

QUERIES

-- List customers(customer id, Name, Contact No.) that have paid their bills using cash method for amount higher than 400.

```
select cust_id, "Name", Contact_No from customer
        natural join
(select cust_id
        from invoice
        where amount_payable >= 400 AND payment_method = 'cash') as r1;
```

-- Find hotel wise the percentage of female employees above 30 years.

```
select hotel_id,
((aged_emp::decimal)*100)/(total_emp::decimal)::numeric(10,2) || '%' as
percentage
        from
(select hotel_id, count(emp_no) as total_emp from employee group by
hotel_id) as r2
        natural join
(select hotel_id,count(extract(year from age(dob))) as aged_emp
        from employee
        where extract(year from age(dob)) > 35
        group by hotel_id) as r1;
```

-- List customer who registered hotel room for more than 8 days.

```
select cust_id, "Name", contact_no, hotel_id, days from customer
        natural join
(select hotel_id, cust_id, (end_date - start_date)::integer as days
        from reservation_on
        where (end_date - start_date)::integer >= 8) as r1;
```

-- List Hotel Room prices in 'Budget' category in mumbai city.

```
select * from
(select hotel_id ,hotel_name ,service ,contact_no from hotel where city =
'Mumbai') as r1
        natural join
(select hotel_id, price from today_price where category_id = '3') as r2;
```

-- List hotels who have "Fine Dining" or "Pool" facilities in New Delhi city.

```
select * from
(select hotel_id ,hotel_name ,service ,contact_no from hotel where city =
'New Delhi') as r1
        natural join
(select distinct hotel_id from room_category where ('Pool' =
any(facilities)) or ('Fine Dining' = any(facilities))) as r2;
```

-- list out all hotel's category wise available rooms

```
select hotel_id, category_id, name, Available_Rooms
        from Today_Price natural join Room_Category
        order by category_ID;
```

-- Give a department wise avg salary and count of all the employees working in hotel with hotel id = '2456324ND1'.

```
select dept_id, name, avg_salary, total_emp
        from department
        natural join
(select hotel_id, dept_id, avg(salary)::numeric(10,2) as avg_salary,
count(emp_no) as total_emp
        from employee
        where hotel_id = '2456324ND1'
        group by hotel_id,dept_id) as r1;
```

-- List Employees who have salary more than its supervisor working in hotel_id = '2546312CN9'.

```
select e1.emp_no, e1."Name", e1.gender, e1.salary as emp_sal, e2.salary as
super_sal from

    (select * from employee where hotel_id = '3201564ND4') as e1

        join

    (select * from employee where hotel_id = '3201564ND4') as e2

on e1.super_eno = e2.emp_no

where e1.salary >= e2.salary;
```

-- Make a list of top 3 hotels in 'Chennai' having the cheapest price.

```
select * from

(select hotel_id, hotel_name, contact_no from hotel where city = 'Chennai')
as r1

        natural join

today_price

order by price limit 3;
```

-- List hotels which have stars <= 5 and having 'Fitness Centre' or 'Swimming Pool' or 'Fine Dining' in jaipur.

```
select hotel_id, hotel_name, contact_no, service

from hotel

    where stars <= 5 and (

        position('Swimming Pool' in service) != 0

        or position('Fine Dining' in service) != 0 or position('Fitness
Centre' in service) != 0);
```

-- List out all employees who are supervisor as well as manager.

```
select Emp_No, Name, hotel_id

from Employee natural join Department

where (Manager_eno=Super_eno);
```

-- List the name, id, city and pincode of hotels in Chennai that are 5 stars and have rate per day less than 3000.

```
select hotel_id, hotel_name, city, pincode
      from (hotel natural join today_price)
     where state = 'Chennai' and stars = 5 and price <= 3000;
```

-- Give the list of hotel wise veg food under 350.

```
select hotel_id, "Name", rate from food
     where category = 'Veg' and rate <= 350;
```

-- Give frequently ordered foods for last two month

```
select hotel_id, max(food_count) as famous_food from
      (select hotel_id, "Name", count(food_no) as food_count
       from ordered_by natural join food
       group by hotel_id, "Name") as r1
     group by hotel_id;
```

-- Find hotel wise employees(emp no, name) which having age grater than 40.

```
select hotel_id, emp_no, "Name", extract(year from age(dob)) as age
      from employee
     where extract(year from age(dob)) > 40;
```