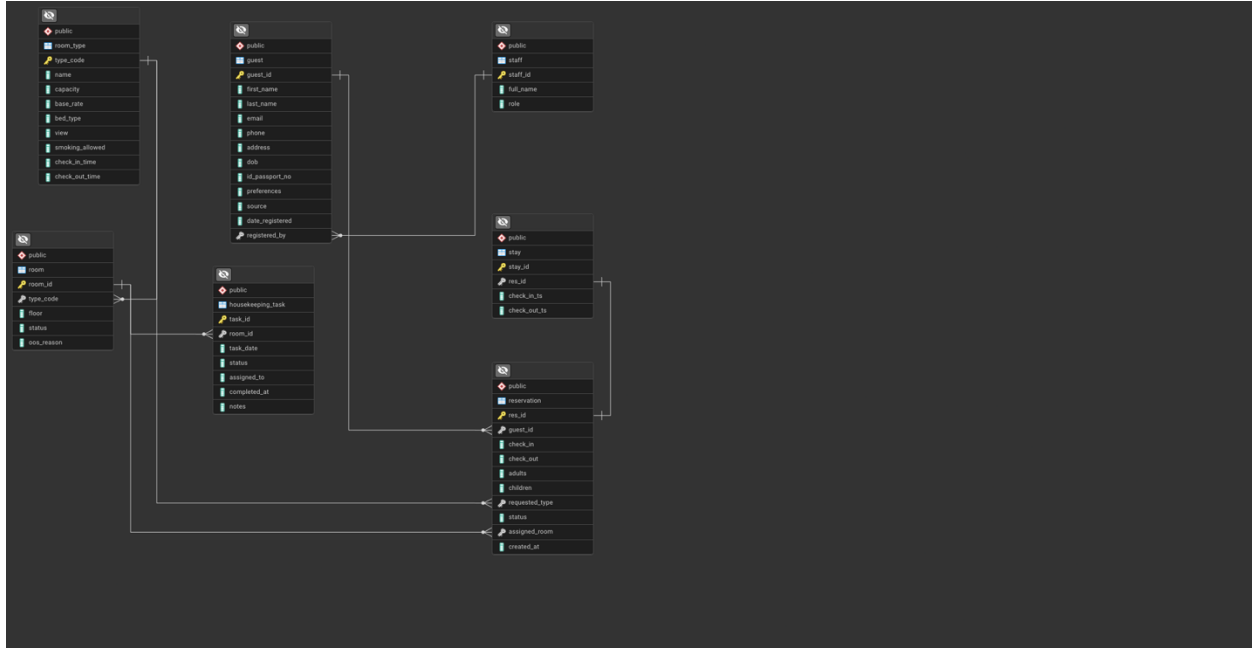# Project 6

6.A) **ER Diagram:**

6.B) **Fifteen queries:**

1) **Today's arrivals**

SELECT r.res_id,

    g.first_name || ' ' || g.last_name AS guest,

    r.adults + r.children AS party_size,

    r.requested_type,

    r.assigned_room

FROM reservation r

JOIN guest g ON g.guest_id = r.guest_id

WHERE r.check_in = CURRENT_DATE

 AND r.status IN ('BOOKED', 'IN_HOUSE')

ORDER BY r.check_in, r.created_at;

2) Dirty rooms with no pending housekeeping task today

```sql
SELECT rm.room_id, rm.type_code, rm.floor

FROM room rm

WHERE rm.status = 'DIRTY'

  AND NOT EXISTS (

    SELECT 1

    FROM housekeeping_task t

    WHERE t.room_id = rm.room_id

      AND t.task_date = CURRENT_DATE

      AND t.status = 'PENDING'

  )

ORDER BY rm.room_id;
```

3) BOOKED reservations starting in the next 3 days

```sql
SELECT r.res_id, g.first_name || ' ' || g.last_name AS guest,

      r.check_in, r.check_out

FROM reservation r

JOIN guest g ON g.guest_id = r.guest_id

WHERE r.status = 'BOOKED'

  AND r.check_in BETWEEN CURRENT_DATE AND CURRENT_DATE + INTERVAL '3 days'

ORDER BY r.check_in, r.created_at;
```

4) Room counts by type and status

```sql
SELECT rm.type_code, rm.status, COUNT(*) AS room_count

FROM room rm

GROUP BY rm.type_code, rm.status
```

ORDER BY rm.type_code, rm.status;

5) **Clean and vacant today rooms**

SELECT rm.room_id, rm.type_code

FROM room rm

WHERE rm.status = 'CLEAN'

  AND NOT EXISTS (

    SELECT 1

    FROM reservation r

    WHERE r.assigned_room = rm.room_id

     AND CURRENT_DATE >= r.check_in

     AND CURRENT_DATE < r.check_out

)

ORDER BY rm.room_id;

6) **DONE housekeeping tasks today per assignee**

SELECT t.assigned_to, COUNT(*) AS done_count

FROM housekeeping_task t

WHERE t.task_date = CURRENT_DATE

 AND t.status = 'DONE'

GROUP BY t.assigned_to

ORDER BY done_count DESC, t.assigned_to;

7) OOS rooms with reason

SELECT rm.room_id, rm.type_code, rm.oos_reason

FROM room rm

```
WHERE rm.status = 'OOS'

AND rm.floor > 1

ORDER BY rm.room_id;
```

8) **Guests with at least 2 completed stays**

```
SELECT g.guest_id, g.first_name || ' ' || g.last_name AS guest,

    COUNT(*) AS completed_stays

FROM reservation r

JOIN guest g ON g.guest_id = r.guest_id

WHERE r.status = 'COMPLETE'

GROUP BY g.guest_id, guest

HAVING COUNT(*) >= 2

ORDER BY completed_stays DESC, guest;
```

9) **Reservations where party size exceeds capacity**

```
SELECT r.res_id, r.assigned_room,

    (r.adults + r.children) AS party_size,

    rt.capacity

FROM reservation r

JOIN room rm   ON rm.room_id   = r.assigned_room

JOIN room_type rt ON rt.type_code = rm.type_code

WHERE (r.adults + r.children) > rt.capacity

ORDER BY (r.adults + r.children - rt.capacity) DESC;
```

10) **Same-day turnovers**

```
SELECT out_r.assigned_room AS room_id,
```

```sql
    g1.first_name || ' ' || g1.last_name AS checkout_guest,

    g2.first_name || ' ' || g2.last_name AS checkin_guest

FROM reservation out_r

JOIN reservation in_r

  ON in_r.assigned_room = out_r.assigned_room

 AND in_r.check_in  = CURRENT_DATE

 AND out_r.check_out = CURRENT_DATE

JOIN guest g1 ON g1.guest_id = out_r.guest_id

JOIN guest g2 ON g2.guest_id = in_r.guest_id

ORDER BY room_id;
```

11) Cancelled reservations in the last 30 days

```sql
SELECT DATE(r.created_at) AS day, COUNT(*) AS cancelled_count

FROM reservation r

WHERE r.status = 'CANCELLED'

  AND r.created_at >= CURRENT_DATE - INTERVAL '30 days'

GROUP BY DATE(r.created_at)

ORDER BY day;
```

12) Rooms sold today by type

```sql
SELECT rm.type_code, COUNT(DISTINCT rm.room_id) AS rooms_sold_today

FROM reservation r

JOIN room rm ON rm.room_id = r.assigned_room

WHERE r.assigned_room IS NOT NULL

 AND CURRENT_DATE >= r.check_in

 AND CURRENT_DATE <  r.check_out
```

GROUP BY rm.type_code

ORDER BY rooms_sold_today DESC, rm.type_code;

13) Yesterday's housekeeping tasks per assignee, by status

SELECT t.assigned_to,

    SUM(CASE WHEN t.status = 'PENDING' THEN 1 ELSE 0 END) AS pending_count,

    SUM(CASE WHEN t.status = 'DONE'   THEN 1 ELSE 0 END) AS done_count

FROM housekeeping_task t

WHERE t.task_date = CURRENT_DATE - INTERVAL '1 day'

GROUP BY t.assigned_to

ORDER BY done_count DESC, pending_count DESC, t.assigned_to;

14) Rooms with overlapping reservations

SELECT r1.assigned_room AS room_id,

    r1.res_id AS res_a, g1.first_name || ' ' || g1.last_name AS guest_a,

    r2.res_id AS res_b, g2.first_name || ' ' || g2.last_name AS guest_b

FROM reservation r1

JOIN reservation r2

 ON r1.assigned_room = r2.assigned_room

 AND r1.res_id < r2.res_id

 AND r1.check_in < r2.check_out

 AND r2.check_in < r1.check_out

JOIN guest g1 ON g1.guest_id = r1.guest_id

JOIN guest g2 ON g2.guest_id = r2.guest_id

WHERE r1.assigned_room IS NOT NULL

ORDER BY room_id, res_a, res_b;

15) Top 5 guests by number of reservations in last 90 days

SELECT g.guest_id,

    g.first_name || ' ' || g.last_name AS guest,

    COUNT(*) AS reservations_90d

FROM reservation r

JOIN guest g ON g.guest_id = r.guest_id

WHERE r.created_at >= CURRENT_DATE - INTERVAL '90 days'

GROUP BY g.guest_id, guest

ORDER BY reservations_90d DESC, guest

LIMIT 5;