ORDER BY clause

See also

Sorts the rows retrieved by a SELECT statement.

```
ORDER BY column_reference [, column_reference...] [ASC|DESC]
```

Description

Use an ORDER BY clause to sort the rows retrieved by a SELECT statement based on the values from one or more columns. In statements that support an ORDER BY clause, use of the clause is optional.

The value for the ORDER BY clause is a comma-separated list of column names. The columns in this list must also be in the SELECT clause of the query statement. Columns in the ORDER BY list can be from one or multiple tables. If the columns used for an ORDER BY clause come from multiple tables, the tables must all be those that are part of a join. They cannot be a table included in the statement only through a SELECT subquery.

A column may be specified in an ORDER BY clause using a number representing the relative position of the column in the SELECT of the statement. Column correlation names can also be used in an ORDER BY clause columns list. Calculations cannot be used directly in an ORDER BY clause. Instead, assign a column correlation name to the calculation and use that name in the ORDER BY clause.

Use ASC (or ASCENDING) to force the sort to be in ascending order (smallest to largest), or DESC (or DESCENDING) for a descending sort order (largest to smallest). When not specified, ASC is the implied default.

The statement below sorts the result set ascending by the year extracted from the LastInvoiceDate column, then descending by the State column, and then ascending by the uppercase conversion of the Company column.

```
SELECT EXTRACT(YEAR FROM LastInvoiceDate) AS YY, State, UPPER(Company)
FROM Customer
ORDER BY YY DESC, State ASC, 3
```

See the section Relational Operators for more information on retrieving data from multiple tables in a single SELECT query.

Applicability

SELECT statements