

4.3 Functies

- Functie: naam gevolgd door () met een of meerdere argumenten
- Vrijwel overal toepasbaar in query
- 6 groepen
 - ▣ Numerieke functies toepasbaar op numerieke gegevens
 - ▣ Alfnumerieke functies toepasbaar op alfanumerieke gegevens
 - ▣ Datumfuncties toepasbaar op datum
 - ▣ Algemene functies toepasbaar op ieder datatype
 - ▣ Conversiefuncties conversie naar een ander datatype
 - ▣ Groepsfuncties toepasbaar op een groep gegevens

4.3.1 Rekenfuncties

Belangrijkste functies om met cijfers te werken:

- `ROUND(n[,m])` rondt n af op m decimale posities
- `CEIL(n)` rondt n naar boven af op een geheel getal
- `FLOOR(n)` rondt n naar beneden af op een geheel getal
- `TRUNC(n[,m])` kapt n af op m decimale posities
- `ABS(n)` de absolute waarde van n
- `SIGN(n)` -1, 0 of 1 als n negatief, nul of positief is
- `SQRT(n)` vierkantswortel uit n (*square root*)
- `POWER(n,m)` n tot de m -de macht
- `MOD(n,m)` rest na deling van n door m

4.3.1 Rekenfuncties

- `SELECT ROUND (345.678) FROM dual`
 - ▣ 346 (standaard: geen cijfers na de komma)
 - `SELECT CEIL(345.678), CEIL(345.278) FROM dual`
 - ▣ 346 346
 - `SELECT FLOOR (345.578) FROM dual`
 - ▣ 345
 - `SELECT ROUND(345.678, 2) FROM dual`
 - ▣ 345.68
 - `SELECT ROUND (345.678, -1) FROM dual`
 - ▣ 350
- (-1=tiental, -2=honderdtal, -3=duizendtal,...)

4.3.1 Rekenfuncties

- `SELECT TRUNC(345.678, 2) FROM dual`
 - ▣ 345.67
- `SELECT ABS(-123) , ABS(0), ABS(456) FROM dual`
 - ▣ 123/0/456
- `SELECT SIGN(-13) , SIGN(0), SIGN(456) FROM dual`
 - ▣ -1/0/1
- `SELECT SQRT(16), SQRT(8), SQRT(4) FROM dual`
 - ▣ 4/2.8284271/2
- `SELECT POWER(2, 3), POWER(-2,3) FROM dual`
 - ▣ $2^3=8$ $-2^3=-8$

4.3.1 Rekenfuncties

- `SELECT MOD(8,3) FROM dual` → 2
- `SELECT SIGN(maandsal-1600)`
`FROM medewerkers`
 - ▣ 1/0/-1/... (welke groter, kleiner dan 1600)
- `SELECT *`
`FROM medewerkers`
`WHERE MOD (mnr, 2) = 0` → *even nummers....*
- `SELECT naam`
`, sysdate-gbdatum "aantalDagenOud"`
`, floor ((sysdate-gbdatum)/7) "aantalWekenOud"`
`, floor (mod(sysdate-gbdatum,7)) as dagen`
`from medewerkers`

4.3.2 Tekstfuncties

- **LENGTH(t)** aantal karakters (lengte) van *t*
select LENGTH('hi all') from dual; → 6
- **ASCII(t)** ascii-waarde eerste karakter van *t*
- **CHR(n)** karakter met ascii-waarde *n*
- **UPPER(t)** *t* in hoofdletters
select UPPER('Hi all') from dual; → HI ALL
- **LOWER(t)** *t* in kleine letters
select UPPER('Hi ALL') from dual; → hi all
- **INITCAP(t)** elke woord in *t* met beginhoofdletter
select INITCAP('Hi jean-marie') from dual → Hi Jean-Marie

4.3.2 Tekstfuncties

- LTRIM(*t*,*k*) verwijdt links van *t* de *k*-karakters, t.e.m. het eerste karakter niet in *k*
stopt als het teken niet gevonden is: ltrim('marteress', 'ram') → teress
- RTRIM(*t*,*k*) verwijdt rechts van *t* de *k*-karakters , na het laatste karakter niet in *k*
stopt als een teken niet gevonden is: rtrim('marteress', 'sen') → marter
- LPAD(*t*,*n*) vult *t* links uit met spaties tot lengte *n*
- LPAD(*t*,*n*,*k*) idem, met *k*-karakter
select lpad('', 10, '-') from dual; → -----**
- RPAD(*t*,*n*) vult *t* rechts aan met spaties tot lengte *n*
- RPAD(*t*,*n*,*k*) idem, met *k*-karakter
select rpad('', 10, '-') from dual; → *-----*

4.3.2 Tekstfuncties

- SUBSTR(*t*,*n*)
geeft deel van *t* vanaf positie *n* tot het einde
select substr('kelder', 2) from dual; → elder
- SUBSTR(*t*,*n*,*m*)
geeft deel van *t* vanaf positie *n*, *m*-karakters lang
select substr('kelder', 2, 3) from dual; → eld
- INSTR(*t*,*k*)
positie eerste voorkomen van *k* in *t*
select instr('kelder', 'e') from dual; → 2
- INSTR(*t*,*k*,*n*)
idem, op of na de *n*-de positie in *t*
select instr('kelder', 'e', 3) from dual; → 5
- INSTR(*t*,*k*,*n*,*m*)
het *m*-de voorkomen van *k*, vanaf de *n*-de positie in *t*
select instr('kelderen', 'e', 3, 2) from dual; → 7

4.3.2 Tekstfuncties

- REPLACE(*t*,*v*) verwijdt uit *t* elk voorkomen van *v* (woorden)
select replace('hello world', 'l') from dual; → heo word
- REPLACE(*t*,*v*,*w*) vervangt in *t* elk voorkomen van string *v* in *w* (woorden)
select replace('hello world', 'l', '') from dual;*
*→ he**o wor*d*
- TRANSLATE(*t*,*v*,*w*) vervangt alle **karakters** uit *v* die in *t* voorkomen door het corresponderende karakter uit *w*
select translate('veer', 've', 'bo') from dual;
→ boor
- CONCAT(*t*1,*t*2) voegt *t*1 en *t*2 samen (equivalent met ||)
select concat('boor', 'machine') from dual;
→ boormachine

4.3.2 Tekstfuncties

- `SELECT UPPER(omschrijving), LOWER(type)`
`FROM cursussen`
 - ▣ `INTRODUCTIE... alg`

- `SELECT anr, naam, INITCAP(locatie)`
`FROM afdelingen ORDER BY LENGTH(naam)`

- `SELECT *`
`FROM medewerkers`
`WHERE LOWER (functie) = 'trainer'`
 - ▣ `SWINNEN` `TRAINER ...`

| ANR | NAAM | INITCAP(LOCATIE) |
|-----|-----------------|------------------|
| 30 | VERKOOP | Genk |
| 20 | OPLEIDINGEN | Hasselt |
| 10 | HOOFDKANTOOR | Maasmechelen |
| 40 | PERSONEELSZAKEN | Leuven |

4.3.2 Tekstfuncties

- `SELECT ASCII('a'), ASCII('z'), chr(77) FROM dual`
 - ▣ 97/122/M
- `SELECT SUBSTR(naam,4), SUBSTR(naam,4,3) FROM afdelingen`
 - ▣ hoofdkantoor → fdkantoor fdk
 - ▣ Personeelszaken → soneelszaken son

4.3.2 Tekstfuncties

□ SELECT naam,
INSTR(naam,'A')
FROM medewerkers

□ ALLARD:

■ 1

□ JACOBS:

■ 2

□ DE COOMAN

■ 8

□ SELECT naam,
INSTR(naam,'A',3)
FROM medewerkers

□ ALLARD:

■ 4

□ JACOBS:

■ 0

□ DE COOMAN

■ 8

□ SELECT naam,
INSTR(naam,'A',1,2)
FROM medewerkers

□ ALLARD:

■ 4

□ JACOBS:

■ 0

□ DE COOMAN

■ 0

4.3.2 Tekstfuncties

□ SELECT

LTRIM(naam,'SDAER')
FROM medewerkers

□ CASPERS

■ CASPERS

□ ALLARD

■ LLARD

□ DEN RUYTER

■ N RUYTER

□ SELECT

RTRIM(naam,'SDAER')
FROM medewerkers

□ CASPERS

■ CASP

□ ALLARD

■ ALL

□ DEN RUYTER

■ DEN RUYT

4.3.2 Tekstfuncties

- `SELECT LPAD(naam,8,'@'), RPAD (naam,10,'=')`
`FROM medewerkers`
 - ▣ `@@JACOBS` `JACOBS=====`

- `SELECT LPAD (maandsal,4) || ' ' || RPAD ('=',maandsal/100,'=')`
`FROM medewerkers`
 - ▣ `1800 =====` *(18 =-tekens)*
 - ▣ `1300 =====` *(13 =-tekens)*
 - ▣ `4000 =====`
 - ▣ `2500 =====`

4.3.2 Tekstfuncties

- `SELECT TRANSLATE(code, 'AESOL', '12345') FROM cursussen`
 - ▣ `SQL → 3Q5`

 - `SELECT REPLACE (omschrijving, 'SQL', 'Visual C#') FROM cursussen`
 - ▣ `INTRODUCTIE SQL → INTRODUCTIE Visual C#`
-
- **OEFENING REKENFUNCTIES & TEKSTFUNCTIES:
p29 oef 1-10 "4.4.3 Functies"**