

Hotel Database – JOIN Queries

1. Display hotel ID, name, and the name of its manager.

The screenshot shows a MySQL command-line interface window. The query entered is:

```
DH8\user
4
5     SELECT
6         b.Branch_id,
7             b.name AS Hotel_name,
8             s.name AS Manager_name
9     FROM Branch b
10    LEFT JOIN Staff s
11        ON b.Branch_id = s.Branch_id
12    WHERE s.job = 'Manager';
13
14
15
16
17
```

The results pane is empty, indicating no data has been returned yet. The interface includes standard navigation buttons (100%, back, forward, search) and tabs for Results and Messages.

2. Display hotel names and the rooms available under them.

```

15
16     SELECT
17         b.name AS Hotel_name,
18         r.Room_num,
19         r.Room_type,
20         r.Nightly_rate
21     FROM Branch b
22     JOIN Room r
23         ON b.Branch_id = r.Branch_id
24     ORDER BY b.name, r.Room_num;
25

```

100 % 5 0 ↑ ↓

Results Messages

	Hotel_name	Room_num	Room_type	Nightly_rate
1	Airport Branch	201	Suite	165.00
2	Airport Branch	205	Suite	1200.00
3	Downtown Branch	101	Single	50.00
4	Downtown Branch	102	Double	80.00

3. Display guest data along with the bookings they made.

```

28
29     SELECT
30         c.Customer_id,
31         c.name AS Guest_name,
32         c.phone,
33         c.email,
34         b.Booking_id,
35         b.Check_in,
36         b.Checkout,
37         b.status,
38         b.total_cost
39     FROM Customer c
40     LEFT JOIN Booking b
41         ON c.Customer_id = b.Customer_id
42     ORDER BY c.Customer_id;

```

100 % 4 0 ↑ ↓

Results Messages

	Customer_id	Guest_name	phone	email	Booking_id	Check_in	Checkout	status	total_cost
1	1	Alice Smith	555-1234	alice@example.com	1	2025-12-20	2025-12-25	NULL	NULL
2	2	Bob Johnson	555-5678	bob@example.com	2	2025-12-24	2025-12-28	NULL	NULL
3	9011	malak	555-0000	mkmalaufi@gmail.com	1001	2025-12-20	2025-12-25	Confirmed	1200.00
4	9012	Guest Two	NULL	NULL	NULL	NULL	NULL	NULL	NULL

4. Display bookings for hotels in 'Hurghada' or 'Sharm El Sheikh'.

```
44
45     SELECT
46         bo.Booking_id,
47         c.name AS Guest_name,
48         r.Room_num,
49         r.Room_type,
50         b.name AS Hotel_name,
51         b.location
52     FROM Booking bo
53     -- Only include customers who have bookings → INNER JOIN
54     INNER JOIN Customer c
55         ON bo.Customer_id = c.Customer_id
56
57     -- Only include bookings that have rooms → INNER JOIN
58     INNER JOIN Booking_room br
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
```

100 % 4 0 ↑ ↓

Results Messages

Booking_id	Guest_name	Room_num	Room_type	Hotel_name	location
------------	------------	----------	-----------	------------	----------

5. Display all room records where room type starts with "S" (e.g., "Suite", "Single").

```
73
74     5. Display all room records where
75
76     SELECT *
77     FROM Room
78     WHERE Room_type LIKE 'S%';
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

100 % 5 0 ↑ ↓

Results Messages

	Room_num	Room_type	Nightly_rate	Branch_id
1	101	Single	50.00	1
2	201	Suite	165.00	2
3	205	Suite	1200.00	2

6. List guests who booked rooms priced between 1500 and 2500 LE.

```

-----p. List guests who booked rooms priced between 1500 and 2500

SELECT DISTINCT
    c.Customer_id,
    c.name AS Guest_name,
    c.phone,
    c.email
FROM Customer c
inner JOIN Booking b ON c.Customer_id = b.Customer_id
inner JOIN Booking_room br ON b.Booking_id = br.Booking_id
inner JOIN Room r ON br.Room_num = r.Room_num
WHERE r.Nightly_rate BETWEEN 1500 AND 2500;

```

The screenshot shows a database interface with a toolbar at the top featuring icons for close, minimize, maximize, and refresh, followed by a status bar showing '5' and '0'. Below is a tabs section with 'Results' and 'Messages' selected. The main area is a table with four columns: 'Customer_id', 'Guest_name', 'phone', and 'email'. The table is currently empty.

7. Retrieve guest names who have bookings marked as 'Confirmed' in hotel "Hilton Downtown".

```

94
95     SELECT DISTINCT
96         c.name AS Guest_name
97     FROM Customer c
98     -- Only include customers who have bookings → INNER JOIN
99     INNER JOIN Booking b ON c.Customer_id = b.Customer_id
100
101    -- Only include bookings linked to rooms → INNER JOIN
102    INNER JOIN Booking_room br ON b.Booking_id = br.Booking_id
103
104    -- Only include rooms assigned to bookings → INNER JOIN
105    INNER JOIN Room r ON br.Room_num = r.Room_num
106
107    -- Only include branches (hotels) for these rooms → INNER JOIN

```

The screenshot shows a database interface with a toolbar at the top featuring icons for close, minimize, maximize, and refresh, followed by a status bar showing '6' and '0'. Below is a tabs section with 'Results' and 'Messages' selected. The main area is a table with one column: 'Guest_name'. The value 'Guest_name' is listed once.

8. Find guests whose bookings were handled by staff member "Mona Ali".

```
116  SELECT DISTINCT
117      c.name AS Guest_name,
118      s.name AS Staff_name,
119      sa.Action_type,
120      sa.Action_date
121  FROM StaffAction sa
122  -- Include only staff linked to actions → INNER JOIN
123  INNER JOIN Staff s ON sa.Staff_id = s.Staff_id
124
125  -- Include only bookings linked to actions → INNER JOIN
126  INNER JOIN Booking b ON sa.Booking_id = b.Booking_id
127
128  -- Include only customers linked to bookings → INNER JOIN
129  INNER JOIN Customer c ON b.Customer_id = c.Customer_id
```

100 % x 7 ▲ 0 ↑ ↓

Results Messages

Guest_name	Staff_name	Action_type	Action_date
------------	------------	-------------	-------------

9. Display each guest's name and the rooms they booked, ordered by room type.

```

136  ||| SELECT
137      c.name AS Guest_name,
138          r.Room_num,
139          r.Room_type
140  FROM Customer c
141  -- Include only customers who have bookings → I
142  INNER JOIN Booking b ON c.Customer_id = b.Cust
143
144  -- Include only bookings that have rooms → INNE
145  INNER JOIN Booking_room br ON b.Booking_id = br
146
147  -- Include only rooms linked to the booking → I
148  INNER JOIN Room r ON br.Room_num = r.Room_num
149
150  ORDER BY ...

```

100 %

✖ 7 ⚠ 0 ↑ ↓

Results

Messages

	Guest_name	Room_num	Room_type
1	Alice Smith	101	Single
2	Bob Johnson	201	Suite
3	malak	101	Single
4	malak	205	Suite

10. For each hotel in 'Cairo', display hotel ID, name, manager name, and contact info.

```
155  
156     br.Branch_id,  
157     br.name AS Hotel_name,  
158     s.name AS Manager_name,  
159     s.salary,  
160     s.Staff_id  
161 FROM Branch br  
162 -- Include staff linked to the branch → LEFT JOIN  
163 LEFT JOIN Staff s  
164     ON br.Branch_id = s.Branch_id AND s.job = 'Manager'  
165 WHERE br.location = 'Cairo';  
166  
167  
168 11. Display all staff members who hold 'Manager'  
169 12. Display all guests and their reviews, even if some  
170  
171  
100 % 10 0 ↑ ↓
```

Results Messages

Branch_id	Hotel_name	Manager_name	salary	Staff_id
-----------	------------	--------------	--------	----------

11. Display all staff members who hold 'Manager' positions.

```
167  
168     SELECT *  
169     FROM Staff  
170     WHERE job = 'Manager';  
171  
100 % 11 0 ↑ ↓
```

Results Messages

Staff_id	name	job	salary	Branch_id
----------	------	-----	--------	-----------

12. Display all guests and their reviews, even if some guests haven't submitted any reviews.

```
173    CREATE TABLE Review (
174        Review_id INT PRIMARY KEY,
175        Customer_id INT,
176        Booking_id INT,
177        Review_text NVARCHAR(255),
178        Rating INT,
179        FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id),
180        FOREIGN KEY (Booking_id) REFERENCES Booking(Booking_id)
181    );
182
183    SELECT
184        c.name AS Guest_name,
185        r.Review_text,
```

100 % ▾ 10 ▲ 0 ↑ ↓

Results Messages

	Customer_name	Review_text	Rating
1	Alice Smith	NULL	NULL
2	Bob Johnson	NULL	NULL
3	malak	NULL	NULL
4	Guest Two	NULL	NULL