

# Práctica 10

Fundamentos de Bases de Datos

Hernández Ferreiro Enrique Ehecatl

López Soto Ramses Antonio

Miguel Torres Eric Giovanni

Quintero Villeda Erik

11 de noviembre de 2019

## Introducción

### Objetivo

Utilizar las consultas avanzadas (instrucciones GROUP BY y HAVING) para resolver las consultas solicitadas.

## Desarrollo

Las consultas solicitadas son las siguientes:

1. El número de empleados que existen por cada puesto.

```
SELECT titulo, COUNT(titulo) numEmpleados
FROM Empleados
GROUP BY titulo;
```

2. El número de empleados que tiene a su cargo cada empleado.

```
SELECT A.idEmpleado, COUNT(B.idEmpleado) numEmpladosACargo
FROM Empleados A INNER JOIN Empleados B
ON A.idEmpleado = B.idEmpleado
GROUP BY A.idEmpleado
ORDER BY A.idEmpleado;
```

3. Nombre de la ciudades que tiene mas de un cliente.

```

SELECT ciudad
FROM Clientes
GROUP BY ciudad
HAVING COUNT(idCliente) > 1;

```

4. El total de productos por categoría con los que cuenta cada proveedor.

```

SELECT idProveedor, nombreCategoria, COUNT(Productos.idProducto) totalProductos
FROM Pedidos INNER JOIN DetallesPedido
ON Pedidos.idPedido = DetallesPedido.idPedido
INNER JOIN Productos
ON DetallesPedido.idProducto = Productos.idProducto
INNER JOIN Categorías
ON Productos.idCategoría = Categorías.idCategoría
GROUP BY idProveedor, nombreCategoria;

```

5. La compañía de envíos que mas pedidos ha despachado de la categoría 'Dairy Products'.

```

SELECT TOP (1) nombreCompania, COUNT(Pedidos.idPedido) AS numPedidos
FROM CompaniasEnvio INNER JOIN Pedidos
ON CompaniasEnvio.idCompaniaEnvio = Pedidos.viaEnvio
INNER JOIN DetallesPedido ON Pedidos.idPedido = DetallesPedido.idPedido
INNER JOIN Productos ON DetallesPedido.idProducto = Productos.idProducto
INNER JOIN Categorías ON Productos.idCategoría = Categorías.idCategoría
WHERE nombreCategoria = 'Dairy Products'
GROUP BY nombreCompania
ORDER BY numPedidos DESC;

```

6. La región que más empleados tiene.

```

SELECT TOP(1) descripcionRegion, COUNT(Empleados.idEmpleado) numEmpleados
FROM Empleados INNER JOIN TerritoriosEmpleado
ON TerritoriosEmpleado.idEmpleado = Empleados.idEmpleado
INNER JOIN Territorios
ON TerritoriosEmpleado.idTerritorio = Territorios.idTerritorio
INNER JOIN Region ON Territorios.idRegion = Region.idRegion
GROUP BY descripcionRegion
ORDER BY numEmpleados DESC;

```

7. Obtener el nombre de los clientes que han realizado los 10 pedidos más caros, ordenar el resultado por el pedido con mas valor.

```
SELECT TOP(10) nombreContacto
FROM Clientes INNER JOIN Pedidos ON Clientes.idCliente = Pedidos.idCliente
ORDER BY cargo DESC;
```

8. Las ganancias por categorías que se han obtenido todos los proveedores.

```
SELECT idProveedor, nombreCategoria, SUM(precioUnitario) totalGanancias
FROM Productos INNER JOIN Categorías
ON Productos.idCategoria = Categorías.idCategoria
GROUP BY idProveedor, nombreCategoria
ORDER BY idProveedor;
```

9. El proveedor con los que cuenta con mas productos descontinuados.

```
SELECT TOP(1) idProveedor, COUNT(descontinuado) AS numDescontinuados
FROM Productos
WHERE descontinuado = 1
GROUP BY idProveedor
ORDER BY numDescontinuados DESC;
```

10. Obtener el total de ganancias que se han obtenido por año y mes de todos los pedidos realizados.

```
SELECT DATENAME(YYYY, fechaPedido) Año, DATENAME(MM, fechaPedido) Mes,
COUNT(Pedidos.idPedido) NúmeroDePedidos, SUM(precioUnitario) Total
FROM Pedidos INNER JOIN DetallesPedido
ON Pedidos.idPedido = DetallesPedido.idPedido
GROUP BY DATENAME(YYYY, fechaPedido), DATENAME(MM, fechaPedido)
ORDER BY Año, Mes;
```

## Conclusión

La práctica de esta semana estuvo sencilla pues, comprendimos de mejor manera cómo se comporta el lenguaje T-SQL, pues al principio no sabíamos con exactitud como combinar las funciones GROUP BY y HAVING ni cómo organizar bien las tablas, pero al final lo logramos entender bien.

En conclusión, las funciones GROUP BY y HAVING nos ayudan a simplificar datos al momento de obtener una tabla resultante por medio de una consulta y con esto el objetivo de la práctica es alcanzado de manera exitosa.