Samenvatting Data

# Hoofdstuk 1: Inleiding van basisbegrippen

**Wat is een databank?**

Een digital opgeslagen archief, ingericht met oog op het makkelijk terugvinden van informatie, en een flexibel gebruik.

**Aan welke voorwaarden moet de databank voldoen om echt te kunne spreken van een databank?**

* Gegevens moeten eenvoudig worden opgeslagen
* Gegevens moeten eenvoudig worden op of doorgezocht
* Gegevens moeten gewijzigd kunnen worden
* Gegevens moeten verwijderd kunnen worden (zonder dat dit de werking van de db beinvloed)

**Wat is een index?**

Een index is de data die uniek is en er voor zorgt dat alle gegevens optimaal doorzoekbaar zijn.

**Wat is een datamodel?**

Hetgene dat beschrijft hoe de gegevens in een database of informatiesysteem gestructureerd zijn.

**Welke soorten datamodellen ken je?**

* Platte datamodel (1ste generatie)
* Hierargische model (1ste generatie)
* Het netwerkmodel (2de generatie)
* Rationele databank (3de generatie)
  + Deze gebruiken wij. Het is een verzameling van tabellen die relatries hebben met elkaar. Het verband tussen de tabellen wordt niet expliciet gedefinieerd maar aangeduid door eigenschappnen tussen verschillende tabellen.
* Objectgeorienteerde databank (4de generatie)

**Wat is het verschil tussen informatie en data?**

|  |  |
| --- | --- |
| **Informatie** | **Data** |
| Gestructureerde en betekenisvolle relatie met data | Betekenisloze gegevens zonder informatie |

**Uit welke componenten staat een gegevensverzameling?**

* Groep (Tabel)
* Gegevens (Kolommen in de tabel)
* Waarden (De data die in de kolommen staat)
* Occurence/Voorkomen (1 rij in de tabel)

# Hoofdstuk 2: Structured Query Language

## 2.1 Kennismaking met SQL

**Structured Querry Language SQL**

Is een query taal om een groep gegevens af te halen of te wijzigen. Deze wordt toegepast op 2 manieren. Interactief (sql commando rechtstreeks ingeven en het resultaat krijgen) en embedded (sql commando binnen een programmeertaal)

**De onderverdelingen in SQL:**

* Gegevensdefinitie (DDL -> Data Definition Language)
  + Tabel maken, wijzigen of verwijderen (Create table, Alter table, Drop table)
* Gegevensmanipulatie (DML -> Data Manipulation Language)
  + Groepen rijen invoegen, bijwerken en verwijderen (Delete, insert, update, commit, rollback)
* Raadpleging of Query
  + Raadplegingen van databasegegevens (selectiequery) op tabelniveau -> select
* Beveiliging of DCL (Data Control Language)
  + Toegang tot de database met een wachtwoord
  + Gevolg -> Privileges (toegang tot gegevens in detail regelen)
    - Objectprivileges -> recht om een bepaald object te benadeen bv insert, select, delete, update table
    - Systeemprivileges -> Recht om bepaalde acties te ondernemen bv aanloggen, tabel maken, ....
  + Create user, alter user, drop user, grant(privileges regelen) en revoke (privileges ontnemen)

## 2.2 Basisbegrippen

**De verschillende basisbegrippen**

|  |  |
| --- | --- |
| Constante | Numeriek, alfanumeriek, datum |
| Variabelen | Kolomvariabelen en systeemvariabelen |
| Operatoren | Rekenkundige operatoren (+, -, \*, /) |
| Alfanumerieke operator | || -> bv ‘Patricia’ || ‘Briers’ -> ‘Patricia Briers’ |
| Vergelijkingsoperatoren | <, >, =, <=, >=, != |
| Logische operatoren | AND, OR, NOT |

**De voorrangsregel bij de operatoren:**

Rekenkundig > Vergelijking > Logisch

Naamgeving van tabellen en kolommen:

Beginnen met een letter, max 30 char, geen verschil tussen kleine en hoofdletters

/\* commentaar \*/ -- verhelderend voorbeeld

# Hoofdstuk 3: SQL\* Plus

## 3.2 De SQL-Buffer

Deze bevat de laatst ingegeven SQL-Commando.

Select \* from cat; -> uitgevoerd en in de buffer

Select \* from cat -> zit in de buffer en wordt nog niet uitgevoerd

Een commando in de buffer uitvoeren doe je met /

## 3.4 SQL\* Plus – Editor

Dit is een regeleditor, die 1 regel actueel wijzigd gemarkeerd met \*

De SQL\*Plus commandos :

* A(ppend) name -> toevoegen aan dehuidige regel
* C(hange) /van /naar -> wijzigen
* Del -> Verwijdert alle regels
* Del 1 -> verwijdert regel 1
* Del 1 5 -> verwijdert regel 1 tot en met 5
* I(nput) -> toevoegen van een nieuwe regel)
* L(ist) -> Geeft alle regels in de buffer
* L 5 of 5 -> geeft regel 5
* L5 10 -> geeft regel 5 tot en met 10
* L\* Geeft de huidige regels

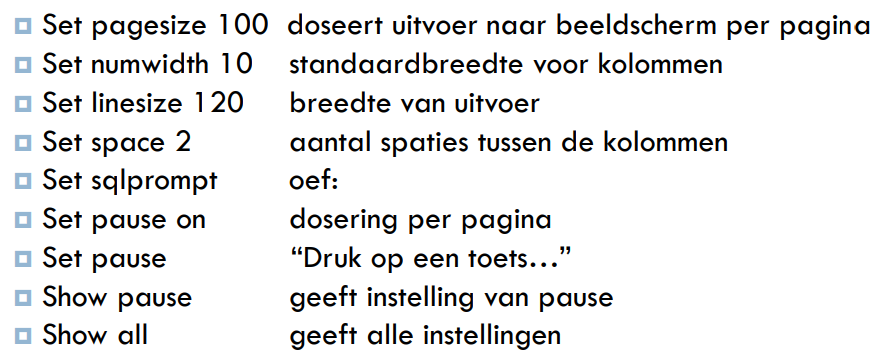
## 3.5 Commando’s bewaren

Inhoud van de buffer bewaren in een txt bestand kan je op verschillende manieren

* Via buffer (afiedt.buf)
* Via Commando (Save met optie append /replace bv save “c:\oracle\oefening append”)
  + Terug ophalen in buffer -> get “c:\oracle\oefening”
  + Onmiddelijk uitvoeren -> start “c:\oracle\oefening”
  + Commando “Spool” -> Volledige schermuitvoer bewaren -> spool test -> get test.lst opent de opgeslagen scherm.

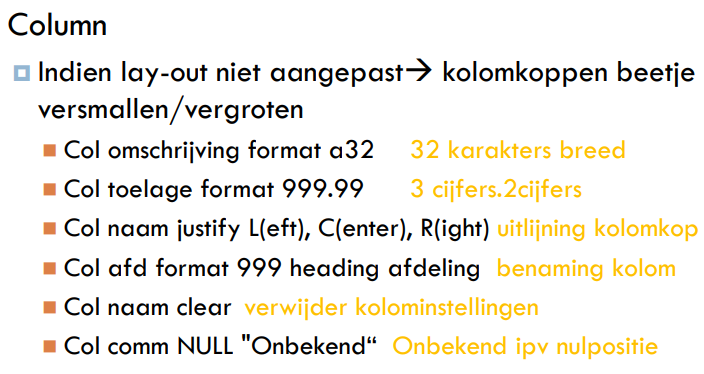
## 3.6 SQL\*Plus – Instellingen

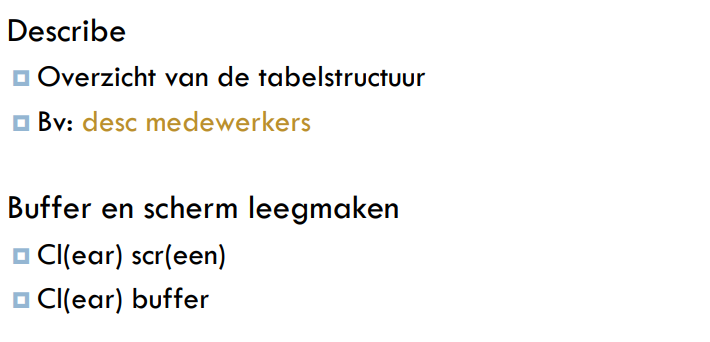
Instellingen voor de omgeving te wijzigen





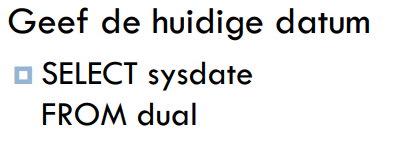
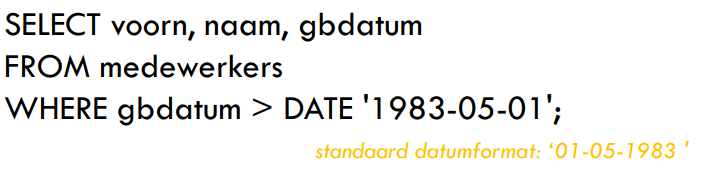
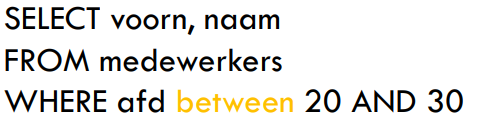
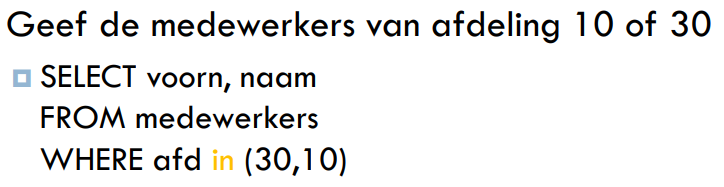
## 3.7 Overige SQL\*Plus – Commando’s

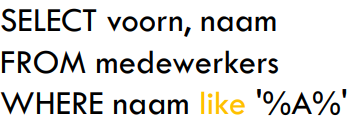


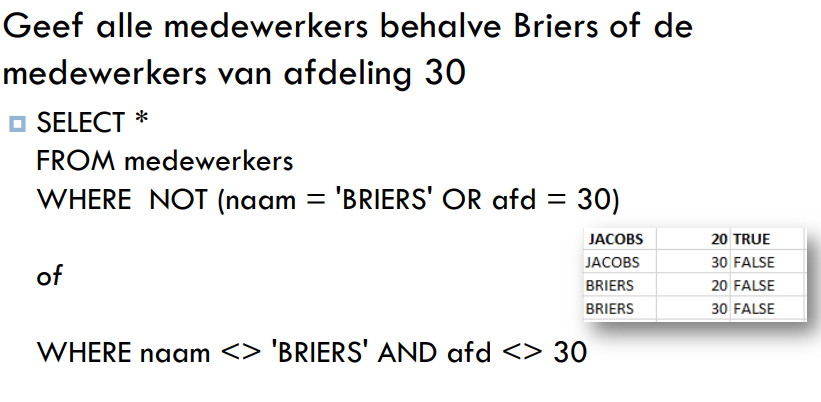
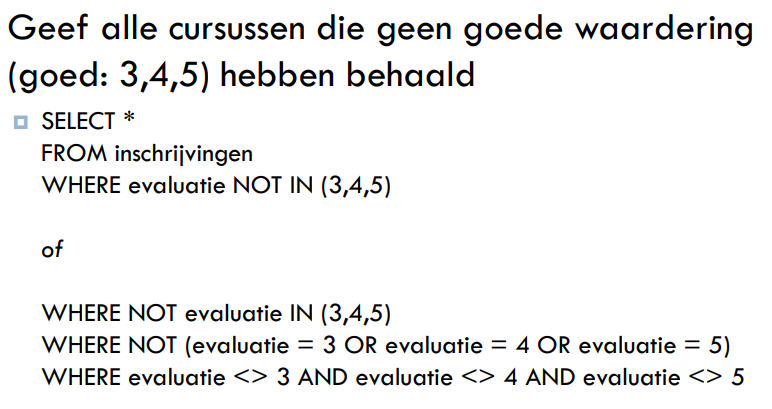


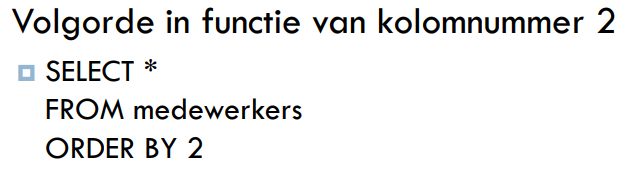
# Hoofdstuk 4: Raadpleging

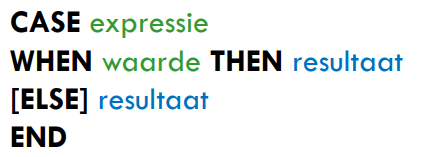
## 4.1 Het select component

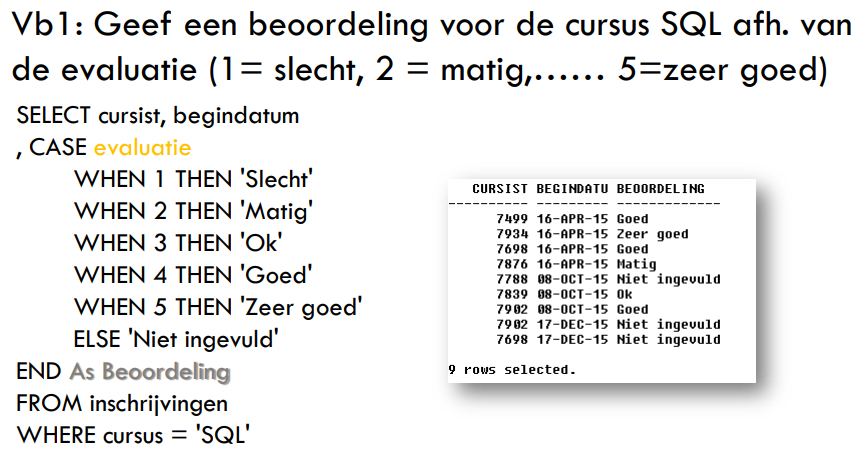


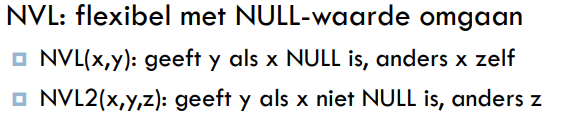




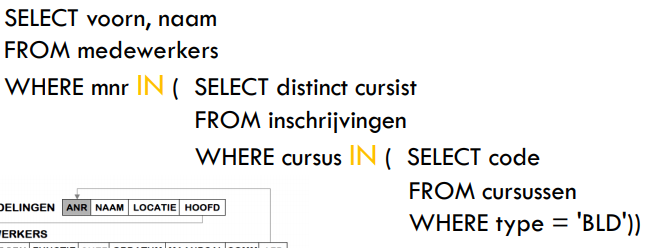






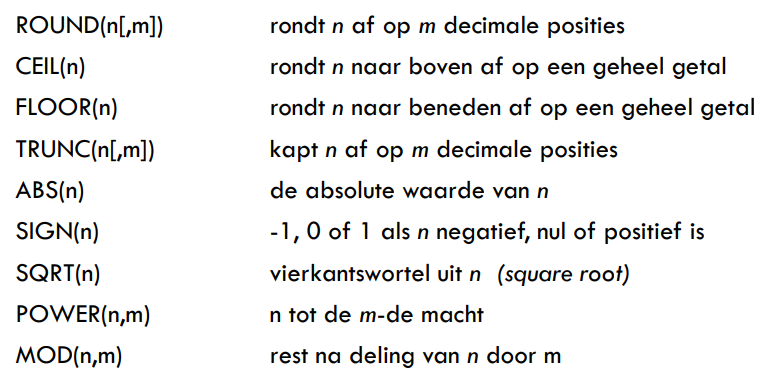


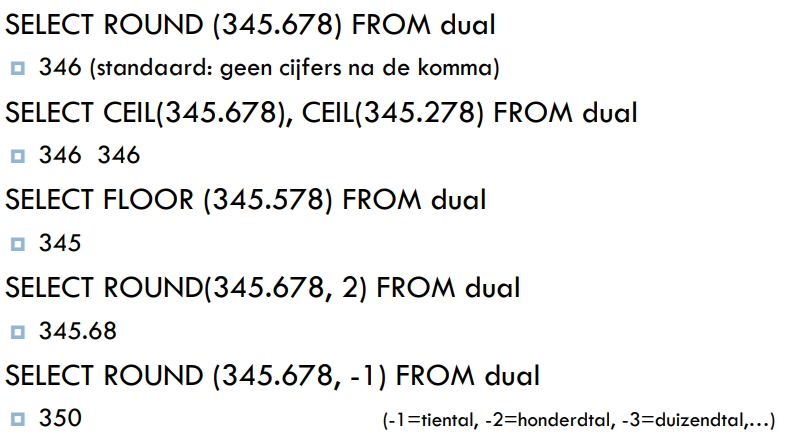
## 4.2 Subquerries



## 4.3 Functies

## 4.3.1 Rekenfuncties

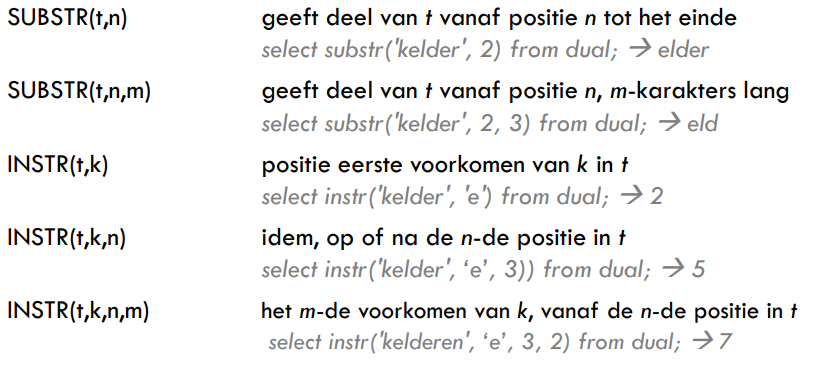




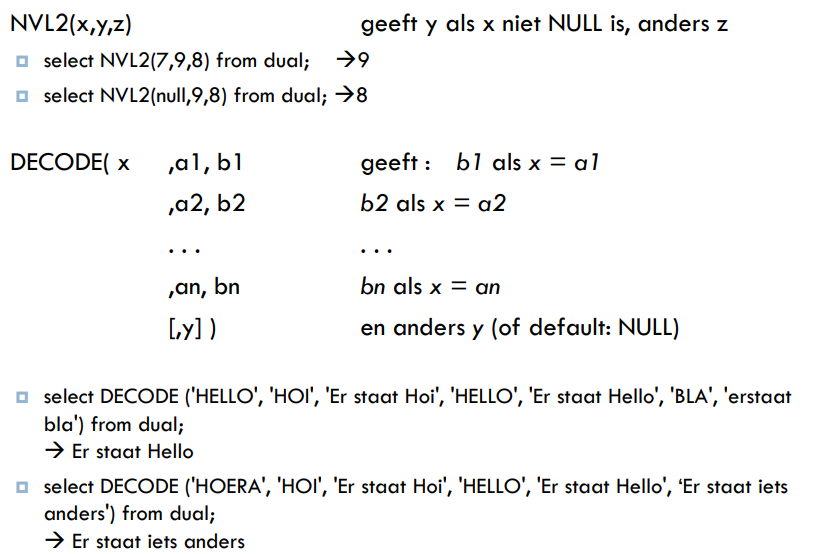
## 4.3.2 Tekstfuncties

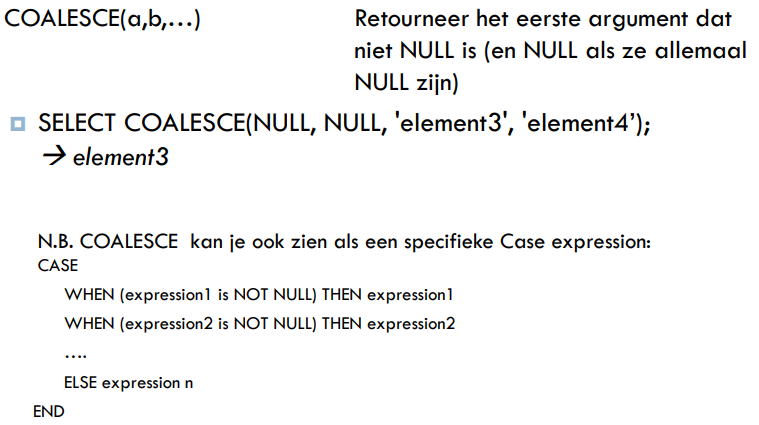




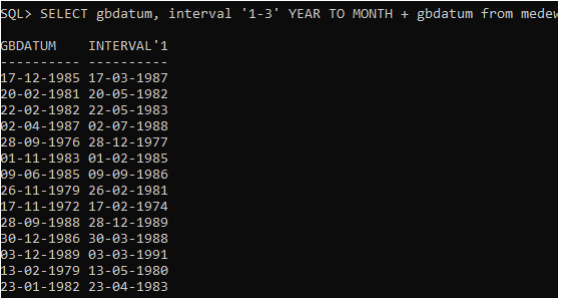


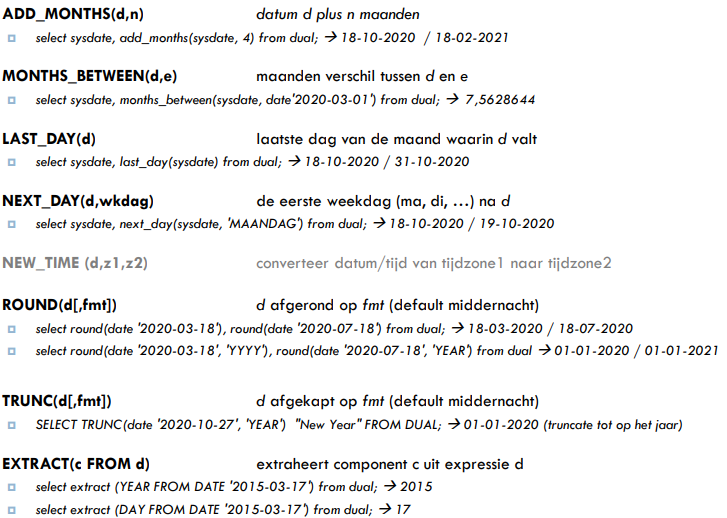
## 4.3.3 Algemene functies



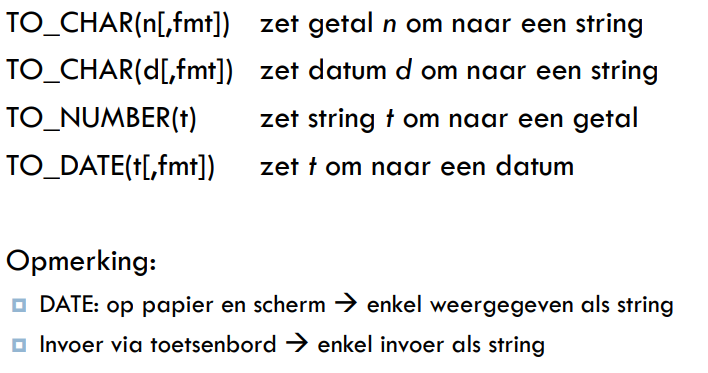


## 4.3.4 Datumfuncties

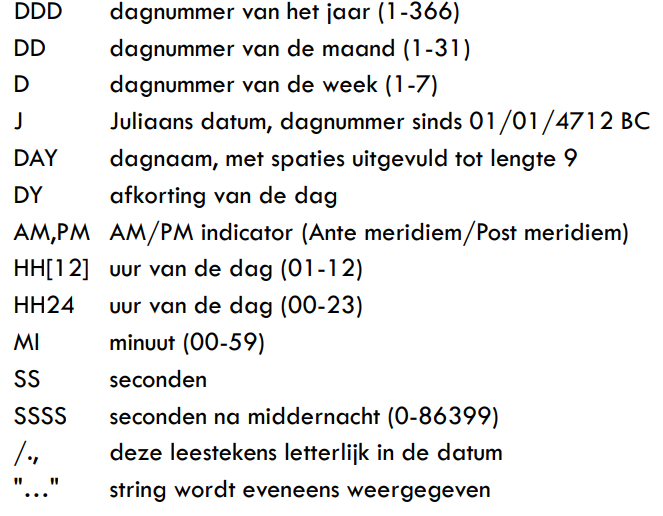


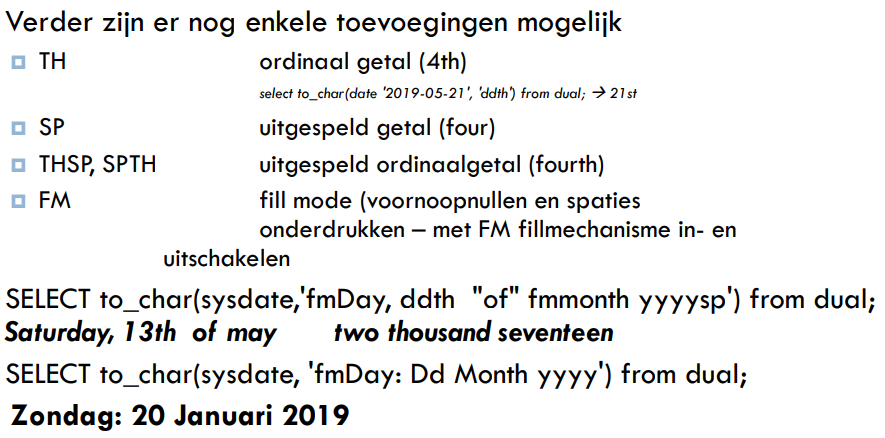




4.3.5 Conversiefuncties

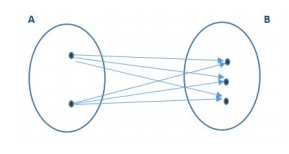






# 5 Raadpleging Deel 2

## 5.1 De Join

**5.1.1 De Equi-join (Inner join)**

Als je deze join doet ga je meerdere tabellen aanspreken na de “FROM”

Select m.naam, a.naam as afdeling from medewerkers m, afdelingen a;

**5.1.2 De auto-join**

Een tabel met zichzelf joinen (er is niet percee een fel verschil met de innerjoin)

Select m.naam as medewerker, c.naam as chef

from medewerkers m, medewerkers c where m.chef = c.mnr;

**5.1.3 De outer-join**

Joint een andere tabel op de eerste tabel.

+ 🡪 Zorgt ervoor dat de gegevens die een nullwaarde in de gejoinde tabel hebben, ook getoond worden

**5.1.4 SQL-Standaard: Inner join (of gewoon join)**

SELECT m.naam ,a.naam afdeling

FROM afdelingen a JOIN medewerkers m JOIN

ON m.afd = a.anr;

5.1.4.1 Right outer join (of right join)

Geeft alle tijen van de rechtertabel inclusief de rijen waar de initiele tabel geen waardes voor heeft

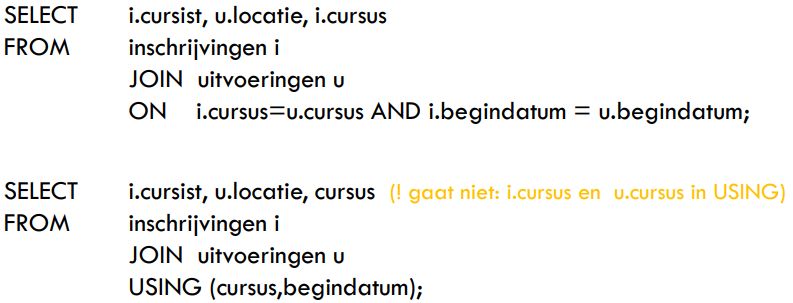
5.1.4.2 Left outer join (of left join)

Geeft alle rijen van de linkertabel incluisief de rijen waar de initiele tabel geen waardes voor heeft

5.1.4.3 Full outer join (of full join)

Geeft rijen van beiden tabellen

5.1.4.4 Using component



5.4.5 Break

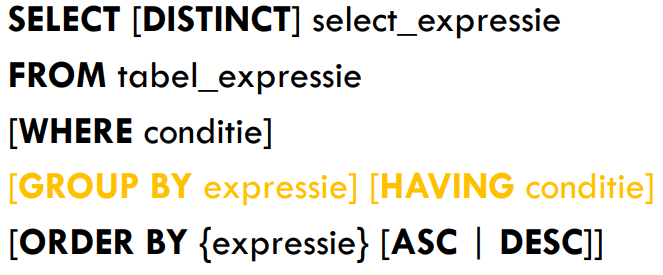
Hiermee kunnen we de kolomwaarden onderdrukken en een regel overslaan of toevoegen

Break on afd skip 1 (skip 1 lijn op elke afdeling)

Break on afd page set (streepjes bij elke nieuwe afdeling)

pause on (nieuwe pagina bij elke afdeling)

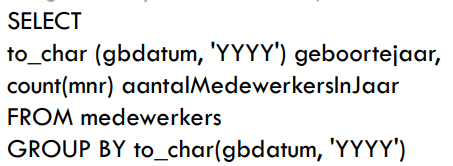
clear breaks (breaks weg doen)



## 5.2 Groepsfuncties

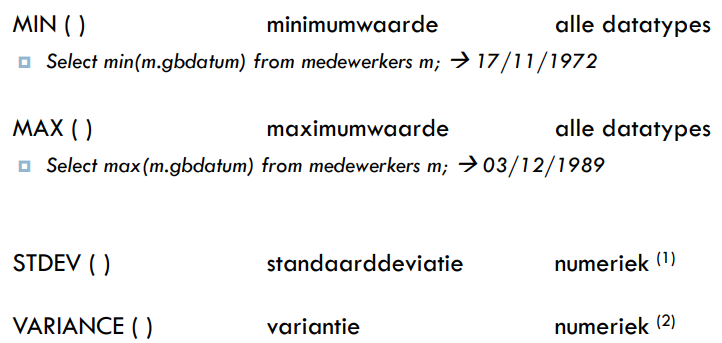
**5.2.1 Group by-component**

Groepeert de table aandehand van de parameter die mee is gegeven.



Hierboven worden alle geboortejaren geselecteerd samen met het aantal werknemers dat in dit jaar geboren werd

**5.2.2 Groepsfuncties**



**5.2.3 Having component**

Het verschil met where:

Where 🡪 op niveau van rijen

Having 🡪 op niveau van de geselecteerde waarden

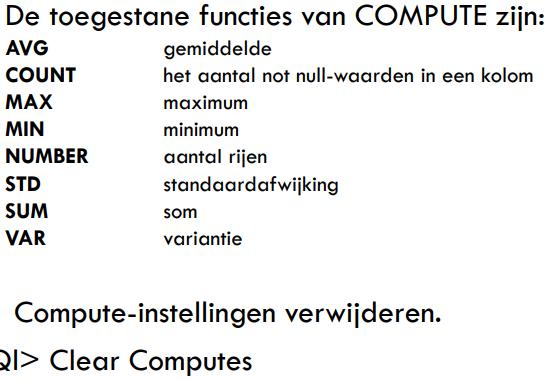
5.2.4 Break and Compute commando

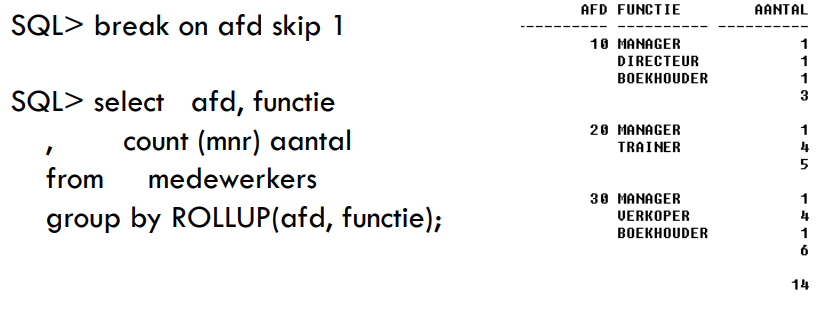
Break on afd skip 1

COMPUTE number label "AANTAL MEDEWERKERS" of functie on afd

Er wordt een total aantal medewerkers berekend van het aantal functies binnen elke afdeling

De COMPUTE clause wordt samen met de select statement gebruikt om ee samengevatte data te berekenen van het resultaat van een groep data.

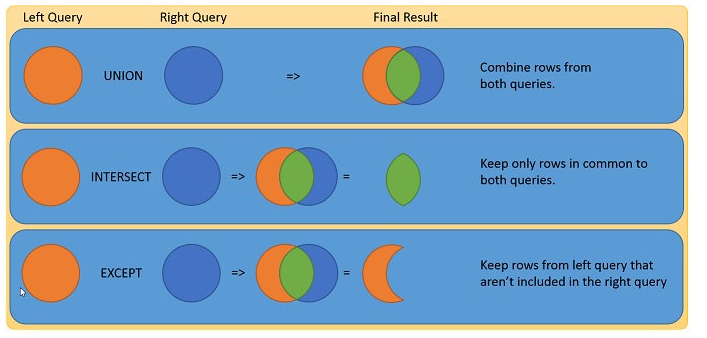


Met rollup(...,...) kan je ook per groep een overzicht krijgen

## 5.3 Gecorreleerde subqueries

## 5.4 Verzalemingoperatoren

De 3 verzamelingsoperatoren:

* Union
* Minus
* Intersect

Het verschil tussen union en union all

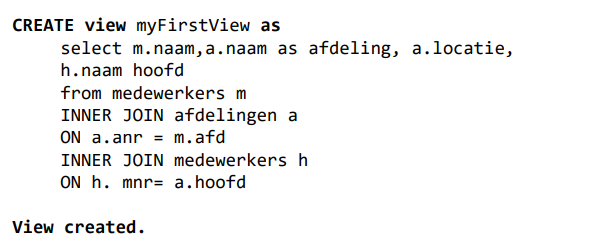
Union 🡪 combineert de rijen van beide querries, en voert automatisch een distinct uit

Union all 🡪 Geeft ook de dubbele waarden

Belangrijk:

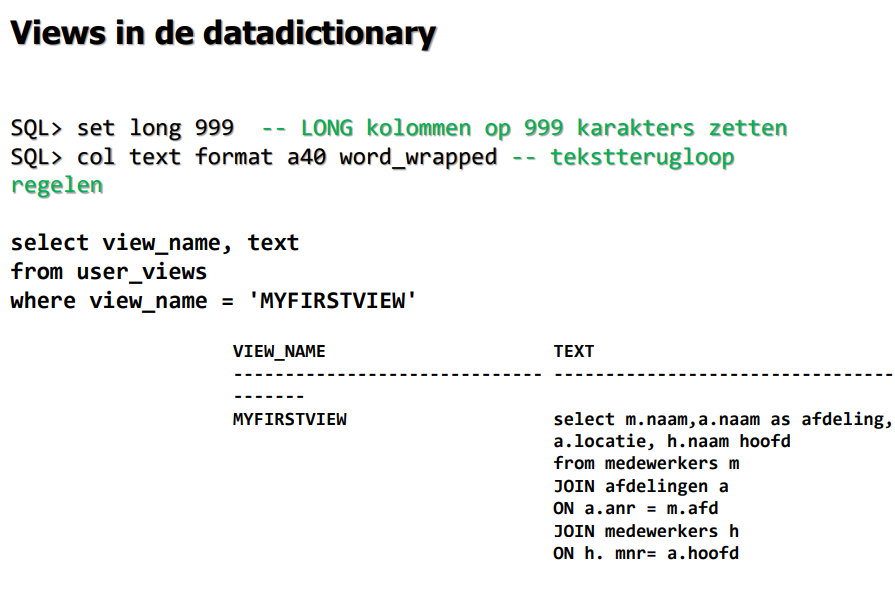
* De resultaatstabel neemt de aliassen over van de Q1
* Q1 en Q2 kunnen apparte voorwaarden hebben en deze zet je voor of na de Q (bv select .. where .. intersect select ... where ....)
* De select parameter mag maar 1 resultaat colomn hebben
* De datatypes van beide selects moeten hetzelfde zijn
* Verzamelingsoperatoren werken NIET met order by in het eerste deel
* De order by telt voor het hele geheel en wordt op het einde van de querrie gezet, je gebruikt de waarde van de eerste select

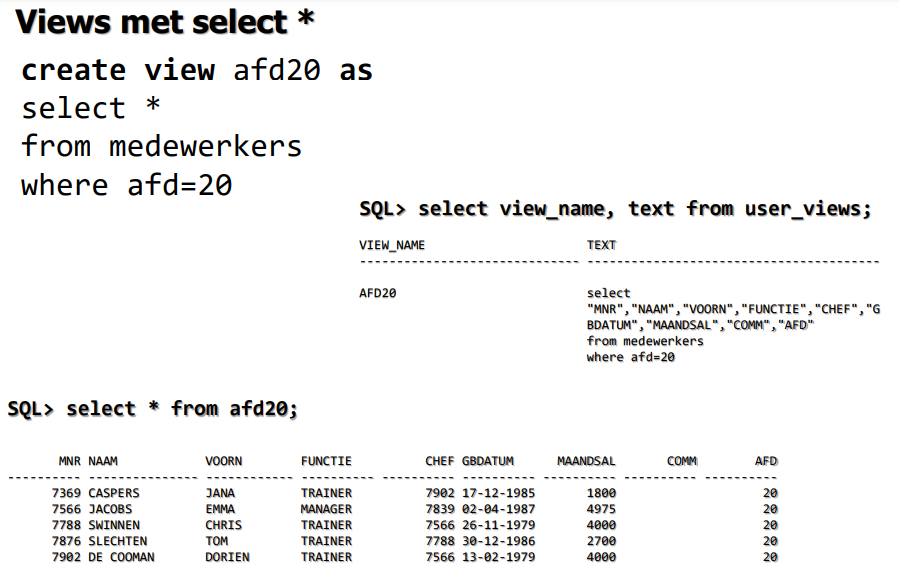
## 5.5 Views (table maken, opgesteld uit andere tables via een join)



select \* from tab; -> laat je alle views (tables) zien

desc MYFIRSTVIEW. -> laat alle kolomnamen zien, descending by name(??)





Drop view afd20 -> verwijder view

Save common querries in a view ->

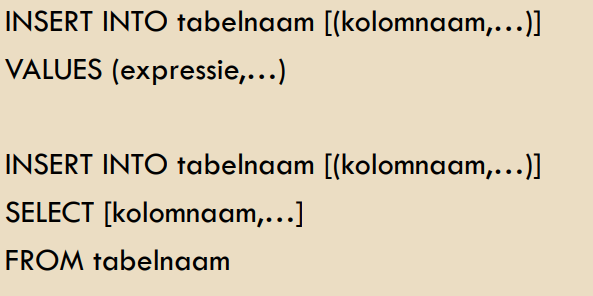
create view VW\_Provincielimburg =

select gemeente from provincies where provinvienaam = ‘Limburg’;

# Hoofdstuk 6: Data Manipulation Language

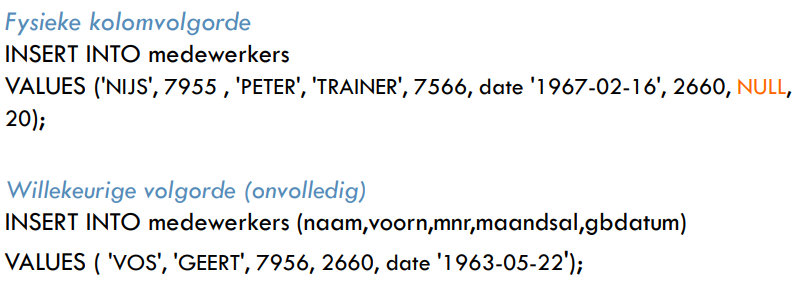
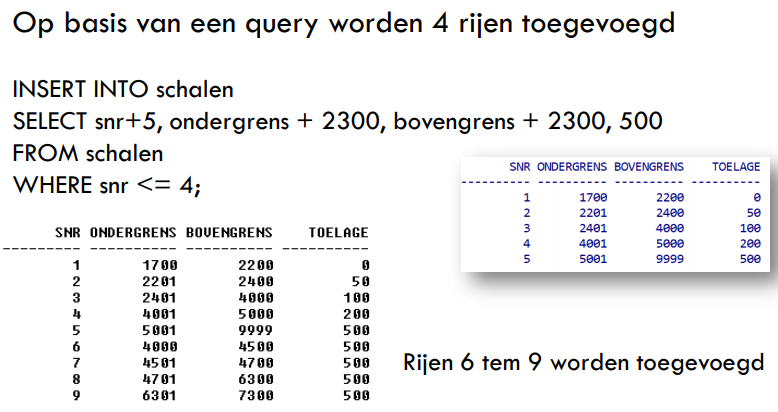
Dit is een taal die gebruikt wordt om datamanipulatie te doen vanuit applicaties zoals forms.

## 6.1 Gegevens invoeren

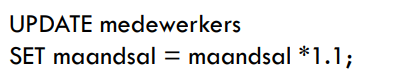


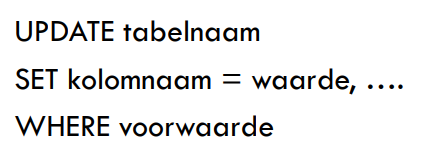
Values -> nieuwe values

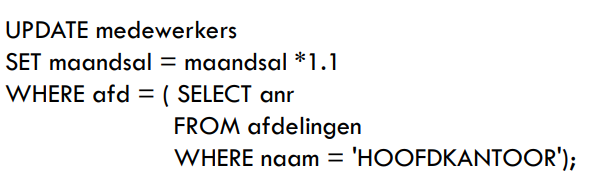
Vanuit een bestaande tabel copieeren



## 6.2 Gegevens wijzigen







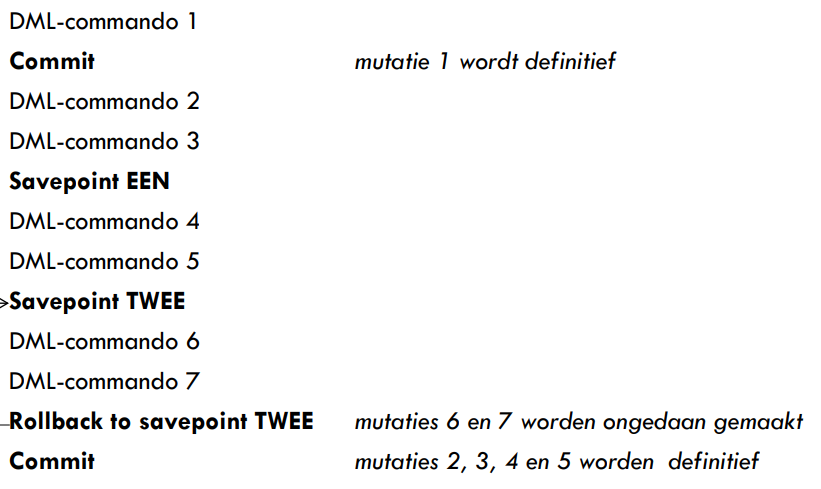
## 6.3 Gegevens verwijderen

## 6.4 Transactieverwerking

Wordt gebruikt om veranderingen te beheren. Na een edit, alter, delete, .... worden deze vaak gebruikt om een veilige manier van datamanipulatie te doen

Commit 🡪 Veranderingen toepassen

Rollback 🡪 wijzigingen ongedaan maken

Savepoint 🡪 markeringspunten maken tussen verschillende aanpassingen

|  |  |
| --- | --- |
| **Data** | **Opgeslagen gegevens** |
| Datamodel | De wijze waarop de gegevens opgeslagen zijn |
| Database Management Systeem (DBMS) | De software waar de db mee aangemaakt of benaderd wordt |
| Gegevensdictionary | Dit bevat hoe gegevens in een informatiesysteem gestructureerd zijn. Het geeft info over de gevens. |