

4HPlus!

Use of Parentheses in an SQL Request

Purpose

To control the evaluation process of a more "complicated" SQL request.

Rules Governing Evaluation of SQL Requests

- If parentheses are nested (parentheses inside of parentheses), then requests are evaluated from the innermost set of parentheses out.
- The remaining non-nested groups of parentheses are evaluated from left to right.

Mathematical Examples

- The computer evaluates from left to right.

$$3 \times 4 + 1 = 13$$

- The order of evaluation can be changed using parentheses.

$$3 \times (4+1) = 15$$

- Extra or non-valid parentheses are ignored.

$$(3 \times (4+1)) = 15$$

Truth Tables and Joining Statements

The selection of data from a database is based on the order of the request and the joining statements used. Joining statements are **And** and **Or**. The information presented in the box below may help in evaluating requests in deciding whether to use an **And** statement or an **Or** statement.

And

T and T = True
T and F = False
F and T = False
F and F = False

Truth Tables

Or

T or T = True
T or F = True
F or T = True
F or F = False

Evaluate Joining Statements with *And*

Gender = M *And* Age = 10

- If Gender and Age are **True** (match the request), the name **is** returned.
- If Gender and Age are **False**, the name **is not** returned.
- If Gender is **False** and Age is **True**, the name **is not** returned.
- If Gender is **False** and Age is **False**, the name **is not** returned.

Evaluate Joining Statements with *Or*

Gender = M *Or* Age = 10

- If Gender is **True** and Age is **True**, the name **is** returned.
- If Gender is **True** and Age is **False**, the name **is** returned.
- If Gender is **False** and Age is **True**, the name **is** returned.
- If Gender is **False** and Age is **False**, the name **is not** returned.

Summary

When using **And** statements, **all criteria linked by And must be true** in order to return the requested information. When using **Or** statements, **only one** criteria linked by **Or** must be true to return the requested information.

Sample Request

A report of all the 10-, 11-, and 12-year old girls in the market beef project is needed.

- *List Members with Age = 10 Or Age = 11 Or Age = 12 And Projects = 10101 And Gender = F* produces the correct list of members.
- If the request is *List Members with Projects = 10101 And Age = 10 Or Age = 11 Or Age = 12 And Gender = F*, it returns incorrect data because it does not understand the criterion.
- Use the parentheses to **"group"** the **"like criteria"** such as age.
List Members with Projects = 10101 And (Age = 10 Or Age = 11 Or Age = 12) And Gender = F returns the correct list.
- Learn to use the parentheses to group "like criteria." Then, the request does not have to be entered in the exact, correct sequence order.
- Example: Produce a list of males in the 4th or 5th grade and females in the 6th or 7th grade. This is possible by grouping the "male" main set of criteria against the "female" main set of criteria. Here is the request: *List members with (Gender = M And (Grade = 4 Or Grade = 5)) Or (Gender = F And (Grade = 6 Or Grade = 7))*.

The exact information is available without worrying about the proper order. **Remember, the key to grouping within parentheses is to use them to group "like information" to select specific records.**

Don't worry too much about getting the correct number of parentheses. Code has been included in this software that adds the correct number of closing parentheses when the Process button is clicked.

All simple requests can still be entered without parentheses just as in the past. But, the addition of this feature makes it literally impossible to not find a way to get the information out of the system provided the data has been entered to begin with!