-- Introductory

Q18. Products with associated supplier names

We’d like to show, for each product, the associated Supplier. Show the ProductID, ProductName, and the CompanyName of the Supplier. Sort by ProductID.

Ans 18 :

select product\_id,product\_name,company\_name

from products

left join suppliers

on suppliers.supplier\_id = products.supplier\_id

order by product\_id

Q19. Orders and the Shipper that was used

We’d like to show a list of the Orders that were made, including the Shipper that was used. Show the OrderID, OrderDate (date only), and CompanyName of the Shipper, and sort by OrderID.

In order to not show all the orders (there’s more than 800), show only those rows with an OrderID of less than 10300

Ans 19 :

select order\_id,order\_date,company\_name

from orders

left join shippers

on shippers.shipper\_id = orders.ship\_via

where order\_id < 10300

order by order\_id

--Advanced

Q32 .High-value customers

We want to send all of our high-value customers a special VIP gift.We're defining high-value customers as those who've made at least 1 order with a total value (not including the discount) equal to $10,000 or more. We only want to consider orders made in the year 2016

Ans 32 : As orders in 2016 not present in the data base , replacing 2016 with 1998

select customers.customer\_id ,company\_name,orders.order\_id,

sum(unit\_price\*quantity) as "Total Order Amount"

from customers

left join orders

on customers.customer\_id = orders.customer\_id

left join order\_details

on orders.order\_id = order\_details.order\_id

where order\_date between '1998-01-01' and '1998-12-31'

group by customers.customer\_id ,company\_name,orders.order\_id

having sum(unit\_price\*quantity) > 10000

order by sum(unit\_price\*quantity) desc

Q33. High-value customers - total orders

The manager has changed his mind. Instead of requiring that customers have at least one individual orders totaling $10,000 or more, he wants to define high-value customers as those who have orders totaling $15,000 or more in 2016. How would you change the answer to the problem above?

Ans 33:

select customers.customer\_id ,company\_name,

sum(unit\_price\*quantity) as "Total Order Amount"

from customers

left join orders

on customers.customer\_id = orders.customer\_id

left join order\_details

on orders.order\_id = order\_details.order\_id

where order\_date between '1998-01-01' and '1998-12-31'

group by customers.customer\_id ,company\_name

having sum(unit\_price\*quantity) > 15000

order by sum(unit\_price\*quantity) desc

Q34. High-value customers - with discount

Change the above query to use the discount when calculating high-value customers. Order by the total amount which includes the discount

Ans34:

select customers.customer\_id ,company\_name -- ,orders.order\_id,

,sum((unit\_price\*quantity)\*(1-discount)) -- as "Total Order Amount"

from customers

left join orders

on customers.customer\_id = orders.customer\_id

left join order\_details

on orders.order\_id = order\_details.order\_id

where order\_date between '1998-01-01' and '1998-12-31'

group by customers.customer\_id ,company\_name-- ,orders.order\_id

having sum((unit\_price\*quantity)\*(1-discount)) > 10000

order by sum((unit\_price\*quantity)\*(1-discount)) desc

Q35. Month-end orders

At the end of the month, sales people are likely to try much harder to get orders, to meet their month-end quotas. Show all orders made on the last day of the month. Order by EmployeeID and OrderID

Ans35 :

select employee\_id,order\_id,order\_date from orders where order\_date like '%02-28%'

union select employee\_id,order\_id,order\_date from orders where order\_date like '%-30%'

union select employee\_id,order\_id,order\_date from orders where order\_date like '%-31%'

Q36 . Orders with many line items

The Northwind mobile app developers are testing an app that customers will use to show orders. In order to make sure that even the largest orders will show up correctly on the app, they'd like some samples of orders that have lots of individual line items. Show the 10 orders with the most line items, in order of total line items.

Ans36 : select order\_id , count(product\_id) from order\_details

group by order\_id

order by count(product\_id) desc limit 10

Q37. Orders - random assortment

The Northwind mobile app developers would now like to just get a random assortment of orders for beta testing on their app. Show a random set of 2% of all orders.

Ans37 : As 2 % is around 17 orders

select order\_id from orders limit 17