```
import mysql.connector
def connect_db():
  try:
    connection = mysql.connector.connect(
      host="localhost",
      user="root",
      password="password",
      database="motel_management"
    )
    return connection
  except mysql.connector.Error as err:
    print(f"Error: {err}")
    return None
connection = connect_db()
if connection:
  cursor = connection.cursor()
  try:
    cursor.execute('''
    CREATE TABLE IF NOT EXISTS Rooms (
      RoomID INT AUTO_INCREMENT PRIMARY KEY,
      RoomNumber VARCHAR(10) NOT NULL UNIQUE,
      RoomType VARCHAR(50) NOT NULL,
      PricePerNight DECIMAL(10, 2) NOT NULL,
      IsAvailable BOOLEAN DEFAULT TRUE
    );
    ''')
    print("Rooms table created or already exists.")
    cursor.execute('''
    CREATE TABLE IF NOT EXISTS Guests (
      GuestID INT AUTO_INCREMENT PRIMARY KEY,
      FirstName VARCHAR(100) NOT NULL,
      LastName VARCHAR(100) NOT NULL,
      PhoneNumber VARCHAR(15),
      Email VARCHAR(100),
      Address TEXT,
      RoomID INT,
      FOREIGN KEY (RoomID) REFERENCES Rooms(RoomID) ON DELETE SET NULL
    );
    ''')
    print("Guests table created or already exists.")
    cursor.execute('''
      CREATE TABLE IF NOT EXISTS Bookings (
        BookingID INT AUTO_INCREMENT PRIMARY KEY,
        GuestID INT NOT NULL,
        RoomID INT NOT NULL,
        CheckInDate DATE NOT NULL,
        CheckOutDate DATE NOT NULL,
        TotalAmount DECIMAL(10, 2) NOT NULL,
```

```
            FOREIGN KEY (GuestID) REFERENCES Guests(GuestID) ON DELETE CASCADE,
            FOREIGN KEY (RoomID) REFERENCES Rooms(RoomID) ON DELETE CASCADE
        );
        ''')
    connection.commit()
    connection.close()
    print("Bookings table created successfully!")
  except mysql.connector.Error as err:
    print(f"Error creating table: {err}")
def insert_rooms():
  connection = connect_db()
  if connection:
    cursor = connection.cursor()
    try:
      cursor.executemany('''
      INSERT INTO Rooms (RoomNumber, RoomType, PricePerNight, IsAvailable)
      VALUES (%s, %s, %s, %s)
      ON DUPLICATE KEY UPDATE RoomType=VALUES(RoomType);
      ''', [
        ('101', 'Deluxe', 150.00, True),
        ('102', 'Special', 200.00, True),
        ('103', 'Common', 100.00, True),
        ('104', 'Deluxe', 150.00, True),
        ('105', 'Special', 200.00, True),
        ('106', 'Common', 100.00, True),
        ('107', 'Suite', 300.00, True),
        ('108', 'Penthouse', 500.00, True)
      ])
      connection.commit()
      print("Rooms inserted successfully!")
    except mysql.connector.Error as err:
      print(f"Error inserting rooms: {err}")
      raise
def main1():
  try:
    insert_rooms()
  except Exception as e:
    print(f"Error occurred during data insertion: {e}")
main1()
def insert_new_guest():
  connection = connect_db()
  if connection:
    cursor = connection.cursor()
    try:
      first_name = input("Enter guest's first name: ")
      last_name = input("Enter guest's last name: ")
      phone = input("Enter guest's phone number: ")
      email = input("Enter guest's email: ")
```

```
        address = input("Enter guest's address: ")
        room_number = input("Enter Room Number to assign (101-110): ")
        cursor.execute("SELECT RoomID, IsAvailable FROM Rooms WHERE RoomNumber = %s",
(room_number,))
        room = cursor.fetchone()
        if room:
            room_id, is_available = room
            if is_available:
                cursor.execute('''
                INSERT INTO Guests (FirstName, LastName, PhoneNumber, Email, Address,
RoomID)
                VALUES (%s, %s, %s, %s, %s, %s)
                ''', (first_name, last_name, phone, email, address, room_id))
                cursor.execute("UPDATE Rooms SET IsAvailable = FALSE WHERE RoomID = %s",
(room_id,))
                connection.commit()
                print(f"Guest {first_name} {last_name} added and assigned to Room Number
{room_number}.")
            else:
                print(f"Room {room_number} is already occupied.")
        else:
            print(f"Room {room_number} does not exist.")
    except mysql.connector.Error as err:
        print(f"Error: {err}")
def total_rooms_occupied():
  connection = connect_db()
  if connection:
    cursor = connection.cursor()
    try:
      cursor.execute("SELECT COUNT(*) FROM Rooms WHERE IsAvailable = FALSE")
      occupied_rooms = cursor.fetchone()[0]
      print(f"Total rooms occupied: {occupied_rooms}")
    except mysql.connector.Error as err:
      print(f"Error: {err}")
def total_bill_of_guest():
  connection = connect_db()
  if connection:
    cursor = connection.cursor()
    try:
      guest_id = int(input("Enter the Guest ID: "))
      cursor.execute('''
      SELECT SUM(TotalAmount)
      FROM Bookings
      WHERE GuestID = %s
      ''', (guest_id,))
      total_bill = cursor.fetchone()[0]
      if total_bill:
        print(f"Total bill for Guest ID {guest_id}: ${total_bill:.2f}")
```

```
            else:
                print("No booking details found for the given Guest ID.")
        except mysql.connector.Error as err:
            print(f"Error: {err}")
def display_guests():
    connection = connect_db()
    if connection:
        cursor = connection.cursor()
        try:
            query = '''
            SELECT Guests.GuestID, Guests.FirstName, Guests.LastName, Guests.PhoneNumber,
                Guests.Email, Guests.Address, Rooms.RoomNumber
            FROM Guests
            LEFT JOIN Rooms ON Guests.RoomID = Rooms.RoomID
            '''
            cursor.execute(query)
            rows = cursor.fetchall()
            if rows:
                print("\n--- Guests Table ---")
                print(f"{'GuestID':<10}{'FirstName':<15}{'LastName':<15}{'PhoneNumber':<15}"
                    f"{'Email':<25}{'Address':<25}{'RoomNumber':<10}")
                print("-" * 100)
                for row in rows:
                    print(f"{row[0]:<10}{row[1]:<15}{row[2]:<15}{row[3]:<15}"
                        f"{row[4]:<25}{row[5]:<25}{row[6]:<10}")
            else:
                print("No guests found in the database.")
        except mysql.connector.Error as err:
            print(f"Error: {err}")
def mark_room_available():
    connection = connect_db()
    if connection:
        cursor = connection.cursor()
        try:
            display_guests()
            guest_id = input("\nEnter the GuestID of the guest checking out: ")
            cursor.execute("SELECT RoomID FROM Guests WHERE GuestID = %s", (guest_id,))
            result = cursor.fetchone()
            if not result:
                print(f"No guest found with GuestID {guest_id}.")
                return
            room_id = result[0]
            cursor.execute("UPDATE Rooms SET IsAvailable = TRUE WHERE RoomID = %s",
(room_id,))
            cursor.execute("DELETE FROM Guests WHERE GuestID = %s", (guest_id,))
            connection.commit()
            print(f"RoomID {room_id} is now available, and GuestID {guest_id} has been
removed.")
```

```
        except mysql.connector.Error as err:
            print(f"Error: {err}")
def display_guests1():
    connection = connect_db()
    if connection:
        cursor = connection.cursor()
        try:
            query = '''
            SELECT Guests.GuestID, Guests.FirstName, Guests.LastName, Guests.PhoneNumber,
                Guests.Email, Guests.Address, Rooms.RoomNumber
            FROM Guests
            LEFT JOIN Rooms ON Guests.RoomID = Rooms.RoomID
            '''
            cursor.execute(query)
            rows = cursor.fetchall()
            if rows:
                print("\n--- Guests Table ---")
                print(f"{'GuestID':<10}{'FirstName':<15}{'LastName':<15}{'PhoneNumber':<15}"
                    f"{'Email':<25}{'Address':<25}{'RoomNumber':<10}")
                print("-" * 100)
                for row in rows:
                    print(f"{row[0]:<10}{row[1]:<15}{row[2]:<15}{row[3]:<15}"
                        f"{row[4]:<25}{row[5]:<25}{row[6]:<10}")
            else:
                print("No guests found in the database.")
        except mysql.connector.Error as err:
            print(f"Error: {err}")
def display_rooms():
    connection = connect_db()
    if connection:
        cursor = connection.cursor()
        try:
            cursor.execute("SELECT * FROM Rooms")
            rows = cursor.fetchall()
            if rows:
                print("\n--- Rooms Table ---")
                print(f"{'RoomID':<10}{'RoomNumber':<15}{'RoomType':<15}{'PricePerNight':<15}
{'IsAvailable':<15}")
                print("-" * 70)
                for row in rows:
                    print(f"{row[0]:<10}{row[1]:<15}{row[2]:<15}{row[3]:<15}{'Yes' if row[4] else
'No':<15}")
            else:
                print("No rooms found in the database.")
        except mysql.connector.Error as err:
            print(f"Error: {err}")
def menu():
    while True:
```

```
        print("\nMenu:")
        print("1. Insert new guest")
        print("2. Total rooms occupied")
        print("3. Total bill of a guest during checkout")
        print("4. Mark room available if guest checked out")
        print("5. Display Rooms Table")
        print("6. Display Guests Table")
        print("7. Exit")
        option = input("Enter your choice: ")
        if option == "1":
            insert_new_guest()
        elif option == "2":
            total_rooms_occupied()
        elif option == "3":
            total_bill_of_guest()
        elif option == "4":
            mark_room_available()
        elif option == "7":
            print("Exiting program.")
        elif option == "5":
                display_rooms()
        elif option == "6":
                display_guests1()
        else:
                print("Invalid choice. Please try again.")
                break
print("Invalid choice, please try again.")
print("Reminder there are only 10 rooms which is from 101 to 110")
menu()
```