Sommario

Text Fields vs. Numeric Fields

SQL requires single quotes around text values (most database systems will also allow double quotes).

However, numeric fields should not be enclosed in quotes:

Example

SELECT * FROM Customers

WHERE CustomerID=1;

Operators in The WHERE Clause

The following operators can be used in the WHERE clause:

Operator	Description	Example
=	Equal	
>	Greater than	
<	Less than	
>=	Greater than or equal	
<=	Less than or equal	
<>	Not equal. Note: In some versions of SQL this operator may be written as !=	
BETWEEN	Between a certain range	
LIKE	Search for a pattern	
IN	To specify multiple possible values for a column	

SQL LIKE Operator

The SQL LIKE Operator

The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

There are two wildcards often used in conjunction with the LIKE operator:

% - The percent sign represents zero, one, or multiple characters

_ - The underscore represents a single character

Note: MS Access uses an asterisk (*) instead of the percent sign (%), and a question mark (?) instead of the underscore (_).

The percent sign and the underscore can also be used in combinations!

LIKE Syntax

SELECT column1, column2, ...

FROM table_name

WHERE columnN LIKE pattern;

Tip: You can also combine any number of conditions using AND or OR operators.

WHERE CustomerName LIKE 'a__%';

The following SQL statement selects all customers with a ContactName that starts with "a" and ends with "o":

Example

SELECT * FROM Customers

WHERE ContactName LIKE 'a%o';

The following SQL statement selects all customers with a CustomerName that does NOT start with "a":

Example

SELECT * FROM Customers

WHERE CustomerName NOT LIKE 'a%';

SQL AND, OR and NOT Operators

The SQL AND, OR and NOT Operators

The WHERE clause can be combined with AND, OR, and NOT operators.

The AND and OR operators are used to filter records based on more than one condition:

The AND operator displays a record if all the conditions separated by AND are TRUE.

The OR operator displays a record if any of the conditions separated by OR is TRUE.

The NOT operator displays a record if the condition(s) is NOT TRUE.

AND Syntax

SELECT column1, column2, ...

FROM table_name

WHERE condition1 AND condition2 AND condition3 ...;

OR Syntax

SELECT column1, column2, ...

FROM table_name

WHERE condition1 OR condition2 OR condition3 ...;

NOT Syntax

SELECT column1, column2, ...

FROM table_name

WHERE NOT condition;

AND Example

The following SQL statement selects all fields from "Customers" where country is "Germany" AND city is "Berlin":

+	Add
-	Subtract
*	Multiply
/	Divide
%	Modulo

SQL Bitwise Operators

Operator	Description
&	Bitwise AND
|	Bitwise OR
^	Bitwise exclusive OR

SQL Comparison Operators

Operator	Description	Example
=	Equal to	
>	Greater than	
<	Less than	
>=	Greater than or equal to	
<=	Less than or equal to	
<>	Not equal to	

SQL Compound Operators

Operator	Description
+=	Add equals
-=	Subtract equals
*=	Multiply equals
/=	Divide equals
%=	Modulo equals
&=	Bitwise AND equals
^-=	Bitwise exclusive equals
|*=	Bitwise OR equals

SQL Logical Operators

Operator	Description	Example
ALL	TRUE if all of the subquery values meet the condition	
AND	TRUE if all the conditions separated by AND is TRUE	

NOT BETWEEN Example

To display the products outside the range of the previous example, use NOT BETWEEN:

Example

SELECT * FROM Products

WHERE Price NOT BETWEEN 10 AND 20;

BETWEEN with IN Example

The following SQL statement selects all products with a price BETWEEN 10 and 20. In addition; do not show products with a CategoryID of 1,2, or 3:

Example

SELECT * FROM Products

WHERE Price BETWEEN 10 AND 20

AND CategoryID NOT IN (1,2,3);

BETWEEN Text Values Example

The following SQL statement selects all products with a ProductName BETWEEN Carnarvon Tigers and Mozzarella di Giovanni:

Example

SELECT * FROM Products

WHERE ProductName BETWEEN 'Carnarvon Tigers' AND 'Mozzarella di Giovanni'

ORDER BY ProductName;

The following SQL statement selects all products with a ProductName BETWEEN Carnarvon Tigers and Chef Anton's Cajun Seasoning:

Example

SELECT * FROM Products

WHERE ProductName BETWEEN "Carnarvon Tigers" AND "Chef Anton's Cajun Seasoning"

ORDER BY ProductName;

NOT BETWEEN Text Values Example

The following SQL statement selects all products with a ProductName NOT BETWEEN Carnarvon Tigers and Mozzarella di Giovanni:

Example

SELECT * FROM Products

WHERE ProductName NOT BETWEEN 'Carnarvon Tigers' AND 'Mozzarella di Giovanni'

ORDER BY ProductName;

Sample Table