

## Precedentie

(...), *, /, +, -,   , < > =, IS LIKE IN, BETWEEN, !=, NOT, AND, OR	volgorde van bewerkingen
---------------------------------------------------------------------	--------------------------

## Vergelijkingsoperatoren

<, >, <=, >=	kleiner dan, groter dan, ... of gelijk aan
!=   <>   ^=	verschillend van
=	is gelijk aan
IS [NOT] NULL	is (niet) gelijk aan null
BETWEEN ondergrens AND bovengrens	tussen twee waarden, inclusief de twee waarden
IN (waarde1, waarde2,...)	is gelijk aan een van de waarden tussen de haakjes
LIKE 'patroon'	komt overeen met een patroon, hoofdlettergevoelig
%	willekeurig aantal karakters
_	één willekeurig teken
'patroon' ESCAPE 'c'	karakter c voor letterlijke tekens

## Logische operatoren

AND	vereist dat beide voorwaarden waar zijn
NOT	vereist dat de voorwaarde niet waar is (o.a. NOT IN, NOT BETWEEN, ...)
OR	vereist dat minstens één voorwaarde waar is

## Groeperende functies

AVG(expr)	Gemiddelde (NULL waarden tellen niet mee)
COUNT(expr)	aantal rijen (NULL waarden tellen niet mee)
MAX(expr)	maximum waarde
MIN(expr)	minimum waarde
SUM(n)	som

## Conditionele functies

CASE WHEN c1 THEN r1 ... ELSE rn END	Als c1 waar is dan r1... en anders rn
COALESCE(a, b, c, ...)	het eerste niet-null argument
DECODE(x, s1, r1,s2, r2,...default)	r1 als x=s1, r2 als x=s2, ...en anders default
NULLIF(expr1, expr2)	Null als expr1=expr2 en anders expr1
NVL(expr1, expr2)	expr2 als expr1 NULL is, anders expr1
NVL2(expr1, expr2, expr3)	expr2 als expr1 niet NULL is, en anders expr3

## Conversiefuncties

TO_CHAR(expr, fmt)	conversie van expr naar een tekst
TO_DATE(char, fmt)	conversie van char naar een datum
TO_NUMBER(char, fmt)	conversie van char naar een getal

## Formaatmodel

"tekst"	tekst, letterlijk in het formaat
Day	dag van de week voluit
DD	numerieke dag van de maand
DY	dag van de week in drie letters
Fm	fill mode: verwijdert opvulspaties of onderdrukt beginnullen
HH of HH12 of HH24	Uur van de dag, of uur (1-12), of uur (0-23)
MI	minuten (0-59)
MON	maand, drie letters
Month	maand, voluit geschreven
MM	maand, in 2 cijfers
RR	jaar in 2 cijfers
SCC of CC	eeuw
SP en SPTH	uitgespeld getal, bijv. 'vier' en 'vierde'
SS	seconden (0-59)
SSSSS	seconden na middernacht
TH en THSP	ordinaal getal, bijv. '4e' en 'vierde'
Year	jaar uitgespeld
YYYY	jaar in 4 cijfers

## Getallen

* + - /	rekenkundige expressies
MOD(deler, deeltal)	modulo, de rest van deler na deling door deeltal
ROUND(getal[, n])	rondt getal af op n cijfers na de komma (geen n is 0, n < 0 is voor de komma).
TRUNC(getal[, n])	kapt getal af op n cijfers na de komma (geen n is 0, n < 0 is voor de komma).

## Tekst

'There''s Oracle SQL'	Tekst, hoofdlettergevoelig, 2 keer ' voor een aanhalingsteken
	concatenatie
CONCAT(c1, c2)	voegt de tekst c1 en c2 samen
INITCAP(tekst)	zet de eerste letter van elk woord in hoofdletters, de rest in kleine letters
INSTR(c1, c2[, start[, n]])	vindt de positie van het n <sup>de</sup> voorkomen van c2 in c1, vanaf positie start. Het eerste karakter heeft positie 1. Negatieve argumenten tellen vanaf het eind.
LENGTH(tekst)	geeft aantal karakters terug
LOWER(tekst)	zet tekst om naar kleine letters
LPAD(c1, n, c2)	vult c1 wordt links aan tot lengte n met karakters van c2
REPLACE(tekst, s, r)	leder voorkomen van s in tekst vervangen door r
RPAD(c1, n, c2)	vult c1 rechts aan tot lengte n met karakters van c2
SUBSTR(tekst, start[, n])	tekstdeel uit tekst vanaf positie start, n karakters lang (of tot het einde). Het eerste karakter heeft positie 1. Negatieve argumenten tellen vanaf het eind.
TRIM([c1 FROM] c2)	verwijdert karakters c1 of spaties aan het begin en eind van c2
UPPER(tekst)	zet tekst om naar hoofdletters

## Datums

datum + n	telt n dagen op bij datum (decimaal getal voor uren, minuten, ...)
datum1 – datum2	geeft aantal dagen (getal) tussen twee datums (kommagetal bij uren, ...)
ADD_MONTHS(datum, n)	telt n maanden op bij datum (n mag negatief zijn)
CURRENT_DATE	datum en tijd in de tijdzone van de huidige sessie (lokaal)
LAST_DAY(datum)	de laatste dag van de maand waarin datum valt
MONTHS_BETWEEN(dt1, dt2)	aantal maanden tussen twee datums, positief als dt1 na dt2 valt (cijfers na de komma stellen een deel van de maand voor).
NEXT_DAY(datum, 'dag')	de eerste weekdag 'dag' (tekst of getal) gelijk aan of later dan datum
ROUND(datum[, 'fmt'])	rondt datum af tot dichtstbijzijnde eenheid fmt (of 'dag')
SYSDATE	datum en tijd van de database
TRUNC(datum[, 'fmt'])	kapt datum af op eenheid fmt (of dag)

## Substitutievariabelen

&var	variabele var, letterlijk in te voegen in de SQL de waarde wordt gevraagd aan de gebruiker indien nodig
'&var'	variabele var, gebruikt als tekst (hoofdlettergevoelig) of datum
&&var	variabele var, onthoud de waarde voor hergebruik
DEFINE var = value	variabele var de waarde value toewijzen, onthoud de waarde
SET VERIFY [ON OFF]	toon wel/niet de SQL met de variabelen ingevuld
UNDEFINE var	variabele var vergeten

## Algemeen

*	alle kolommen
AS "Kolom naam"	kolom alias, AS is optioneel
DESCRIBE	weergave tabelstructuur
DISTINCT	onderdrukt dubbels
DUAL	ingebouwde dummy tabel
FROM tabel1 t1	gebruik data uit tabel1, met alias t1
GROUP BY	waarop moeten rijen gegroepeerd worden
HAVING voorwaarde	filtert de rijen na groepering met de voorwaarde
tabel1 t1 JOIN tabel2 t2 USING veldnaam	
tabel1 t1 JOIN tabel2 t2 ON voorwaarde	
...LEFT OUTER JOIN...	alle velden linkertabel worden getoond
...RIGHT OUTER JOIN...	alle velden rechtertabel worden getoond
...FULL OUTER JOIN...	alle velden van beide tabellen worden getoond
...CROSS JOIN...	product van twee tabellen
tabel1 t1, tabel2 t2	product van twee tabellen
ORDER BY criteria1, criteria2, ...	sorteert (oplopend) op basis van kolom, kolomalias, kolompositie of een expressie.
criteria [ASC DESC] [NULLS FIRST LAST]	af- of oplopend, met lege waarden eerst of laatst.
SELECT t1.kolom	selectie uit tabel(len)
SHOW ALL	toon alle systeemvariabelen.
WHERE voorwaarde	filtert de rijen met de voorwaarde