

Problem: List all customers

```
SELECT *  
  
FROM Customer
```

Problem: List the first name, last name, and city of all customers

```
SELECT FirstName, LastName, City  
  
FROM Customer
```

Problem: List the customers in Sweden

```
SELECT Id, FirstName, LastName, City, Country, Phone  
  
FROM Customer  
  
WHERE Country = 'Sweden'
```

Problem: Update the city to Sydney for supplier Pavlova, Ltd.

```
UPDATE Supplier  
  
SET City = 'Sydney'  
  
WHERE CompanyName = 'Pavlova, Ltd.'
```

Problem: Delete all products with unit price higher than \$50.

```
DELETE FROM Product  
  
WHERE UnitPrice > 50
```

Problem: Discontinue products over \$50.

```
UPDATE Product  
  
SET IsDiscontinued = 1  
  
WHERE UnitPrice > 50
```

Problem: Discontinue products over \$50.

UPDATE Product

SET IsDiscontinued = 1

WHERE Id = 46

Problem: Supplier Norske Meierier (Id = 15) has moved: update their city, phone and fax.

UPDATE Supplier

SET City = 'Oslo', Phone = '(0)1-953530', Fax = '(0)1-953555'

WHERE Id = 15

Problem: Delete products over \$50.

DELETE Product

WHERE UnitPrice > 50

Problem: List all suppliers in alphabetical order

SELECT CompanyName, ContactName, City, Country

FROM Supplier

ORDER BY CompanyName

Problem: List all suppliers in reverse alphabetical order

SELECT CompanyName, ContactName, City, Country

FROM Supplier

ORDER BY CompanyName DESC

**Problem: List all customers ordered by country, then by city within each country
Ordering by one or more columns is possible.**

SELECT FirstName, LastName, City, Country

FROM Customer

ORDER BY Country, City

Problem: List top 10 most expensive products

```
SELECT TOP 10 Id, ProductName, UnitPrice, Package  
FROM Product  
ORDER BY UnitPrice DESC
```

Problem: Get all but the 10 most expensive products sorted by price

```
SELECT Id, ProductName, UnitPrice, Package  
FROM Product  
ORDER BY UnitPrice DESC
```

Problem: List all supplier countries in alphabetical order.

```
SELECT DISTINCT Country  
FROM Supplier  
ORDER BY COUNTRY
```

Problem: List all supplier countries in alphabetical order

```
SELECT DISTINCT Country  
FROM Supplier  
ORDER BY COUNTRY
```

Problem: Find the cheapest product

```
SELECT MIN(UnitPrice)  
FROM Product
```

Problem: Find the largest order placed in 2014

```
SELECT MAX(TotalAmount)
FROM Order
WHERE YEAR(OrderDate) = 2014
```

Problem: Find the last order date in 2013

```
SELECT MAX(OrderDate)
FROM Order1
WHERE YEAR(OrderDate) = 2013
```

Problem: Find the number of customers

```
SELECT COUNT(Id)
FROM Customer
```

Problem: Compute the total amount sold in 2013

```
SELECT SUM(TotalAmount)
FROM Order1
WHERE YEAR(OrderDate) = 2013
```

Problem: Compute the total amount sold in 2013

```
SELECT AVG(TotalAmount)
FROM Order1
```

Problem: Get customer named Thomas Hardy

```
SELECT Id, FirstName, LastName, City, Country
FROM Customer
WHERE FirstName = 'Thomas' AND LastName = 'Hardy'
```

Problem: List all customers from Spain or France

```
SELECT Id, FirstName, LastName, City, Country  
FROM Customer  
WHERE Country = 'Spain' OR Country = 'France'
```

Problem: List all customers that are not from the USA

```
SELECT Id, FirstName, LastName, City, Country  
FROM Customer  
WHERE NOT Country = 'USA'
```

Problem: List all orders that not between \$50 and \$15000

```
SELECT Id, OrderDate, CustomerId, TotalAmount  
FROM Order1  
WHERE NOT (TotalAmount >= 50 AND TotalAmount <= 15000)  
ORDER BY TotalAmount DESC
```

Problem: List all products between \$10 and \$20

```
SELECT Id, ProductName, UnitPrice  
FROM Product  
WHERE UnitPrice BETWEEN 10 AND 20  
ORDER BY UnitPrice
```

Problem: Get the number of orders and amount sold between Jan 1, 2013 and Jan 31, 2013.

```
SELECT COUNT(Id), SUM(TotalAmount)  
FROM Order1  
WHERE OrderDate BETWEEN '1/1/2013' AND '1/31/2013'
```

Problem: List all suppliers from the USA, UK, OR Japan

```
SELECT Id, CompanyName, City, Country
FROM Supplier
WHERE Country IN ('USA', 'UK', 'Japan')
```

Problem: List all products that are not exactly \$10, \$20, \$30, \$40, or \$50

```
SELECT Id, ProductName, UnitPrice
FROM Product
WHERE UnitPrice NOT IN (10,20,30,40,50)
```

**Problem: List all customers that are from
the same countries as the suppliers.**

```
SELECT Id, FirstName, LastName, Country
FROM Customer
WHERE Country IN
    (SELECT Country
     FROM Supplier)
```

Problem: List all products with names that start with 'Ca'

```
SELECT Id, ProductName, UnitPrice, Package
FROM Product
WHERE ProductName LIKE 'Ca%'
```

**Problem: List all products that start with 'Cha' or 'Chan'
and have one more character.**

```
SELECT Id, ProductName, UnitPrice, Package
FROM Product
WHERE ProductName LIKE 'Cha_' OR ProductName LIKE 'Chan_'
```

Problem: List all suppliers that do have a fax number

```
SELECT Id, CompanyName, Phone, Fax  
FROM Supplier  
WHERE Fax IS NULL
```

Problem: List the number of customers in each country

```
SELECT COUNT(Id), Country  
FROM Customer  
GROUP BY Country
```

Problem: List the number of customers in each country

```
SELECT COUNT(Id), Country  
FROM Customer  
GROUP BY Country  
ORDER BY COUNT(Id) DESC
```

Problem: List the total amount ordered for each customer*

```
SELECT SUM(O.TotalPrice), C.FirstName, C.LastName  
FROM order1 O JOIN Customer C  
ON O.CustomerId = C.Id  
GROUP BY C.FirstName, C.LastName  
ORDER BY SUM(O.TotalPrice) DESC
```

Problem: List the number of customers in each country. Only include countries with more than 10 customers.

```
SELECT COUNT(Id), Country  
FROM Customer  
GROUP BY Country  
HAVING COUNT(Id) > 10
```

**Problem: List the number of customers in each country, except the USA, sorted high to low.
Only include countries with 9 or more customers.**

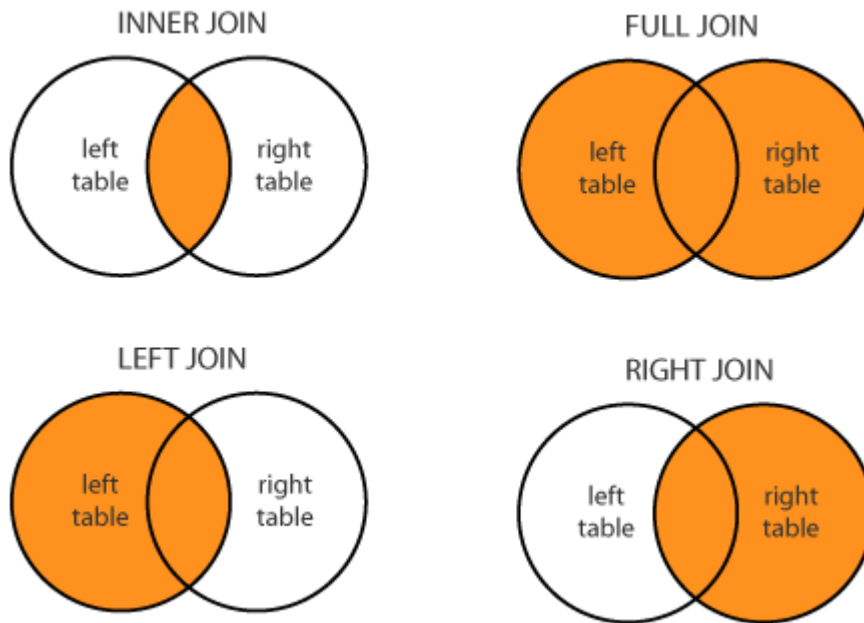
```
SELECT COUNT(Id), Country
FROM Customer
WHERE Country <>'USA' // <> not opt
GROUP BY Country
HAVING COUNT(Id) >= 9
ORDER BY COUNT(Id) DESC
```

**Problem: List total customers in each country.
Display results with easy to understand column headers. //Alias**

```
SELECT COUNT(C.Id) AS TotalCustomers, C.Country AS Nation
FROM Customer C
GROUP BY C.Country
```

**Problem: List the total amount ordered by customer
with easy to read column headers***

```
SELECT C.Id AS Identifier, C.LastName + ', ' + C.FirstName AS CustomerName,
      SUM(O.TotalAmount) AS TotalSpent
FROM order1 O JOIN Customer C ON O.CustomerId = C.Id
GROUP BY C.Id, C.LastName + ', ' + C.FirstName
ORDER BY TotalSpent DESC
```

- (INNER) JOIN: Select records that have matching values in both tables.
- LEFT (OUTER) JOIN: Select records from the first (left-most) table with matching right table records.
- RIGHT (OUTER) JOIN: Select records from the second (right-most) table with matching left table records.
- FULL (OUTER) JOIN: Selects all records that match either left or right table records.

Problem: List all orders with customer information*

```
SELECT OrderNumber, TotalAmount, FirstName, LastName, City, Country
FROM order1 JOIN Customer
ON order1.CustomerId = Customer.Id
```

The general LEFT OUTER JOIN syntax is:

```
SELECT OrderNumber, TotalAmount, FirstName, LastName, City, Country
FROM Customer C LEFT JOIN order1 O
ON O.CustomerId = C.Id
ORDER BY TotalAmount
```

Problem: List customers that have not placed orders

```
SELECT TotalAmount, FirstName, LastName, City, Country
FROM order1 O RIGHT JOIN Customer C
ON O.CustomerId = C.Id
WHERE TotalAmount IS NULL
```

Problem: Match all customers and suppliers by country

```
SELECT C.FirstName, C.LastName, C.Country AS CustomerCountry,
S.Country AS SupplierCountry, S.CompanyName
FROM Customer C FULL JOIN Supplier S
ON C.Country = S.Country
ORDER BY C.Country, S.Country
```

Problem: List all contacts, i.e., suppliers and customers.

```
SELECT 'Customer' As Type,
FirstName + ' ' + LastName AS ContactName,
City, Country, Phone
FROM Customer
UNION
SELECT 'Supplier',
ContactName, City, Country, Phone
FROM Supplier
```

Problem: List products with order quantities greater than 100.

```
SELECT ProductName
FROM Product
WHERE Id IN (SELECT ProductId
FROM OrderItem
WHERE Quantity > 100)
```

Problem: Match customers that are from the same city and country

```
SELECT B.FirstName AS FirstName1, B.LastName AS LastName1,  
       A.FirstName AS FirstName2, A.LastName AS LastName2,  
       B.City, B.Country  
FROM Customer A, Customer B  
WHERE A.Id <> B.Id  
      AND A.City = B.City  
      AND A.Country = B.Country  
ORDER BY A.Country
```

Auto Increment:

```
CREATE TABLE Persons (  
    Personid int NOT NULL AUTO_INCREMENT,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    PRIMARY KEY (Personid)  
);
```