Bills
CREATE TABLE Bills (
    Bill_id int NOT NULL AUTO_INCREMENT,
    Book_id int DEFAULT NULL references
Bookings.Book_id,
    RoomNo int DEFAULT Null references rooms.RoomNo,
    Bill_date date DEFAULT NULL,
    GST int DEFAULT 42,
    ST int DEFAULT 26,
    Amount decimal DEFAULT NULL,
    -- CREATED BY: SAGAR YADAV
    PRIMARY KEY (bill_id)
) ENGINE=InnoDB AUTO_INCREMENT = 101;
-- CREATED BY: SAGAR YADAV

# DROP database The_Taj_Hotel;
-- CREATED BY: SAGAR YADAV
commit
```



Python Source Code(Main)

```
# Project Name : Hotel Management System
# Made by : SAGAR YADAV

# Importing Libraries:
import mysql.connector
from datetime import date, datetime

# Global variables:
hotel_name = ''
addr = ''
phone= ''
email = ''
gst=42
st =13

# 1. Function for Getting Hotel Details:
def hotelDetails():
    global hotel_name
    global addr
    global phone
    global email
    global gst
    global st
    conn = mysql.connector.connect(
        host='localhost', database='The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    sql = "select * from Hotel_Details;"
    cursor.execute(sql)
    records = cursor.fetchmany(6)
    for record in records:
        if record[1]=='Hotel_Name':
            hotel_name = record[2]
        if record[1] == 'Address':
            addr = record[2]
        if record[1] == 'Phone':
            phone = record[2]
        if record[1] == 'Email':
```



```
email = record[2]
if record[1] == 'GST':
    gst = record[2]
if record[1] == 'ST':
    st = record[2]
print("\n\n")
print("\tDATE : ", date.today())
print("\t ~~~~~ HOTEL DETAILS ~~~~~ \n")
print("\tHotel Name      : ", hotel_name)
print("\tAddress         : ", addr)
print("\tPhone No.       : ", phone)
print("\tEmail           : ", email)
print("\tCurrent GST     : ", gst)
print("\tCurrent ST      : ", st)
greet()
wait = input("\nEnter any key to continue.....\n")
```

2. Function for Changing Hotel Details:

```
def change_hotelDetails():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    clear()
    print('Change Hotel Details : ')
    print('-'*75)
    print('1.Hotel Name')
    print('2.Hotel Address')
    print('3.Phone Number(s)')
    print('4.Email ID')
    print('5.Current GST Rate')
    print('6.Current Service Rate')
    print('7.Return to main menu')
    choice = int(input('Enter your choice :'))
    if choice == 7:
        main_menu()
    else :
        print("\n\n\tInvalid input")
        wait = input('\n\n\n')
        .....Press any key to start again.....')
```



```
main_menu()
value = input('Enter new value :')
sql ="update hotel_Details \
      set ENTRY = '{}'\ \
      WHERE id = {};" .format(value, choice)
cursor.execute(sql)
conn.commit()
conn.close()
greet()
wait = input('\n\n\n Record Updated \
.....Press Enter to continue.....')

# 3. Function to check whether the room exist
def room_exist(room_no):
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    sql ="select * from rooms \
          where RoomNo = {};" .format(room_no)
    cursor.execute(sql)
    record = cursor.fetchone()
    return record

# 4. Function to check whether the customer exist
def customer_exist(cust_no):
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    sql = "select * from customers \
           where CustomerID = {};" .format(cust_no)
    cursor.execute(sql)
    record = cursor.fetchone()
    return record

# 5. Function to add or create a new room
def add_room():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
```



```
cursor = conn.cursor()
clear()
print('Add New Room - Screen')
print('-'*75)
room_no = input('\n Enter Room No.(Starts from 101) :> ')
if room_exist(room_no) is not None:
    print("\t\tNOTE: Room No",room_no,
          "already exists in our database")
    room_no = input('\n Enter Another Room No. :> ')
print("\nFor Room no.", room_no, "-->")
room_type_choice = int(input('''Available Room Types -
1. AC
2. Non-AC
3. DELUX
4. Super Delux
5. Queen Delight
6. Kings Special
7. Super Rich Special
Select your choice:> '''))
if room_type_choice == 1:
    room_type = 'AC'
if room_type_choice == 2:
    room_type = 'Non-AC'
if room_type_choice == 3:
    room_type = 'Delux'
if room_type_choice == 4:
    room_type = 'Super Delux'
if room_type_choice == 5:
    room_type = 'Queen Delight'
if room_type_choice == 6:
    room_type = 'King Special'
if room_type_choice == 7:
    room_type = 'Super Rich Special'
if room_type_choice not in (1,2,3,4,5,6,7):
    print("Invalid Input.....")
print("\n\t\tBed Selection ----- Menu")
print("\nFor Room no.", room_no, "-->")
room_bed_choice = int(input('''\nAvialable Room Bed
Types are:
1. Single
```



```
2. Double
3. Triple
Select your bed choice :> ''')
if room_bed_choice == 1:
    room_bed = 'Single bed'
if room_bed_choice == 2:
    room_bed = 'Double bed'
if room_bed_choice == 3:
    room_bed = 'Triple bed'
if room_bed_choice not in (1,2,3,4,5,6,7):
    print("Invalid Input.....")
print("\n\t\tRent -----")
Menu")
print('''Available room rents for "Single bed" are:
1. AC : ₹3499/night
2. Non-AC : ₹1499/night
3. Delux : ₹4799/night
4. Super Delux : ₹5499/night
5. Queen Delight : ₹6999/night
6. King Special : ₹8999/night
7. Super Rich Special : ₹10499/night
\t\tNOTE: Rent changes as per bed there is addition of
Rs.1000 per bed(type)''')
wait = input('\n Press Enter to continue....')
if room_type == 'AC':
    room_rent_selection = 1
elif room_type == 'Non-AC':
    room_rent_selection = 2
elif room_type == 'Delux':
    room_rent_selection = 3
elif room_type == 'Super Delux':
    room_rent_selection = 4
elif room_type == 'Queen Delight':
    room_rent_selection = 5
elif room_type == 'King Special':
    room_rent_selection = 6
elif room_type == 'Super Rich Special':
    room_rent_selection = 7
else :
    pass
# 1. AC Room rent
```



```
if room_rent_selection == 1:
    if room_bed == 'Single bed':
        room_rent = 3499
    elif room_bed == 'Double bed':
        room_rent = 3499 + 1000
    elif room_bed == 'Triple bed':
        room_rent = 3499 + 2000
# 2. Non-AC Room rent
if room_rent_selection == 2:
    if room_bed == 'Single bed':
        room_rent = 1499
    elif room_bed == 'Double bed':
        room_rent = 1499 + 1000
    elif room_bed == 'Triple bed':
        room_rent = 1499 + 2000
# 3. Delux Room rent
if room_rent_selection == 3:
    if room_bed == 'Single bed':
        room_rent = 4799
    elif room_bed == 'Double bed':
        room_rent = 4799 + 1000
    elif room_bed == 'Triple bed':
        room_rent = 4799 + 2000
# 4. Super delux Room rent
if room_rent_selection == 4:
    if room_bed == 'Single bed':
        room_rent = 5499
    elif room_bed == 'Double bed':
        room_rent = 5499 + 1000
    elif room_bed == 'Triple bed':
        room_rent = 5499 + 2000
# 5. Queen delight Room rent
if room_rent_selection == 5:
    if room_bed == 'Single bed':
        room_rent = 6999
    elif room_bed == 'Double bed':
        room_rent = 6999 + 1000
    elif room_bed == 'Triple bed':
        room_rent = 6999 + 2000
# 6. King Special Room rent
if room_rent_selection == 6:
```



```
if room_bed == 'Single bed':
    room_rent = 8499
elif room_bed == 'Double bed':
    room_rent = 8499 + 1000
elif room_bed == 'Triple bed':
    room_rent = 8499 + 2000
# 7. Super rich special Room rent
if room_rent_selection == 7:
    if room_bed == 'Single bed':
        room_rent = 10499
    elif room_bed == 'Double bed':
        room_rent = 10499 + 1000
    elif room_bed == 'Triple bed':
        room_rent = 10499 + 2000
if room_rent_selection not in (1,2,3,4,5,6,7):
    print("Invalid Input.....")
    add_room()
print('\nFor ---->')
print("Room Number : ", room_no)
print("Room Bed Type : ", room_bed)
print("Room Type : ", room_type)
print("Your Rent will be (in Rs) : ", room_rent)
room_rent = str(room_rent)
sql = 'insert into rooms(RoomNo, room_type, \
    rent, bed_type, room_status) values \
    ('+room_no+', "'+ room_type.upper()+'", \
    '+room_rent+', "'+room_bed.upper()+'", "Free");'
cursor.execute(sql)
conn.commit()
print("\n Room added successfully....")
conn.close()
greet()
wait = input('\n\n\n Press Enter to continue....')

# 6. Function to modify room's data
def modify_room():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    clear()
```



```
print(' Change Room Information ')
print('-'*75)
print("NOTE: Rooms starts from 101...")
room_no = int(input('Enter Room No. :'))
while True:
    check = room_exist(room_no)
    if check == None:
        print('Room doesn`t exist....')
        print("""Select any one from below -
1. Want to enter another room no.
2. Return to main menu
3. Close application or Exit""")
        choice = int(input("Enter your choice :> "))
        if choice == 1:
            room_no = input('Enter Another Room no to modify
:> ')
        elif choice == 2:
            main_menu()
        elif choice == 3:
            greet()
            break
    else:
        print("Invalid Input!")
        print("Exiting.....")
        greet()
        break
    elif check != None:
        print("\n\tAvailable details of Room No.",room_no)
        print('')
        Room Type      : {}
        Room Bedding   : {}
        Room Rent      : {}
        Room Status    : {}''.format(check[1],
check[3],check[2],check[4]))
        wait = input("Press Enter to continue.....")
        print("\n\t Choose what you want to modify")
        print()
        print('1.Room Type')
        print('2.Room Rent')
        print('3.Room Bed')
        print('4.Room Status (free/occupied)')
```



```
choice = int(input('Enter your choice : '))
field_name = ''
if choice == 1: # Room Type
    print('''Available room types are:
AC
Non-AC
Delux
Super Delux
Queen Delight
King Special
Super Rich Special''')
    field_name = 'Room_Type'
if choice == 2: # Room rent
    print('''Available room rents for single bed
are:
AC          ₹3499/night
Non-AC      ₹1499/night
Delux       ₹4799/night
Super Delux ₹5999/night
Queen Delight ₹6999/night
King Special ₹8999/night
Super Rich Special ₹10499/night''')
    field_name = 'rent'
if choice == 3: # Room's bed type
    print('''Available Bed Types are:
Single Bed
Double Bed
Triple Bed ''')
    field_name = 'bed_type'
if choice == 4: # Room's status
    print('''Available Status are:
Free
Occupied''')
    field_name = 'Room_Status'
value = input('Enter new value(Type the new
value):> ')
if value.isdigit():
    sql = "update rooms set {} = {} \
           where RoomNo = {}".format(field_name, value,
room_no)
else:
```



```
        sql = "update rooms set {} = '{}' \
                where RoomNo = {}".format(field_name,
value.upper(), room_no)
        cursor.execute(sql)
        conn.commit()
        greet()
        wait = input('''\n\n\n Record Updated \n
.....Press Enter to continue.....''')
        break

# 7. Function to add customer
def add_customer():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    clear()
    print('Add New Customer - Screen')
    print('-'*75)
    name = input('\n Enter Customer Name :> ')
    address = input('\n Enter Customer Address:> ')
    phone = input('\n Enter Customer Contact No. :> ')
    email = input('\n Enter Customer Email ID :')
    id_proof = input('\n Enter Customer ID Proof type\
        (e.g,Aadhar/Passport/DL/VoterID) :> ')
    id_proof_no = input('\n Enter Customer ID proof No. :>
')
    males = input('\n Enter Total Males :> ')
    females = input('\n Enter Total Females :> ')
    children = input('\n Enter Total Children :> ')
    noOfRoomMates = males + females + children
    sql1 = 'insert into customers\
        (CustomerName,address,Contact_No,email,id_proof,id_num
ber\
        ,males,females,children) values \
        ("'+name+'","' + address.upper()+'","'+phone+'",\
        "'+email.upper()+'","'+id_proof.upper()+'","'+id
_proof_no.upper()+'",\
        '+males+', '+females+', '+children+');'
    cursor.execute(sql1)
```



```
sql2 = "select * from customers where id_number = '{}'"
;.format(id_proof_no)
cursor.execute(sql2)
data = cursor.fetchone()
print('\n\n\nCustomer Added successfully
.....')
print("Customer Details ---")
print("Customer No. :", data[0])
print("Name      :", data[1])
print("Address   :", data[2])
print("Contact No.  :", data[3])
print("Email     :", data[4])
print("ID Proof   :", data[5])
print("ID Proof No.  :", data[6])
print("Males      :", data[7])
print("Female     :", data[8])
print("Children   :", data[9])
conn.commit()
conn.close()
greet()
wait = input('\n\n\n Press Enter to continue....')

# 8. Function to modify customer details
def modify_customer():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    clear()
    print(' Change Customer Information ')
    print('-'*75)
    cust_no = input('Enter Customer No., whose
information you want to change :>')
    while True:
        check = customer_exist(cust_no)
        if check == None:
            print('Customer doesn`t exist for this Customer
Number, ',cust_no,'....')
            print("""Now, Select any one option from below -
1. Want to enter another Customer No.
2. Return to main menu
3. Exit""")
            choice = int(input('Enter your choice : '))
            if choice == 1:
                modify_customer()
            elif choice == 2:
                return
            else:
                exit()
        else:
            print('Customer Found')
            print('Customer Details ---')
            print("Customer No. :", data[0])
            print("Name      :", data[1])
            print("Address   :", data[2])
            print("Contact No.  :", data[3])
            print("Email     :", data[4])
            print("ID Proof   :", data[5])
            print("ID Proof No.  :", data[6])
            print("Males      :", data[7])
            print("Female     :", data[8])
            print("Children   :", data[9])
            conn.commit()
            conn.close()
            greet()
            wait = input('\n\n\n Press Enter to continue....')
```



```
3. Close application or Exit""")
choice = int(input("Enter your choice :> "))
if choice == 1:
    cust_no = input('Enter Another Customer No. to
modify :> ')
elif choice == 2:
    main_menu()
elif choice == 3:
    break
else:
    print("Invalid Input!")
    print("Exiting.....")
    break
elif check != None:
    print("\n\tAvailable details of Customer
No.",cust_no,'.....')
    print('''
Name : {}
Address : {}
Phone No. : {}
Email : {}
ID Proof : {}
ID Proof No. : {}
No. of males : {}
No. of females : {}
No. of children : {}'''.format(check[1],check[2],check[3],
check[4],check[5],check[6],check[7],check[8],check
[9]))
    wait = input("\nPress Enter to continue.....")
    print("\n\t Choose what you want to modify")
    print()
    print('1.Name')
    print('2.Address')
    print('3.Phone No')
    print('4.Email ID')
    print('5.ID Proof')
    print('6.ID Proof No')
    print('7.Males')
    print('8.Females')
    print('9.Childeren')
```



```
choice = int(input('Enter your choice :'))
field_name = ''
if choice == 1:
    field_name = 'CustomerName'
if choice == 2:
    field_name = 'Address'
if choice == 3:
    field_name = 'Contact_No'
if choice == 4:
    field_name = 'email'
if choice == 5:
    field_name = 'id_proof'
if choice == 6:
    field_name = 'ID_Number'
if choice == 7:
    field_name = 'males'
if choice == 8:
    field_name = 'females'
if choice == 9:
    field_name = 'children'
value = input('Enter new value :')
if value.isdigit():
    sql = sql = 'update customers set {} = {} where
CustomerID ={};'.format(
        field_name, value, cust_no)
else:
    sql = sql = 'update customers set {} = "{}"
where CustomerID ={};'.format(
        field_name, value, cust_no)
cursor.execute(sql)
conn.commit()
greet()
wait = input(
    '\n\n\n Record Updated .....Press
Enter to continue.....')
break

# 9. Function to book room
def room_booking():
    conn = mysql.connector.connect(
        host = 'localhost', database ='the_taj_hotel',
```



```
        user = 'root', password = 'SAGAR YADAV')
cursor = conn.cursor()
clear()
print("Room Booking ----- Menu")
print("*"*20)
room_no = int(input("\nEnter Room No. :> "))
while True:
    query1 = "SELECT * From rooms where roomno =
{};".format(room_no)
    cursor.execute(query1)
    data = cursor.fetchone()
    query2 = "Select book_id, roomno, bill_date from
bills where roomno = {}".format(room_no)
    cursor.execute(query2)
    record = cursor.fetchone()
    if record == None:
        rmno = 0
        pass
    else:
        book_id, rmno, billdate = record[0],
record[1], record[2]
        if data != None and data[4] in ('free', 'Free'):
            print("\nDetails for the Room No.", room_no)
            room_no = data[0]
            roomtype = data[1]
            roomrent = data[2]
            roombed = data[3]
            print("Room No.      :", room_no)
            print("Room Type     :", roomtype)
            print("Room Rent     :", roomrent)
            print("Room Bedding  :", roombed)
            cust_no = int(input("Enter Customer Id or No.
:> "))
            query2 = "SELECT * From customers where
customerID = {}".format(cust_no)
            cursor.execute(query2)
            data2 = cursor.fetchone()
            cName = data2[1]
            if customer_exist(cust_no) != None :
                if rmno == room_no:
```



```
print("""The Date of Occupancy should
be
equal to or greater than the date
""", billdate)
DateOfBooking =
datetime.strptime(input('Enter date of occupancy (yyyy-mm-
dd) :> '), '%Y-%m-%d')
if DateOfBooking.date() >= billdate:
    pass
else :
    print("Please enter a right
date")
DateOfBooking = input('Enter again
date of occupancy (yyyy-mm-dd) :> ')
else:
    DateOfBooking = input('Enter date of
occupancy (yyyy-mm-dd) :> ')
advance = input('Enter advance amount :>
')
sql1 = 'update rooms \
set Room_status = "Occupied",\
CustomerID = {} \
where Roomno =
{};'.format(cust_no,room_no)
sql2 = 'insert into bookings\
(roomno,customerid,DateOfBooking,advan-
ce,customername) \
values ({} , {}, "{}", {}, {},
"{}");'.format(room_no,
                cust_no, DateOfBooking, advance,
cName)
cursor.execute(sql1)
cursor.execute(sql2)
print('\n\n\nRoom no ', room_no, 'a
',roomtype,' Room is booked for---')
print('Customer No. :', cust_no)
print("Customer Name :", cName)
conn.commit()
conn.close()
wait = input('\n Press Enter to
continue....')
```



```
        break
    else :
        print("\tNOTE:Customer ID you entered is
not present in our database.")
        elif data == None:
            print("There is no such room found in our
database, Room No.", room_no)
            room_no = int(input("\nEnter Another Room No.
:> "))
        else :
            print("NOTE: This room (", room_no,) is not
available for booking")
            print("It is Already Booked!.....")
            room_no = int(input("\nEnter Another Room No.
:> "))

# 10. Function to generate bill
def bill_generation():
    global gst
    global st
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    print("BILL GENERATION ----- SCREEN")
    print(35*'')
    room_no = input('Enter Room No. for which you want to
generate bill :> ')
    while True:
        query1 = "SELECT r.room_type, b.customerid,
dateofleaving,\n
                dateofbooking, advance, r.rent, b.book_id,
r.bed_type, r.roomno \
                FROM bookings b, rooms r \
                WHERE r.roomno = b.roomno and r.roomno =
{};".format(room_no)
        cursor.execute(query1)
        data = cursor.fetchall()
        checkRoomExist = room_exist(room_no)
        if checkRoomExist == None:
```



```
        print("\nNOTE: This room doesn't exist in our
database,Room no.", room_no)
        print("\tYou have to create a new room")
        print("Exiting Bill Generation Menu.....")
        wait = input('\nPress Enter to continue....')
        break
    else :
        pass
    if data == []:
        print("NOTE:Enter Another Room No., there is
no booking for this room no.",room_no)
        wait = input('\n Press Enter to continue....')
        choice = int(input('''Select any one option-
1. Want to enter another room no
2. Return to Main Menu
3. Return back
:> '''))
        if choice == 1:
            room_no = input('Enter Another Room No.
for which you want to generate bill :> ')
            elif choice == 2:
                main_menu()
                pass
            elif choice == 3:
                break
            else:
                print("Invalid Input!")
                print("Exiting.....")
                break
        else:
            for rm_t, c_id, dl, db, adv, rent, bookid,
bedtype, rmno in data:
                room_type, cust_id, dolv, roombed, roomno
= rm_t, c_id, dl, bedtype, rmno
                doo, advance, rent, book_id = db, adv,
rent, bookid
                if dolv != None :
                    print("\n\tNOTE: This room has already a
bill...")
                    wait = input('\n Press Enter to
continue....')
```



```
choice = int(input('''Select any one
option-
        1. Want to enter another room id
        2. Return to Main Menu
        3. Close application or Exit
        :> '''))
if choice == 1:
    room_no = input('Enter Another Room
no. for which you want to generate bill :> ')
elif choice == 2:
    main_menu()
    pass
elif choice == 3:
    break
else:
    print("Invalid Input!")
    print("Exiting.....")
    break
elif dolv == None:
    query2 = "SELECT customername, address\
              FROM customers\
              Where customerid =
{};".format(cust_id)
    cursor.execute(query2)
    data = cursor.fetchone()
    cname, caddr = data[0], data[1]
    print('''\tRoom Detail or Booking Details
\t-----''')
    -
    Room No.          : {}
    Room Name         : {}
    Room Bedding      : {}
    Room Rent(Per night) : {}
    Customer ID       : {}
    Customer Name      : {}
    Customer Address    : {}
    Advance paid       :
{}''' .format(roomno, room_type, roombed, rent, cust_id,
cname, caddr, advance))
    wait = input('\nPress Enter to
continue....')
```



```
clear()
print('Bill Generation ')
print('-'*75)
print('Room occupied :',room_no)
dol = date.today()
total_days = (dol-doo).days
amount = total_days*rent
gst_amount = amount*int(gst)/100
st_amount = amount*int(st)/100
payable_amount = total_days * rent -
advance + gst_amount + st_amount
if dol == doo:
    if room_type == "SUPER RICH SPECIAL":
        charge = 1500
    elif room_type == "KING SPECIAL":
        charge = 1200
    elif room_type == "QUEEN DELIGHT":
        charge = 1000
    elif room_type == "SUPER DELUX":
        charge = 800
    elif room_type == "DELUX":
        charge = 550
    elif room_type == "AC":
        charge = 275
    elif room_type == "NON-AC":
        charge = 150
    payable_amount = rent - advance +
gst_amount + st_amount + charge
    print("You have been charged by
Rs.", charge, 'as you are leaving same day..')
else:
    pass
print('Date of Occupancy :',doo, '\nDate
of Leaving :',dol)
    print('Total Payable Days : ', total_days)
    print('Room Rent Per Day : ', rent)
    print('Total Amount :',amount)
    print('Advance :',advance, '\nGST : {}'
.format(gst,gst_amount), '\nService Tax :
{}' .format(st,st_amount))
    print('Amount Payable :',payable_amount)
```



```
sql1 = 'update rooms set room_status
="free" \
        where roomno =' +room_no +';'
sql2 = 'update bookings set DateOfLeaving
=' +str(dol)+ ' \
        where roomno=' +room_no+ ' and customerid
=' +cust_id +';'
sql3 = 'insert into
bills(book_id,amount,bill_date,gst,st,roomno) \
        values(' +str(book_id)+ ',' +str(payable_amou
nt)+ ',' +str(dol)+ ',' +str(gst)+ ',' +str(st)+ ',' +str(roomno
)+');'
cursor.execute(sql1)
cursor.execute(sql2)
cursor.execute(sql3)
conn.commit()
conn.close()
print("Bill Generated successfully....")
print('This room is Free now, Room
No.:',roomno)
wait = input('\n\n Press Enter to
continue....')
break

# 11. Function to search rooms
def search_rooms():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    room_no = input('Enter Room No :')
    check = room_exist(room_no)
    if check is not None:
        sql ='select * from rooms where roomno =' +room_no
        '+';
        cursor.execute(sql)
        record = cursor.fetchone()
        clear()
        print('*'*75)
        print('Room Info ---')
        print('Room No. :',record[0])
```



```
print('Room Type :',record[1])
print('Room Rent :',record[2])
print('Room Bed :',record[3])
print('Room Status :',record[4])
print('Booked for Customer No. :', record[5])
conn.close()
else :
    print("\nThis Room doesn't exist.....")
wait = input('\n\n\nPress Enter to continue.....')

# 12. Function to search customer's details
def search_customer():
    while True:
        conn = mysql.connector.connect(
            host='localhost', database = 'The_Taj_Hotel',
            user='root', password='SAGAR YADAV')
        cursor = conn.cursor()
        clear()
        print('Search Customer DataBase')
        print(' - *75')
        print("\nBy which data you want to search--")
        print('1.by Customer ID')
        print('2.by Customer Name')
        print('3.by Customer Address')
        print('4.by Customer Phone')
        print('5.by Customer Email')
        print('6.by ID Proof')
        print('7.by ID Proof no.')
        print('8.Return to back menu')
        choice = int(input('\n\tEnter your choice by which u
want to search : '))
        field_name = ''
        if choice ==1:
            field_name = 'CustomerID'
        if choice ==2:
            field_name = 'CustomerName'
        if choice ==3:
            field_name = 'address'
        if choice ==4:
            field_name = 'Contact_No'
        if choice ==5:
```



```
    field_name = 'email'
if choice ==6:
    field_name = 'id_proof'
if choice ==7:
    field_name = 'id_number'
if choice ==8:
    break
value = input('Enter value that you want to search :')
if field_name =='CustomerID':
    sql = 'select * from customers where '+ field_name
+ ' = '+ value + ';'
else:
    sql = 'select * from customers where ' + field_name
+ ' like "%' + value + '%";'
cursor.execute(sql)
records = cursor.fetchall()
# checking for record
if records == []:
    print('*' * 45)
    print('Search Result for {} = {} : NOT
FOUND'.format(field_name,value))
    wait = input('\n\n\nPress Enter to continue.....')
    break
else :
    pass
clear()
print('Search Result for {} =
{}'.format(field_name,value))
print('*'*80)
for record in records:
    print('''
Customer ID      : {}
Customer Name   : {}
Address          : {}
Contact No.     : {}
Email            : {}
ID Proof         : {}
ID Proof No.    : {}'''.format(
        record[0], record[1], record[2], record[3],
record[4], record[5], record[6]))
conn.close()
```



```
greet()
wait = input('\n\n\nPress Enter to continue.....')
break

# 13. Function to search for bookings
def search_booking():
    while True:
        conn = mysql.connector.connect(
            host='localhost', database = 'The_Taj_Hotel',
            user='root', password='SAGAR YADAV')
        cursor = conn.cursor()
        cust_no = input('Enter Customer No :')
        sql = 'select
book_id,r.roomno,c.Customername,DateOfBooking,advance from
bookings b, customers c, rooms r where b.roomno = r.roomno
AND b.customerid = '+cust_no+' and DateOfLeaving is NULL;'
        cursor.execute(sql)
        records = cursor.fetchall()
        clear()
        print('Booking information for customer ID
:{}'.format(cust_no))
        print('*'*50)
        if records == []:
            print("NOTE: There is no booking for this Customer
no.",cust_no)
            wait = input('\n Press Enter to continue....')
            choice = int(input('''Select any one option-
1. Want to enter another room no
2. Return to Main Menu
3. Return to Search menu
:> '''))
            if choice == 1:
                cust_no = input('Enter Another Customer No.:> ')
            elif choice == 2:
                main_menu()
                pass
            elif choice == 3:
                break
            else:
                print("Invalid Input!")
                print("Returning to Search menu.....")
```



```
        break
else:
    for record in records:
        print(''\tCustomer ID :\t{}'
        Room Number :\t{}'
        Customer` s Name :\t{}'
        Date Of Booking :\t{}'
        Advance Payment :\t{}''.format(
            record[0], record[1], record[2], record[3],
record[4]))
    conn.close()
    greet()
    wait = input('\n\n\nPress Enter to continue.....')
break

# 14. Function to search bills
def search_bills():
    while True:
        conn = mysql.connector.connect(
            host='localhost', database = 'The_Taj_Hotel',
            user='root', password='SAGAR YADAV')
        cursor = conn.cursor()
        bill_no = input('Enter Bill No :')
        sql = ' select bills.bill_id, bills.amount, bill_date,
gst, st, b.book_id, DateOfBooking,\n
            DateOfleaving, advance, b.CustomerName, address,
Contact_No, email, bills.roomno, r.room_type,\n
            r.bed_type, r.rent, c.customerid, id_proof,
id_number \
            from bills, bookings b, customers c , rooms r \
            where bills.book_id = b.book_id \
            and b.roomno = r.roomno and b.customerid =
c.customerid AND NOT DateOfLeaving is NULL AND \
            bill_id = '+bill_no +';'
        cursor.execute(sql)
        record = cursor.fetchone()
        clear()
        print('Bill information for Bill No
:{}''.format(bill_no))
        print('*'*140)
        # checking record
```



```
if record == None:  
    print("NOTE :There is no record for Bill No. ",  
bill_no, "in our database.")  
    wait = input('\n\n\nPress Enter to continue.....')  
    greet()  
    break  
else :  
    pass  
print('Bill Number : ', record[0])  
print('Bill\'s Total Amount : ', 'Rs.',record[1])  
print('Bill Date : ', record[2])  
print('GST Charged : ', 'Rs.',record[3])  
print('Service Tax Charged : ', 'Rs.',record[4])  
print('Booking ID : ', record[5])  
print('Room no. Used : ', record[13])  
print('Room Name or type : ', record[14])  
print('Room Bedding : ', record[15])  
print('Room Rent(per night) : ', 'Rs.',record[16])  
print('Date of Occupancy : ', record[6])  
print('Date of Leaving : ', record[7])  
print('Advance Paid : ', 'Rs.',record[8])  
print('Customer\'s ID : ', record[17])  
print('Customer\'s Name : ', record[9])  
print('Customer\'s Address : ', record[10])  
print('Customer\'s Phone : ', record[11])  
print('Customer\'s Email ID : ', record[12])  
print('Customer\'s ID Proof : ', record[18])  
print('Customer\'s ID Proof No. : ', record[19])  
conn.close()  
wait = input('\n\n\nPress Enter to continue.....')  
break  
  
# 15. Function to select what you want to search  
def search_menu():  
    while True:  
        clear()  
        print(' Search Menu')  
        print('-'*75)  
        print("\n1.Room Status")  
        print('\n2.Booking Status')  
        print('\n3.Customer Details')
```



```
print('\n4.Bills')
print('\n5.Back to Main Menu')
print('\n\n')
choice = int(input('Enter your choice ...: '))
if choice == 1:
    search_rooms()
if choice == 2:
    search_booking()
if choice == 3:
    search_customer()
if choice == 4:
    search_bills()
if choice == 5:
    greet()
    break

# 16. Function to take report of room
def report_room_status():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    sql = 'select * from rooms'
    cursor.execute(sql)
    records = cursor.fetchall()
    clear()
    print('Rooms Status - Report')
    print('-'*75)
    print('{:15s} {:18s} {:15s} {:15s} {:20s}
{:10s}'.format('Room No', 'Room Type', 'Rent','Bedding',
'Room Status', 'Booked for CustomerID'))
    for no,rtype,rent,bed,status,customerid in records:
        print('{:<10d} {:21s} {:<7.2f} {:>18s} {:>12s}
{:>20s}'.format(no, rtype, rent, bed, status,
str(customerid)))
    conn.close()
    greet()
    wait = input('\n\n\n Press Enter to continue....')

# 17. Function to take report of booking status
def report_booking_status():
```



```
conn = mysql.connector.connect(  
    host='localhost', database = 'The_Taj_Hotel',  
    user='root', password='SAGAR YADAV')  
cursor = conn.cursor()  
sql = ''' SELECT  
b.book_id,b.roomno,DateOfBooking,DateOfLeaving,advance,  
b.Customername,address,Contact_No  
    FROM bookings b, customers c  
WHERE b.customerid = c.customerid;  
'''  
  
cursor.execute(sql)  
records = cursor.fetchall()  
clear()  
print(' Booking Status - Report')  
print('*'*75)  
print('{:9s} {:15s} {:18s} {:10s} {:15s} {:20s} {:15s}  
{:15s}'.format(  
    'BookID', 'RoomNo', 'DOO', 'DOL', 'Advance',  
'CustomerName', 'Address', 'Phone'))  
for BookID, Rno, doo, dol, advance, Cname, addr, phone  
in records:  
    print('{:} {:>10d} {:>20s} {:>12s} {:>14.2f}  
{:>20s} {:>14s} {:>18d}'.format(  
        BookID, Rno, str(doo), str(dol), advance,  
Cname, addr, phone))  
conn.close()  
greet()  
wait = input('\n\n\n Press Enter to continue....')  
  
# 18. Function for report menu of report functions  
def report_menu():  
    while True:  
        clear()  
        print('Report Menu')  
        print('*'*35)  
        print("\n1.Room Status")  
        print("\n2.Booking Status")  
        print("\n3.Bill Collections")  
        print("\n4.Back to Main Menu")  
        print("\n\n")  
        choice = int(input('Enter your choice ...: '))
```



```
if choice == 1:
    report_room_status()
if choice == 2:
    report_booking_status()
if choice == 3:
    reportCollection()
if choice == 4:
    break

# 19. Function for Collections
def reportCollection():
    conn = mysql.connector.connect(
        host='localhost', database = 'The_Taj_Hotel',
        user='root', password='SAGAR YADAV')
    cursor = conn.cursor()
    clear()
    print("Bill Collections ----- Menu")
    print('*'*45)
    print("1.Total Bill Collection")
    print("2.Collection from individual customer")
    print("3.Back to Report menu")
    choice = int(input("\nEnter your choice from above (1
or 2) :> "))
    # FOR TOTAL COLLECTION
    if choice == 1:
        print("\nTotal Collection")
        print('-'*35)
        sql = "SELECT sum(Amount) from bills;"
        cursor.execute(sql)
        data = cursor.fetchone()
        print("Total Collections we have, Rs.", data[0])
        wait = input('\n\n\n Press Enter to continue....')
    # FOR INDIVIDUAL CUSTOMER
    if choice == 2:
        print("\nCollection from individual customer")
        print('-'*35)
        custNo = input("Enter Customer No. :> ")
        sql1 = "SELECT DateofLeaving, advance \
                FROM bookings WHERE customerID =
                {};" .format(custNo)
        cursor.execute(sql1)
```



```
records = cursor.fetchall()
count = cursor.rowcount
print("Rows :", count)
for record in records:
    dateofleaving = record[0]
    advance = record[1]
    if count == 1 and dateofleaving == None:
        collection = advance
        print("Collection from Customer No.", custNo,
":> Rs. ",collection)
    elif count == 1 and dateofleaving != None :
        sql2 = "SELECT amount\
                From bills bl, bookings bk\
                WHERE bl.book_id = bk.book_id and
customerid = {}".format(custNo)
        cursor.execute(sql2)
        data = cursor.fetchone()
        print("Collections from CustomerNo.", custNo,
"we have, Rs.", data[0])
    elif count > 1 and dateofleaving == None:
        sql3 = "SELECT bk.book_id, amount\
                From bookings bk, bills bl\
                Where bk.book_id = bl.book_id and
customerid = {}".format(custNo)
        cursor.execute(sql3)
        records = cursor.fetchall()
        for record in records:
            bill = record[1]
            collection = bill + advance
            print("Collection from Customer No.", custNo,
":> Rs. ",collection)
    elif count > 1 and dateofleaving != None:
        bill = 0
        sql3 = "SELECT bk.book_id, amount\
                From bookings bk, bills bl\
                Where bk.book_id = bl.book_id and
customerid = {}".format(custNo)
        cursor.execute(sql3)
        records = cursor.fetchall()
        for record in records:
```



```
        bill +=  
record[1]  
        collection = bill  
        print("Collection from Customer No.", custNo,  
":> Rs. ",collection)  
        break  
    if count == 0 :  
        print("No record found for Customer No.", custNo)  
    if choice == 3:  
        report_menu()  
    greet()  
    wait = input('\n\n\n Press Enter to continue....')  
  
# 20. Function for Main Screen  
def main_menu():  
    while True:  
        clear()  
        print(' H O T E L   M A N A G E M E N T   S Y S T E  
M ')  
        print('-'*75)  
        print("\n1. Add New Room")  
        print('\n2. Add Customer')  
        print('\n3. Modify Room Information')  
        print('\n4. Modify Customer Information')  
        print('\n5. Room Booking')  
        print('\n6. Bill Generation')  
        print('\n7. Search Database')  
        print('\n8. Report Menu')  
        print('\n9. Hotel Details')  
        print('\n10. Change Hotel Details')  
        print('\n11. Close application')  
        print('\n\n')  
        choice = int(input('Enter your choice ...: '))  
        if choice == 1: # Add New Room  
            add_room()  
        if choice == 2: # Add Customer  
            add_customer()  
        if choice == 3: # Modify Room Information  
            modify_room()  
        if choice == 4: # Modify Customer Information  
            modify_customer()
```



```
if choice == 5:    # Room Booking
    room_booking()
if choice == 6:    # Bill Generation
    bill_generation()
if choice == 7:    # Search Database
    search_menu()
if choice == 8:    # Report Menu
    report_menu()
if choice == 9:    # Hotel Details
    hotelDetails()
if choice ==10:   # Change Hotel Details
    change_hotelDetails()
if choice ==11:
    greet()
    break

# 21. Function to clear the screen
def clear():
    for _ in range(30):
        print()

# 22. Function to greet
greet = lambda : print("\n\t\tThanks for
using....\n\t\tCreated by : SAGAR YADAV")

#
#     __main__
if __name__ == "__main__":
    hotelDetails()
    main_menu()

# END OF PROGRAM
# THANKS FOR USING
# CREATED BY : SAGAR YADAV
```



Python program for : Creating database or tables)

```
# Project Name          : Hotel Management System
# Created by           : SAGAR YADAV

'''                      READ ME FIRST
Run this file if you are not able to run "The_Taj_Hotel by
Sagar yadav.sql" file.
Then use the another python file named
"Hotel_Management_System" by Sagar yadav.py"
Change the password from 'cnx' variable and enter your own
password
Change the user if you have imported 'The_Taj_Hotel.sql'
file in another user, other than root user.'''


from __future__ import print_function
import mysql.connector
from mysql.connector import errorcode

DB_NAME = 'The_Taj_Hotel'
TABLES = {}

# Creating Tables with thier Structures - Table Rooms
TABLES['Rooms'] = (
    '''create table Rooms (
        RoomNo int PRIMARY key,
        Room_Type varchar(25) default 'AC',
        -- CREATED BY: SAGAR YADAV
        Rent float not null default '2500.00',
        Bed_Type varchar(15),
        Room_Status varchar(15),
        CustomerID int default null) ENGINE=InnoDB''')


# Creating Tables with thier Structures - Table Customers
TABLES['Customers'] = (
    '''create table Customers (
        CustomerID int primary key AUTO_INCREMENT,
```



```
CustomerName varchar(50) Not Null,  
Address varchar(70) Not null,  
Contact_No bigint default '1234567890',  
-- CREATED BY: SAGAR YADAV  
Email varchar(50) default 'Not Available',  
ID_Proof varchar(50) not null default 'Aadhar  
Card',  
Id_Number varchar(20) not null unique,  
Males int default '1',  
Females int default '0',  
Children int default '0')  
AUTO_INCREMENT = 101 ENGINE=InnoDB''')  
# Created by Sagar Yadav  
# Creating Tables with thier Structures - Table: Bookings  
TABLES['Bookings'] = (  
'''CREATE TABLE Bookings (  
    Book_id int primary key AUTO_INCREMENT,  
    RoomNo int references Rooms(RoomNo),  
    DateOfBooking date Not null,  
    -- CREATED BY: SAGAR YADAV  
    DateOfLeaving date default null,  
    Advance int DEFAULT '00',  
    CustomerID varchar(50) Not null references  
customers.CustomerID,  
    CustomerName varchar(50) Not Null)  
ENGINE=InnoDB AUTO_INCREMENT = 101 ''')  
  
# Creating Tables with thier Structures - Table  
Hotel_Details  
TABLES['Hotel_Details'] = (  
'''CREATE TABLE Hotel_Details (  
    ID int primary key,  
    -- CREATED BY: SAGAR YADAV  
    Field_name varchar(30) Not NULL,  
    Entry varchar(100) not NULL) ENGINE=InnoDB''')  
  
# Table structure for table Bills  
TABLES['Bills'] = (  
'''CREATE TABLE Bills (  
    Bill_id int NOT NULL AUTO_INCREMENT,  
    Book_id int DEFAULT NULL references Bookings.Book_id,
```



```
RoomNo int DEFAULT Null references rooms.RoomNo,
Bill_date date DEFAULT NULL,
GST int DEFAULT 42,
ST int DEFAULT 26,
Amount decimal DEFAULT NULL,
-- CREATED BY: SAGAR YADAV
PRIMARY KEY (bill_id)
AUTO_INCREMENT = 101 ENGINE=InnoDB''')

cnx = mysql.connector.connect(user='root',password =
'SAGAR YADAV')
cursor = cnx.cursor()
# Created by Sagar Yadav
def create_database(cursor):
    try:
        cursor.execute("CREATE DATABASE {} DEFAULT
CHARACTER SET 'utf8'".format(DB_NAME))
    except mysql.connector.Error as err:
        print("Failed creating database: {}".format(err))
        exit(1)
try:
    cursor.execute("DROP DATABASE IF Exists
The_Taj_Hotel;")
    cursor.execute("USE {}".format(DB_NAME))
except mysql.connector.Error as err:
    print("Database {} does not exists.".format(DB_NAME))
    if err(errno == errorcode.ER_BAD_DB_ERROR):
        create_database(cursor)
        print("Database {} created
successfully.".format(DB_NAME))
        cnx.database = DB_NAME
    else:
        print(err)
        exit(1)
# Created by Sagar Yadav

for table_name in TABLES:
    table_description = TABLES[table_name]
    try:
        print("Creating table {}: ".format(table_name),
end='')
```



```
        cursor.execute(table_description)
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_TABLE_EXISTS_ERROR:
            print("already exists.")
        else:
            print(err.msg)
    else:
        print("OK")

# Dumping data for table `Hotel_Details`
Hotel_Details = ('''INSERT INTO Hotel_Details (ID,
Field_name, Entry) VALUES
(1, 'Hotel_Name', "The Taj Hotel"),
(2, 'Address', 'ABC Sector, Delhi -22'),
(3, 'Phone', '011-8000111,12,13'),
(4, 'Email', 'thetajhotel_delhi@gmail.com'),
(5, 'GST', '42'),
(6, 'ST', '16')'''')

cursor.execute(Hotel_Details)
print("Data dumped successfully in Table: Hotel_details")
cnx.commit()
cursor.close()
cnx.close()
```



Data structure of tables (after importing database)

STRUCTURE OF TABLE: HOTEL_DETAILS

```
mysql> DESC HOTEL_DETAILS;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ID    | int   | NO   | PRI  | NULL    |        |
| Field_name | varchar(30) | NO   |      | NULL    |        |
| Entry   | varchar(100) | NO   |      | NULL    |        |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

STRUCTURE OF TABLE: ROOMS

```
mysql> USE THE_TAJ_HOTEL;
Database changed
mysql> DESC ROOMS;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| RoomID | int   | NO   | PRI  | NULL    | auto_increment |
| RoomNo | int   | NO   | UNI  | NULL    |        |
| Room_Type | varchar(25) | YES  |      | AC      |        |
| Rent   | float  | NO   |      | 2500    |        |
| Bed_Type | varchar(15) | YES  |      | NULL    |        |
| Room_Status | varchar(15) | YES  |      | NULL    |        |
| CustomerID | int   | YES  |      | NULL    |        |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.56 sec)
```

STRUCTURE OF TABLE: OF CUSTOMERS

```
mysql> DESC CUSTOMERS;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CustomerID | int   | NO   | PRI  | NULL    | auto_increment |
| CustomerName | varchar(50) | NO   |      | NULL    |        |
| Address | varchar(70) | NO   |      | NULL    |        |
| Contact_No | bigint | YES  |      | 1234567890 |        |
| Email   | varchar(50) | YES  |      | Not Available |        |
| ID_Proof | varchar(50) | NO   |      | Aadhar Card |        |
| Id_Number | varchar(20) | NO   | UNI  | NULL    |        |
| Males   | int   | YES  |      | 1       |        |
| Females | int   | YES  |      | 0       |        |
| Children | int   | YES  |      | 0       |        |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.04 sec)
```



Follow & Connect with me on :





STRUCTURE OF TABLE: BOOKINGS

```
mysql> DESC BOOKINGS;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Book_id | int | NO | PRI | NULL | auto_increment |
| RoomID | int | YES | | NULL | |
| DateOfBooking | date | NO | | NULL | |
| DateOfLeaving | date | YES | | NULL | |
| Advance | int | YES | | 0 | |
| CustomerID | varchar(50) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)
```

STRUCTURE OF TABLE: BILLS

```
mysql> DESC BILLS;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Bill_id | int | NO | PRI | NULL | auto_increment |
| Book_id | int | YES | | NULL | |
| Amount | decimal(10,0) | YES | | NULL | |
| Bill_date | date | YES | | NULL | |
| GST | int | YES | | 42 | |
| ST | int | YES | | 16 | |
| RoomNo | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.03 sec)
```

DATA OF TABLE HOTEL DETAILS

```
mysql> SELECT * FROM HOTEL_DETAILS;
+-----+-----+
| ID | Field_name | Entry |
+-----+-----+
| 1 | Hotel_Name | The Taj Hotel |
| 2 | Address | ABC Sector, Delhi -22 |
| 3 | Phone | 011-8000111,12,13 |
| 4 | Email | thetajpalace_delhi@gmail.com |
| 5 | GST | 42 |
| 6 | ST | 16 |
+-----+-----+
6 rows in set (0.20 sec)
```



DATA OF ALL REMAINING TABLES

```
mysql> SELECT * FROM ROOMS;  
Empty set (0.00 sec)  
  
mysql> SELECT * FROM CUSTOMERS;  
Empty set (0.01 sec)  
  
mysql> SELECT * FROM BOOKINGS;  
Empty set (0.01 sec)  
  
mysql> SELECT * FROM BILLS;  
Empty set (0.02 sec)
```



Pictures of Complete Program

Pictures of Hotel Management System by Sagar Yadav.py

The image shows two side-by-side screenshots of a code editor, likely PyCharm, displaying a Python script named `Hotel_Management_System.py`. The script is used to manage hotel details and rooms.

Top Screenshot:

```
1  # Project Name      : Hotel Management System
2  # Created by        : SAGAR YADAV
3
4  """ ~~~~~ HOW TO RUN ~~~~~
5  Make sure that you have installed "mysql.connector" library or module.
6  Also make sure you have imported the sql script, named "The_Taj_Hotel.sql", or
7  Run the "Creating Table for running HMS file.py" file first
8  then use this file "Hotel_Management_System by Sagar yadav.py"
9  Change the password from every 'conn' variable and enter your own password.
10 Change the user if you have imported 'The_Taj_Hotel.sql' file in another user, other than root user.
11 Change the host if you have another host.
12 Run it on Desktop or Laptop for best results.
13 For further guidance check out the document.
14 """
15
16
17 # Importing Libraries:
18 import mysql.connector
19 from datetime import date, datetime
20
21 # Global variables:
22 hotel_name = ''
23 addr = ''
24 phone = ''
25 email = ''
26 gst = 42
27 st = 13
28
29 # 1. Function for Getting Hotel Details:
30 def hotelDetails():
```

Bottom Screenshot:

```
2/   st = 13
28
29 # 1. Function for Getting Hotel Details:
30 def hotelDetails():
31     global hotel_name
32     global addr
33     global phone
34     global email
35     global gst
36     global st
37     conn = mysql.connector.connect(
38         host='localhost', database='The_Taj_Hotel',
39         user='root', password='SAGAR YADAV')
40     cursor = conn.cursor()
41     sql = "select * from Hotel_Details;"
42     cursor.execute(sql)
43     records = cursor.fetchmany(6)
44     for record in records:
45         if record[1] == 'Hotel_Name':
46             hotel_name = record[2]
47         if record[1] == 'Address':
48             addr = record[2]
49         if record[1] == 'Phone':
50             phone = record[2]
51         if record[1] == 'Email':
52             email = record[2]
53         if record[1] == 'GST':
54             gst = record[2]
55         if record[1] == 'ST':
56             st = record[2]
```



```
File: Hotel_Management_System by Sagar yadav.py X
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
56     st = record[2]
57     print("\n\n")
58     print("\tDATE : ", date.today())
59     print("\t ~~~~~ HOTEL DETAILS ~~~~~ \n")
60     print("\tHotel Name : ", hotel_name)
61     print("\tAddress : ", addr)
62     print("\tPhone No. : ", phone)
63     print("\tEmail : ", email)
64     print("\tCurrent GST : ", gst)
65     print("\tCurrent ST : ", st)
66     greet()
67     wait = input("\nEnter any key to continue..... >")
68
# 2. Function for Changing Hotel Details:
69 def change_hotelDetails():
70     conn = mysql.connector.connect(
71         host='localhost', database = 'The_Taj_Hotel',
72         user='root', password='SAGAR YADAV')
73     cursor = conn.cursor()
74     clear()
75     print('Change Hotel Details : ')
76     print('*' *75)
77     print('1.Hotel Name')
78     print('2.Hotel Address')
79     print('3.Phone Number(s)')
80     print('4.Email ID')
81     print('5.Current GST Rate')
82     print('6.Current Service Rate')
83     print('7.Return to main menu')
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111

Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit  ⌂  ⌂
```



```
File: Hotel_Management_System by Sagar yadav.py X
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111

choice = int(input('Enter your choice :'))

if choice == 7:
    | main_menu()
else :
    print("\n\n\tInvalid input"
    wait = input('\n\n\n'
    | .....Press any key to start again.....')
    main_menu()

value = input('Enter new value :')

sql ="update hotel_Details \
    set ENTRY = '{}'\ \
    WHERE id = {} ;".format(value, choice)

cursor.execute(sql)
conn.commit()
conn.close()
greet()
wait = input('\n\n\n Record Updated \
    .....Press Enter to continue.....')

# 3. Function to check whether the room exist
def room_exist(room_no):
    conn = mysql.connector.connect(
Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit  ⌂  ⌂
```



```
# Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
109 # 3. Function to check whether the room exist
110 def room_exist(room_no):
111     conn = mysql.connector.connect(
112         host='localhost', database = 'The_Taj_Hotel',
113         user='root', password='SAGAR YADAV')
114     cursor = conn.cursor()
115     sql = "select * from rooms \
116           where RoomNo = {}".format(room_no)
117     cursor.execute(sql)
118     record = cursor.fetchone()
119     return record
120
121 # 4. Function to check whether the customer exist
122 def customer_exist(cust_no):
123     conn = mysql.connector.connect(
124         host='localhost', database = 'The_Taj_Hotel',
125         user='root', password='SAGAR YADAV')
126     cursor = conn.cursor()
127     sql = "select * from customers \
128           where CustomerID = {}".format(cust_no)
129     cursor.execute(sql)
130     record = cursor.fetchone()
131     return record
132
133 # 5. Function to add or create a new room
134 def add_room():
135     conn = mysql.connector.connect(
136         host='localhost', database = 'The_Taj_Hotel',
137         user='root', password='SAGAR YADAV')
138     cursor = conn.cursor()
139
140     print('Add New Room - Screen')
141     print('*'*75)
142     room_no = input('\nEnter Room No.(Starts from 101) :> ')
143
144     if room_exist(room_no) is not None:
145         print("\t\tNOTE: Room No{},room_no,\n\t\talready exists in our database")
146         room_no = input('\nEnter Another Room No. :> ')
147
148     print("\nFor Room no.", room_no, "-->")
149     room_type_choice = int(input('Available Room Types -\n\t1. AC\n\t2. Non-AC\n\t3. DELUX\n\t4. Super Delux\n\t5. Queen Delight\n\t6. Kings Special\n\t7. Super Rich Special\n\tSelect your choice:> '))
150
151     if room_type_choice == 1:
152         room_type = 'AC'
153     if room_type_choice == 2:
154         room_type = 'Non-AC'
155     if room_type_choice == 3:
156         room_type = 'Delux'
```



```
Hotel_Management_System by Sagar yadav.py
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
164     room_type = 'Delux'
165     if room_type_choice == 4:
166         room_type = 'Super Delux'
167     if room_type_choice == 5:
168         room_type = 'Queen Delight'
169     if room_type_choice == 6:
170         room_type = 'King Special'
171     if room_type_choice == 7:
172         room_type = 'Super Rich Special'
173     if room_type_choice not in (1,2,3,4,5,6,7):
174         print("Invalid Input.....")
175
176     print("\n\t\tBed Selection ----- Menu")
177     print("For Room no.", room_no, "-->")
178     room_bed_choice = int(input("\nAvailable Room Bed Types are:
179         1. Single
180         2. Double
181         3. Triple
182         Select your bed choice :- > "))
183     if room_bed_choice == 1:
184         room_bed = 'single bed'
185     if room_bed_choice == 2:
186         room_bed = 'Double bed'
187     if room_bed_choice == 3:
188         room_bed = 'Triple bed'
189     if room_bed_choice not in (1,2,3,4,5,6,7):
190         print("Invalid Input.....")
191
192     print("\n\t\tRent ----- Menu")
Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂  ⌂
```



```
Hotel_Management_System by Sagar yadav.py
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
190
191
192     print("\n\t\tRent ----- Menu")
193
194
195     print(''Available room rents for "Single bed" are:
196         1. AC      : ₹3499/night
197         2. Non-AC  : ₹1499/night
198         3. Delux   : ₹4799/night
199         4. Super Delux : ₹5499/night
200         5. Queen Delight : ₹6999/night
201         6. King Special  : ₹8999/night
202         7. Super Rich Special : ₹10499/night
203         \t\tNOTE: Rent changes as per bed there is addition of Rs.1000 per bed(type)'')
204     wait = input('\n Press Enter to continue....')
205
206     if room_type == 'AC':
207         room_rent_selection = 1
208
209     elif room_type == 'Non-AC':
210         room_rent_selection = 2
211
212     elif room_type == 'Delux':
213         room_rent_selection = 3
214
215     elif room_type == 'Super Delux':
216         room_rent_selection = 4
217
218     elif room_type == 'Queen Delight':
219         room_rent_selection = 5
Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂  ⌂
```



```
Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
217 elif room_type == 'Queen Delight':
218     | room_rent_selection = 5
219
220 elif room_type == 'King Special':
221     | room_rent_selection = 6
222
223 elif room_type == 'Super Rich Special':
224     | room_rent_selection = 7
225
226 else :
227     | pass
228
229 # 1. AC Room rent
230 if room_rent_selection == 1:
231     if room_bed == 'Single bed':
232         | room_rent = 3499
233     elif room_bed == 'Double bed':
234         | room_rent = 3499 + 1000
235     elif room_bed == 'Triple bed':
236         | room_rent = 3499 + 2000
237
238 # 2. Non-AC Room rent
239 if room_rent_selection == 2:
240     if room_bed == 'Single bed':
241         | room_rent = 1499
242     elif room_bed == 'Double bed':
243         | room_rent = 1499 + 1000
244     elif room_bed == 'Triple bed':
245         | room_rent = 1499 + 2000
246
247 # 3. Delux Room rent
248 if room_rent_selection == 3:
249     if room_bed == 'Single bed':
250         | room_rent = 4799
251     elif room_bed == 'Double bed':
252         | room_rent = 4799 + 1000
253     elif room_bed == 'Triple bed':
254         | room_rent = 4799 + 2000
255
256 # 4. Super delux Room rent
257 if room_rent_selection == 4:
258     if room_bed == 'Single bed':
259         | room_rent = 5499
260     elif room_bed == 'Double bed':
261         | room_rent = 5499 + 1000
262     elif room_bed == 'Triple bed':
263         | room_rent = 5499 + 2000
264
265 # 5. Queen delight Room rent
266 if room_rent_selection == 5:
267     if room_bed == 'Single bed':
268         | room_rent = 6999
269     elif room_bed == 'Double bed':
270         | room_rent = 6999 + 1000
271     elif room_bed == 'Triple bed':
272         | room_rent = 6999 + 2000
```

Ln 339, Col 26 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit


```
Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
243 elif room_bed == 'Double bed':
244     | room_rent = 1499 + 1000
245 elif room_bed == 'Triple bed':
246     | room_rent = 1499 + 2000
247
248 # 3. Delux Room rent
249 if room_rent_selection == 3:
250     if room_bed == 'Single bed':
251         | room_rent = 4799
252     elif room_bed == 'Double bed':
253         | room_rent = 4799 + 1000
254     elif room_bed == 'Triple bed':
255         | room_rent = 4799 + 2000
256
257 # 4. Super delux Room rent
258 if room_rent_selection == 4:
259     if room_bed == 'Single bed':
260         | room_rent = 5499
261     elif room_bed == 'Double bed':
262         | room_rent = 5499 + 1000
263     elif room_bed == 'Triple bed':
264         | room_rent = 5499 + 2000
265
266 # 5. Queen delight Room rent
267 if room_rent_selection == 5:
268     if room_bed == 'Single bed':
269         | room_rent = 6999
270     elif room_bed == 'Double bed':
271         | room_rent = 6999 + 1000
272     elif room_bed == 'Triple bed':
273         | room_rent = 6999 + 2000
```

Ln 339, Col 26 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit



```
# Hotel_Management_System by Sagar yadav.py
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
272 elif room_bed == 'Triple bed':
273     room_rent = 6999 + 2000
274
275 # 6. King Special Room rent
276 if room_rent_selection == 6:
277     if room_bed == 'Single bed':
278         room_rent = 8499
279     elif room_bed == 'Double bed':
280         room_rent = 8499 + 1000
281     elif room_bed == 'Triple bed':
282         room_rent = 8499 + 2000
283
284 # 7. Super rich special Room rent
285 if room_rent_selection == 7:
286     if room_bed == 'Single bed':
287         room_rent = 10499
288     elif room_bed == 'Double bed':
289         room_rent = 10499 + 1000
290     elif room_bed == 'Triple bed':
291         room_rent = 10499 + 2000
292
293 if room_rent_selection not in (1,2,3,4,5,6,7):
294     print("Invalid Input.....")
295     add_room()
296
297 print('\nFor ----->')
298 print("Room Number           :", room_no)
299 print("Room Bed Type        :", room_bed)
300 print("Room Type             :", room_type)
301 print("Your Rent will be (in Rs) : ", room_rent)

Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit   ⌂ ⌂
```



```
❸ Hotel_Management_System by Sagar yadav.py
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
301 print("Your Rent will be (in Rs) : ", room_rent)
302 room_rent = str(room_rent)
303
304 sql = 'insert into rooms(RoomNo, room_type,\n    rent, bed_type, room_status) values \
305         ('+room_no+', "'+ room_type.upper()+'",\n            '+room_rent+', "'+room_bed.upper()+'", "Free");'
306 cursor.execute(sql)
307 conn.commit()
308 print("\n Room added successfully....")
309 conn.close()
310 greet()
311 wait = input('\n\n\n Press Enter to continue....')
312
313 # 6. Function to modify room's data
314 def modify_room():
315     conn = mysql.connector.connect(
316         host='localhost', database = 'The_Taj_Hotel',
317         user='root', password='SAGAR YADAV')
318     cursor = conn.cursor()
319     clear()
320     print(' Change Room Information ')
321     print('*'*75)
322     print("NOTE: Rooms starts from 101...")
323     room_no = int(input('Enter Room No. :'))
324     while True:
325         check = room_exist(room_no)
326         if check == None:
327             print('Room doesn't exist....')
328             print("""Select any one from below -""")
329
Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit   ⌂ ⌂
```



```
❸ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚭ modify_room
330     print("""Select any one from below -
331     1. Want to enter another room no.
332     2. Return to main menu
333     3. Close application or Exit""")
334     choice = int(input("Enter your choice : > "))
335     if choice == 1:
336         room_no = input("Enter Another Room no to modify : > ")
337         main_menu()
338     elif choice == 2:
339         main_menu()
340     elif choice == 3:
341         greet()
342         break
343     else:
344         print("Invalid Input!")
345         print("Exiting.....")
346         greet()
347         break
348
349     elif check != None:
350         print("\n\tAvailable details of Room No.",room_no)
351         print('''
352             Room Type      : {}
353             Room Bedding   : {}
354             Room Rent       : {}
355             Room Status     : {}''.format(check[1],
356                                         check[3],check[2],check[4]))
357         wait = input("Press Enter to continue.....")
358         print("\n\t Choose what you want to modify")
359         print()
360         print('1. Room Type')
```

In 339, Col 26 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit

```
❸ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚭ modify_room
359     print('1.Room Type')
360     print('2.Room Rent')
361     print('3.Room Bed')
362     print('4.Room Status (free/occupied)')
363     choice = int(input('Enter your choice : '))
364     field_name = ''
365
366     if choice == 1: # Room Type
367         print('Available room types are:')
368         AC
369         Non-AC
370         Delux
371         Super Delux
372         Queen Delight
373         King Special
374         Super Rich Special'''
375     field_name = 'Room_Type'
376
377     if choice == 2: # Room rent
378         print('Available room rents for single bed are:')
379         AC           ₹3499/night
380         Non-AC       ₹1499/night
381         Delux        ₹4799/night
382         Super Delux ₹5999/night
383         Queen Delight ₹6999/night
384         King Special ₹8999/night
385         Super Rich Special ₹10499/night'''
386     field_name = 'rent'
387
388     if choice == 3: # Room's bed type
389         print('Available Bed Types are:
```

In 339, Col 26 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit



```
❖ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❖ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
385     field_name = 'rent'
386
387     if choice == 3: # Room's bed type
388         print('Available Bed Types are:
389             Single Bed
390             Double Bed
391             Triple Bed ')
392         field_name = 'bed_type'
393
394     if choice == 4: # Room's status
395         print('Available Status are:
396             Free
397             Occupied')
398         field_name = 'Room_Status'
399
400     value = input('Enter new value(Type the new value):> ')
401
402     if value.isdigit():
403         sql = "update rooms set {} = {} \
404               where RoomNo = {}".format(field_name, value, room_no)
405     else:
406         sql = "update rooms set {} = '{}' \
407               where RoomNo = {}".format(field_name, value.upper(), room_no)
408
409     cursor.execute(sql)
410     conn.commit()
411     greet()
412     wait = input('\n\n\n Record Updated \n
413 .....Press Enter to continue.....')
414
415     break
Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit   ⌂   ⌂

❖ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❖ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
412     wait = input('\n\n\n Record Updated \n
413 .....Press Enter to continue.....')
414
415
416     # 7. Function to add customer
417     def add_customer():
418         conn = mysql.connector.connect(
419             host='localhost', database = 'The_Taj_Hotel',
420             user='root', password='SAGAR YADAV')
421         cursor = conn.cursor()
422         clear()
423         print('Add New Customer - Screen')
424         print('*'*75)
425         name = input('\nEnter Customer Name :> ')
426         address = input('\nEnter Customer Address:> ')
427         phone = input('\nEnter Customer Contact No. :> ')
428         email = input('\nEnter Customer Email ID :')
429         id_proof = input('\nEnter Customer ID Proof type\
430 | (e.g,Aadhar/Passport/DL/VoterID) :> ')
431         id_proof_no = input('\nEnter Customer ID proof No. :> ')
432         males = input('\nEnter Total Males :> ')
433         females = input('\nEnter Total Females :> ')
434         children = input('\nEnter Total Children :> ')
435         noOfRoomMates = males + females + children
436
437         sql1 = 'insert into customers\
438             (CustomerName,address,Contact_No,email,id_proof,id_number\
439             ,males,females,children) values \
440             ("'+name+'","' + address.upper()+'","'+phone+'",\
Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit   ⌂   ⌂
```



```
❸ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
438     (CustomerName,address>Contact_No,email,id_proof,id_number\
439     ,males,females,children) values \
440     ("+name+", "+ address.upper() +", "+phone+", \
441     "+email.upper() +", "+id_proof.upper() +", "+id_proof_no.upper() +", \
442     '+males+', '+females+', '+children+') ;
443
444 cursor.execute(sql1)
445 sql2 = "select * from customers where id_number = '{}' ".format(id_proof_no)
446 cursor.execute(sql2)
447 data = cursor.fetchone()
448 print("\n\n\nCustomer Added successfully .....")
449 print("Customer Details ---")
450 print("Customer No. :", data[0])
451 print("Name      :", data[1])
452 print("Address   :", data[2])
453 print("Contact No.:", data[3])
454 print("Email     :", data[4])
455 print("ID Proof   :", data[5])
456 print("ID Proof No.:", data[6])
457 print("Males      :", data[7])
458 print("Female    :", data[8])
459 print("Children   :", data[9])
460 conn.commit()
461 conn.close()
462 greet()
463 wait = input('\n\n\n Press Enter to continue....')
464
465 # 8. Function to modify customer details
466 def modify_customer():
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
```

Ln 339, Col 26 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂

```
❸ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
464
465 # 8. Function to modify customer details
466 def modify_customer():
467     conn = mysql.connector.connect(
468         host='localhost', database = 'The_Taj_Hotel',
469         user='root', password='SAGAR YADAV')
470     cursor = conn.cursor()
471     clear()
472     print(' Change Customer Information ')
473     print('*'*75)
474     cust_no = input('Enter Customer No., whose information you want to change :>')
475     while True:
476         check = customer_exist(cust_no)
477         if check == None:
478             print('Customer doesn\'t exist for this Customer Number, ',cust_no,'....')
479             print("""Now, Select any one option from below -
480             1. Want to enter another Customer No.
481             2. Return to main menu
482             3. Close application or Exit""")
483             choice = int(input("Enter your choice :> "))
484             if choice == 1:
485                 cust_no = input('Enter Another Customer No. to modify :> ')
486             elif choice == 2:
487                 main_menu()
488             elif choice == 3:
489                 break
490             else:
491                 print("Invalid Input!")
492                 print("Exiting.....")
493                 break
```

Ln 339, Col 26 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂



```
❸ Hotel_Management_System by Sagar yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar yadav.py > ⚡ modify_room
493     break
494 elif check != None:
495     print("\n\tAvailable details of Customer No.",cust_no,'.....')
496     print('''
497         Name : {}
498         Address : {}
499         Phone No. : {}
500         Email : {}
501         ID Proof : {}
502         ID Proof No. : {}
503         No. of males : {}
504         No. of females : {}
505         No. of children : {} '''.format(check[1],check[2],check[3],
506                                         check[4],check[5],check[6],check[7],check[8],check[9]))
507     wait = input("\nPress Enter to continue....")
508
509     print("\n\t Choose what you want to modify")
510     print()
511
512     print('1.Name')
513     print('2.Address')
514     print('3.Phone No')
515     print('4.Email ID')
516     print('5.ID Proof')
517     print('6.ID Proof No')
518     print('7.Males')
519     print('8.Females')
520     print('9.Children')
521     choice = int(input('Enter your choice :'))
522     field_name = ''
```

```
❸ Hotel_Management_System by Sagar yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar yadav.py > ⚡ modify_room
522     field_name = ''
523 if choice == 1:
524     field_name = 'CustomerName'
525 if choice == 2:
526     field_name = 'Address'
527 if choice == 3:
528     field_name = 'Contact_No'
529 if choice == 4:
530     field_name = 'email'
531 if choice == 5:
532     field_name = 'id_proof'
533 if choice == 6:
534     field_name = 'ID_Number'
535 if choice == 7:
536     field_name = 'males'
537 if choice == 8:
538     field_name = 'females'
539 if choice == 9:
540     field_name = 'children'
541
542     value = input('Enter new value :')
543
544     if value.isdigit():
545         sql = sql = 'update customers set {} = {} where CustomerID ={}'.format(
546             field_name, value, cust_no)
547     else:
548         sql = sql = 'update customers set {} = "{}" where CustomerID ={}'.format(
549             field_name, value, cust_no)
550
551     cursor.execute(sql)
```



```
 551     cursor.execute(sql)
 552     conn.commit()
 553     greet()
 554     wait = input(
 555         '\n\n\n Record Updated .....Press Enter to continue.....')
 556     break
 557
 558 # 9. Function to book room
 559 def room_booking():
 560     conn = mysql.connector.connect(
 561         host = 'localhost', database ='the_taj_hotel',
 562         user = 'root', password = 'SAGAR YADAV'
 563     )
 564     cursor = conn.cursor()
 565     clear()
 566     print("Room Booking ----- Menu")
 567     print("*#*#0")
 568     room_no = int(input("\nEnter Room No. :> "))
 569     while True:
 570         query1 = "SELECT * From rooms where roomno = {}".format(room_no)
 571         cursor.execute(query1)
 572         data = cursor.fetchone()
 573         status = data[4]
 574
 575         query2 = "Select book_id, roomno, bill_date from bills where roomno = {}".format(room_no)
 576         cursor.execute(query2)
 577         record = cursor.fetchone()
 578         if record == None:
 579             rmno = 0
 580             pass
 581         else:
 582             book_id, rmno, billdate = record[0], record[1], record[2]
 583
 584             if data != None and status in ('free', 'Free'):
 585                 print("\nDetails for the Room No.", room_no)
 586                 room_no = data[0]
 587                 roomtype = data[1]
 588                 roomrent = data[2]
 589                 roombed = data[3]
 590                 print("Room No.      :",room_no)
 591                 print("Room Type    :",roomtype)
 592                 print("Room Rent    :",roomrent)
 593                 print("Room Bedding :",roombed)
 594                 cust_no = int(input("Enter Customer Id or No. :> "))
 595                 query2 = "SELECT * From customers where customerID = {}".format(cust_no)
 596                 cursor.execute(query2)
 597                 data2 = cursor.fetchone()
 598                 cname = data2[1]
 599
 600                 if customer_exist(cust_no) != None :
 601                     if rmno == room_no:
 602                         print("""The Date of Occupancy should be
 603 equal to or greater than the date :"",, billdate)
 604 DateOfBooking = datetime.strptime(input('Enter date of occupancy (yyyy-mm-dd) :> '), '%Y-%m-%d')
 605 if DateOfBooking.date() >= billdate:
 606                         pass
 607                     else :
 608                         print("Please enter a right date")
 609                         DateOfBooking = input('Enter again date of occupancy (yyyy-mm-dd) :> ')
```



```
❸ Hotel_Management_System by Sagar_yadav.py X
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
609     DateOfBooking = input('Enter again date of occupancy (yyyy-mm-dd) :> ')
610     else:
611         DateOfBooking = input('Enter date of occupancy (yyyy-mm-dd) :> ')
612     advance = input('Enter advance amount :> ')
613     sql1 = 'update rooms \
614             set Room_Status = "Occupied",\
615                 CustomerID = {} \
616                 where Roomno = {}'.format(cust_no, room_no)
617     sql2 = 'insert into bookings\
618             (roomno,customerid,DateOfBooking,advance,customername) \
619             values ({}, {}, "{}", {}, "{}")'.format(room_no,
620                                                 cust_no, DateOfBooking, advance, cName)
621     cursor.execute(sql1)
622     cursor.execute(sql2)
623     print('\n\nRoom no ', room_no, 'a ', roomtype, ' Room is booked for---')
624     print('Customer No. :', cust_no)
625     print("Customer Name :", cName)
626     conn.commit()
627     conn.close()
628     wait = input('\n Press Enter to continue....')
629     break
630 else :
631     print("\tNOTE:Customer ID you entered is not present in our database.")
632
633 elif data == None:
634     print("There is no such room found in our database, Room No.", room_no)
635     room_no = int(input("\nEnter Another Room No. :> "))
636 else :
637     print("NOTE: This room (", room_no,") is not available for booking")
638     print("\tIt is Already Booked!.....")
639
640 # 10. Function to generate bill
641 def bill_generation():
642     global gst
643     global st
644
645     conn = mysql.connector.connect(
646         host='localhost', database = 'The_Taj_Hotel',
647         user='root', password='SAGAR YADAV')
648     cursor = conn.cursor()
649
650     room_no = input('Enter Room No. for which you want to generate bill :> ')
651     while True:
652         query1 = "SELECT r.room_type, b.customerid, dateofleaving, \
653                   dateofbooking, advance, r.rent, b.book_id, r.bed_type, r.roomno \
654                   FROM bookings b, rooms r \
655                   WHERE r.roomno = b.roomno and r.roomno = {}".format(room_no)
656         cursor.execute(query1)
657         data = cursor.fetchall()
658         checkRoomExist = room_exist(room_no)
659         if checkRoomExist == None:
660             print("\nNOTE: This room doesn't exist in our database,Room no.", room_no)
661             print("\tYou have to create a new room")
662             print("Exiting Bill Generation Menu.....")
663             wait = input('\nPress Enter to continue....')
664             break
665         else :
666             pass
```



```
❸ Hotel_Management_System by Sagar_yadav.py X
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
667 if data == []:
668     print("NOTE:Enter Another Room No., there is no booking for this room no.",room_no)
669     wait = input('\n Press Enter to continue....')
670     choice = int(input(''Select any one option-
671     1. Want to enter another room no
672     2. Return to Main Menu
673     3. Return back
674     :> '''))
675     if choice == 1:
676         room_no = input('Enter Another Room No. for which you want to generate bill :> ')
677     elif choice == 2:
678         main_menu()
679         pass
680     elif choice == 3:
681         break
682     else:
683         print("Invalid Input!")
684         print("Exiting.....")
685         break
686
687     else:
688         for rm_t, c_id, dl, db, adv, rent, bookid, bedtype, rmno in data:
689             room_type, cust_id, dolv, roombed, roomno = rm_t, c_id, dl, bedtype, rmno
690             doo, advance, rent, book_id = db, adv, rent, bookid
691             if dolv != None :
692                 print("\n\tNOTE: This room has already a bill...")
693                 wait = input('\n Press Enter to continue....')
694                 choice = int(input(''Select any one option-
695                     1. Want to enter another room id
696                     2. Return to Main Menu
697                     3. Close application or Exit
698                     :> '''))
699                 if choice == 1:
700                     room_no = input('Enter Another Room no. for which you want to generate bill :> ')
701                 elif choice == 2:
702                     main_menu()
703                     pass
704                 elif choice == 3:
705                     break
706                 else:
707                     print("Invalid Input!")
708                     print("Exiting.....")
709                     break
710
711             elif dolv == None:
712                 query2 = "SELECT customername, address\
713                             FROM customers\
714                             Where customerid = {}".format(cust_id)
715                 cursor.execute(query2)
716                 data = cursor.fetchone()
717                 cname, caddr = data[0], data[1]
718
719                 print(''\tRoom Detail or Booking Details
720 \t-----\n\tRoom No. : {}'''.format(roomno))
721
722
Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit  ⌂ ⌂
```



```
❸ Hotel_Management_System by Sagar_yadav.py X
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚒ modify_room
693     print("\n\tNOTE: This room has already a bill...")
694     wait = input('\n Press Enter to continue....')
695     choice = int(input(''Select any one option-
696         1. Want to enter another room id
697         2. Return to Main Menu
698         3. Close application or Exit
699         :> '''))
700     if choice == 1:
701         room_no = input('Enter Another Room no. for which you want to generate bill :> ')
702     elif choice == 2:
703         main_menu()
704         pass
705     elif choice == 3:
706         break
707     else:
708         print("Invalid Input!")
709         print("Exiting.....")
710
711     elif dolv == None:
712         query2 = "SELECT customername, address\
713                         FROM customers\
714                         Where customerid = {}".format(cust_id)
715         cursor.execute(query2)
716         data = cursor.fetchone()
717         cname, caddr = data[0], data[1]
718
719         print(''\tRoom Detail or Booking Details
720 \t-----\n\tRoom No. : {}'''.format(roomno))
721
722
Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit  ⌂ ⌂
```



```
Hotel_Management_System by Sagar yadav.py
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
722     Room No.          : {}
723     Room Name         : {}
724     Room Bedding      : {}
725     Room Rent(Per night) : {}
726     Customer ID       : {}
727     Customer Name     : {}
728     Customer Address   : {}
729     Advance paid       : {}''.format(roomno, room_type, roombed, rent, cust_id, cname, caddr, advance))
730     wait = input('\nPress Enter to continue....')

731
732
733     clear()
734     print('Bill Generation ')
735     print('-'*75)
736     print('Room occupied  :',room_no)
737     dol = date.today()
738     total_days = (dol-doo).days
739     amount = total_days*rent
740     gst_amount = amount*int(gst)/100
741     st_amount = amount*int(st)/100
742     payable_amount = total_days * rent - advance + gst_amount + st_amount
743
744     print('Date of Occupancy :',doo, '\nDate of Leaving :',dol)
745     print('Total Payable Days : ', total_days)
746     print('Room Rent Per Day : ', rent)
747     print('Total Amount : ',amount)
748     print('Advance :',advance,'GST ({}): {}'.format(gst,gst_amount), '\nService Tax ({}): {}'.format(st,st_amount))
749     print('Amount Payable : ', payable_amount)

Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⚙  ⌂

Hotel_Management_System by Sagar yadav.py
Main files > Hotel_Management_System by Sagar yadav.py > modify_room
749     print('Amount Payable :',payable_amount)
750
751     sql1 = 'update rooms set room_status ="free" \
752     where roomno ='+room_no+';
753     sql2 = 'update bookings set DateOfLeaving ="' +str(dol)+'" \
754     where roomno=' +room_no+ ' and customerid =' +cust_id+';
755     sql3 = 'insert into bills(book_id,amount,bill_date,gst,st,roomno) \
756     values(' +str(book_id)+',' +str(payable_amount)+',' +str(dol)+',"'+str(gst)+',' +str(st)+',' +str
757     (roomno)+');'
758     cursor.execute(sql1)
759     cursor.execute(sql2)
760     cursor.execute(sql3)
761     conn.commit()
762     conn.close()
763     print("Bill Generated successfully....")
764     print('This room is Free now, Room No.:',roomno)
765     wait = input('\n\n Press Enter to continue....')
766     break
767
768 # 11. Function to search rooms
769 def search_rooms():
770     conn = mysql.connector.connect(
771         host='localhost', database = 'The_Taj_Hotel',
772         user='root', password='SAGAR YADAV')
773     cursor = conn.cursor()
774     room_no = input('Enter Room No :')
775     check = room_exist(room_no)
776     if check is not None:
777         sql = 'select * from rooms where roomno =' +room_no+';'
    cursor.execute(sql)
Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⚙  ⌂
```





```
# Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
830 value = input('Enter value that you want to search :')
831 if field_name == 'CustomerID':
832     sql = 'select * from customers where ' + field_name + ' = ' + value + ';'
833 else:
834     sql = 'select * from customers where ' + field_name + ' like "%' + value + '%"'
835
836 cursor.execute(sql)
837 records = cursor.fetchall()
838
839 # checking for record
840 if records == []:
841     print('*'*45)
842     print('Search Result for {} = {} : NOT FOUND'.format(field_name,value))
843     wait = input('\n\n\nPress Enter to continue.....')
844     break
845 else :
846     pass
847
848 clear()
849 print('Search Result for {} = {}'.format(field_name,value))
850 print('*'*80)
851
852 for record in records:
853     print('''
854         Customer ID : {}
855         Customer Name : {}
856         Address : {}
857         Contact No. : {}
858         Email : {}''')

Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit   ⚙   ⌂

# Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
859
860     ID Proof : {}
861     ID Proof No. : {}''.format(
862         record[0], record[1], record[2], record[3], record[4], record[5], record[6]))
863
864 conn.close()
865 greet()
866 wait = input('\n\n\nPress Enter to continue.....')
867     break
868
869 # 13. Function to search for bookings
870 def search_booking():
871     while True:
872         conn = mysql.connector.connect(
873             host='localhost', database = 'The_Taj_Hotel',
874             user='root', password='SAGAR YADAV')
875         cursor = conn.cursor()
876         cust_no = input('Enter Customer No :')
877         sql = 'select book_id,r.roomno,c.Customername,DateOfBooking,advance from bookings b, customers c, rooms r where b.roomno = r.roomno AND b.customerid = '+cust_no+' and DateOfLeaving is NULL;'
878         cursor.execute(sql)
879         records = cursor.fetchall()
880         clear()
881         print('Booking information for customer ID :{}'.format(cust_no))
882         print('*'*50)
883
884         if records == []:
885             print("NOTE: There is no booking for this Customer no.",cust_no)
886             wait = input('\n Press Enter to continue....')
887             choice = int(input('Select any one option-'))

Ln 339, Col 26   Spaces: 2   UTF-8   CRLF   Python  3.10.6 64-bit   ⚙   ⌂
```



The image shows two tabs of Python code for a Hotel Management System, likely from a Jupyter Notebook or similar environment. The left tab contains code for modifying room bookings, and the right tab contains code for searching bills.

Left Tab: modify_room

```
884     print("NOTE: There is no booking for this Customer no.",cust_no)
885     wait = input('\n Press Enter to continue....')
886     choice = int(input("Select any one option-
887     1. Want to enter another room no
888     2. Return to Main Menu
889     3. Return to Search menu
890     :> ''"))
891     if choice == 1:
892         cust_no = input('Enter Another Customer No.:> ')
893     elif choice == 2:
894         main_menu()
895         pass
896     elif choice == 3:
897         break
898     else:
899         print("Invalid Input!")
900         print("Returning to Search menu.....")
901         break
902     else:
903         for record in records:
904             print("\tCustomer ID :\t{}"
905             Room Number :\t{}"
906             Customer's Name :\t{}"
907             Date Of Booking :\t{}"
908             Advance Payment :\t{}''.format(
909                 record[0], record[1], record[2], record[3], record[4]))
910         conn.close()
911     greet()
912     wait = input('\n\n\nPress Enter to continue.....')
913     break
```

Right Tab: search_bill

```
913     break
914
915 # 14. Function to search bills
916 def search_bill():
917     while True:
918         conn = mysql.connector.connect(
919             host='localhost', database = 'The_Taj_Hotel',
920             user='root', password='SAGAR YADAV')
921         cursor = conn.cursor()
922         bill_no = input('Enter Bill No :')
923         sql = ' select bills.bill_id, bills.amount, bill_date, gst, st, b.book_id, DateOfBooking, \
924             DateOfLeaving, advance, b.CustomerName, address, Contact_No, email, bills.roomno, r.room_type, \
925             r.bed_type, r.rent, c.customerid, id_proof, id_number \
926             from bills, bookings b, customers c , rooms r \
927             where bills.book_id = b.book_id \
928             and b.roomno = r.roomno and b.customerid = c.customerid AND NOT DateOfLeaving is NULL AND \
929             bill_id = '+bill_no +';'
930         cursor.execute(sql)
931         record = cursor.fetchone()
932         clear()
933         print('Bill information for Bill No :{}'.format(bill_no))
934         print('***140')
935
936 # checking record
937 if record == None:
938     print("NOTE :There is no record for Bill No. ", bill_no, "in our database.")
939     wait = input('\n\n\nPress Enter to continue.....')
940     greet()
941     break
942     '
```



The screenshot shows two tabs open in a code editor:

Hotel_Management_System by Sagar yadav.py

```
939     wait = input('\n\n\nPress Enter to continue.....')
940     greet()
941     break
942 else :
943     pass
944
945     print('Bill Number      : ', record[0])
946     print('Bill\'s Total Amount   : ', 'Rs.',record[1])
947     print('Bill Date        : ', record[2])
948     print('GST Charged     : ', 'Rs.',record[3])
949     print('Service Tax Charged : ', 'Rs.',record[4])
950     print('Booking ID       : ', record[5])
951     print('Room no. Used    : ', record[13])
952     print('Room Name or type : ', record[14])
953     print('Room Bedding      : ', record[15])
954     print('Room Rent(per night) : ', 'Rs.',record[16])
955     print('Date of Occupancy  : ', record[6])
956     print('Date of Leaving    : ', record[7])
957     print('Advance Paid      : ', 'Rs.',record[8])
958     print('Customer\'s ID      : ', record[17])
959     print('Customer\'s Name     : ', record[9])
960     print('Customer\'s Address   : ', record[10])
961     print('Customer\'s Phone     : ', record[11])
962     print('Customer\'s Email ID  : ', record[12])
963     print('Customer\'s ID Proof   : ', record[18])
964     print('Customer\'s ID Proof No. : ', record[19])
965
966     conn.close()
967     wait = input('\n\n\nPress Enter to continue.....')
968     break
```

Hotel_Management_System by Sagar yadav.py

```
968     break
969
970 # 15. Function to select what you want to search
971 def search_menu():
972     while True:
973         clear()
974         print(' Search Menu')
975         print('-'*75)
976         print("\n1.Room Status")
977         print("\n2.Booking Status")
978         print("\n3.Customer Details")
979         print("\n4.Bills")
980         print("\n5.Back to Main Menu")
981         print("\n\n")
982         choice = int(input('Enter your choice ...: '))
983         if choice == 1:
984             search_rooms()
985         if choice == 2:
986             search_booking()
987         if choice == 3:
988             search_customer()
989         if choice == 4:
990             search_bills()
991         if choice == 5:
992             greet()
993             break
994
995 # 16. Function to take report of room
996 def report_room_status():
997     conn = mysql.connector.connect(
```



```
# Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
997     conn = mysql.connector.connect(
998         host='localhost', database = 'The_Taj_Hotel',
999         user='root', password='SAGAR YADAV')
1000     cursor = conn.cursor()
1001     sql = 'select * from rooms'
1002     cursor.execute(sql)
1003     records = cursor.fetchall()
1004     clear()
1005     print('Rooms Status - Report')
1006     print('-'*75)
1007
1008     print('{:15s} {:18s} {:15s} {:15s} {:10s}'.format('Room No', 'Room Type', 'Rent', 'Bedding', 'Room
Status', 'Booked for CustomerID'))
1009     for no,rtype,rent,bed,status,customerid in records:
1010         # if customerid == None:
1011         #   customerid = 0
1012         print('{:<10d} {:<21s} {:>18s} {:>12s} {:>20s}'.format(no, rtype, rent, bed, status, str(customerid)))
1013     conn.close()
1014     greet()
1015     wait = input('\n\n\n Press Enter to continue....')
1016
1017 # 17. Function to take report of booking status
1018 def report_booking_status():
1019     conn = mysql.connector.connect(
1020         host='localhost', database = 'The_Taj_Hotel',
1021         user='root', password='SAGAR YADAV')
1022     cursor = conn.cursor()
1023     sql = """ SELECT b.book_id,b.roomno,DateOfBooking,DateOfLeaving,advance, b.Customername,address,Contact_No
FROM bookings b, customers c
WHERE b.customerid = c.customerid."""
1024
1025     WHERE b.customerid = c.customerid;
1026     ...
1027
1028     cursor.execute(sql)
1029     records = cursor.fetchall()
1030     clear()
1031     print(' Booking Status - Report')
1032     print('-'*75)
1033     #{:9s} {:15s} {:18s} {:15s} {:15s} {:20s}
1034     #{:10s} {:10s} {:20s} {:20s} {:20s} {:15s}
1035     print('{:9s} {:15s} {:10s} {:15s} {:20s} {:15s} {:15s}'.format(
1036         'BookID', 'RoomNo', 'DOO', 'DOL', 'Advance', 'CustomerName', 'Address', 'Phone'))
1037     for BookID, Rno, doo, dol, advance, Cname, addr, phone in records:
1038         print('{:>10d} {:>20s} {:>12s} {:>14.2f} {:>20s} {:>14s} {:>18d}'.format(
1039             BookID, Rno, str(doo), str(dol), advance, Cname, addr, phone))
1040     conn.close()
1041     greet()
1042     wait = input('\n\n\n Press Enter to continue....')
1043
1044 # 18. Function for report menu of report functions
1045 def report_menu():
1046     while True:
1047         clear()
1048         print('Report Menu')
1049         print('*'*35)
1050         print("\n1.Room Status")
1051         print("\n2.Booking Status")
1052         print("\n3.Bill Collections")
1053         print("\n4.Back to Main Menu")
1054         print("\n\n")
1055         choice = int(input("Enter your choice : "))

Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂ ⌂
```

```
# Hotel_Management_System by Sagar_yadav.py
Main files > Hotel_Management_System by Sagar_yadav.py > modify_room
1025     WHERE b.customerid = c.customerid;
1026     ...
1027
1028     cursor.execute(sql)
1029     records = cursor.fetchall()
1030     clear()
1031     print(' Booking Status - Report')
1032     print('-'*75)
1033     #{:9s} {:15s} {:18s} {:15s} {:15s} {:20s}
1034     #{:10s} {:10s} {:20s} {:20s} {:20s} {:15s}
1035     print('{:9s} {:15s} {:10s} {:15s} {:20s} {:15s} {:15s}'.format(
1036         'BookID', 'RoomNo', 'DOO', 'DOL', 'Advance', 'CustomerName', 'Address', 'Phone'))
1037     for BookID, Rno, doo, dol, advance, Cname, addr, phone in records:
1038         print('{:>10d} {:>20s} {:>12s} {:>14.2f} {:>20s} {:>14s} {:>18d}'.format(
1039             BookID, Rno, str(doo), str(dol), advance, Cname, addr, phone))
1040     conn.close()
1041     greet()
1042     wait = input('\n\n\n Press Enter to continue....')
1043
1044 # 18. Function for report menu of report functions
1045 def report_menu():
1046     while True:
1047         clear()
1048         print('Report Menu')
1049         print('*'*35)
1050         print("\n1.Room Status")
1051         print("\n2.Booking Status")
1052         print("\n3.Bill Collections")
1053         print("\n4.Back to Main Menu")
1054         print("\n\n")
1055         choice = int(input("Enter your choice : "))

Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂ ⌂
```



The image shows two tabs of a Python script for a Hotel Management System, each with various icons for file operations, search, and navigation.

Tab 1: Hotel_Management_System by Sagar yadav.py

```
1054 choice = int(input('Enter your choice ...: '))
1055 if choice == 1:
1056     report_room_status()
1057 if choice == 2:
1058     report_booking_status()
1059 if choice == 3:
1060     reportCollection()
1061 if choice == 4:
1062     break
1063
1064 # 19. Function for Collections
1065 def reportCollection():
1066     conn = mysql.connector.connect(
1067         host='localhost', database = 'The_Taj_Hotel',
1068         user='root', password='SAGAR YADAV')
1069     cursor = conn.cursor()
1070     clear()
1071     print("Bill Collections ----- Menu")
1072     print('*'*45)
1073     print("1.Total Bill Collection")
1074     print("2.Collection from individual customer")
1075     print("3.Back to Report menu")
1076     choice = int(input("\nEnter your choice from above (1 or 2) :: "))
1077     # FOR TOTAL COLLECTION
1078     if choice == 1:
1079         print("\nTotal Collection")
1080         print('*'*35)
1081         sql = "SELECT sum(Amount) from bills;"
1082         cursor.execute(sql)
1083         data = cursor.fetchone()
```

Tab 2: Hotel_Management_System by Sagar yadav.py

```
1083 data = cursor.fetchone()
1084 print("Total Collections we have, Rs.", data[0])
1085
1086 wait = input('\n\n\n Press Enter to continue....')
1087
1088
1089 # FOR INDIVIDUAL CUSTOMER
1090 if choice == 2:
1091     print("\nCollection from individual customer")
1092     print('*'*35)
1093     custNo = input("Enter Customer No. :: ")
1094     sql1 = "SELECT Dateofleaving, advance \
1095           FROM bookings WHERE customerID = {};" .format(custNo)
1096     cursor.execute(sql1)
1097     records = cursor.fetchall()
1098     count = cursor.rowcount
1099     print("Rows ::", count)
1100     for record in records:
1101         dateofleaving = record[0]
1102         advance = record[1]
1103
1104         if count == 1 and dateofleaving == None:
1105             collection = advance
1106             print("Collection from Customer No.", custNo, "=> Rs. ",collection)
1107
1108         elif count == 1 and dateofleaving != None :
1109             sql2 = "SELECT amount\
1110                   From bills bl, bookings bk\
1111                   WHERE bl.book_id = bk.book_id and customerid ={};" .format(custNo)
```



```
❖ Hotel_Management_System by Sagar yadav.py ×
Main files > ❖ Hotel_Management_System by Sagar yadav.py > ⚙ modify_room
1109     sql12 = "SELECT amount"
1110         From bills bl, bookings bk\|
1111             | WHERE bl.book_id = bk.book_id and customerid ={};".format(custNo)
1112             cursor.execute(sql12)
1113             data = cursor.fetchone()
1114             print("Collections from CustomerNo.", custNo, "we have, Rs.", data[0])
1115
1116
1117 elif count > 1 and dateoffleaving == None:
1118     sql13 = "SELECT bk.book_id, amount\|
1119             | From bookings bk, bills bl\|
1120                 | Where bk.book_id = bl.book_id and customerid ={};".format(custNo)
1121             cursor.execute(sql13)
1122             records = cursor.fetchall()
1123             for record in records:
1124                 bill = record[1]
1125                 collection = bill + advance
1126                 print("Collection from Customer No.", custNo, ":> Rs. ",collection)
1127
1128 elif count > 1 and dateoffleaving != None:
1129     bill = 0
1130     sql13 = "SELECT bk.book_id, amount\|
1131             | From bookings bk, bills bl\|
1132                 | Where bk.book_id = bl.book_id and customerid ={};".format(custNo)
1133             cursor.execute(sql13)
1134             records = cursor.fetchall()
1135             for record in records:
1136                 bill += record[1]
1137                 collection = bill
1138                 print("collection from Customer No.", custNo, ":> Rs. ",collection)

Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⚡  ⌂

❖ Hotel_Management_System by Sagar yadav.py ×
Main files > ❖ Hotel_Management_System by Sagar yadav.py > ⚙ modify_room
1138     print("Collection from Customer No.", custNo, ":> Rs. ",collection)
1139
1140     break
1141
1142 if count == 0 :
1143     print("No record found for Customer No.", custNo)
1144 if choice == 3:
1145     report_menu()
1146
1147 greet()
1148 wait = input('\n\n\n Press Enter to continue...')
1149
1150 # 20. Function for Main Screen
1151 def main_menu():
1152     while True:
1153         clear()
1154         print(' H O T E L   M A N A G E M E N T   S Y S T E M ')
1155         print('*'*75)
1156         print("\n1. Add New Room")
1157         print("\n2. Add Customer")
1158         print("\n3. Modify Room Information")
1159         print("\n4. Modify Customer Information")
1160         print("\n5. Room Booking")
1161         print("\n6. Bill Generation")
1162         print("\n7. Search Database")
1163         print("\n8. Report Menu")
1164         print("\n9. Hotel Details")
1165         print("\n10. Change Hotel Details")
1166         print("\n11. Close application")
1167         print("\n\n")

Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⚡  ⌂
```



```
❸ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚙ modify_room
1167     print('\n\n')
1168     choice = int(input('Enter your choice ...: '))
1169     if choice == 1: # Add New Room
1170         add_room()
1171     if choice == 2: # Add Customer
1172         add_customer()
1173     if choice == 3: # Modify Room Information
1174         modify_room()
1175     if choice == 4: # Modify Customer Information
1176         modify_customer()
1177     if choice == 5: # Room Booking
1178         room_booking()
1179     if choice == 6: # Bill Generation
1180         bill_generation()
1181     if choice == 7: # Search Database
1182         search_menu()
1183     if choice == 8: # Report Menu
1184         report_menu()
1185     if choice == 9: # Hotel Details
1186         hotelDetails()
1187     if choice == 10: # Change Hotel Details
1188         change_hotelDetails()
1189     if choice == 11:
1190         greet()
1191     break
1192
1193 # 21. Function to clear the screen
1194 def clear():
1195     for _ in range(30):
1196         print()
Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂ ⌂

❸ Hotel_Management_System by Sagar_yadav.py ×
Main files > ❸ Hotel_Management_System by Sagar_yadav.py > ⚙ modify_room
1183     if choice == 8: # Report Menu
1184         report_menu()
1185     if choice == 9: # Hotel Details
1186         hotelDetails()
1187     if choice == 10: # Change Hotel Details
1188         change_hotelDetails()
1189     if choice == 11:
1190         greet()
1191     break
1192
1193 # 21. Function to clear the screen
1194 def clear():
1195     for _ in range(30):
1196         print()
1197
1198 # 22. Function to greet
1199 greet = lambda : print("\n\t\tThanks for using....\n\t\tCreated by : SAGAR YADAV")
1200
1201 #
1202 if __name__ == "__main__":
1203     hotelDetails()
1204     main_menu()
1205     greet()
1206
1207
Ln 339, Col 26  Spaces: 2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂ ⌂
```



Pictures of Creating Table for running HMS.py

```
# Project Name : Hotel Management System
# Created by : SAGAR YADAV
...
# Creating Tables with thier Structures - Table Rooms
TABLES['Rooms'] = (
    """create table Rooms (
        RoomNo int PRIMARY key,
        Room_Type varchar(25) default 'AC',
        -- CREATED BY: SAGAR YADAV
        Rent float not null default '2500.00',
        Bed_Type varchar(15),
        Room_Status varchar(15),
        CustomerID int default null ENGINE=InnoDB'''")
...
# Creating Tables with thier Structures - Table Customers
TABLES['Customers'] = (
    """create table Customers (
        CustomerID int primary key AUTO_INCREMENT,
        CustomerName varchar(50) Not Null,
        Address varchar(70) Not null,
        Contact_No bigint default '1234567890',
        -- CREATED BY: SAGAR YADAV
        Email varchar(50) default 'Not Available',
        ID_Proof varchar(50) not null default 'Aadhar Card',
        Id_Number varchar(20) not null unique,
        Males int default '1',
        Females int default '0',
        Children int default '0'
        AUTO_INCREMENT = 101 ENGINE=InnoDB'''")
# Created by Sagar Yadav
# Creating Tables with thier Structures - Table: Bookings
TABLES['Bookings'] = (
    """CREATE TABLE Bookings (
        Book_id int primary key AUTO_INCREMENT,
        RoomNo int references Rooms(RoomNo),
        DateOfBooking date Not null,
        CREATED_BY : SAGAR YADAV
```



SAGAR YADAV

HOTEL MANAGEMENT SYSTEM



```
File Edit Selection View Go Run Terminal Help Creating Tables for running HMS file.py - Complete project - Visual Studio Code
EXPLORER ...
COMPLE...
Main files
Creating Tables for r...
Hotel_Management_...
The_Taj_Hotel.sql
Hotel Management S...
Read me first.pdf
Read me first.rtf
Main files > Creating Tables for running HMS file.py ...
47     BOOK_id int primary key AUTO_INCREMENT,
48     RoomNo int references Rooms(RoomNo),
49     DateOfBooking date Not null,
50     -- CREATED BY: SAGAR YADAV
51     DateOfLeaving date default null,
52     Advance int DEFAULT '0',
53     CustomerID varchar(50) Not null references customers.CustomerID,
54     CustomerName varchar(50) Not Null)
55     ENGINE=InnoDB AUTO_INCREMENT = 101 '')
56
57 # Creating Tables with thier Structures - Table Hotel_Details
58 TABLES['Hotel_Details'] = (
59     '''CREATE TABLE Hotel_Details (
60         ID int primary key,
61         -- CREATED BY: SAGAR YADAV
62         Field_name varchar(30) Not NULL,
63         Entry varchar(100) not NULL) ENGINE=InnoDB'''')
64
65 # Table structure for table Bills
66 TABLES['Bills'] = (
67     '''CREATE TABLE Bills (
68         Bill_id int NOT NULL AUTO_INCREMENT,
69         Book_id int DEFAULT NULL references Bookings.Book_id,
70         RoomNo int DEFAULT Null references rooms.RoomNo,
71         Bill_date date DEFAULT NULL,
72         GST int DEFAULT 42,
73         ST int DEFAULT 26,
74         Amount decimal DEFAULT NULL,
75         -- CREATED BY: SAGAR YADAV
76         PRIMARY KEY (bill_id)
77         AUTO_INCREMENT = 101 ENGINE=InnoDB'''')
78
79 cnx = mysql.connector.connect(user='root',password = 'SAGAR YADAV')
80 cursor = cnx.cursor()
81 # Created by Sagar Yadav
82 def create_database(cursor):
83     try:
84         cursor.execute("CREATE DATABASE {} DEFAULT CHARACTER SET 'utf8'".format(DB_NAME))
85     except mysql.connector.Error as err:
86         print("Failed creating database: {}".format(err))
87         exit(1)
88     try:
89         cursor.execute("USE {}".format(DB_NAME))
90     except mysql.connector.Error as err:
91         print("Database {} does not exists.".format(DB_NAME))
92         if err.errno == errorcode.ER_BAD_DB_ERROR:
93             create_database(cursor)
94             print("Database {} created successfully.".format(DB_NAME))
95             cnx.database = DB_NAME
96         else:
97             print(err)
98             exit(1)
99 # Created by Sagar Yadav
```

Follow & Connect with me on :





The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a "COMPLETE PROJECT" folder containing "Main files". Inside "Main files" are "Creating Tables for r...", "Hotel_Management_...", "The_Taj_Hotel.sql", "Hotel Management S...", "Read me first.pdf", and "Read me first.rtf".
- Code Editor:** Displays a Python script titled "Creating Tables for running HMS file.py". The code is as follows:

```
100 for table_name in TABLES:
101     table_description = TABLES[table_name]
102     try:
103         print("Creating table {}:".format(table_name), end='')
104         cursor.execute(table_description)
105     except mysql.connector.Error as err:
106         if err.errno == errorcode.ER_TABLE_EXISTS_ERROR:
107             print("already exists.")
108         else:
109             print(err.msg)
110     else:
111         print("OK")
112
113 # Dumping data for table `Hotel_Details`
114 Hotel_Details = ('''INSERT INTO Hotel_Details (ID, Field_name, Entry) VALUES
115     (1, 'Hotel_Name', "The Taj Hotel"),
116     (2, 'Address', 'ABC Sector, Delhi -22'),
117     (3, 'Phone', '011-8000111,12,13'),
118     (4, 'Email', 'thetajhotel_delhi@gmail.com'),
119     (5, 'GST', '42'),
120     (6, 'ST', '16')'''')
121
122 cursor.execute(Hotel_Details)
123 print("Data dumped successfully in Table: Hotel_details")
124 cnx.commit()
125 cursor.close()
126 cnx.close()
```

Status Bar: Shows "Ln 1, Col 1" and other settings like "Spaces: 4", "UTF-8", "CRLF", "Python", "3.10.6 64-bit".

Pictures of The Taj Hotel.sql

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a "COMPLETE..." folder containing "Main files". Inside "Main files" are "Creating Tables for r...", "Hotel_Management_...", "The_Taj_Hotel.sql", "Hotel Management S...", "Read me first.pdf", and "Read me first.rtf".
- Code Editor:** Displays an SQL script titled "The_Taj_Hotel.sql". The code is as follows:

```
1 # The_Taj_Hotel SQL Script
2 # Created by: SAGAR YADAV
3 /*
4 Execute this script in your SQL database server first.
5 */
6 # Creating Database
7 drop database if exists The_Taj_Hotel;
8 CREATE database The_Taj_Hotel;
9 -- CREATED BY: SAGAR YADAV
10 # Using the database "The_Taj_Hotel"
11 use The_Taj_Hotel;
12
13 # Creating Tables with thier Structures - Table Rooms
14 create table Rooms (
15     -- RoomID int primary key AUTO_INCREMENT,
16     RoomNo int PRIMARY key,
17     Room_Type varchar(25) default 'AC',
18     Rent float not null default '2500.00',
19     Bed_Type varchar(15),
20     Room_Status varchar(15),
21     CustomerID int default null,
22     ENGINE=InnoDB AUTO_INCREMENT = 101;
23     -- CREATED BY: SAGAR YADAV
24 # Creating Tables with thier Structures - Table Customers
25 create table Customers (
26     CustomerID int primary key AUTO_INCREMENT,
27     CustomerName varchar(50) Not Null,
```

Status Bar: Shows "Ln 1, Col 1" and other settings like "Spaces: 2", "UTF-8", "LF", "SQL".



SAGAR YADAV

HOTEL MANAGEMENT SYSTEM



The_Taj_Hotel.sql - Complete project - Visual Studio Code

```
File Edit Selection View Go Run Terminal Help The_Taj_Hotel.sql - Complete project - Visual Studio Code
EXPLORER ... Main files > The_Taj_Hotel.sql
COMPLETE PROJECT Main files
Main files Creating Tables for r... Hotel_Management_...
The_Taj_Hotel.sql Hotel Management S...
Read me first.pdf Read me first.rtf
> OUTLINE > TIMELINE
Ln 1, Col 1 Spaces: 2 UTF-8 LF SQL ⚡ 🔍
The_Taj_Hotel.sql - Complete project - Visual Studio Code
File Edit Selection View Go Run Terminal Help The_Taj_Hotel.sql - Complete project - Visual Studio Code
EXPLORER ... Main files > The_Taj_Hotel.sql
COMPLETE PROJECT Main files
Main files Creating Tables for r... Hotel_Management_...
The_Taj_Hotel.sql Hotel Management S...
Read me first.pdf Read me first.rtf
> OUTLINE > TIMELINE
Ln 1, Col 1 Spaces: 2 UTF-8 LF SQL ⚡ 🔍
25 create table Customers (
26     CustomerID int primary key AUTO_INCREMENT,
27     CustomerName varchar(50) Not Null,
28     Address varchar(70) Not null,
29     Contact_No bigint default '1234567890',
30     Email varchar(50) default 'Not Available',
31     ID_Proof varchar(50) not null default 'Aadhar Card',
32     Id_Number varchar(20) not null unique,
33     Males int default '1',
34     Females int default '0',
35     Children int default '0'
36     AUTO_INCREMENT = 101 ENGINE=InnoDB;
37     -- CREATED BY: SAGAR YADAV
38 # Creating Tables with thier Structures - Table: Bookings
39 CREATE TABLE Bookings (
40     Book_id int primary key AUTO_INCREMENT,
41     RoomNo int references Rooms(RoomNo),
42     DateOfBooking date Not null,
43     DateOfLeaving date default null,
44     Advance int DEFAULT '00',
45     CustomerID varchar(50) Not null references customers.CustomerID,
46     CustomerName varchar(50) Not Null
47     AUTO_INCREMENT = 101 ENGINE=InnoDB;
48     -- CREATED BY: SAGAR YADAV
49 # Creating Tables with thier Structures - Table Hotel_Details
50 CREATE TABLE Hotel_Details (
51     ID int primary key,
52     Field_name varchar(30) Not NULL,
53     Entry varchar(100) not NULL
54     ENGINE=InnoDB;
55     -- CREATED BY: SAGAR YADAV
56 # Dumping data for table `Hotel_Details`
57 INSERT INTO Hotel_Details (ID, Field_name, Entry) VALUES
58 (1, 'Hotel_Name', "The Taj Hotel"),
59 (2, 'Address', 'ABC Sector, Delhi -22'),
60 (3, 'Phone', '011-8000111,12,13'),
61 (4, 'Email', 'thetajhotel_delhi@gmail.com'),
62 (5, 'GST', '42'),
63 (6, 'ST', '16');
64     -- CREATED BY: SAGAR YADAV
65 # Table structure for table Bills
66 CREATE TABLE Bills (
67     Bill_id int NOT NULL AUTO_INCREMENT,
68     Book_id int DEFAULT NULL references Bookings.Book_id,
69     RoomNo int DEFAULT Null references rooms.RoomNo,
70     Bill_date date DEFAULT NULL,
71     GST int DEFAULT 42,
72     ST int DEFAULT 26,
73     Amount decimal DEFAULT NULL,
74     -- CREATED BY: SAGAR YADAV
75     PRIMARY KEY (bill_id)
76     ENGINEN=InnoDB AUTO_INCREMENT = 101;
```

Follow & Connect with me on :



SAGAR YADAV

77



```
File Edit Selection View Go Run Terminal Help The_Taj_Hotel.sql - Complete project - Visual Studio Code
EXPLORER ... The_Taj_Hotel.sql x
Main files > The_Taj_Hotel.sql
59 (2, 'Address', 'ABC Sector, Delhi -22'),
60 (3, 'Phone', '011-8000111,12,13'),
61 (4, 'Email', 'thetajhotel_delhi@gmail.com'),
62 (5, 'GST', '42'),
63 (6, 'ST', '16');
64
65 -- CREATED BY: SAGAR YADAV
66 # Table structure for table Bills
67 CREATE TABLE Bills (
68     Bill_id int NOT NULL AUTO_INCREMENT,
69     Book_id int DEFAULT NULL references Bookings.Book_id,
70     RoomNo int DEFAULT Null references rooms.RoomNo,
71     Bill_date date DEFAULT NULL,
72     GST int DEFAULT 42,
73     ST int DEFAULT 26,
74     Amount decimal DEFAULT NULL,
75     -- CREATED BY: SAGAR YADAV
76     PRIMARY KEY (bill_id)
77     ENGINE=InnoDB AUTO_INCREMENT = 101;
78     -- CREATED BY: SAGAR YADAV
79
80 # DROP database The_Taj_Hotel;
81 -- CREATED BY: SAGAR YADAV
82 commit
83
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF SQL ⚙️ 🔍

Follow & Connect with me on :





Follow & Connect with me on :





Running of Program(Output)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/powershell>

PS C:\Users\Sagar Yadav> python -u "c:\Users\Sagar Yadav\Desktop\Hotel Management System Project\Hotel_Management_System by Sagar yadav.py"

```
DATE : 2022-10-14
~~~~~ HOTEL DETAILS ~~~~~

Hotel Name : The Taj Hotel
Address : ABC Sector, Delhi -22
Phone No. : 011-8000111,12,13
Email : thetajhotel_delhi@gmail.com
Current GST : 42
Current ST : 16

Thanks for using....
Created by : SAGAR YADAV

Enter any key to continue..... >
```

HOTEL MANAGEMENT SYSTEM

1. Add New Room
2. Add Customer
3. Modify Room Information
4. Modify Customer Information
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Add New Room - Screen

Enter Room No.(Starts from 101) :> 101

For Room no. 101 -->
Available Room Types -
1. AC
2. Non-AC
3. DELUX
4. Super Delux
5. Queen Delight
6. Kings Special
7. Super Rich Special
Select your choice:> 1

Bed Selection ----- Menu

For Room no. 101 -->

Available Room Bed Types are:
1. Single
2. Double
3. Triple
Select your bed choice :> 1

Rent ----- Menu

Available room rents for "Single bed" are:
1. AC : ₹3499/night
2. Non-AC : ₹1499/night
3. Delux : ₹4799/night
4. Super Delux : ₹5499/night
5. Queen Delight : ₹6999/night
6. King Special : ₹8999/night
7. Super Rich Special : ₹10499/night

NOTE: Rent changes as per bed there is addition of Rs.1000 per bed(type)

Press Enter to continue....



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
6. Kings Special
7. Super Rich Special
Select your choice:> 1

Bed Selection ----- Menu
For Room no. 101 -->

Available Room Bed Types are:
1. Single
2. Double
3. Triple
Select your bed choice :> 1

Rent ----- Menu
Available room rents for "Single bed" are:
1. AC : ₹3499/night
2. Non-AC : ₹1499/night
3. Delux : ₹4799/night
4. Super Delux : ₹5499/night
5. Queen Delight : ₹6999/night
6. King Special : ₹8999/night
7. Super Rich Special : ₹10499/night
NOTE: Rent changes as per bed there is addition of Rs.1000 per bed(type)

Press Enter to continue.....

For ----->
Room Number : 101
Room Bed Type : Single bed
Room Type : AC
Your Rent will be (in Rs) : 3499

Room added successfully.....

Thanks for using....
Created by : SAGAR YADAV

Press Enter to continue.....
Ln 603, Col 17  Spaces:2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂  ⌂

② 0 Δ 0 PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Add New Room - Screen
-----
Enter Room No.(Starts from 101) :> 102

For Room no. 102 -->
Available Room Types -
1. AC
2. Non-AC
3. DELUX
4. Super Delux
5. Queen Delight
6. Kings Special
7. Super Rich Special
Select your choice:> 7

Bed Selection ----- Menu
For Room no. 102 -->

Available Room Bed Types are:
1. Single
2. Double
3. Triple
Select your bed choice :> 1

Rent ----- Menu
Available room rents for "Single bed" are:
1. AC : ₹3499/night
2. Non-AC : ₹1499/night
3. Delux : ₹4799/night
4. Super Delux : ₹5499/night
5. Queen Delight : ₹6999/night
6. King Special : ₹8999/night
7. Super Rich Special : ₹10499/night
NOTE: Rent changes as per bed there is addition of Rs.1000 per bed(type)

Press Enter to continue.....
Ln 603, Col 17  Spaces:2  UTF-8  CRLF  Python  3.10.6 64-bit  ⌂  ⌂
② 0 Δ 0
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

6. Kings Special
7. Super Rich Special
Select your choice:> 7

Bed Selection ----- Menu

For Room no. 102 -->

Avialable Room Bed Types are:

1. Single
2. Double
3. Triple

Select your bed choice :> 1

Rent ----- Menu

Available room rents for "Single bed" are:

1. AC	: ₹499/night
2. Non-AC	: ₹1499/night
3. Delux	: ₹4799/night
4. Super Delux	: ₹5499/night
5. Queen Delight	: ₹6999/night
6. King Special	: ₹8999/night
7. Super Rich Special	: ₹10499/night

NOTE: Rent changes as per bed there is addition of Rs.1000 per bed(type)

Press Enter to continue....

For ---->
Room Number : 102
Room Bed Type : Single bed
Room Type : Super Rich Special
Your Rent will be (in Rs) : 10499

Room added successfully....

Thanks for using....
Created by : SAGAR YADAV

Press Enter to continue....

Ln 603, Col 17 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Add New Room - Screen

Enter Room No.(Starts from 101) :> 101
NOTE: Room No 101 already exists in our database

Enter Another Room No. :> 104

For Room no. 104 -->

Available Room Types -

1. AC
2. Non-AC
3. DELUX
4. Super Delux
5. Queen Delight
6. Kings Special
7. Super Rich Special

Select your choice:> 5

Bed Selection ----- Menu

For Room no. 104 -->

Avialable Room Bed Types are:

1. Single
2. Double
3. Triple

Select your bed choice :> 1

Rent ----- Menu

Available room rents for "Single bed" are:

1. AC	: ₹499/night
2. Non-AC	: ₹1499/night
3. Delux	: ₹4799/night
4. Super Delux	: ₹5499/night
5. Queen Delight	: ₹6999/night
6. King Special	: ₹8999/night
7. Super Rich Special	: ₹10499/night

NOTE: Rent changes as per bed there is addition of Rs.1000 per bed(type)

Press Enter to continue....

Ln 603, Col 17 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit



The screenshot shows two terminal windows side-by-side. The top window displays the Hotel Management System menu:

```
HOTEL MANAGEMENT SYSTEM
-----
1. Add New Room
2. Add Customer
3. Modify Room Information
4. Modify Customer Information
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 2
```

The bottom window shows the process of adding a new customer:

```
Add New Customer - Screen
-----
Enter Customer Name :: Sagar yadav
Enter Customer Address:: ABCDEFGH
Enter Customer Contact No. :: 1122334455
Enter Customer Email ID :shivayasagar04@gmail.com
Enter Customer ID Proof type (e.g,Aadhar/Passport/DL/VoterID) :: Aadhar
Enter Customer ID proof No. :: 1234-5678-9000
Enter Total Males ::1
Enter Total Females :: 0
Enter Total Children :: 0

Customer Added successfully .....
Customer Details: --
Customer No.: 101
Name : Sagar yadav
Address : ABCDEFGH
Contact No. : 1122334455
Email : SHIVAYASAGAR04@GMAIL.COM
ID Proof : Aadhar
ID Proof No. : 1234-5678-9000
Males : 1
Female : 0
Children : 0

Thanks for using...
Created by : SAGAR YADAV
```



SAGAR YADAV
HOTEL MANAGEMENT SYSTEM



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
```

Add New Customer - Screen

```
Enter Customer Name ::> Utkarsh
Enter Customer Address::> xyz
Enter Customer Contact No. ::> 1234567890
Enter Customer Email ID :asasasasa@notmail.com
Enter Customer ID Proof type (e.g,Aadhar/Passport/DL/VoterID) ::> Passport
Enter Customer ID proof No. ::> UTYKA1111
Enter Total Males ::>1
Enter Total Females ::> 0
Enter Total Children ::> 0

Customer Added successfully .....
Customer Details ---
Customer No. : 102
Name : Utkarsh
Address : XYZ
Contact No. : 1234567890
Email : ASASASAS@NOTMAIL.COM
ID Proof : PASSPORT
ID Proof No. : UTYKA1111
Males : 1
Female : 0
Children : 0
```

```
Ln 603, Col 17 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
```

```
Enter Customer Contact No. ::> 1231231230
Enter Customer Email ID :notmail@mail.com
Enter Customer ID Proof type (e.g,Aadhar/Passport/DL/VoterID) ::> VoterID
Enter Customer ID proof No. ::> DEV7410
Enter Total Males ::>1
Enter Total Females ::> 0
Enter Total Children ::> 0

Customer Added successfully .....
Customer Details ---
Customer No. : 103
Name : Devanshu
Address : QWERTY
Contact No. : 1231231230
Email : NOTMAIL@MAIL.COM
ID Proof : VOTERID
ID Proof No. : DEV7410
Males : 1
Female : 0
Children : 0

Thanks for using....
Created by : SAGAR YADAV

Press Enter to continue...
```

```
Ln 603, Col 17 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
```

```
Room Booking ----- Menu
*****
Enter Room No. ::> 102
Details for the Room No. 102
Room No. : 102
Room Type : SUPER RICH SPECIAL
Room Rent : 10499.0
Room Bedding : SINGLE BED
Enter Customer Id or No. ::> 101
Enter date of occupancy (yyyy-mm-dd) ::> 2022-10-10
Enter advance amount ::> 11000

Room no 102 a SUPER RICH SPECIAL Room is booked for---
Customer No. : 101
Customer Name : Sagar yadav

Press Enter to continue...|
```

```
Ln 603, Col 17 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂
```

Follow & Connect with me on :





```
Room Booking ----- Menu
*****
Enter Room No. :> 103

Details for the Room No. 103
Room No. : 103
Room Type : KING SPECIAL
Room Rent : 8499.0
Room Bedding : SINGLE BED
Enter Customer Id on No. :> 104
Enter date of occupancy (yyyy-mm-dd) :> 2022-09-29
Enter advance amount :> 15000

Room no 103 a KING SPECIAL Room is booked for---
Customer No. : 104
Customer Name : Prashant

Press Enter to continue....|
```



```
File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar.yadav.py - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 6
BILL GENERATION ----- SCREEN
*****
Enter Room No. For which you want to generate bill :> 102
Room Detail or Booking Details
-----
Room No. : 102
Room Name : SUPER RICH SPECIAL
Room Bedding : SINGLE BED
Room Rent(Per night) : 8499.0
Customer ID : 101
Customer Name : Sagar yadav
Customer Address : ABCDEFGH
Advance paid : 11000

Press Enter to continue....|
```



```
Bill Generation
-----
Room occupied : 102
Date of Occupancy : 2022-10-10
Date of Leaving : 2022-10-15
Total Payable Days : 5
Room Rent Per Day : 8499.0
Total Amount : 42495.0
Advance : 11000
GST : 42
Service Tax : 16
Amount Payable : 71942.1
Bill Generated successfully....
This room is Free now, Room No.: 102

Press Enter to continue....|
```

master ⌘ 0 △ 0 Ln 1, Col 1 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂



```
HOTEL MANAGEMENT SYSTEM
-----
1. Add New Room
2. Add Customer
3. Modify Room Information
4. Modify Customer Information
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 7
```

```
Search Menu
-----
1.Room Status
2.Booking Status
3.Customer Details
4.Bills
5.Back to Main Menu

Enter your choice ...: 1

Search Menu
-----
1.Room Status
2.Booking Status
3.Customer Details
4.Bills
5.Back to Main Menu

Enter your choice ...: 1
Enter Room No :101

-----
Room Info ---
Room No. : 101
Room Type : AC
Room Rent : 3499.0
Room Bed : SINGLE BED
Room Status : Free
Booked for Customer No. : None

Press Enter to continue.....|
```





The screenshot shows two instances of Visual Studio Code running side-by-side. Both instances have the title bar "File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar_yadav.py - Visual Studio Code". The left instance displays the search results for a customer named "Sagar yadav". The right instance shows a menu system and bill information for Bill No 101.

Search Result for CustomerName = Sagar yadav

```
*****  
Customer ID : 101  
Customer Name : Sagar yadav  
Address : ABCDEFGH  
Contact No. : 1122334455  
Email : SHIVAYASAGAR04@GMAIL.COM  
ID Proof : Aadhar  
ID Proof No. : 1234-5678-9000  
*****
```

Thanks for using....
Created by : SAGAR YADAV

Press Enter to continue.....

master* File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar_yadav.py - Visual Studio Code Ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Search Menu

```
*****  
1.Room Status  
2.Booking Status  
3.Customer Details  
4.Bills  
5.Back to Main Menu  
*****
```

Enter your choice ...: 4
Enter Bill No :101

Bill information for Bill No :101

```
*****  
Bill Number : 101  
Bill's Total Amount : Rs. 71942  
Bill Date : 2022-10-15  
GST Charged : Rs. 42  
Service Tax Charged : Rs. 16  
Booking ID : 101  
Room no. Used : 102  
Room Name or type : SUPER RICH SPECIAL  
Room Bedding : SINGLE BED  
Room Rent(per night) : Rs. 10409.0  
Date of Occupancy : 2022-10-10  
Date of Leaving : 2022-10-15  
Advance Paid : Rs. 11000  
Customer's ID : 101  
Customer's Name : Sagar yadav  
Customer's Address : ABCDEFGH  
Customer's Phone : 1122334455  
Customer's Email ID : SHIVAYASAGAR04@GMAIL.COM  
Customer's ID Proof : Aadhar  
Customer's ID Proof No. : 1234-5678-9000  
*****
```

Press Enter to continue.....

master* File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar_yadav.py - Visual Studio Code Ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit



SAGAR YADAV
HOTEL MANAGEMENT SYSTEM



```
File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar_yadav.py - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
HOTEL MANAGEMENT SYSTEM
-----
1. Add New Room
2. Add Customer
3. Modify Room Information
4. Modify Customer Information
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 8

Report Menu
*****
1.Room Status
2.Booking Status
3.Bill Collections
4.Back to Main Menu

Enter your choice ...: 1
Ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⚡ 🌐

Rooms Status - Report
-----
Room No    Room Type     Rent      Bedding   Room Status   Booked for CustomerID
101        AC           3499.00   SINGLE BED  Free          None
102        SUPER RICH SPECIAL 10499.00  SINGLE BED  free          101
103        KING SPECIAL   8499.00   SINGLE BED  Occupied     104
104        QUEEN DELIGHT    6999.00   SINGLE BED  Free          None

Thanks for using...
Created by : SAGAR YADAV

Press Enter to continue...| Ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⚡ 🌐

Report Menu
*****
1.Room Status
2.Booking Status
3.Bill Collections
4.Back to Main Menu

Enter your choice ...: 2
Ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⚡ 🌐

Booking Status - Report
-----
BookID  RoomNo    DOO      DOL      Advance  CustomerName  Address  Phone
101     102       2022-10-10 2022-10-15 11000.00  Sagar_yadav  ABCDEFGH  1122334455
102     103       2022-09-29  None      15000.00  Prashant     QMERTZ    1234567891

Thanks for using...
Created by : SAGAR YADAV

Press Enter to continue...| Ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⚡ 🌐
```

Follow & Connect with me on :



SAGAR YADAV

89



```
Report Menu
*****
1.Room Status
2.Booking Status
3.Bill Collections
4.Back to Main Menu

Enter your choice ...: 3

Bill Collections ----- Menu
*****
1.Total Bill collection
2.Collection from individual customer
3.Back to Report menu

Enter your choice from above (1 or 2) :> 1

Total Collection
-----
Total Collections we have, Rs. 71942

Press Enter to continue...|
```

ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂

```
Bill Collections ----- Menu
*****
1.Total Bill Collection
2.Collection From individual customer
3.Back to Report menu

Enter your choice from above (1 or 2) :> 2

Collection from individual customer
-----
Enter Customer No. ::> 101
Rows : 1
Collections from CustomerNo. 101 we have, Rs. 71942

Thanks for using...
Created by : SAGAR YADAV

Press Enter to continue...|
```

ln 749, Col 50 Spaces:2 UTF-8 CRLF Python 3.10.6 64-bit ⌂ ⌂



SAGAR YADAV
HOTEL MANAGEMENT SYSTEM



```
File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar.yadav.py - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
3. Modify Room Information
4. Modify Customer Information
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 9

DATE : 2022-10-15
~~~~~ HOTEL DETAILS ~~~~~
Hotel Name : The Taj Hotel
Address : ABC Sector, Delhi -22
Phone No. : 011-8000111,12,13
Email : thetajhotel_delhi@gmail.com
Current GST : 42
Current ST : 16

Thanks for using...
Created by : SAGAR YADAV
Enter any key to continue..... >

File Edit Selection View Go Run Terminal Help Hotel_Management_System by Sagar.yadav.py - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
HOTEL MANAGEMENT SYSTEM
-----
1. Add New Room
2. Add Customer
3. Modify Room Information
4. Modify Customer Information
5. Room Booking
6. Bill Generation
7. Search Database
8. Report Menu
9. Hotel Details
10. Change Hotel Details
11. Close application

Enter your choice ...: 11

Thanks for using...
Created by : SAGAR YADAV

Thanks for using...
Created by : SAGAR YADAV
PS C:\Users\Sagar.Yadav>
Ln 749, Col 50 Spaces: 2 UTF-8 CRLF Python 3.10.6 64-bit
```

Follow & Connect with me on :



SAGAR YADAV 91



Database after running Program(Hotel Management system by Sagar yadav.py)

```
MySQL 8.0 Command Line Client

mysql> select * from customers;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| CustomerID | CustomerName | Address | Contact_No | Email | ID_Proof | Id_Number | Males | Females | Children |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 101 | Sagar yadav | ABCDEFGH | 1122334455 | SHIVAYASAGR04@GMAIL.COM | AADHAR | 1234-5678-9000 | 1 | 0 | 0 |
| 102 | Utkarsh | XYZ | 1234567890 | ASASASASA@NOTMAIL.COM | PASSPORT | UTYKA1111 | 1 | 0 | 0 |
| 103 | Devanshu | QWERTY | 1231231230 | NOTMAIL@MAIL.COM | VOTERID | DEV7410 | 1 | 0 | 0 |
| 104 | Prashant | QWERTZ | 1234567891 | 1234P@SAMPLW.COM | AADHAR | 1470-0000-0000 | 1 | 0 | 0 |
| 105 | Ankit | ABCDEF | 2222211111 | SAMPLE@MAIL.COM | AADHAR | 1470-1470-0000 | 1 | 0 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> select * from rooms;
+-----+-----+-----+-----+-----+
| RoomNo | Room_Type | Rent | Bed_Type | Room_Status | CustomerID |
+-----+-----+-----+-----+-----+
| 101 | AC | 3499 | SINGLE BED | Free | NULL |
| 102 | SUPER RICH SPECIAL | 10499 | SINGLE BED | free | 101 |
| 103 | KING SPECIAL | 8499 | SINGLE BED | Occupied | 104 |
| 104 | QUEEN DELIGHT | 6999 | SINGLE BED | Free | NULL |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from bookings;
+-----+-----+-----+-----+-----+-----+-----+
| Book_id | RoomNo | DateOfBooking | DateOfLeaving | Advance | CustomerID | CustomerName |
+-----+-----+-----+-----+-----+-----+-----+
| 101 | 102 | 2022-10-10 | 2022-10-15 | 11000 | 101 | Sagar yadav |
| 102 | 103 | 2022-09-29 | NULL | 15000 | 104 | Prashant |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from bills;
+-----+-----+-----+-----+-----+-----+
| Bill_id | Book_id | RoomNo | Bill_date | GST | ST | Amount |
+-----+-----+-----+-----+-----+-----+
| 101 | 101 | 102 | 2022-10-15 | 42 | 16 | 71942 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```