

```

/*
Mostrar para cada Supplier, la cantidad de productos
involucrados en ventas y la cantidad involucrada en
compras (purchases)
*/

SELECT S.SupplierID,S.CompanyName, (SELECT COUNT(DISTINCT
D.ProductID)
                                from OrderDetails D
                                inner join Products
P on D.ProductID = P.ProductID
                                where P.SupplierID
= S.SupplierID
                                ) as
#CantProdVentas,
                                (SELECT
COUNT(DISTINCT PD.ProductID)
                                from PurchaseDetail
PD
                                inner join Products
P on PD.ProductID = P.ProductID
                                where P.SupplierID
= S.SupplierID
                                ) as
#CantProdCompras
from Suppliers s

/*
Mostrar aquellos clientes que realizaron ordenes cuyos productos
poseen un precio unitario mayor que el precio máximo de los
productos
que participaron en compras de los suppliers que contienen una
'y' en su nombre
*/

SELECT distinct C.CompanyName
FROM Customers C
INNER JOIN Orders O on O.CustomerID = C.CustomerID
INNER JOIN OrderDetails D ON O.OrderID = D.OrderID
WHERE D.UnitPrice > (
                                SELECT MAX(Tprod.UnitPrice)
                                FROM (
                                SELECT DISTINCT
pd.ProductID, pd.UnitPrice    --obtengo productos y su precio
                                unitario
                                FROM PurchaseDetail
PD
                                INNER JOIN Products p
ON p.ProductID = pd.ProductID
                                inner join Suppliers
S ON S.SupplierID = P.SupplierID
                                where s.CompanyName
LIKE '%y%') as TProd
                                )

```

```

/*
Mostrar aquellos empleados que no hicieron Ordenes en los años
1995 ni 1997 o que tienen clientes que pagaron más de 25.000 $
entre todas sus ordenes
*/

SELECT E.FirstName,E.LastName
FROM Employees E
WHERE NOT EXISTS (
    SELECT *
    from Orders O
    where YEAR(O.ORDERDATE) IN (1995,1997)
    AND E.EmployeeID = O.EmployeeID)
    OR e.EmployeeID in (
        SELECT distinct
o1.EmployeeID
        FROM Orders o1
        WHERE exists (
            SELECT O.CustomerID
            FROM
Orders O
            inner
join OrderDetails D on D.OrderID = O.OrderID
            where
O.EmployeeID = o1.EmployeeID
            group
by O.CustomerID
        HAVING SUM(D.UnitPrice * d.Quantity) > 25000))

select *
from PurchaseDetail

/*
Agregar la columna "Rate" de tipo VARCHAR(30) a la tabla de
Clientes.
Actualizar el campo Rate con el valor "ORO, PLATA, BRONCE"
según: ORO: >= 30 Ordenes , PLATA Entre 15 y 29 Ordenes,
BRONCE menos de 15 ordenes
*/

ALTER TABLE Customers
ADD rate varchar(30)

UPDATE Customers SET
rate = (SELECT (CASE
    WHEN count(o.OrderID) < 15 then 'BRONCE'
    WHEN count(o.OrderID) BETWEEN 15 and 29
then 'PLATA'
    ELSE 'ORO' END)
    FROM Orders O
    WHERE O.CustomerID = C.CustomerID)

```

FROM Customers C

/*

Llevar a un MR en 3FN la siguiente tabla y escribir la consulta que muestre para cada Destino, la cantidad de viajeros. Mostrar tambien los Destinos que no tuvieron viajeros.

VIAJES(idPaquete, FechaCompra, precioPagado, fechaInicioPaquete, fechaFinPaquete, codCliente, nomCliente, codDestino, nomDestino, codViajero, nomViajero, edadViajero)

Clientes (codCliente, nomCliente)

Destinos (codDestino, nomDestino)

Viajeros (codViajero, nomViajero, edadviajero)

Paquetes (idPaquete, fechaCompra, codCliente, precioPagado, fechaInicioPaquete, fechaFinPaquete)

PaqViajero (idPaquete, codDestino, codViajero)

*/

```
SELECT D.codDestino, D.nomDestino, count(PV.codViajero) AS
TotalViajeros
from Destinos D
LEFT JOIN paqViajero PV on D.codDestino = PV.codDestino
GROUP BY D.codDestino, D.nomDestino
```