**SQL Practice Problems: Retail I**

**Problem Statement**

The dataset has customer-level and order-level details for a retail business. Use both the tables to solve problems

1.

Find the order count placed by every customer.

Expected Output: First\_name, Last\_name, count of orders columns in exact same sequence. The output should be sorted by order count in descending order

select first\_name, last\_name,

count(o.id) as order\_count

from customers c

left join orders o

on c.id = o.cust\_id

group by 1,2

order by 3 desc

2.

Find city-wise order count.  
Expected Output: City and count of orders in exact same sequence. The output should be sorted by order count in descending order

select city,

count(o.id) as order\_count

from customers c

left join orders o

on c.id = o.cust\_id

group by 1

order by 2 desc

3.

Find "favourite" customers based on the order count and the total cost of orders.  
A customer is considered as a favorite if he or she has placed more than 3 orders and with a total cost of orders of more than $100 (excluding).

Expected Ouput: Customer\_id, first\_name, last\_name, count of orders, sum of total\_order\_cost as total\_cost columns in exact same sequence. The output should be sorted by customer id in ascending order

select customers.id,

customers.first\_name,

customers.last\_name,

count(orders.id) as order\_count,

sum (total\_order\_cost) as total\_cost

from orders

inner join customers on orders.cust\_id =

customers.id

group by 1,2,3

having sum(total\_order\_cost)>100

and count(orders.id)>3

order by 1 asc

4.

Company wants to downsize the operation in cities where the order count is below 4(excluding). Help management to identify the cities which are having low order count

Expected Output: City names in descending order

select city from customers c

left join orders o

on c.id = o.cust\_id

group by 1

having count(o.id)<4

order by 1 desc

5.

Identify the customers who have not placed any order

Expected Output: First\_name, Last\_name, count of orders columns in exact same sequence. The output should be sorted by first\_name in descending order than last name in descending order

select first\_name, last\_name , count(o.id) as order\_count

from customers c

left join orders o

on c.id = o.cust\_id

group by 1, 2

having count(o.id) = 0

order by 1 desc,

2 desc