

# Ejercicios SQL

Agustin Huczok

20/9/2021

#PPT ##1 Todas las ordenes de compra con los datos de los empleados, clientes y proveedores:

```
SELECT o.* , c.CustomerName, s.ShipperName FROM orders o INNER JOIN Customers c ON c.CustomerID=o.CustomerID INNER JOIN Shippers s ON s.ShipperID=o.ShipperID
```

##2 La cantidad de ordenes de compra por nombre de cliente, si la cantidad es mayor a 3:

```
SELECT c.CustomerName, o.CustomerID , COUNT (o.OrderID) as Compras FROM Customers c INNER JOIN Orders o ON o.CustomerID=c.CustomerID GROUP BY c.CustomerName HAVING Compras>3 ORDER BY Compras ASC
```

##3 FALTA LA SUMA El detalle de los productos y categorias por orden de compra, y calcular el total en pesos por cada compra:

```
SELECT o.OrderID, od.ProductID, p.Price, c.CategoryName, FROM Orders o INNER JOIN OrderDetails od ON o.OrderID=od.OrderID INNER JOIN Products p ON p.ProductID=od.ProductID INNER JOIN Categories c ON c.CategoryID=p.CategoryID
```

##4 Los empleados que no estan asignados en ninguna orden de compra:

#Guia ##1 Todas las ventas realizadas por MargaretPeacock al cliente Frankenversand.La salida debe contener los siguientes campos: Nombre y Apellido del Vendedor, Nombre del Cliente, ID de la Orden de Compra (tabla Orders), Nombre del Producto (tabla Products), Presentación (Unit, de tabla Products), Cantidad (tabla OrderDetails), Precio (tabla Products) y Total (calculado como Precio\*Cantidad):

```
SELECT e.LastName as Apellido_Vendedor, e.FirstName as Nombre_Vendedor, c.CustomerName as Cliente ,o.OrderID, p.ProductName as Nombre_Producto, p.unit as Presentacion, od.Quantity as Cantidad, p.price as Precio, round(p.Price*od.Quantity,2) as Total FROM Orders o INNER JOIN OrderDetails od ON o.OrderID=od.OrderID INNER JOIN Customers c ON c.CustomerID=o.CustomerID INNER JOIN Products p ON p.ProductID=od.ProductID INNER JOIN Employees e ON e.EmployeeID=o.EmployeeID WHERE Nombre_Vendedor="Margaret" AND Cliente="Frankenversand"
```

##2 Un descuento de 12% para aquellos productos con precio mayor a 75, un descuento del 8% para productos con precio entre 50 y 75 (inclusive), y 4% para productos con precio mayor o igual a 30 y hasta 49.99.La salida debe contener los siguientes campos: Nombre del Producto, Presentación (Unidades), Precio Original, Descuento (%), Descuento (\$) y Precio con Descuento:

```
SELECT ProductName as Nombre_Producto, Unit as Presentacion, Price as Precio_Original, CASE WHEN price>75 THEN "12%" WHEN (price>=50 AND price<=75) THEN "8%" WHEN (price>30 AND price<50) THEN "4%" ELSE "0%" END as "Descuento %", round(price(CASE WHEN price>75 THEN 0.12 WHEN (price>=50 AND price<=75) THEN 0.08 WHEN (price>30 AND price<50) THEN 0.04 ELSE 0 END),2) as "Descuento $", round(price(CASE WHEN price>75 THEN 0.88 WHEN (price>=50 AND price<=75) THEN 0.92 WHEN (price>30 AND price<50) THEN 0.96 ELSE 1 END),2) as Precio_final FROM Products ORDER BY "Descuento %" DESC
```

##3 El precio máximo y mínimo por cada categoría de los productos. La salida debe contener los siguientes campos: Nombre de la categoría, Precio Máximo, Precio Mínimo y debe ser ordenado por el precio máximo de forma descendente:

```
SELECT c.CategoryName , MIN(p.Price) as Minimo, MAX(p.Price) as Maximo FROM categories c INNER JOIN products p ON p.categoryID=c.CategoryID GROUP BY CategoryName ORDER BY Maximo DESC
```

##4

Todas las ventas realizadas a los clientes de España. La salida debe contener los siguientes campos: Nombre del Cliente, ciudad y país del cliente, ID de la Orden de Compra (tabla Orders), Nombre del Producto (tabla Products), Presentación (Unit, de tabla Products), Cantidad (tabla OrderDetails), Precio (tabla Products) y Total (calculado como Precio\*Cantidad):

```
SELECT c.CustomerName as Nombre_Cliente, c.City as Ciudad, c.Country as Pais, o.OrderID, p.ProductName as Producto, p.Unit as Presentacion, od.Quantity as Cantidad, p.price as Precio, round(price*quantity,2) as Total From Customers c INNER JOIN Orders o ON o.CustomerID=c.CustomerID INNER JOIN OrderDetails od ON od.OrderID=o.OrderID INNER JOIN Products p ON p.ProductID=od.ProductID WHERE c.Country="Spain"
```

##5 Un recargode 7% para los productos que se venden en caja y un 5% para productos que se venden en botellas o paquetes, para el resto de los productos el recargo es del 2%. La salida debe contener los siguientes campos: Nombre del Producto, Presentación (Unidades), Precio Original, Recargo(%), Recargo(\$) y Precio con Recargo:

```
SELECT ProductName as Nombre_Producto, Unit as Presentacion, price as Precio_Original, CASE WHEN Unit LIKE "%box%" THEN "7%" WHEN Unit LIKE "%bottle%" OR Unit LIKE "%pkgs%" THEN "5%" ELSE "2%" END as "Recargo%", round(Price(CASE WHEN Unit LIKE "%box%" THEN 0.07 WHEN Unit LIKE "%bottle%" OR Unit LIKE "%pkgs%" THEN 0.05 ELSE 0.02 END),2) as "Recargo$", round(Price(CASE WHEN Unit LIKE "%box%" THEN 1.07 WHEN Unit LIKE "%bottle%" OR Unit LIKE "%pkgs%" THEN 1.05 ELSE 1.02 END),2) as Precio_Con_Recargo From Products
```

##6 La cantidad de órdenes de compra y el total gastado por cliente, para los clientes de Estados Unidos. La salida debe contener los siguientes campos: Nombre del Cliente, Cantidad de órdenes, Total (calculado como suma total del Precio\*Cantidad):

```
SELECT c.CustomerName as Nombre_Cliente, COUNT(DISTINCT(o.OrderID)) as Cantidad_De_Ordenes, round(SUM(p.price*od.quantity),2) as Total_Gastado From Customers c INNER JOIN orders o ON o.customerID=c.CustomerID INNER JOIN orderdetails od ON o.orderid=od.orderid INNER JOIN products p ON p.productID=od.productid where c.country="USA" group by o.customerid
```

##7 Un comisión del 20% para los empleados que realizaron ventas por más de \$100.000, del 10% para los que tuvieron ventas entre \$40.000 y \$99.999 y del 5% para los que vendieron entre \$30.000 y 39000. La salida debe contener los siguientes campos : Nombre y Apellido del Vendedor, Cantidad de ventas (cantidad de órdenes, n

```
SELECT e.LastName as Apellido_Vendedor, e.FirstName as Nombre_Vendedor, COUNT(o.OrderID) as Cantidad_Ventas, round(SUM(p.price*od.quantity),2) as Total_Vendido, CASE WHEN SUM(p.price*od.quantity) > 100000 THEN "20%" WHEN SUM(p.price*od.quantity) > 40000 THEN "10%" WHEN SUM(p.price*od.quantity) > 30000 THEN "5%" ELSE "0%" END as "Comision(%)" FROM Employees e INNER JOIN Orders o ON o.EmployeeID=e.EmployeeID INNER JOIN OrderDetails od ON od.OrderID=o.OrderID INNER JOIN Products p ON p.ProductID=od.ProductID GROUP BY e.LastName ORDER BY total_vendido DESC
```

##8 Total de productos vendidos cuyos proveedores son de Japón. La salida debe contener los siguientes campos: Nombre del Proveedor, ciudad y país del proveedor, Contacto del proveedor, Nombre del Producto (tabla Products), Nombre de la categoría del producto (tabla Categories), Cantidad (tabla OrderDetails), Precio (tabla Products) y Total (calculado como Precio\*Cantidad):

```
SELECT s.supplierName as Proveedor, s.city as Ciudad, s.country as Pais ,s.Contactname as Contacto, p.Productname as Producto, c.CategoryName as Categoria, od.quantity as Cantidad, p.Price as Precio, round(p.price*od.quantity,2) as Total FROM Suppliers s INNER JOIN Products p ON p.SupplierID=s.SupplierID INNER JOIN Categories c ON c.CategoryID=p.CategoryID INNER JOIN OrderDetails od ON p.ProductID=od.ProductID WHERE s.Country="Japan"
```

##9 La cantidad de proveedores y de productos, agrupando por cada país y ciudad de los proveedores. La salida debe contener los siguientes campos: País y ciudad de los proveedores (Tabla Suppliers), Cantidad de proveedores, Cantidad de productos:

```
SELECT s.City as Ciudad, s.Country as Pais, COUNT(DISTINCT(s.SupplierName)) as Cantidad_Proveedores, COUNT(DISTINCT(p.ProductID)) as Cantidad_Productos FROM Suppliers s INNER JOIN Products p ON p.SupplierID=s.SupplierID GROUP BY s.SupplierName ORDER BY Pais, Cantidad_Productos DESC
```

##10 Los clientes que gastaron más de \$14.000 en el total de todas sus compras. La salida debe contener los siguientes campos: Nombre del Cliente, ciudad y país del cliente, Cantidad de compras, Total (calculado como la suma total del Precio Cantidad): *SELECT c.CustomerName as Cliente, c.City as Ciudad, c.Country as Pais, COUNT(DISTINCT(o.OrderID)) as "Cantidad de compras", round(SUM(p.priceod.quantity),2) as "Total" FROM Customers c INNER JOIN Orders o ON o.CustomerID=c.CustomerID INNER JOIN OrderDetails od ON od.OrderID=o.OrderID INNER JOIN Products p ON p.ProductID=od.ProductID GROUP BY c.CustomerName HAVING "Total" >14000 ORDER BY Total DESC*