|  |  |
| --- | --- |
| **Rekenen** |  |
| 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* (vergelijken) |
| SIGN(n) | -1, 0 of 1 als *n* negatief, nul of positief is |
| SQRT(n) | vierkantswortel uit *n* |
| POWER(n,m) | n tot de *m*-de macht |
| MOD(n,m) | rest na deling van *n* door m |
| **Alfanumeriek** |  |
| LENGTH(t) | aantal karakters (lengte) van *t* |
| ASCII(‘t’) | ascii-waarde eerste karakter van *t* |
| CHR(n) | karakter met ascii-waarde *n* |
| UPPER(t) | *t* in hoofdletters |
| LOWER(t) | *t* in kleine letters |
| INITCAP(t) | elke woord in *t* met beginhoofdletter |
| LTRIM(t,’k’) | verwijdert links van *t* de *k*-karakters |
| RTRIM(t,’k’) | verwijdert rechts van *t* de *k*-karakters |
| LPAD(t,n) | vult *t* links uit met spaties tot lengte *n* |
| LPAD(t,n,k) | idem, met *k*-karakter |
| RPAD(t,n) | vult *t* rechts aan met spaties tot lengte *n* |
| RPAD(t,n,k) | idem, met *k*-karakter |
| SUBSTR(t,n)  SUBSTR(naam, 8) | geeft deel van *t* vanaf positie *n* tot het einde  //dit schrijft de naam vanaf de 8ste positie. |
| SUBSTR(t,n,m) | geeft deel van *t* vanaf positie *n*, *m*-karakters lang |
| INSTR(t,’k’)  INSTR(naam, ‘kik‘) | positie eerste voorkomen van *k* in *t*  //dit geef het getal positie wanneer ‘kik‘ is voorgekomen |
| INSTR(t,’k’,1) | idem, op of na de 1-de positie in *t* |
| INSTR(t,k,1,2) | het *m*-de voorkomen van *k*, op of na de *n*-de positie in *t* |
| REPLACE(t,v) | *verwijdert* uit *t* elk voorkomen van *v* (woorden) |
| REPLACE(t,v,w) | *vervangt* in *t* elk voorkomen van string *v* in *w* (woorden) |
| TRANSLATE(t,v,w)  TRANSLATE(tabelnaam, ‘XXxx#’, ‘XXxx-‘) | vervangt alle *karakters* uit *v* die in *t* voorkomen door het corresponderende karakter uit *w* |
| CONCAT(t1,t2) | concateneert t1 en t2 (equivalent met ||) |
| **Andere** |  |
| || | Iendien we iets willen toevoegen |
| INITCAP(x) | Alles in kleine letters buiten de eerste |
| UPPER(x) | Alles in hoofdletters |
| LOWER(x) | Alles in kleine letters |
| DECODE(tabelnaam,  XX, ‘Y’,  XX, ‘Y’,  XX, ‘Y’,  XX, ‘Y’,  ‘Y’) | Decode een tabel |
| CASE xx WHEN ‘xx THEN yy WHEN ‘xx THEN yy WHEN ‘xx THEN yy WHEN ‘xx THEN yy ELSE yy END; | Indien het beide strings zijn zet ‘ ‘ bij beide |
| DISTINCT |  |
| INPUT |  |
| LIKE % % |  |
| BETWEEN XX AND XX |  |
| IN (XX, XX) |  |
| NULLIF(a,b) | geeft NULL als a=b, anders a |
| col histogram format a25; | //maak een colom die histogram noemt, maak deze 25 chars lang |
| X UNION X | Colum + colum |
| X MINUS X | Colum - colum |
| GREATEST(a,b,…) | grootste waarde uit de argumenten |
| LEAST(a,b, …) | kleinste waarde uit de argumenten |
| COALESCE(a,b,…) | Retourneert het eerste argument dat niet NULL is (en NULL als ze allemaal NULL zijn) |
| NVL(x,y) | geeft *y* als *x* NULL is, anders *x* zelf |
| create or replace view XX as  select … | Maak een nieuwe view aan |
| m.tabelnaam, a.tabelnaam  from tabel1 m  join table2 a  on m.tabelnaam = a.tabelnaam | Op de laatste rij komt de connectie te staan.  (bv m.afd = a.anr) |
| Right join | Geeft alles van rechter kollom |
| Left join | Geeft alles van linker kollom |
| ADD\_MONTHS(d,n) | datum *d* plus *n* maanden |
| MONTHS\_BETWEEN(d,e) | maanden verschil tussen *d* en *e* |
| LAST\_DAY(d) | laatste dag van de maand waarin *d* valt |
| NEXT\_DAY(d,wkdag) | de eerste weekdag (ma, di, …) na *d* |
| NEXT\_TIME (d,z1,z2) | converteert datum/tijd van tijdzone1 naar tijdzone2 |
| ROUND(d[,fmt]) | *d* afgerond op *fmt* (default middernacht) |
| TRUNC(d[,fmt]) | *d* afgekapt op *fmt* (default middernacht) |
| EXTRACT(c FROM d) | extraheert component c uit expressie d |
| TO\_CHAR(n[,fmt]) | zet getal *n* om naar een string |
| TO\_CHAR(d[,fmt]) | zet datum *d* om naar een string |
| TO\_NUMBER(t) | zet string *t* om naar een getal |
| TO\_DATE(t[,fmt]) | zet *t* om naar een datum |
| DROP VIEW XX |  |
| update tabel  set tabelnaam = ‘xx’  where xx |  |
| delete from tabel  where code = tabelnaam; |  |
| insert into tabel  (mnr, voorn, naam, gbdatum, functie, maandsal, chef)  values  (7999, 'Willem',' 'Revis', date '1983-01-21', 'BOEKHOUDER', 2950, 7782) |  |
| set nls\_date\_format = “dd-mm-yyyy” | standaard datumformaat |
| Set nls\_language=Dutch | taal voor boodschappen |
| Set nls\_currency='EUR' | symbool voor geldbedrag |
| Set nls\_numeric\_characters=', ' | ',' decimale scheiding en  blanco: scheiding voor duizendtal |
| DATE '1977-05-01' | *standaard datumformat: '01-MAY-77 '* |
| DDSP | Zet cijfers om in woorden |
|  |  |
| ORDER BY | Sorteren op |
| GROUP BY | Groeperen op |
| … having count(tabelnaam) > 1 | If statement  Indien groter dan 1 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |