**Ejercicios de SQL sobre BD Northwind (W3Schools)**

En la bbdd  formacion existe un esquema llamado northwind que es el que contiene las tablas. Podeis usar la que querais. Yo os recomiendo Dbeaver <https://dbeaver.io/download/>

Esta es **una colección de ejercicios con el lenguaje SQL** para que puedas usar y reforzar lo que vemos en el curso de BD ([BD - Bases de Datos](https://www.apuntesinformaticafp.com/modulos/bd.html)).

**Ejercicios con una tabla**

**--------------------------------**

**Para el ejercicio he utilizado la siguiente estructura:**

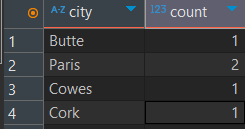
**N. Pregunta**

**Código (consulta)**

**Resultado**

**--------------------------------**

1. Revisa las tablas Customers (clientes), Employees (empleados) y Orders (pedidos)
   1. **select** \* **from** customers;
   2. **select** \* **from** Employees;
   3. **select** \* **from** Orders;
2. ¿Cuántos clientes hay registrados?
   1. **select** **count**(customer\_id) **from** customers;
      * 91 clientes registrados.
3. ¿Cuántos pedidos?
   1. **select** **count**(order\_id) **from** Orders;
      * 830 pedidos.
4. ¿Qué clientes viven en Londes y su nombre (CustomerName) empieza por A?
   1. **select** contact\_name **from** customers **where** city = 'London' **AND** **lower**(contact\_name) **similar** **to** 'a%';
      * Ann Devon como resultado.
5. ¿Cuántos clientes hay que son de Londres?
   1. **select** **count**(customer\_id) **from** customers **where** city = 'London';
      * 5
6. ¿Cuántos clientes hay en cada ciudad?
   1. **select** city, **count**(customer\_id) **from** customers **group** **by** city;
      * En este caso el resultado es muy extenso así que muestro una fracción->



1. ¿Cuántos empleados nacieron después de 1 de Junio del 1965?
   1. **select** \* **from** employees **where** birth\_date > '1965-06-01';
      * employer\_id 9 -> Dodsworth Anne
2. Hazme un informe que diga «El empleado N nación el N», siendo N, nombre (FirstName y Last Name) y día de nacimiento (BirthDate)
   1. **create** **or** **replace** **procedure** getEmployeeData (empl\_id **integer**)
   2. **language** plpgsql
   3. **as** **$$**
   4. **declare**
   5. emp\_name employees.first\_name%**type**;
   6. emp\_lastName employees.last\_name%**type**;
   7. emp\_birthDate employees.birth\_date%**type**;
   8. **begin**
   9. **select** first\_name,last\_name,birth\_date **into** emp\_name, emp\_lastName, emp\_birthDate **from** employees **where** employee\_id = empl\_id;
   10. **raise** **notice** 'El empleado ,%,%, nació el, %', emp\_name, emp\_lastName, emp\_birthDate;
   11. **end**
   12. **$$**;
   13. **call** getEmployeeData(1);
       * Tras llamarla con el id 1 de ejemplo recibo:
       * El empleado ,Nancy,Davolio, nació el, 1948-12-08
3. Hazme el informe anterior, pero sólo para los empleados con id 8, 7, 3 y 10
   1. **create** **or** **replace** **procedure** getSomeEmpData (emp\_array **INTEGER**[])
   2. **language** plpgsql
   3. **as** **$$**
   4. **declare**
   5. emp\_name employees.first\_name%**type**;
   6. emp\_lastName employees.last\_name%**type**;
   7. emp\_birthDate employees.birth\_date%**type**;
   8. **begin**
   9. **for** i **in** 1 .. **array\_length**(emp\_array,1)
   10. **loop**
   11. **select** first\_name,last\_name,birth\_date
   12. **into** emp\_name, emp\_lastName, emp\_birthDate
   13. **from** employees
   14. **where** employee\_id = emp\_array[i];
   15. **IF** emp\_name **IS** **NULL** **THEN**
   16. **RAISE** **NOTICE** 'El empleado con id % no existe', emp\_array[i];
   17. **ELSE**
   18. **raise** **notice** 'El empleado ,%,%, nació el, %', emp\_name, emp\_lastName, emp\_birthDate;
   19. **END** **IF**;
   20. **end** **loop**;
   21. **end**;
   22. **$$**;
   24. **CALL** getSomeEmpData(**ARRAY**[8,7,3,10]);
       * He implementado el procedimiento para que acepte un array con los IDs que sean, parece que el ID 10 no existe así que aviso en ese caso de que el empleado con dicho id no existe.
       * Texto

         El contenido generado por IA puede ser incorrecto.
4. Dime los paises que tengan más de 5 clientes, ordenados por el nombre de país
   1. **select** country, **count**(customer\_id) **from** customers **group** **by** country **having** **count**(country) > 5 **order** **by** country;
      * Una captura de pantalla de un videojuego

        El contenido generado por IA puede ser incorrecto.

**Ejercicios con más de una tabla**

1. Dime el nombre del cliente del pedido 10360
   1. **SELECT** customers.contact\_name
   2. **FROM** customers
   3. **INNER** **JOIN** orders **ON** customers.customer\_id = orders.customer\_id
   4. **where** order\_id = 10360;
      * Frédérique Citeaux
2. Dime el nombre completo de los clientes con los pedidos 10360, 10253 y 10440
   1. **SELECT** customers.contact\_name
   2. **FROM** customers
   3. **INNER** **JOIN** orders **ON** customers.customer\_id = orders.customer\_id
   4. **WHERE** orders.order\_id **IN** (10360, 10361, 10362);
      * Frédérique Citeaux, Laurence Lebihan, Horst Kloss
3. Dime las ciudades y número de pedidos de clientes en esa ciudad
   1. Una captura de pantalla de un celular

      El contenido generado por IA puede ser incorrecto.**select** ship\_city, **count**(order\_id)
   2. **from** orders
   3. **inner** **join** customers **on** orders.customer\_id = customers.customer\_id
   4. **group** **by** ship\_city;
4. ¿Cuales son las ciudades desde las que hay más de 7 pedidos?
   1. **select** ship\_city, **count**(order\_id)
   2. **from** orders
   3. **inner** **join** customers **on** orders.customer\_id = customers.customer\_id
   4. **group** **by** ship\_city
   5. **having** **count**(order\_id) > 7;
      * Una captura de pantalla de un videojuego

        El contenido generado por IA puede ser incorrecto.
5. ¿Cuales son los tres países desde los que tengo más pedidos?
   1. **select** ship\_city, **count**(order\_id) **as** *order\_count*
   2. **from** orders
   3. **inner** **join** customers **on** orders.customer\_id = customers.customer\_id
   4. **group** **by** ship\_city
   5. **order** **by** *order\_count* **desc**
   6. **limit** 3;
      * Tabla

        El contenido generado por IA puede ser incorrecto.
6. Necesito un informe con el Nombre completo de los Empleados y el número de pedidos que registraron
   1. **CREATE** **OR** **REPLACE** **PROCEDURE** getEmpOrders()
   2. **LANGUAGE** plpgsql
   3. **AS** **$$**
   4. **DECLARE**
   5. emp\_name employees.first\_name%**TYPE**;
   6. emp\_lastname employees.last\_name%**TYPE**;
   7. emp\_orders **INTEGER**;
   8. r\_employee RECORD;
   9. **BEGIN**
   10. **FOR** r\_employee **IN**
   11. **SELECT**
   12. e.first\_name,
   13. e.last\_name,
   14. **COUNT**(**DISTINCT** o.order\_id) **as** total\_orders
   15. **FROM** employees e
   16. **LEFT** **JOIN** orders o **ON** e.employee\_id = o.employee\_id
   17. **GROUP** **BY** e.first\_name, e.last\_name, e.employee\_id
   18. **LOOP**
   19. emp\_name := r\_employee.first\_name;
   20. emp\_lastname := r\_employee.last\_name;
   21. emp\_orders := r\_employee.total\_orders;
   23. **RAISE** **NOTICE** 'El empleado % % ha registrado %, pedidos.',
   24. emp\_name,
   25. emp\_lastname,
   26. emp\_orders;
   27. **END** **LOOP**;
   28. **END**;
   29. **$$**;
   30. **CALL** getEmpOrders();
       * El empleado Robert King ha registrado 72, pedidos.
       * El empleado Laura Callahan ha registrado 104, pedidos.
       * El empleado Anne Dodsworth ha registrado 43, pedidos.
7. En el informe anterior, sólo necesito los empleados cuyo nombre comience por M
   1. **CREATE** **OR** **REPLACE** **PROCEDURE** getEmpOrdersByM()
   2. **LANGUAGE** plpgsql
   3. **AS** **$$**
   4. **DECLARE**
   5. emp\_name employees.first\_name%**TYPE**;
   6. emp\_lastname employees.last\_name%**TYPE**;
   7. emp\_orders **INTEGER**;
   8. r\_employee RECORD;
   9. **BEGIN**
   10. **FOR** r\_employee **IN**
   11. **SELECT**
   12. e.first\_name,
   13. e.last\_name,
   14. **COUNT**(**DISTINCT** o.order\_id) **as** total\_orders
   15. **FROM** employees e
   16. **LEFT** **JOIN** orders o **ON** e.employee\_id = o.employee\_id
   17. **GROUP** **BY** e.first\_name, e.last\_name, e.employee\_id
   18. **HAVING** **lower**(e.first\_name) **similar** **to** 'm%'
   19. **LOOP**
   20. emp\_name := r\_employee.first\_name;
   21. emp\_lastname := r\_employee.last\_name;
   22. emp\_orders := r\_employee.total\_orders;
   24. **RAISE** **NOTICE** 'El empleado % % ha registrado %, pedidos.',
   25. emp\_name,
   26. emp\_lastname,
   27. emp\_orders;
   28. **END** **LOOP**;
   29. **END**;
   30. **$$**;
   31. **CALL** getEmpOrdersByM();
       * El empleado Margaret Peacock ha registrado 156, pedidos.
       * El empleado Michael Suyama ha registrado 67, pedidos.
8. Quiero saber el número de pedido, qué empleado (sólo el nombre) lo registró y el cliente.
   1. **SELECT**
   2. *e*.first\_name **AS** *"Nombre Empleado"*,
   3. *o*.order\_id **AS** *"Número de pedido"*,
   4. *c*.\*
   5. **FROM** customers *c*
   6. **INNER** **JOIN** orders *o* **ON** *c*.customer\_id = *o*.customer\_id
   7. **INNER** **JOIN** employees *e* **ON** *o*.employee\_id = *e*.employee\_id;
9. ¿Hay algún cliente que haya hecho más de un pedido registrado por el mismo empleado?
   1. **SELECT**
   2. *c*.contact\_name,
   3. *c*.customer\_id **AS** *"ID Cliente"*,
   4. **COUNT**(\*) **AS** *"Número de pedidos"*
   5. **FROM** customers *c*
   6. **INNER** **JOIN** orders *o* **ON** *c*.customer\_id = *o*.customer\_id
   7. **INNER** **JOIN** employees *e* **ON** *o*.employee\_id = *e*.employee\_id
   8. **GROUP** **BY** *c*.customer\_id
   9. **HAVING** **COUNT**(\*) > 1
   10. **ORDER** **by** *"Número de pedidos"* **DESC**;
       * Tabla

         El contenido generado por IA puede ser incorrecto.
10. Quiero saber los clientes que hayan hecho más de un pedido y que hayan sido registrado por un Empleado cuyo nombre sea Margaret.
    1. **SELECT**
    2. *e*.first\_name,
    3. *c*.customer\_id **AS** *"ID Cliente"*,
    4. **COUNT**(\*) **AS** *"Número de pedidos"*
    5. **FROM** customers *c*
    6. **INNER** **JOIN** orders *o* **ON** *c*.customer\_id = *o*.customer\_id
    7. **INNER** **JOIN** employees *e* **ON** *o*.employee\_id = *e*.employee\_id
    8. **GROUP** **BY** *c*.customer\_id, *e*.first\_name
    9. **HAVING** **COUNT**(\*) > 1 **and** *e*.first\_name = 'Margaret'
    10. **ORDER** **by** *"Número de pedidos"* **DESC**;
    11. **select** \* **from** customers **order** **by** contact\_name;

**Ejercicios con Subconsultas**

1. ¿Cuál es el producto con el precio mínimo más bajo? (usando subconsultas)
   1. **select** \* **from** products **where** unit\_price **in** (**select** **MIN**(unit\_price) **from** products);
      * **producto id 33**
2. ¿Cuál es el producto cuyo precio sea al menos 10 veces el pedido mínimo (quantity) de los pedidos (OrderDetails)?
   1. **select** \* **from** products **where** unit\_price **in** (**select** **MIN**(unit\_price)\*10 **from** products);
3. ¿Cuáles son los registros de productos (Products) cuyo precio (price) sea mayor que el máximo de los precios de los productos cuyo id sea 3, 6, 9 y 10?
   1. **select** \* **from** products **where** unit\_price >( **SELECT** **MAX**(unit\_price) **FROM** products
   2. **WHERE** product\_id **IN** (3, 6, 9, 10));
4. ¿Cuáles son los registros de productos (Products) cuyo ProductID sea un valor que esté en Shippers?ShipperID?
   1. **SELECT** \* **FROM** products **WHERE** product\_id **IN** (**SELECT** shipper\_id **FROM** shippers);
5. ¿Qué clientes (Customers) tenemos registrados, que estén en ciudades de nuestros proveedores (Suppliers)?
   1. **select** \* **from** customers **where** city **in** (**select** city **from** suppliers);
6. Seleccionar el nombre de la compañía del cliente, nombre del  
   contacto, el código de la orden de compra, la fecha de la orden de  
   compra, el código del producto,cantidad pedida del producto, nombre  
   del producto y el nombre de la compañía proveedora, usas  
   JOIN.Solamente las compañías proveedoras que comienzan con la  
   letra de la A hasta la letra G, además la cantidad pedida del producto  
   debe estar entre 23 y 187.
7. Pregunta que de valor al negocio, es decir, con los datos que dispongo, ¿que querria saber? Por ejemplo, clientes más rentables.... empresa de transporte que más uso… (kpis)

**MANUAL DE SQL**

<https://www.w3schools.com/sql/default.asp>

**EJEMPLOS SQL AVANZADOS**

<https://learnsql.es/blog/25-ejemplos-de-consultas-sql-avanzadas/>

<https://josejuansanchez.org/bd/ejercicios-consultas-sql/index.html>

**PRUEBA SQL**

Necesitamos hacer el cálculo del NPS por tienda y tipo de pregunta. Para ello clasificamos las respuestas según el valor en:

Promotores: Clientes cuya respuesta tiene un valor entre 9 y 10.  
Pasivos: Clientes cuya respuesta tiene un valor entre 7 y 8.  
Detractores: Clientes cuya respuesta tiene un valor entre 0 y 6.

El modelo de datos proporcionado es el siguiente:

**Tabla Input 1: ANSWER**cliente\_id (INTEGER), pregunta\_id (INTEGER),  codigo\_tienda (INTEGER), respuesta (INTEGER)

Tabla Input 2: QUESTION  
pregunta\_id (INTEGER), pregunta\_tipo (INTEGER), pregunta\_desc (VARCHAR (500))

**Tabla Output (3): : NPS\_VALUES**codigo\_tienda (INTEGER), pregunta\_id (INTEGER), promotores (INTEGER),  pasivos (INTEGER), detractores (INTEGER)

**EJERCICIOS:** Necesitamos que nos envíe la sentencia sql que permite crear la tabla resultante con el total de Promotores, Pasivos y Detractores que hay por Tienda y tipo de pregunta.

* Sentencias SQLs de las DDLs de las tres tablas del modelo de datos; la DDL debe incluir una breve descripción de cada campo. (2 Puntos)
* Sentencia SQL que permite construir los registros de la tabla Output a partir de los registros de las dos tablas Input, con los totales de Promotores, Pasivos y Detractores que hay por Tienda y tipo de pregunta. (5 Puntos)
* Sentencia SQL que borra la tabla Output si existe y la vuelve a crear a partir de los registros de las dos tablas Input con la sentencia SQL generada en el ejercicio 2. (3 Puntos)

**PRUEBA SQL**

Dada las tablas de las bbdd northwind:  
INPUT 1 - orders  
INPUT 2 - shippers

**tabla OUTPUT:**company\_name - string -- Nombre de la compañia de envio  
año - integer  -- año de pedido  
perc\_retrasado - float --porcentaje de pedidos retrasados contra total de pedidos del año  
perc\_ontime -float  --porcentaje de pedidos a tiempo contra total de pedidos del año  
perc adelantado -float --porcentaje de pedidos adelantados contra total de pedidos del año

Se considera un pedido retrasado cuando la fecha de envio es superior a la fecha requerida.  
Se considera un pedido a tiempo cuando la fecha de envio y la requerida son iguales.  
Se considera un pedido adelantado cuando la fecha de envio es anterior a la fecha requerida.

**1- Sentencia SQL de la ddl de la tabla OUTPUT.  
2- Sentencia SQL que genera la tabla OUTPUT a partir de las tablas INPUT.   
3- Sentencia/s SQL para insertar los datos en la tabla OUTPUT. Considerar que la tabla no se puede borrar (drop). Controlar que si se volviera a ejecutar el script no habría duplicados.**