

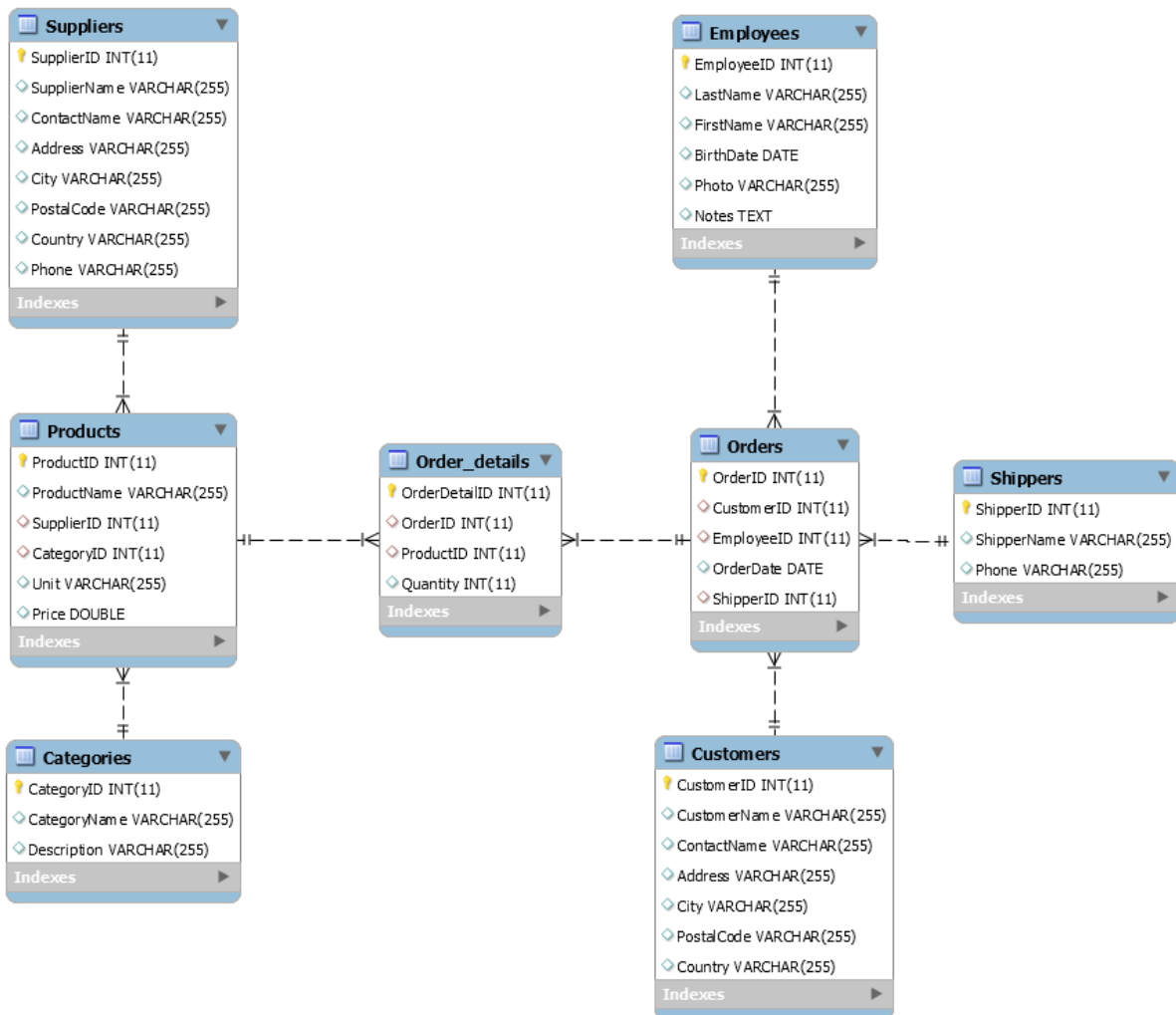
Base de datos: w3schools

[SQL Tryit Editor v1.6](#)

Script MySQL: [w3schools-database/w3schools.sql at master ·](#)

[AndrejPHP/w3schools-database · GitHub](#)

Script MS SQL Server:



De acuerdo al diagrama de base de datos previo, escribir las sentencias SQL que permitan obtener lo solicitado en cada ejercicio.

1. Por cada cliente, indicar el nombre del cliente y la cantidad de ítems que ha adquirido en total.

```
SELECT CustomerName, SUM(Quantity)
FROM Customers C
      JOIN Orders O ON C.CustomerID = O.CustomerID
      JOIN OrderDetails OD ON O.OrderID = OD.OrderID
GROUP BY CustomerName
```

2. Indicar el nombre del cliente que más ítems ha adquirido.

```
SELECT CustomerName, MAX(Total)
FROM (
    SELECT CustomerName, SUM(Quantity) Total
    FROM Customers C
        JOIN Orders O ON C.CustomerID = O.CustomerID
        JOIN OrderDetails OD ON O.OrderID = OD.OrderID
    GROUP BY CustomerName) Result
```

3. Por cada cliente, indicar el nombre del cliente, los nombres de los productos adquiridos y la cantidad de ítems por producto adquirido en total.

```
SELECT CustomerName, ProductName, SUM(Quantity) [ItemsTotal]
FROM Customers C
    JOIN Orders O ON C.CustomerID = O.CustomerID
    JOIN OrderDetails OD ON O.OrderID = OD.OrderID
    JOIN Products P ON OD.ProductID = P.ProductID
GROUP BY CustomerName, ProductName
```

4. Por cada país de procedencia del cliente, indicar el nombre del país y la cantidad de pedidos atendidos.

```
SELECT Country, COUNT (OrderID) Quantity
FROM Customers C JOIN Orders O ON C.CustomerID = O.CustomerID
GROUP BY Country
```

5. Indicar el país de procedencia del cliente con la menor cantidad de pedidos atendidos.

```
SELECT Country, MIN(Quantity)
FROM
    (SELECT Country, COUNT (OrderID) Quantity
    FROM Customers C JOIN Orders O ON C.CustomerID = O.CustomerID
    GROUP BY Country) Result
```

6. Por cada cliente indicar el nombre del cliente y la relación de los nombres de las categorías de los productos que ha adquirido.

```
SELECT DISTINCT CustomerName, CategoryName
FROM Customers C
    JOIN Orders O ON C.CustomerID = O.CustomerID
    JOIN OrderDetails OD ON O.OrderID = OD.OrderID
    JOIN Products P ON P.ProductID = OD.ProductID
    JOIN Categories CA ON CA.CategoryID = P.CategoryID
ORDER BY 1, 2
```

7. Por cada proveedor, indicar el nombre del proveedor y la cantidad de ítems de productos vendidos de dicho proveedor.

```
SELECT SupplierName, SUM(Quantity) Total
FROM Products P
    JOIN OrderDetails OD ON P.ProductID = OD.ProductID
    JOIN Suppliers S ON P.SupplierID = S.SupplierID
GROUP BY SupplierName
```

8. Indicar el nombre del proveedor con la menor cantidad de ítems de productos vendidos.

```
SELECT SupplierName, MIN(Total) FROM
(SELECT SupplierName, SUM(Quantity) Total
FROM Products P
    JOIN OrderDetails OD ON P.ProductID = OD.ProductID
    JOIN Suppliers S ON P.SupplierID = S.SupplierID
GROUP BY SupplierName)
```

9. Por cada categoría, indicar el nombre de la categoría y la cantidad de ítems de productos vendidos para dicha categoría.

```
SELECT CategoryName, SUM(Quantity) as Total
FROM Products P
    JOIN OrderDetails OD ON P.ProductID = OD.ProductID
    JOIN Categories C ON P.CategoryID = C.CategoryID
GROUP BY CategoryName
```

10. Indicar el nombre de la categoría con la mayor cantidad de ítems de productos vendidos.

```
SELECT CategoryName, MAX(Total)
FROM (
    SELECT CategoryName, SUM(Quantity) as Total
    FROM Products P
        JOIN OrderDetails OD ON P.ProductID = OD.ProductID
        JOIN Categories C ON P.CategoryID = C.CategoryID
    GROUP BY CategoryName
)
```

11. Para cada empleado que atendió al menos cinco pedidos, indicar el nombre y apellido del empleado, y la cantidad de pedidos que atendió.

```
SELECT FirstName, LastName, Count(OrderID) Quantity
FROM Employees E
    JOIN Orders O ON E.EmployeeID = O.EmployeeID
GROUP BY FirstName
HAVING Quantity > 4
```

12. Para cada orden, indicar el número de orden y el nombre del cliente que lo realizó.

```
SELECT OrderId, CustomerName
FROM Customers C
    JOIN Orders O ON O.CustomerID = C.CustomerID
```

13. Listar los nombres de los productos cuyo precio unitario sea mayor a 18.

```
SELECT ProductName
FROM Products
WHERE Price > 18
```

14. Listar los nombres de los clientes que son del país de 'Mexico'.

```
SELECT CustomerName
FROM Customers
WHERE Country='Mexico'
```

15. Listar los nombres de los productos y el nombre de su categoría.

```
SELECT ProductName, CategoryName
FROM Products
    JOIN Categories ON Products.CategoryID = Categories.CategoryID
```

16. Listar los nombres de los productos y el nombre de su proveedor.  

```
SELECT ProductName, SupplierName
FROM Products P
JOIN Suppliers S ON P.SupplierID = S.SupplierID
```
17. Listar los nombres de los productos y el nombre del proveedor que lo comercializa.  

```
SELECT ProductName, SupplierName
FROM Products P
JOIN Suppliers S ON P.SupplierID = S.SupplierID
```
18. Indicar la cantidad de clientes que han realizado pedidos.  

```
SELECT COUNT(DISTINCT CustomerID) FROM Orders
```
19. Listar el código de las categorías y la cantidad de productos en dicha categoría  

```
SELECT CategoryID, COUNT(ProductName) Quantity
FROM Products
GROUP BY CategoryID
```
20. Listar el nombre de las categorías y la cantidad de productos en dicha categoría  

```
SELECT CategoryName, COUNT(ProductID) Quantity
FROM Categories C
JOIN Products P ON C.CategoryID = P.CategoryID
GROUP BY CategoryName
```
21. Indicar el producto con el mayor precio unitario.  

```
SELECT ProductName, MAX(Price)
FROM Products
```
22. Lista los nombres de las categorías y el precio promedio de sus productos.  

```
SELECT CategoryName, AVG(Price)
FROM Products P, Categories C
WHERE P.CategoryID = C.CategoryID
GROUP BY CategoryName
```
23. Lista los nombres de las categorías, el nombre del producto con mayor precio para cada categoría y el precio del producto.  

```
SELECT CategoryName, ProductName, MAX(Price)
FROM Products P, Categories C
WHERE P.CategoryID = C.CategoryID
GROUP BY CategoryName
```
24. Lista los nombres de las categorías y la cantidad de sus productos para aquellas categorías que tienen más de cinco (5) productos.  

```
SELECT CategoryName, COUNT(ProductName) Quantity
FROM Products P, Categories C
WHERE P.CategoryID=C.CategoryID
GROUP BY CategoryName
HAVING COUNT(ProductName)> 5
```

25. Listar los nombres de los clientes que no tienen órdenes registradas.
- ```
SELECT CustomerName
FROM Customers
WHERE CustomerID NOT IN ( SELECT DISTINCT CustomerID
                        FROM Orders
                        ORDER BY 1)
```
26. Listar los nombres de los países y la cantidad de proveedores por país para aquellos países con más de dos (2) proveedores.
- ```
SELECT Country, COUNT(SupplierID)
FROM Suppliers
GROUP BY Country
HAVING COUNT(SupplierID) > 2
```
27. Por cada cliente que ha hecho por lo menos un pedido, listar el nombre del cliente y la cantidad de pedidos que ha solicitado.
- ```
SELECT CustomerName, COUNT(OrderID) Quantity
FROM Orders O, Customers C
WHERE O.CustomerID = C.CustomerID
GROUP BY CustomerName
ORDER BY 1
```
28. Indicar el nombre del cliente con más pedidos.
- ```
SELECT CustomerName, MAX(Quantity)
FROM ( SELECT CustomerName, COUNT(OrderID) Quantity
      FROM Orders O, Customers C
      WHERE O.CustomerID = C.CustomerID
      GROUP BY CustomerName
      ORDER BY 1)
```
29. Por cada Shipper indicar el nombre y la cantidad de pedidos atendidos.
- ```
SELECT ShipperName, COUNT(OrderID) Quantity
FROM Shippers S
JOIN Orders O ON S.ShipperID = O.ShipperID
GROUP BY ShipperName
```
30. Indicar el nombre del Shipper con mayor cantidad de pedidos atendidos.
- ```
SELECT ShipperName, MAX(Quantity)
FROM ( SELECT ShipperName, COUNT(OrderID) Quantity
      FROM Shippers S
      JOIN Orders O ON S.ShipperID = O.ShipperID
      GROUP BY ShipperName)
```
31. Para cada cliente, indicar el nombre del cliente y la cantidad de pedidos realizados.
- ```
SELECT CustomerName, COUNT(OrderID) Quantity
FROM Customers C
JOIN Orders O ON C.CustomerID = O.CustomerID
GROUP BY CustomerName
```
32. Listar los nombres de los empleados que no han atendido pedidos.
- ```
SELECT FirstName, LastName FROM Employees WHERE EmployeeID NOT IN
(SELECT DISTINCT E.EmployeeID FROM Employees E JOIN Orders O ON
E.EmployeeID = O.EmployeeID ORDER BY 1)
```

33. Indicar la categoría con mayor cantidad de productos vendidos.

```
SELECT CategoryName, MAX(Quantity) FROM
  (SELECT CategoryName, COUNT(OrderDetailID) Quantity
   FROM Products P
   JOIN OrderDetails OD
     ON P.ProductID = OD.ProductID
   JOIN Categories C
     ON C.CategoryID = P.CategoryID
   GROUP BY P.CategoryID)
```