Select:

Simple Syntax:

SELECT [DISTINCT] expression1 [alias_1] [, expression2 [alias_2], ...]
FROM table_name1 [,table_name2, ...]
[WHERE condition] [GROUP BY grouping_expr1 [, grouping_expr2, ...]]
[HAVING having_condition]
[ORDER BY order_expr1 [, order_expr2, ...]]

Select Basic Example:

SELECT lib_id FROM Readers;
/* Gibt alle lib_ids von der Tabelle Readers aus. */

SELECT * FROM Readers;
/* Gibt alles von Readers aus. */

Distinct:

SELECT DISTINCT Geburtsdatum FROM Student;
/* Vermeidet doppelte Reihen*/

Pseudonyms:

SELECT s.Vorname, s.Nachname FROM Student s;

Statische Funktionen:

1. COUNT
2. MIN
3. MAX
4. SUM
5. AVG
6. STDDEV
7. VARIANCE

COUNT(DISTINCT expression)
/* Distinct kann verwendet warden */

Numerische Ausdrücke:

1. +,-,*,/
2. Mod
3. NULL
4. Skalarfunktion NVL → ersetzt alle NULL-Werte mit dem angegebenen Wert

Skalarfunktion

SELECT NVL(lueguenr, 'Keine Liga') FROM Spieler;
/* ersetzt alle NULL-Werte mit “Keine Liga” */

SELECT/Date expressions

Datdate differences:

SELECT sysdate – pen_date FROM penalties;

Input differences:

TO_DATE('29/09/2011','DD/MM/YYYY‘)

Output function:

TO_CHAR(SYSDATE,'DD/MM/YYYY HH24:I')

Where:

Bedingungen zur eingrenzug der Suche:

Operators

=, <=, >=, <> (oder !=, oder ^=)

Alphanumeric comparison

length, ltrim, substr, upper

Comparison with NULL

IS NULL / IS NOT NULL

Combine

AND, OR, NOT

Mehrere Werte vergleichen nur mit =, <, > :

WHERE (c1,c2) = (w1,w1) → c1 = w1 AND c2 = w2

(NOT) BETWEEN …. AND

WHERE preis BETWEEN 0 AND 1000

(NOT) LIKE

WHERE a1 LIKE '%scH%'

(NOT) IN

Matrikelnr IN (1234, 2345, 3456)

Subquery (Select in Select):

(NOT) IN:

WHERE Matrikelnr IN (1234, 2345, 3456);

WHERE Matrikelnr IN (SELECT Matrikelnr FROM Student);

ALL/ANY (Wenn alle oder mind. ein Element die Bedingung erfüllt):

SELECT empno, sal FROM emp WHERE sal > ALL (2000, 3000, 4000);

(NOT) EXISTS (Es gibt mind. ein Eintrag der dir Bedingung erfüllt)

WHERE EXISTS (SELECT Matrikelnr FROM Student);

Order By:

Desc = Absteigend

Asc = Aufsteigend (Standard)

SELECT name, salary, bonus FROM employee ORDER BY salary, bonus;

Having:

Nur mit GROUP BY

SELECT to_char(order_date,'yyyy') y, count(*) FROM carpenter GROUP BY to_char(order_date,'yyyy') HAVING count(*) > 1;

Joins:

SELECT pl.plno, pl.name, pe.amount FROM players pl, penalties pe WHERE pl.plno = pe.plno

Most used Join: comparison =

(+): also list the when the column contains NULL.

SELECT pl.plno, pl.name, pe.amount FROM players pl, penalties pe WHERE pl.plno = pe.plno(+);