



SOICT

Báo cáo bài tập cuối kỳ

Thiết kế cơ sở dữ liệu Website đặt phòng khách sạn

Nhóm 4

Thành viên



Tạ Hồng Phúc



Nguyễn Mạnh Tùng



Bùi Quang Hưng

Nội dung



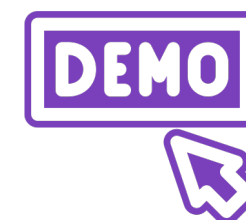
Mục tiêu



Câu lệnh SQL



Quy trình thiết kế



Website demo

Mục tiêu

1. Thiết kế cơ sở dữ liệu website đặt phòng khách sạn
2. Xây dựng website demo một số chức năng dựa trên cơ sở dữ liệu đã xây dựng





Quy trình thiết kết

01

Phân tích các yêu cầu

02

Khảo sát thực tế

03

Xác định các thực thể và thuộc tính

04

Xây dựng sơ đồ thực thể liên kết

05

Xây dựng sơ đồ quan hệ logic

06

Phân tích chi tiết sơ đồ quan hệ



Các yêu cầu

- Tra cứu phòng theo nhu cầu của khách hàng
- Ghi nhận thông tin thuê phòng
- Ghi nhận thông tin check in, check out
- Cập nhật giá phòng, dịch vụ
- Tính tiền phải thanh toán khi đặt phòng
- Cung cấp số liệu báo cáo
- Thêm, sửa, xóa các thông tin



Khảo sát thực tế

- Một khách sạn có nhiều hạng phòng: classic, standard, luxury.
- Có nhiều khách sạn với các mức đánh giá khác nhau: 1, 2, 3, 4, 5 sao.
- Phòng thuê có thể có nhiều trạng thái: sẵn sàng, có khách, đã đặt trước, bảo trì,...
- Mỗi khách sạn có quy định, dịch vụ đi kèm khác nhau
- Thông tin khách hàng: họ tên, email, số điện thoại

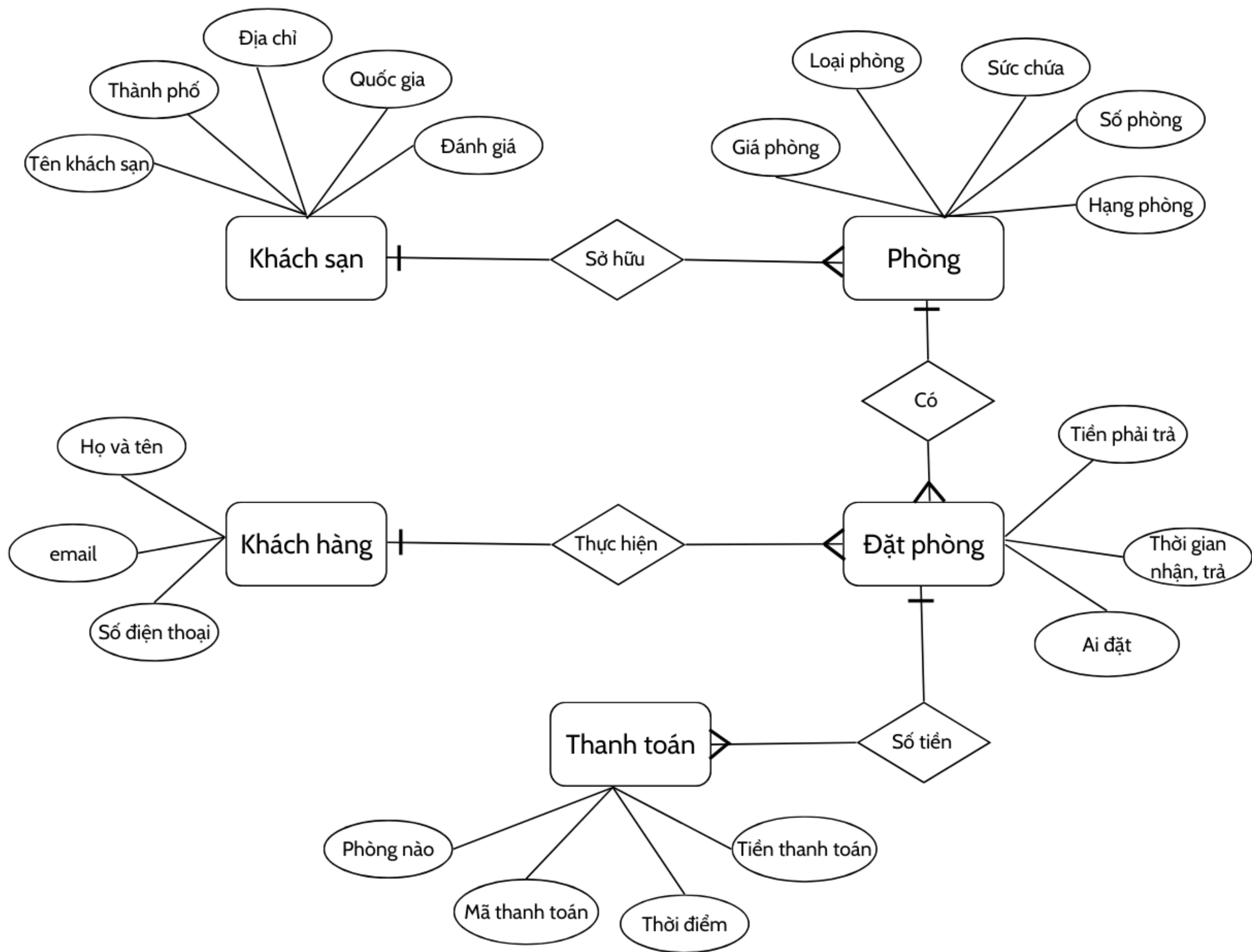


Các thực thể và thuộc tính

- Khách sạn: Tên, địa chỉ, đánh giá
- Phòng: Giá phòng, loại phòng, sức chứa, hạng phòng
- Khách hàng: Họ và tên, email, số điện thoại
- Đặt phòng: Thời gian checkin-checkout, tiền phải trả
- Thanh toán: Số tiền, thời điểm thanh toán

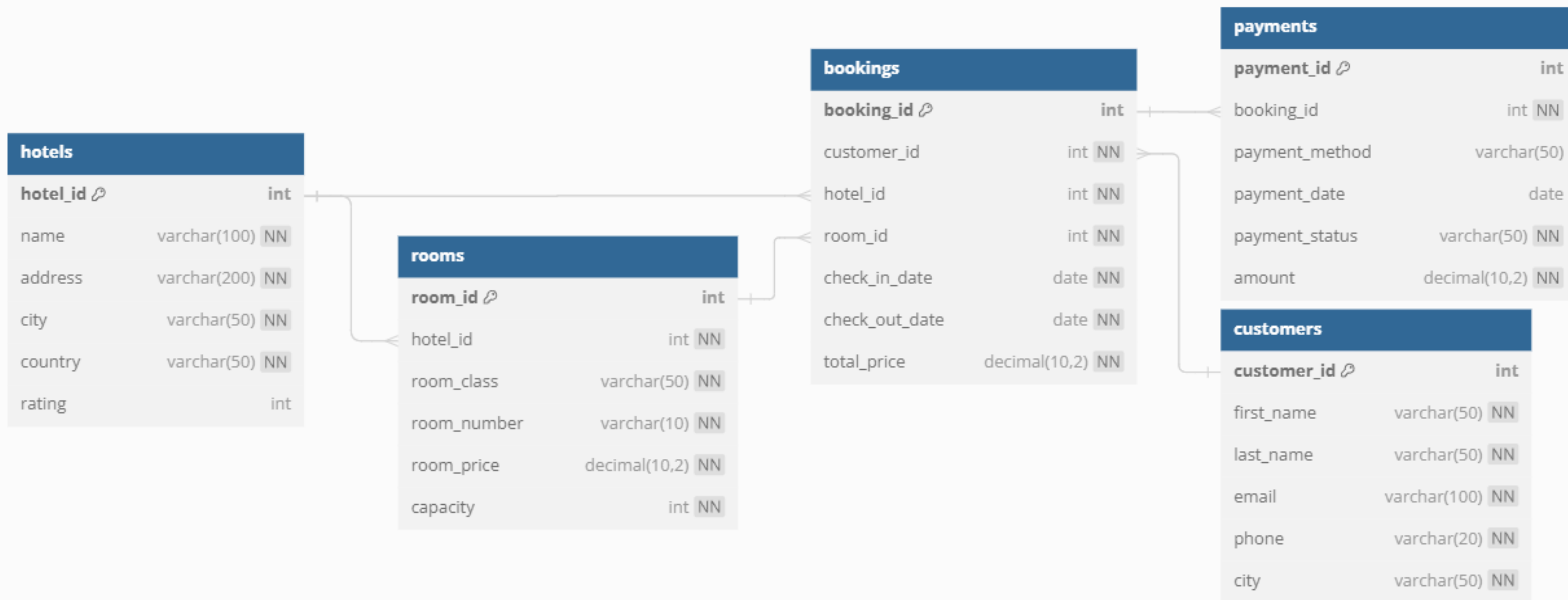


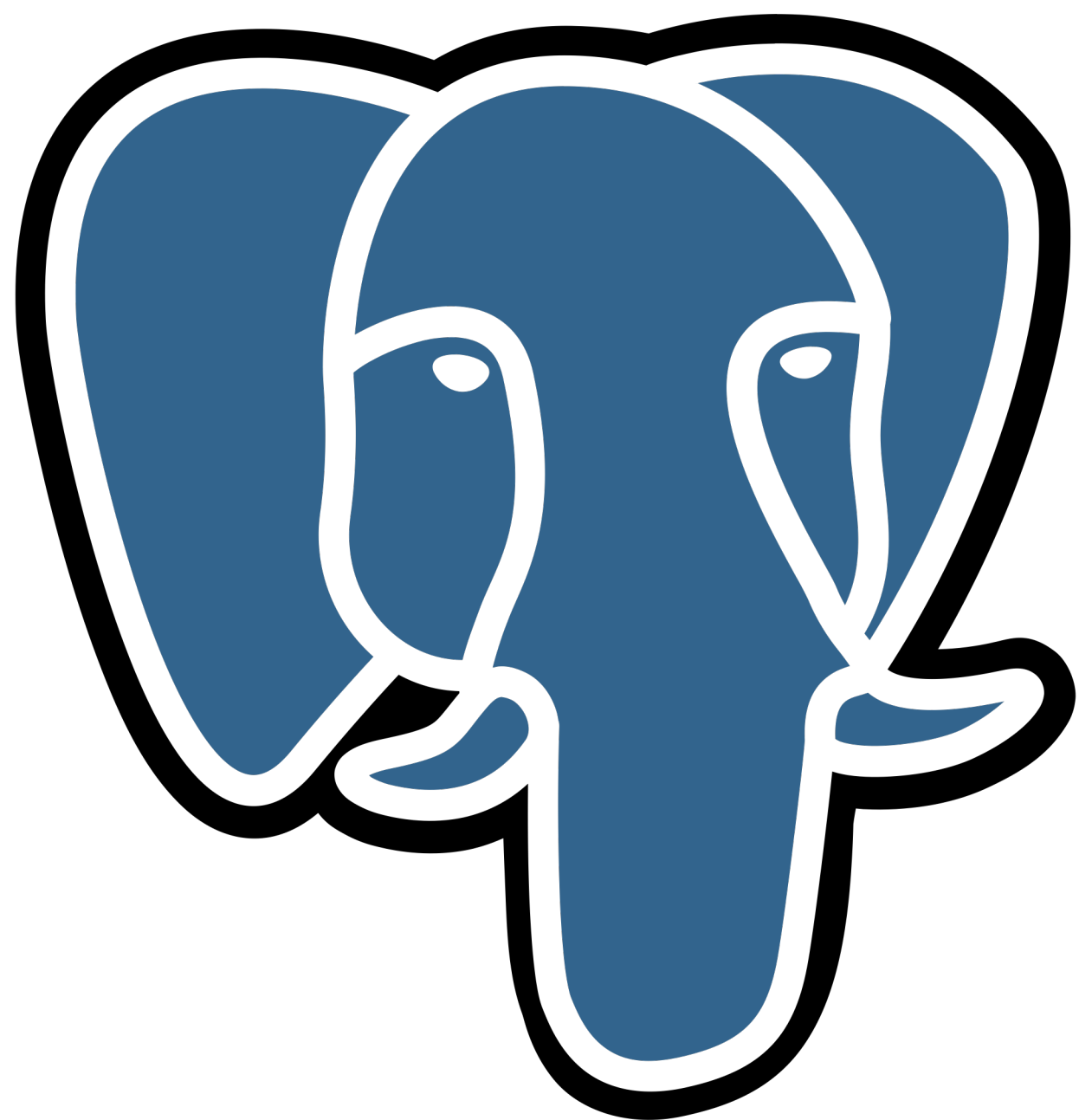
Sơ đồ thực thể liên kết





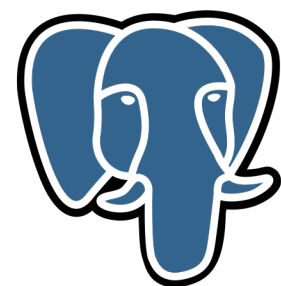
Sơ đồ quan hệ logic





Các câu lệnh SQL





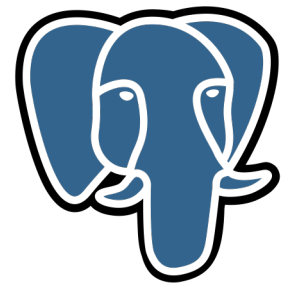
Tối ưu câu lệnh

```
1 SELECT
2     hotel_name,
3     booking_count
4 FROM (
5     SELECT
6         h.name AS hotel_name,
7         COUNT(b.booking_id) AS booking_count,
8         ROW_NUMBER() OVER (ORDER BY COUNT(b.booking_id) DESC) AS rank
9     FROM
10        hotels h
11    JOIN
12        rooms r ON h.hotel_id = r.hotel_id
13    JOIN
14        bookings b ON r.room_id = b.room_id
15    GROUP BY
16        h.name
17 ) AS ranked
18 WHERE
19     rank = 1;
```

	hotel_name character varying (100) 🔒	booking_count bigint 🔒
1	Khach san Melia Ha Noi	9
Total rows: 1 of 1 Query complete 00:00:00.069		

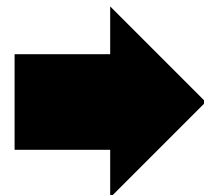
```
1 SELECT
2     h.name AS hotel_name,
3     COUNT(b.booking_id) AS booking_count
4 FROM
5     hotels h
6 JOIN
7     rooms r ON h.hotel_id = r.hotel_id
8 JOIN
9     bookings b ON r.room_id = b.room_id
10 GROUP BY
11     h.name
12 ORDER BY
13     booking_count DESC
14 LIMIT 1;
```

	hotel_name character varying (100) 🔒	booking_count bigint 🔒
1	Khach san Melia Ha Noi	9
Total rows: 1 of 1 Query complete 00:00:00.059		



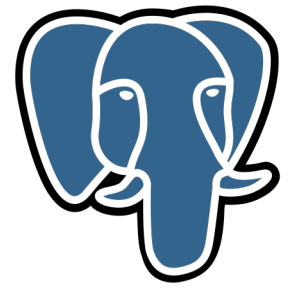
Truy vấn

Khách sạn có
doanh thu cao
nhất năm hiện tại



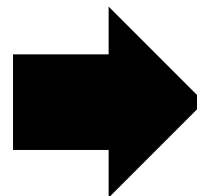
```
1 WITH HotelRevenues AS (  
2     SELECT  
3         h.hotel_id,  
4         h.name AS hotel_name,  
5         h.city AS hotel_city,  
6         SUM(p.amount) AS total_revenue,  
7         EXTRACT(YEAR FROM b.check_in_date) AS booking_year  
8     FROM  
9         hotels h  
10        JOIN bookings b ON h.hotel_id = b.hotel_id  
11        JOIN payments p ON b.booking_id = p.booking_id  
12    WHERE  
13        EXTRACT(YEAR FROM b.check_in_date) = EXTRACT(YEAR FROM CURRENT_DATE)  
14    GROUP BY  
15        h.hotel_id, h.name, booking_year  
16 )  
17 SELECT  
18     hotel_id,  
19     hotel_name,  
20     hotel_city,  
21     total_revenue  
22 FROM  
23     HotelRevenues  
24 WHERE  
25     total_revenue = (SELECT MAX(total_revenue) FROM HotelRevenues);
```

hotel_id [PK] integer	hotel_name character varying (100)	hotel_city character varying (50)	total_revenue numeric
9	Khach san Lavis 18 Residence	Ho Chi Minh	5600.00



Truy vấn

Đưa ra khách sạn
có phòng 'Luxury'
rẻ nhất của từng
thành phố



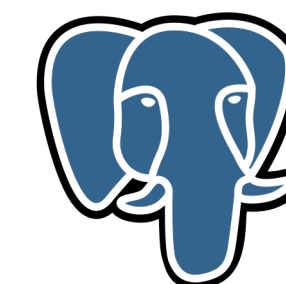
```
1  WITH cheapest_luxury_rooms AS (  
2      SELECT  
3          h.city,  
4          MIN(r.room_price) AS min_price  
5      FROM hotels h  
6      INNER JOIN rooms r ON h.hotel_id = r.hotel_id  
7      WHERE r.room_class = 'Luxury'  
8      GROUP BY h.city  
9  )  
10  
11  SELECT  
12      h.name AS hotel_name,  
13      h.address AS hotel_address,  
14      clr.min_price AS cheapest_luxury_price  
15  FROM hotels h  
16  INNER JOIN rooms r ON h.hotel_id = r.hotel_id  
17  INNER JOIN cheapest_luxury_rooms clr ON h.city = clr.city  
18  WHERE r.room_class = 'Luxury' AND r.room_price = clr.min_price  
19  ORDER BY h.city;  
20
```

	hotel_name character varying (100)	hotel_address character varying (200)	cheapest_luxury_price numeric
1	Khách sạn West Hotel	88-90-92 Hai Ba Trung	580.00
2	Khách sạn Monarque	236 Vo Nguyen Giap, Phuoc My, Son Tra	450.00
3	Khách sạn Melia Ha Noi	44B Ly Thuong Kiet, Tran Hung Dao, Hoan Kiem	500.00
4	Khách sạn Pan Pacific Ha Noi	1 Thanh Nien, Truc Bach, Ba Dinh	500.00
5	Khách sạn Avani Hai Phong Harbour View	12 Tran Phu Street, Ngo Quyen	500.00

```

1 CREATE OR REPLACE VIEW booked_rooms AS
2 SELECT
3     r.room_id,
4     r.room_number,
5     r.room_class,
6     r.room_price,
7     h.hotel_id,
8     h.name AS hotel_name,
9     b.booking_id,
10    b.customer_id,
11    c.first_name AS customer_first_name,
12    c.last_name AS customer_last_name,
13    b.check_in_date,
14    b.check_out_date
15 FROM
16     rooms r
17 JOIN
18     bookings b ON r.room_id = b.room_id
19 JOIN
20     hotels h ON r.hotel_id = h.hotel_id
21 JOIN
22     customers c ON b.customer_id = c.customer_id
23 WHERE
24     h.city = 'Ha Noi'
25     AND EXTRACT(MONTH FROM b.check_in_date) = 7;
26
27 select * from booked_rooms;

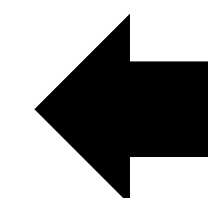
```



View

Danh sách các
phòng đã được
đặt trong tháng 7
ở Hà nội

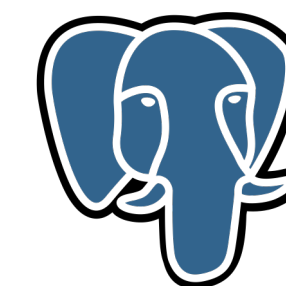
	room_id integer	room_number character varying (10)	room_class character varying (50)	room_price numeric (10,2)	hotel_id integer	hotel_name character varying (100)	booking_id integer	customer_id integer
1	5	301	Luxury	500.00	1	Khach san Melia Ha Noi	157	71
2	7	102	Classic	120.00	2	Khach san Lotte Ha Noi	87	2
3	10	301	Standard	300.00	2	Khach san Lotte Ha Noi	158	72
4	15	301	Luxury	600.00	3	Khach san Sofitel Legend Metropole	159	73
5	20	301	Standard	250.00	4	Khach san JW Marriott Ha Noi	160	74
6	25	301	Luxury	500.00	5	Khach san Pan Pacific Ha Noi	161	75



```

1 CREATE OR REPLACE VIEW top_5_hotel_revenue AS
2 WITH hotel_revenue_summary AS (
3     SELECT
4         h.hotel_id,
5         h.name AS hotel_name,
6         SUM(p.amount) AS total_revenue,
7         RANK() OVER (ORDER BY SUM(p.amount) DESC) AS revenue_rank
8     FROM
9         hotels h
10    JOIN
11        bookings b ON h.hotel_id = b.hotel_id
12    JOIN
13        payments p ON b.booking_id = p.booking_id
14    GROUP BY
15        h.hotel_id, h.name
16 )
17 SELECT
18     hotel_id,
19     hotel_name,
20     total_revenue,
21     revenue_rank
22 FROM
23     hotel_revenue_summary
24 WHERE
25     revenue_rank <= 5
26 ORDER BY
27     total_revenue DESC;
28 select * from top_5_hotel_revenue;

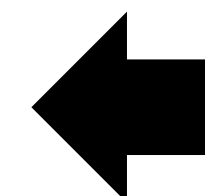
```



View

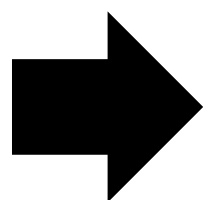
Danh sách các
phòng đã được
đặt trong tháng 7
ở Hà nội

	hotel_id integer	hotel_name character varying (100)	total_revenue numeric	revenue_rank bigint
1	9	Khach san Lavis 18 Residence	5600.00	1
2	3	Khach san Sofitel Legend Metropole	4900.00	2
3	8	Khach san Queen Ann	4685.00	3
4	4	Khach san JW Marriott Ha Noi	4375.00	4
5	7	Khach san Pullman Sai Gon Centre	3680.00	5



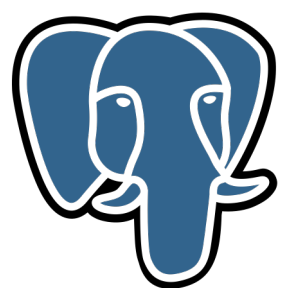
Function

truy xuất danh
sách các khách
sạn và thông tin
về các loại phòng
có sẵn trong mỗi
khách sạn



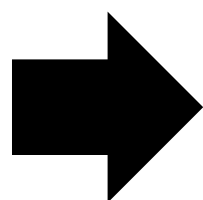
	hotel_id integer	hotel_name character varying	hotel_address character varying	hotel_city character varying	hotel_country character varying	hotel_rating integer	room_class character varying	room_price numeric	room_capacity integer
1	1	Khách sạn Melia Ha Noi	44B Ly Thuong Kiet, Tran Hung Dao, Hoan Kiem	Ha Noi	Vietnam	4	Luxury	500.00	6
2	1	Khách sạn Melia Ha Noi	44B Ly Thuong Kiet, Tran Hung Dao, Hoan Kiem	Ha Noi	Vietnam	4	Standard	150.00	2
3	2	Khách sạn Lotte Ha Noi	54 Lieu Giai, Cong Vi, Ba Dinh	Ha Noi	Vietnam	4	Classic	120.00	2

```
1 CREATE OR REPLACE FUNCTION get_distinct_hotels_and_rooms()
2 RETURNS TABLE (
3     hotel_id INT,
4     hotel_name VARCHAR(100),
5     hotel_address VARCHAR(200),
6     hotel_city VARCHAR(50),
7     hotel_country VARCHAR(50),
8     hotel_rating INT,
9     room_class VARCHAR(50),
10    room_price DECIMAL(10, 2),
11    room_capacity INT
12 )
13 AS $$
14 BEGIN
15     RETURN QUERY
16     SELECT DISTINCT ON (h.hotel_id, r.room_class)
17         h.hotel_id,
18         h.name AS hotel_name,
19         h.address AS hotel_address,
20         h.city AS hotel_city,
21         h.country AS hotel_country,
22         h.rating AS hotel_rating,
23         r.room_class,
24         r.room_price,
25         r.capacity
26     FROM
27         hotels h
28         JOIN rooms r ON h.hotel_id = r.hotel_id
29     ORDER BY
30         h.hotel_id, r.room_class;
31 END;
32 $$ LANGUAGE plpgsql;
33 SELECT * FROM get_distinct_hotels_and_rooms();
```



Function

trả về danh sách
các phòng có sẵn
dựa trên các tham
số đầu vào



```
1 CREATE OR REPLACE FUNCTION get_available_room(  
2     p_hotel_id INT,  
3     p_room_class VARCHAR(50),  
4     p_check_in_date DATE,  
5     p_check_out_date DATE  
6 )  
7 RETURNS TABLE (  
8     room_id INT,  
9     room_number VARCHAR(10),  
10    room_class VARCHAR(50),  
11    room_price DECIMAL(10, 2),  
12    hotel_id INT,  
13    capacity INT  
14 )  
15 AS $$  
16 BEGIN  
17     RETURN QUERY  
18     SELECT  
19         r.room_id,  
20         r.room_number,  
21         r.room_class,  
22         r.room_price,  
23         r.hotel_id,  
24         r.capacity  
25     FROM  
26         rooms r  
27     WHERE  
28         r.hotel_id = p_hotel_id  
29         AND r.room_class = p_room_class  
30         AND r.room_id NOT IN (  
31             SELECT  
32                 b.room_id  
33             FROM  
34                 bookings b  
35             WHERE  
36                 (b.check_in_date >= p_check_in_date AND b.check_in_date < p_check_out_date)  
37                 OR (b.check_out_date > p_check_in_date AND b.check_out_date <= p_check_out_date)  
38                 OR (b.check_in_date < p_check_in_date AND b.check_out_date > p_check_out_date)  
39             );  
40 END;  
41 $$ LANGUAGE plpgsql;  
42 SELECT * FROM get_available_room(1, 'Luxury', '2025-08-20', '2025-08-25');
```

	room_id integer	room_number character varying	room_class character varying	room_price numeric	hotel_id integer	capacity integer
1	5	301	Luxury	500.00	1	6

```

1 CREATE OR REPLACE FUNCTION trg_prevent_double_booking()
2 RETURNS TRIGGER AS $$
3 DECLARE
4     existing_booking INT;
5 BEGIN
6     SELECT COUNT(*)
7     INTO existing_booking
8     FROM bookings
9     WHERE room_id = NEW.room_id
10        AND check_in_date <= NEW.check_out_date
11        AND check_out_date >= NEW.check_in_date;
12 IF existing_booking > 0 THEN
13     RAISE EXCEPTION 'Không thể đặt phòng vì phòng đã được đặt trong khoảng thời gian này!';
14 END IF;
15 RETURN NEW;
16 END;
17 $$ LANGUAGE plpgsql;
18 CREATE TRIGGER prevent_double_booking
19 BEFORE INSERT ON bookings
20 FOR EACH ROW
21 EXECUTE FUNCTION trg_prevent_double_booking();

```



Ngăn chặn
“double booking”

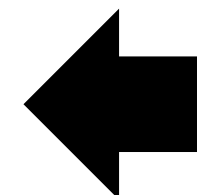
```

2 INSERT INTO bookings (customer_id, hotel_id, room_id, check_in_date, check_out_date, total_price) VALUES
3 (1, 1, 1, '2024-05-05', '2024-08-08', 450.00);

```

Data Output Messages Notifications

ERROR: Không thể đặt phòng vì phòng đã được đặt trong khoảng thời gian này!
CONTEXT: PL/pgSQL function trg_prevent_double_booking() line 12 at RAISE



```

1 CREATE OR REPLACE FUNCTION trg_update_prices()
2 RETURNS TRIGGER AS $$
3 BEGIN
4     UPDATE bookings
5     SET total_price = (NEW.check_out_date - NEW.check_in_date) * rooms.room_price
6     FROM rooms
7     WHERE bookings.room_id = rooms.room_id
8     AND bookings.booking_id = NEW.booking_id;
9     UPDATE payments
10    SET amount = bookings.total_price
11    FROM bookings
12    WHERE payments.booking_id = bookings.booking_id
13    AND payments.booking_id = NEW.booking_id;
14    RETURN NEW;
15 END;
16 $$ LANGUAGE plpgsql;
17 CREATE TRIGGER update_prices
18 AFTER UPDATE OF check_in_date, check_out_date ON bookings
19 FOR EACH ROW
20 WHEN (OLD.check_in_date IS DISTINCT FROM NEW.check_in_date OR OLD.check_out_date IS DISTINCT FROM NEW.check_out_date)
21 EXECUTE FUNCTION trg_update_prices();

```



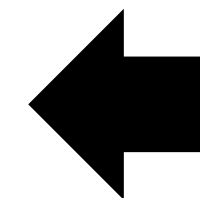
Cập nhật tiền
phải thanh toán
khi có thay đổi

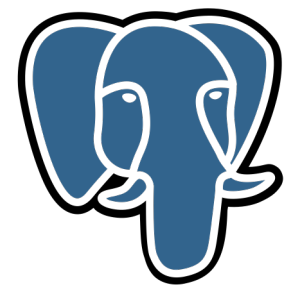
```

1 --UPDATE bookings SET check_out_date = '2024-08-09' WHERE booking_id = 167;
2 SELECT b.booking_id, b.customer_id, b.room_id, b.check_in_date, b.check_out_date,
3        p.payment_id, p.payment_method, p.payment_date, p.payment_status, p.amount
4 FROM bookings b
5 JOIN payments p ON b.booking_id = p.booking_id
6 WHERE b.booking_id = 167;

```

Data Output Messages Notifications										
	id	customer_id	room_id	check_in_date	check_out_date	payment_id	payment_method	payment_date	payment_status	amount
	id	integer	integer	date	date	integer	character varying (50)	date	character varying (50)	numeric (10,2)
1	167	1	1	2024-08-05	2024-08-09	84	Internet Banking	2024-06-20	Completed	600.00





Index

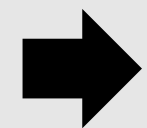
tăng tốc độ truy vấn khi tìm kiếm các khách sạn theo thành phố

```
1 --SELECT * FROM hotels WHERE city = 'Ha Noi';  
2 CREATE INDEX idx_hotels_city ON hotels (city);  
3 |
```

Khi chưa có Index

Query complete 00:00:00.112

Khi có Index



Query complete 00:00:00.079

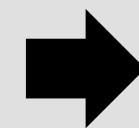
cải thiện hiệu suất khi tìm kiếm các đơn đặt phòng của một khách hàng cụ thể

```
1 --SELECT * FROM bookings WHERE customer_id = 2;  
2 CREATE INDEX idx_bookings_customer_id ON bookings (customer_id);  
3 |
```

Khi chưa có Index

Query complete 00:00:00.120

Khi có Index



Query complete 00:00:00.067

**website
demo**



Thank you