1. Select CategoryName and Description from the Categories table sorted by CategoryName.

Select c.CategoryName, c.Description

From Categories c

Order by c.CategoryName

1. Select ContactName, CompanyName, ContactTitle, and Phone from the Customers table sorted by Phone.

Select c.ContactName, c.CompanyName, c.ContactTitle, c.Phone

From Customers c

Order by c.Phone

1. Create a report showing employees' first and last names and hire dates sorted from newest to oldest employee.

Select e.FirstName + ' '+ e.LastName as NameOfEmployee, e.HireDate

From Employees e

Order by e.HireDate desc

1. Create a report showing Northwind's orders sorted by Freight from most expensive to cheapest. Show OrderID, OrderDate, ShippedDate, CustomerID, and Freight.

Select o.OrderID, o.OrderDate, o.ShippedDate, o.CustomerID, o.Freight

From Orders o

Order by o.Freight desc

1. Select CompanyName, Fax, Phone, HomePage and Country from the Suppliers table sorted by Country in descending order and then by CompanyName in ascending order.

Select s.CompanyName, s.Fax,s.Phone, s.HomePage, s.Country

From Suppliers s

Order by s.Country desc, s.CompanyName asc;

1. Create a report showing all the company names and contact names of Northwind's customers in Buenos Aires.

Select c.CompanyName, c.ContactName, c.City

From Customers c

Where c.City= 'Buenos Aires'

1. Create a report showing the product name, unit price and quantity per unit of all products that are out of stock.

Select c.ProductName, c.UnitPrice, c.QuantityPerUnit

From Products c

Where c.UnitsInStock=0;

1. Create a report showing the order date, shipped date, customer id, and freight of all orders placed on May 19, 1997.

Select o.OrderDate, o.ShippedDate,o.CustomerID, o.Freight

From Orders o

where o.ShippedDate='1997-05-19'

1. Create a report showing the first name, last name, and country of all employees not in the United States.

Select e.FirstName, e.LastName, e.Country

From Employees e

where e.Country not Like 'USA';

1. Create a report that shows the employee id, order id, customer id, required date, and shipped date of all orders that were shipped later than they were required.

Select o.EmployeeID, o.OrderID, o.CustomerID, o.RequiredDate, o.ShippedDate

From Orders o

where o.ShippedDate >= o.RequiredDate

1. Create a report that shows the city, company name, and contact name of all customers who are in cities that begin with "A" or "B."

Select c.City, c.CompanyName, c.ContactName

From Customers c

where c.City Like 'A%' or c.City Like 'B%';

1. Create a report that shows all orders that have a freight cost of more than $500.00.

Select \*

From Orders

where Freight>500.00

1. Create a report that shows the product name, units in stock, units on order, and reorder level of all products that are up for reorder

Select p.ProductName, p.UnitsInStock, p.UnitsOnOrder, p.ReorderLevel

From Products p

Where p.ReorderLevel>0;

1. Create a report that shows the company name, contact name and fax number of all customers that have a fax number.

Select c.CompanyName, c.ContactName, c.Fax

From Customers c

Where c.Fax is not null;

1. Create a report that shows the first and last name of all employees who do not report to anybody

Select e.FirstName , e.LastName

From Employees e

Where e.ReportsTo is null;

1. Create a report that shows the company name, contact name and fax number of all customers that have a fax number. Sort by company name.

Select CompanyName,ContactName, Fax

From Customers

Where Fax is not null

Order by CompanyName;

1. Create a report that shows the city, company name, and contact name of all customers who are in cities that begin with "A" or "B." Sort by contact name in descending order

Select City ,CompanyName,ContactName

From Customers

Where City Like 'A%' or City Like 'B%'

Order by ContactName DESC;

1. Create a report that shows the first and last names and birth date of all employees born in the 1950s.

Select FirstName,LastName,BirthDate

From Employees

Where YEAR(BirthDate) = 1950

1. Create a report that shows the product name and supplier id for all products supplied by Exotic Liquids, Grandma Kelly's Homestead, and Tokyo Traders. *Hint*: you will need to first do a separate SELECT on the Suppliers table to find the supplier ids of these three companies.

Select p.ProductName, p.SupplierID, s.CompanyName

From Products p inner join Suppliers s

on p.SupplierID =s.SupplierID

where s.CompanyName in( 'Exotic Liquids', 'Tokyo Traders')

1. Create a report that shows the shipping postal code, order id, and order date for all orders with a ship postal code beginning with "02389".

Select ShipPostalCode, OrderID, OrderDate

From Orders

where ShipPostalCode Like '02389%'

1. Create a report that shows the contact name and title and the company name for all customers whose contact title does not contain the word "Sales".

Select ContactName, ContactTitle, CompanyName

From Customers

where ContactTitle not like '%Sales%'

1. Create a report that shows the first and last names and cities of employees from cities other than Seattle in the state of Washington.

Select FirstName,LastName, City

From Employees

where City not Like 'Seattle' and Region Like 'WA'

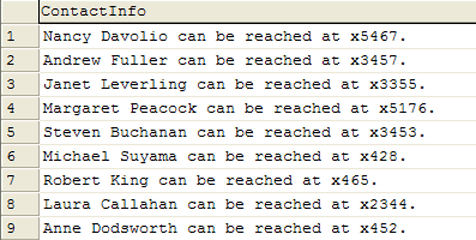
1. Create a report that shows the company name, contact title, city and country of all customers in Mexico or in any city in Spain except Madrid.

Select CompanyName,ContactTitle, City, Country

From Customers

where Country Like 'Mexico' or City in (Select City from Customers where Country like 'Spain' and City not Like 'Madrid')

1. Write a SELECT statement that outputs the following.



Select FirstName + ' ' + LastName + 'can be reached at x'+ Extension + '.' as ContactInfo

1. Create a report that shows the units in stock, unit price, the total price value of all units in stock, the total price value of all units in stock rounded down, and the total price value of all units in stock rounded up. Sort by the total price value descending.

Select ProductName,UnitsInStock, UnitPrice, SUM(UnitsInStock \* UnitPrice) as TotalPriceUnitsInStock

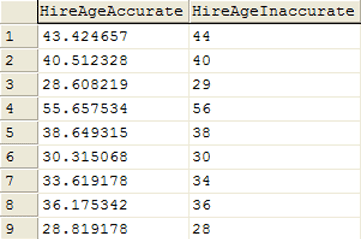
From Products

Group by UnitsInStock,UnitPrice, ProductName

Order by TotalPriceUnitsInStock desc;

1. SQL SERVER AND MYSQL USERS ONLY: In an earlier demo, you saw a report that returned the age of each employee when hired. That report was not entirely accurate as it didn't account for the month and day the employee was born. Fix that report, showing both the original (inaccurate) hire age and the actual hire age. The result will look like this.

Select YEAR(HireDate)-YEAR(BirthDate) as HireAgeInaccurate ??HireAgeAccurate

From Employees

1. Create a report that shows the first and last names and birth month (as a string) for each employee born in the current month.

Select FirstName,LastName , cast (MONTH(BirthDate) as varchar) as BirthMonth

From Employees

where datepart(mm,BirthDate) =month(getdate())

1. Create a report that shows the contact title in all lowercase letters of each customer contact.

Select LOWER(ContactTitle) as CustomerContactTitle

From Customers

1. The above SELECT statement will return the following results



1. Create a report that shows all products by name that are in the Seafood category.

Select p.ProductName

From Products p inner join Categories c

on p.CategoryID=c.CategoryID

where c.CategoryName like 'Seafood'

1. Create a report that shows all companies by name that sell products in CategoryID 8.

Select s.CompanyName, p.ProductName

From Suppliers s inner join Products p

on s.SupplierID =p.SupplierID

inner join Categories c

on c.CategoryID =p.CategoryID

where c.CategoryID=8;

1. Create a report that shows all companies by name that sell products in the Seafood category.

Select s.CompanyName, p.ProductName

From Suppliers s inner join Products p

on s.SupplierID =p.SupplierID

inner join Categories c

on c.CategoryID =p.CategoryID

where c.CategoryName like 'Seafood'

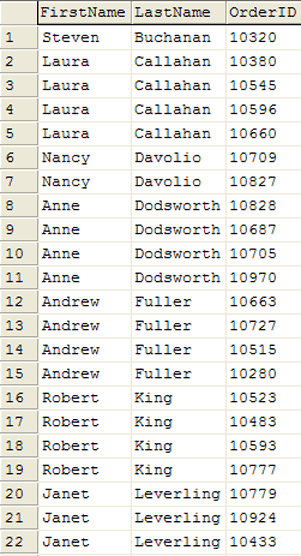
1. Create a report that shows the order ids and the associated employee names for orders that shipped after the required date. It should return the following. There should be 37 rows returned.

Select e.FirstName, e.LastName, o.OrderID

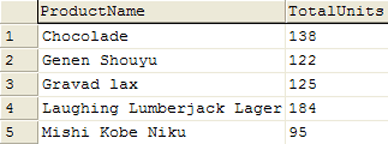
from orders o inner join Employees e

on o.EmployeeID = e.EmployeeID

where Day(o.ShippedDate) > Day(o.RequiredDate)



1. Create a report that shows the total quantity of products (from the Order\_Details table) ordered. Only show records for products for which the quantity ordered is fewer than 200. The report should return the following 5 rows.



Select p.ProductName, SUM(od.Quantity) As TotalUnits

From [Order Details] od inner join Products p

on od.ProductID = p.ProductID

group by p.ProductName

having SUM(od.Quantity)<200

1. Create a report that shows the total number of orders by Customer since December 31, 1996. The report should only return rows for which the NumOrders is greater than 15. The report should return the following 5 rows.

Select c.CompanyName, Count(o.OrderID) as NumOrders

From Orders o inner join Customers c

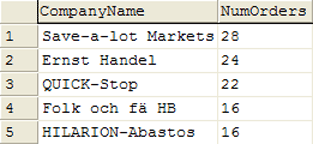
on o.CustomerID =c.CustomerID

where o.OrderDate >='1996-12-31'

group by c.CompanyName

having Count(o.OrderID) >15

order by NumOrders desc;



1. Create a report that shows the company name, order id, and total price of all products of which Northwind has sold more than $10,000 worth. There is no need for a GROUP BY clause in this report.

????

Select s.CompanyName, od.OrderID, Sum(od.UnitPrice \* od.Quantity) as TotalPrice

From Suppliers s inner join Products p

on s.SupplierID =p.SupplierID

inner join [Order Details] od

on od.ProductID=p.ProductID

group by s.CompanyName, od.OrderID

having Sum(od.UnitPrice \* od.Quantity)>10000

