**ARINDA HILLARY**

**DBMS**

**LAB 2**

**3/1/2024**

1. **List the price and type of all rooms at the Palace Hotel.**

SELECT h.hotelNo, h.hotelName, r.type, r.price

FROM Hotel h, Room r

WHERE h.hotelNo = r.hotelNo AND h.hotelName='Palace hotel'

A white rectangular object with a black stripe

Description automatically generated with medium confidence

1. **List all guests currently staying at the Palace Hotel.**Show full details of the guest including guestName and guestAddress.  
   (Hint: Use the GETDATE() function to get the current date)

SELECT g.guestName, g.guestAddress, h.hotelName

FROM Guest g, Booking b, Hotel h

WHERE h.hotelNo = b.hotelNo AND h.hotelName = 'Palace hotel' AND g.guestNo=b.guestNo AND (GETDATE() >= dateFrom AND (GETDATE() <= dateTo OR dateTo IS NULL))

A screenshot of a computer

Description automatically generated

The key assumption here is that if the dateTo is null, it most likely means that the guest has not checked out yet if they are checked in.

1. **List the details of all occupied rooms at the Palace Hotel today, including the name of the guest staying in the room.**

SELECT r.roomNo, r.type, r.price, g.guestName

FROM Room r

JOIN Booking b ON r.roomNo = b.roomNo AND r.hotelNo = b.hotelNo

JOIN Guest g ON b.guestNo = g.guestNo

JOIN Hotel h ON r.hotelNo = h.hotelNo

WHERE h.hotelName = 'Palace hotel' AND GETDATE() BETWEEN b.dateFrom AND b.dateTo

A screenshot of a computer

Description automatically generated

1. **List the details of all unoccupied rooms at the Palace Hotel today.**Show the room type and price as well.

SELECT r.roomNo, r.type, r.price

FROM Room r

WHERE r.hotelNo = (SELECT hotelNo FROM Hotel WHERE hotelName = 'Palace hotel') AND r.roomNo NOT IN (

SELECT b.roomNo

FROM Booking b

WHERE b.hotelNo = r.hotelNo AND GETDATE() BETWEEN b.dateFrom AND b.dateTo

);

A screenshot of a computer

Description automatically generated

1. **What is the lost income for each hotel?**(use current date)

SELECT h.hotelNo, h.hotelName, SUM(r.price) AS lostIncome

FROM Hotel h

JOIN Room r ON h.hotelNo = r.hotelNo

LEFT JOIN Booking b ON r.roomNo = b.roomNo AND b.dateFrom <= GETDATE() AND b.dateTo >= GETDATE()

GROUP BY h.hotelNo, h.hotelName;

A screenshot of a computer

Description automatically generated

1. **What is the most commonly booked room type for each hotel in Fairfield over the whole history of the hotel? Include the number of bookings in the report.**

SELECT h.hotelName, r.Type, COUNT(\*) AS NumberOfBookings FROM Hotel h

JOIN Room r ON h.hotelNo = r.hotelNo

JOIN Booking b ON r.roomNo = b.roomNo

WHERE h.City LIKE '%Fairfield%'

GROUP BY h.hotelName, r.Type

ORDER BY NumberOfBookings DESC

A screenshot of a computer

Description automatically generated

1. **Write SQL statements to Insert rows into each of the tables used for this lab.**(It’ll suffice if you show only 2 rows insertion)

INSERT INTO Hotel (hotelNo, hotelName,City ) VALUES

('H7', 'Tempe Hotel', 'Tempe,AZ'),

('H8', 'Boston Hotel', 'Boston,MA');

INSERT INTO Room (RoomNo, HotelNo, Type, Price) VALUES

('H7R1', 'H7', 'S', 150),

('H7R2', 'H8', 'D', 600);

INSERT INTO Guest (guestNo, guestName, guestAddress) VALUES

('G6', 'Collin Arnold', '2000 N Court St Fairfield, IA'),

('G7', 'Samuel Leon', '389 Lettleton ave, Hingham, MA');

INSERT INTO Booking (hotelNo, guestNo, dateFrom, dateTo, roomNo) VALUES

('H7', 'G6', '2024-01-15', '2024-01-20', 'H7R1'),

('H7', 'G7', '2024-02-10', '2024-02-15', 'H7R2');

A screenshot of a computer

Description automatically generated

1. **Update the price of every room in the Palace Hotel by 5% more.**

UPDATE Room

SET price = price \* 1.05

WHERE HotelNo = (SELECT HotelNo FROM Hotel WHERE HotelName = 'Palace hotel')

A screenshot of a computer

Description automatically generated

1. **Create a separate table with the same structure as the Booking table to hold archive records. Using the INSERT statement, copy the records from the Booking table to the archive table relating to bookings before 1st January 2024.  Then delete all bookings before 1st January 2024 from the Booking table.**

CREATE TABLE BookingArchive ( hotelNo VARCHAR(10), guestNo VARCHAR(10), dateFrom DATE, dateTo DATE, roomNo VARCHAR(10)

)

INSERT INTO BookingArchive ( hotelNo, guestNo, dateFrom, dateTo, roomNo)

SELECT hotelNo, guestNo, dateFrom, dateTo, roomNo

FROM Booking

WHERE dateFrom < '2024-01-01'

DELETE FROM Booking

WHERE dateFrom < '2024-01-01'

A screenshot of a computer

Description automatically generated

1. **Consider a relational database with two tables: Orders and Customers. The Orders table contains information about orders placed by customers, while the Customers table stores details about each customer.**   
   **The Orders table has a foreign key constraint referencing the Customers table, ensuring referential integrity between the two tables. This referential integrity constraint is configured with an ON DELETE action of CASCADE.**  
   **What is the meaning of this action?**

In a database with Orders and Customers tables, ON DELETE CASCADE ensures that deleting a customer record also deletes associated orders, maintaining data integrity between the tables.