

Ejercicios SQL Bootcamp Data Engineer - EDVAi

Consigna: escribir la query/consulta necesaria para llegar al resultado.

SELECT DISTINCT

1. Obtener una lista de todas las categorías distintas:

```
select distinct category_name  
from categories c
```

2. Obtener una lista de todas las regiones distintas para los clientes:

```
select distinct region  
from customers c
```

3. Obtener una lista de todos los títulos de contacto distintos:

```
select distinct contact_title  
from customers c
```

ORDER BY

4. Obtener una lista de todos los clientes, ordenados por país:

```
select *  
from customers c  
order by country asc
```

5. Obtener una lista de todos los pedidos, ordenados por id del empleado y fecha del pedido:

```
select *  
from orders o  
order by employee_id, order_date
```

INSERT INTO

6. Insertar un nuevo cliente en la tabla Customers:

```
insert into customers values  
( 'NEW', 'Test Company', 'Fantasy Name', 'Operator' )
```

7. Insertar una nueva región en la tabla Región:

```
insert into region (region_id ,region_description)  
values ( '6', 'SouthNorthern' );
```

NULL -COALESCE

8. Obtener todos los clientes de la tabla Customers donde el campo Región es NULL:

```
select *  
from customers c  
where region is null
```

9. Obtener Product_Name y Unit_Price de la tabla Products, y si Unit_Price es NULL, use el precio estándar de \$10 en su lugar:

```
select product_name, coalesce(unit_price, 10)  
from products p
```

INNER JOIN

10. Obtener el nombre de la empresa, el nombre del contacto y la fecha del pedido de todos los pedidos:

```
select c.company_name, c.contact_name, order_date
from customers c inner join orders o
on c.customer_id = o.customer_id
```

11. Obtener la identificación del pedido, el nombre del producto y el descuento de todos los detalles del pedido y productos:

```
select od.order_id, p.product_name, od.discount
from order_details od inner join products p
on od.product_id = p.product_id
```

LEFT JOIN

12. Obtener el identificador del cliente, el nombre de la compañía, el identificador y la fecha de la orden de todas las órdenes y aquellos clientes que hagan match :

```
select c.customer_id, c.company_name, o.order_id, o.order_date
from orders o left join customers c
on o.customer_id = c.customer_id
```

13. Obtener el identificador del empleados, apellido, identificador de territorio y descripción del territorio de todos los empleados y aquellos que hagan match en territorios:

```
select e.employee_id, e.last_name, t.territory_id, t.territory_description
from employees e left join employee_territories et
on e.employee_id = et.employee_id
left join territories t
on et.territory_id = t.territory_id
```

14. Obtener el identificador de la orden y el nombre de la empresa de todas las órdenes y aquellos clientes que hagan match:

```
select o.order_id, c.company_name
from orders o left join customers c
on o.customer_id = c.customer_id
```

RIGHT JOIN

15. Obtener el identificador de la orden, y el nombre de la compañía de todas las órdenes y aquellos clientes que hagan match:

```
select o.order_id, c.company_name
from customers c right join orders o
on c.customer_id = o.customer_id
```

16. Obtener el nombre de la compañía, y la fecha de la orden de todas las órdenes y aquellos transportistas que hagan match. Solamente para aquellas ordenes del año 1996:

```
select s.company_name, o.order_date
from shippers s right join orders o
on s.shipper_id = o.ship_via
where o.order_date between '1996-01-01' and '1996-12-31'
```

FULL OUTER JOIN

17. Obtener nombre y apellido del empleados y el identificador de territorio, de todos los empleados y aquellos que hagan match o no de employee_territories:

```
select e.first_name, e.last_name, et.territory_id
from employees e full outer join employee_territories et
on e.employee_id = et.employee_id
```

18. Obtener el identificador de la orden, precio unitario, cantidad y total de todas las órdenes y aquellas órdenes detalles que hagan match o no:

```
select o.order_id, od.unit_price, od.quantity,
       round( od.unit_price*od.quantity*(1-od.discount) ) as total
from orders o full outer join order_details od
on o.order_id = od.order_id
```

UNION

19. Obtener la lista de todos los nombres de los clientes y los nombres de los proveedores:

```
select c.company_name as nombre from customers c
union
select s.company_name from suppliers s
```

20. Obtener la lista de los nombres de todos los empleados y los nombres de los gerentes de departamento:

SUBQUERIES

21. Obtener los productos del stock que han sido vendidos:

```
select p.product_name, p.product_id
  from products p
 where p.product_id in
 (
  select distinct od.product_id
    from order_details od
 )
```

22. Obtener los clientes que han realizado un pedido con destino a Argentina:

```
select c.company_name
  from customers c
 where c.customer_id in
 (
  select o.customer_id
    from orders o
   where o.ship_country='Argentina'
 )
```

23. Obtener el nombre de los productos que nunca han sido pedidos por clientes de Francia:

```
select p.product_name
  from products p
 where p.product_id not in
 (
  select distinct od.product_id
    from orders o inner join order_details od
     on o.order_id = od.order_id
   where o.customer_id in (
     select c.customer_id from customers c
    where c.country='France')
 )
```

GROUP BY

24. Obtener la cantidad de productos vendidos por identificador de orden:

```
select o.order_id, sum(od.quantity)
  from orders o inner join order_details od
   on o.order_id = od.order_id
 group by o.order_id
```

25. Obtener el promedio de productos en stock por producto

```
select p.product_name, avg(p.units_in_stock)
  from products p
 group by p.product_name
```

HAVING

26. Cantidad de productos en stock por producto, donde haya más de 100 productos en stock

```
select p.product_name, sum(p.units_in_stock)
  from products p
 group by p.product_name
 having sum(p.units_in_stock) > 100
```

27. Obtener el promedio de pedidos por cada compañía y solo mostrar aquellas con un promedio de pedidos superior a 10:

CASE

28. Obtener el nombre del producto y su categoría, pero muestre "Discontinued" en lugar del nombre de la categoría si el producto ha sido descontinuado:

```
select p.product_name,  
       case  
         when p.discontinued = 1 then 'Discontinued'  
         else (select c.category_name from categories c where  
c.category_id=p.category_id)  
         end as product_category  
from products p
```

29. Obtener el nombre del empleado y su título, pero muestre "Gerente de Ventas" en lugar del título si el empleado es un gerente de ventas (Sales Manager):

```
select e.first_name, last_name,  
       case  
         when e.title = 'Sales Manager' then 'Gerente de Ventas'  
         else e.title  
         end as job_title  
from employees e
```