

# SQL Queries for Northwind Database

```
/**
```

```
1. For each product list its name, unit price, and how many units we have in stock.
```

```
**/
```

```
SELECT
```

```
    ProductName, UnitPrice, UnitsInStock
```

```
FROM
```

```
    Products
```

```
;
```

```
/**
```

```
2. List the product name and units in stock for any product that has a units in stock greater than 10 and less than 50.
```

```
**/
```

```
SELECT
```

```
    ProductName, UnitsInStock
```

```
FROM
```

```
    Products
```

```
WHERE
```

```
    UnitsInStock > 10 AND
```

```
    UnitsInStock < 50
```

```
;
```

```
/**
```

3. List the product name, unit price for each product with a unit price greater than

\$100. Sort the list with the largest unit price on top.

```
**/
```

```
SELECT
```

```
    ProductName, UnitPrice
```

```
FROM
```

```
    Products
```

```
WHERE
```

```
    UnitPrice > 100
```

```
ORDER BY
```

```
    UnitPrice DESC
```

```
;
```

```
/**
```

4. Create a list of products that should be re-ordered (Note: the products should not be discontinued

(discontinued: 1=True; 0=False and total on hand and products on order should be less than the reorder level).

```
**/
```

```
SELECT
```

```
    ProductName, UnitsInStock, UnitsOnOrder, ReorderLevel, Discontinued
```

```
FROM
```

```
    Products
```

```
WHERE
```

```
UnitsInStock+UnitsOnOrder < ReorderLevel AND  
Discontinued <> 1  
;
```

```
/* 5.Create a list of products that have been discontinued */
```

```
SELECT  
    ProductName  
FROM  
    Products  
WHERE  
    Discontinued = 1  
;
```

```
/* 6. Create a list of all the products (prod_id and name) if all the following  
are true:
```

```
Supplierid = 2, 5, 16, 8, or 9
```

```
Categoryid = 1, 2, or 4
```

```
Unitprice > 15.00 */
```

```
SELECT  
    ProductName, SupplierID, CategoryID, UnitPrice  
FROM  
    Products  
WHERE  
    (SupplierID = '2' OR SupplierID = '5' OR SupplierID = '16' OR SupplierID =  
'8' OR SupplierID = '9')
```

```

AND
(CategoryID = '1' OR CategoryID = '2' OR CategoryID = '4')
AND
(UnitPrice > '15.00')
;
--Note the importance of using single quotes in the query

```

```

/* 8. Create a list of product names that have the second letter of the name =
?? */

```

```

SELECT
    ProductName
FROM
    Products
WHERE
    ProductName LIKE '_h%'
;
-- very handy clause. Here _ mean single character and % mean more than one
character or literally
-- anything

```

```

/* 9. Create a list of product names that have the second letter of the name =
?? and the
last letter = ?? */

```

```

SELECT

```

```
        ProductName
FROM
        Products
WHERE
        ProductName LIKE '_a%'
;
```

/\* 10. List all the customers that have one of the following fields NULL (Region or Fax).

Also the title of the contact should be Owner Sort the list by contact name\*/

```
SELECT
        *
FROM
        Customers
WHERE
        (Region IS NULL OR
        Fax IS NULL) AND
        ContactTitle = 'Owner'
ORDER BY
        ContactName
;
```

/\* 11. List each employee's name (first and last in one column) and their birthdate. Sort the list by birthdate. \*/

```
SELECT
    CONCAT(FirstName, ' ', LastName) AS fullName,
    BirthDate
FROM
    Employees
;
```

```
/* 12. Which employees were born in 1963? */
```

```
SELECT
    CONCAT(FirstName, ' ', LastName) AS FullName,
    BirthDate
FROM
    Employees
WHERE
    YEAR(BirthDate) = '1963'
;
```

```
/* 13. How many employees does Northwind have? */
```

```
SELECT
    COUNT(EmployeeID) AS countOfEmp
FROM
    Employees
;
```

```
/*14. For each customer (customer id) list the date of the first order they
placed and the
date of the last order they placed. */
```

```
SELECT
    CustomerID,
    MIN(OrderDate) AS firstOrderDate,
    MAX(OrderDate) AS lastOrderDate
FROM
    Orders
GROUP BY
    CustomerID
;
```

```
/* 15. Using question 14, only list customers where there last order was in 2011.
Sort the
list by customer. */
```

```
-- Solution 1: By optimizing previous query
```

```
SELECT
    CustomerID,
    MAX(OrderDate) AS lastOrderDate
FROM
    Orders
GROUP BY
    CustomerID
```

HAVING

YEAR(MAX(OrderDate)) = 2011

;

-- Returns empty table

-- Solution 2: By using que14 answer as SubQuery

SELECT

que\_14.CustomerID,

que\_14.lastOrderDate

FROM

(SELECT

CustomerID,

MIN(OrderDate) AS firstOrderDate,

MAX(OrderDate) AS lastOrderDate

FROM

Orders

GROUP BY

CustomerID) AS que\_14

WHERE

YEAR(que\_14.lastOrderDate) = YEAR(2011)

ORDER BY

que\_14.CustomerID

;

-- Still returns empty table

--Let's check if there are any orderDate having 2011 as OrderDateYear

SELECT



```

        *
FROM
    Orders
WHERE
    YEAR(OrderDate) = YEAR(2011)
;
-- Looks like there are not any orders from YEAR(2011)

```

```

/* 16. Which employees were born in the month of July? */

```

```

SELECT
    CONCAT(FirstName, ' ', LastName) AS FullName,
    BirthDate
FROM
    Employees
WHERE
    MONTH(BirthDate) = 07
;

```

```

/* 17. How many orders has Northwind taken? */

```

```

SELECT
    COUNT(DISTINCT(OrderID)) AS numberOfOrdersTaken
FROM
    Orders

```

```
;
```

```
/* 18. How many orders were placed per year? */
```

```
SELECT
```

```
    YEAR(OrderDate) as yearsOfOrder,
```

```
    COUNT(DISTINCT(OrderID)) AS numberOfOrders
```

```
FROM
```

```
    Orders
```

```
GROUP BY
```

```
    YEAR(OrderDate)
```

```
;
```

```
/* 19. How many orders by month for each year? Make sure the list is in order by  
year and month? */
```

```
SELECT
```

```
    YEAR(OrderDate) as yearsOfOrder,
```

```
    CASE
```

```
        WHEN MONTH(OrderDate) = 1 THEN 'Jan'
```

```
        WHEN MONTH(OrderDate) = 2 THEN 'Feb'
```

```
        WHEN MONTH(OrderDate) = 3 THEN 'Mar'
```

```
        WHEN MONTH(OrderDate) = 4 THEN 'Apr'
```

```
        WHEN MONTH(OrderDate) = 5 THEN 'May'
```

```
        WHEN MONTH(OrderDate) = 6 THEN 'Jun'
```

```
        WHEN MONTH(OrderDate) = 7 THEN 'Jul'
```

```

        WHEN MONTH(OrderDate) = 8 THEN 'Aug'
        WHEN MONTH(OrderDate) = 9 THEN 'Sep'
        WHEN MONTH(OrderDate) = 10 THEN 'Oct'
        WHEN MONTH(OrderDate) = 11 THEN 'Nov'
        WHEN MONTH(OrderDate) = 12 THEN 'Dec'
    END AS monthOfYear,

    COUNT(DISTINCT(OrderDate)) AS numberOfOrders
FROM
    Orders
GROUP BY
    MONTH(OrderDate),
    YEAR(OrderDate)
ORDER BY
    YEAR(OrderDate),
    MONTH(OrderDate)
;

```

/\* 20. Using question 19, list only the months where Northwind have less than 25 orders.\*/

```

SELECT
    YEAR(OrderDate) as yearsOfOrder,
    CASE
        WHEN MONTH(OrderDate) = 1 THEN 'Jan'
        WHEN MONTH(OrderDate) = 2 THEN 'Feb'
        WHEN MONTH(OrderDate) = 3 THEN 'Mar'
        WHEN MONTH(OrderDate) = 4 THEN 'Apr'
        WHEN MONTH(OrderDate) = 5 THEN 'May'
    
```

```

        WHEN MONTH(OrderDate) = 6 THEN 'Jun'
        WHEN MONTH(OrderDate) = 7 THEN 'Jul'
        WHEN MONTH(OrderDate) = 8 THEN 'Aug'
        WHEN MONTH(OrderDate) = 9 THEN 'Sep'
        WHEN MONTH(OrderDate) = 10 THEN 'Oct'
        WHEN MONTH(OrderDate) = 11 THEN 'Nov'
        WHEN MONTH(OrderDate) = 12 THEN 'Dec'

    END AS monthOfYear,

    COUNT(DISTINCT(OrderDate)) AS numberOfOrders
FROM
    Orders
GROUP BY
    MONTH(OrderDate),
    YEAR(OrderDate)
HAVING
    --Just added HAVING to filter as per requirement
    COUNT(DISTINCT(OrderID)) < 25
ORDER BY
    YEAR(OrderDate),
    MONTH(OrderDate)
;

```

```

/* 21. List the total amount of sales for all orders.*/

```

```

SELECT
    SUM(salesPrice) AS [Total Sales Price(no discount included)]
FROM
    (SELECT

```

```

        OrderID,
        (UnitPrice*Quantity) AS salesPrice
FROM
    [Order Details] AS tempTable
;

```

/\* 22. For each order detail, list theorderid, productid, and the total sale price (include the discount).\*/

```

SELECT
    OrderID, ProductID,
    (UnitPrice* Quantity*(1-Discount)) AS discountedPrice
FROM
    [Order Details]
;

```

/\* 23. List the total amount of sales for all orders. (with discounts included).\*/

```

SELECT
    SUM(discountedPrice) AS [Total Sales Price(discount included)]
FROM
    (SELECT
        OrderID, ProductID,
        (UnitPrice* Quantity*(1-Discount)) AS discountedPrice
    FROM

```

```
        [Order Details]
    ) AS tempTable
;
```

/\* 24. How old is each employee? List the oldest at the top of the list.\*/

```
SELECT
    CONCAT(FirstName, ' ', LastName) AS [Name of Employee],
    DATEDIFF(year, BirthDate, GETDATE()) AS Age
FROM
    Employees
ORDER BY
    Age DESC
;
```

/\* 25. Create a list of suppliers (companyname, contactname) and the products (product name) they supply.

Sort the list by supplier, then product \*/

```
SELECT
    s.CompanyName, s.ContactName,
    p.ProductName
FROM
    Suppliers as s
LEFT JOIN
    Products as p
```

ON

s.SupplierID = p.SupplierID

;

/\*26. Create a list of customers (companyname) and some information about each order

(orderid, orderdate, shipdate) they have placed.\*/

SELECT

c.CompanyName,

o.OrderID, o.OrderDate, o.ShippedDate

FROM

Orders AS o

LEFT JOIN

Customers AS c

ON

c.CustomerID = o.CustomerID

;

/\* 27. Create list of products that were shipped to customers on Jun 1997. \*/

-- Selecting orders that are shipped to customers in June'1997

SELECT

\*

FROM

Orders

WHERE

YEAR(ShippedDate) = 1997 AND

MONTH(ShippedDate) = 6

;

--Joining Orders and Order Details table to fetch product ID

SELECT

o.ShippedDate,

od.ProductID

FROM

(SELECT

\*

FROM

Orders

WHERE

YEAR(ShippedDate) = 1997 AND

MONTH(ShippedDate) = 6

) AS o

JOIN

[Order Details] AS od

ON

o.OrderID = od.OrderID

;

--Searching fetched ProductID from Products Table

SELECT

p.ProductName,

o\_merged.ShippedDate



```

FROM
    (SELECT
        o.ShippedDate,
        od.ProductID
    FROM
        (SELECT
            *
        FROM
            Orders
        WHERE
            YEAR(ShippedDate) = 1997 AND
            MONTH(ShippedDate) = 6
        ) AS o
    JOIN
        [Order Details] AS od
    ON
        o.OrderID = od.OrderID) AS o_merged
JOIN
    Products AS p
ON
    o_merged.ProductID = p.ProductID
;

```

/\* 28. Create a list of customers that have ordered Tofu. Make sure to list each customer only once.\*/

```

SELECT

```

```

        DISTINCT(c.ContactName),
        p.ProductName,
        od.OrderID
FROM
    Products AS p
JOIN
    [Order Details] as od
ON
    od.ProductID = p.ProductID
JOIN
    Orders AS o
ON
    od.OrderID = o.OrderID
JOIN
    Customers AS c
ON
    o.CustomerID = c.CustomerID
WHERE
    p.ProductName = 'Tofu'
;

```

```

/* 29. Create a list of customers that have placed and order in 1996 and 1998.
Sort the list

```

```

by customer contact. */

```

```

SELECT
    DISTINCT(joinedTable.ContactName)

```

```
FROM
    (SELECT
        c.CompanyName, c.ContactName,
        o.OrderDate
    FROM
        Orders AS o
    JOIN
        [Order Details] AS od
    ON
        o.OrderID = od.OrderID
    JOIN
        Customers AS c
    ON
        o.CustomerID = c.CustomerID
    WHERE
        YEAR(o.OrderDate) = '1998' OR
        YEAR(o.OrderDate) = '1996'
    ) AS joinedTable
ORDER BY
    joinedTable.ContactName
;
```