

SQL

Les 0

Structured Query Language

Programma

Les 0 – Introductieopdracht

Les 1

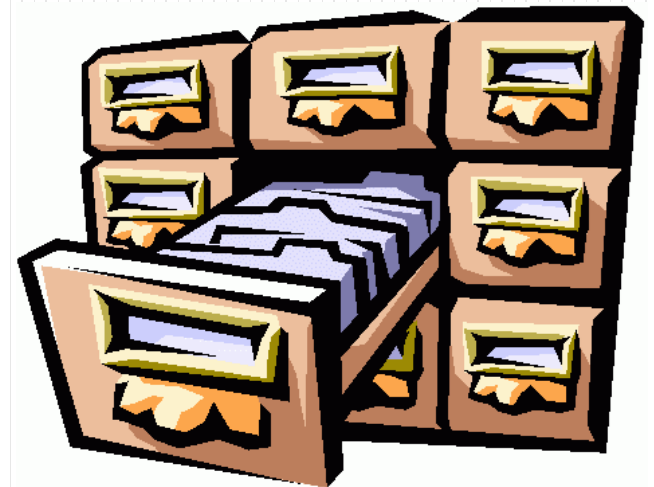
Les 2

Les 3

Schriftelijke toets

Wat is een database?

Soort kaartenbak

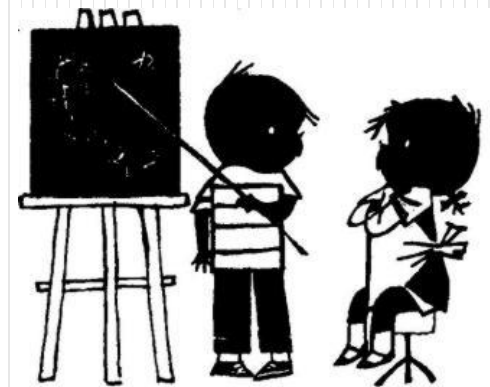


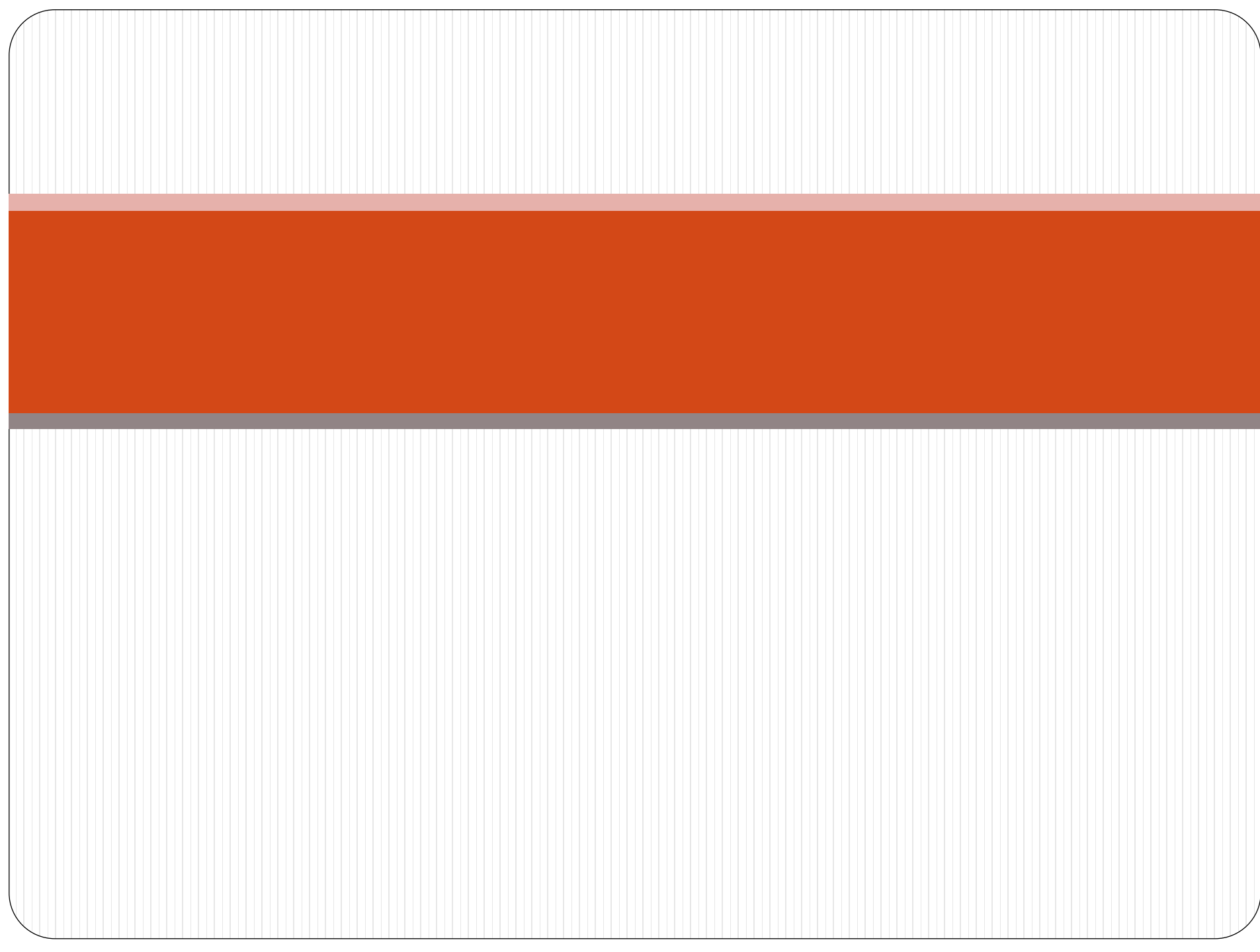
Wat is SQL

Onderzoek in eigen woorden (Jip en Janneke taal):

- wat is SQL
- waarvoor wordt het gebruikt
- geef aan wat de opbouwstructuur van een SELECT-statement is (Basisopbouw)
- wat betekenen de volgende begrippen:
 - tabelnaam
 - attribuut
 - record

Lever dit vandaag op 1 A4tje voorzien van je naam in





SQL

Les 1

SELECT, *, DISTINCT, AS, WHERE, = “text-criteria”,
ORDER BY

Termen

Tabelnaam: (Entiteit) hetgeen/verzamelnaam waar je gegevens van wilt verzamelen.

Attribuut: veldnaam of wel kolomnaam. Geeft aan wat je precies van de entiteit wilt vastleggen.

Record: 1 volledig item uit een tabel, welke uit alle attributen van die tabel bestaat.

(Ongeschreven) regels van SQL

SQL termen en functies in HOOFDLETTERS

Een statement wordt altijd afgesloten met ;

Help

[Http://www.w3schools.com/SQL](http://www.w3schools.com/SQL)

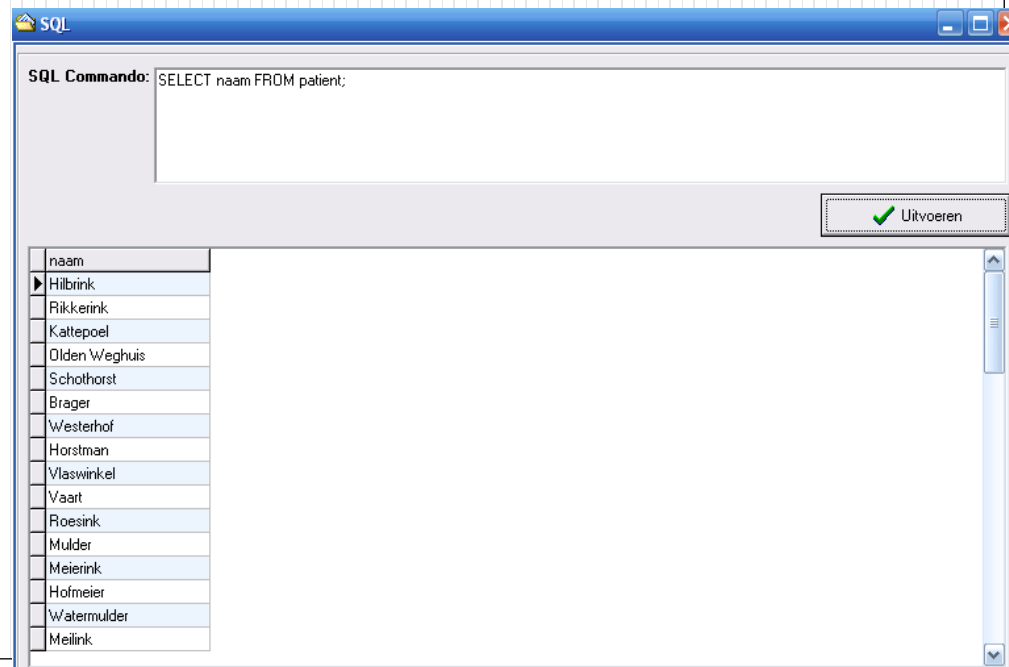
SELECT-statement

SELECT ... FROM ...

Er wordt iets uit een tabel
geselecteerd

Voorbeeld:

```
SELECT naam FROM patient;
```



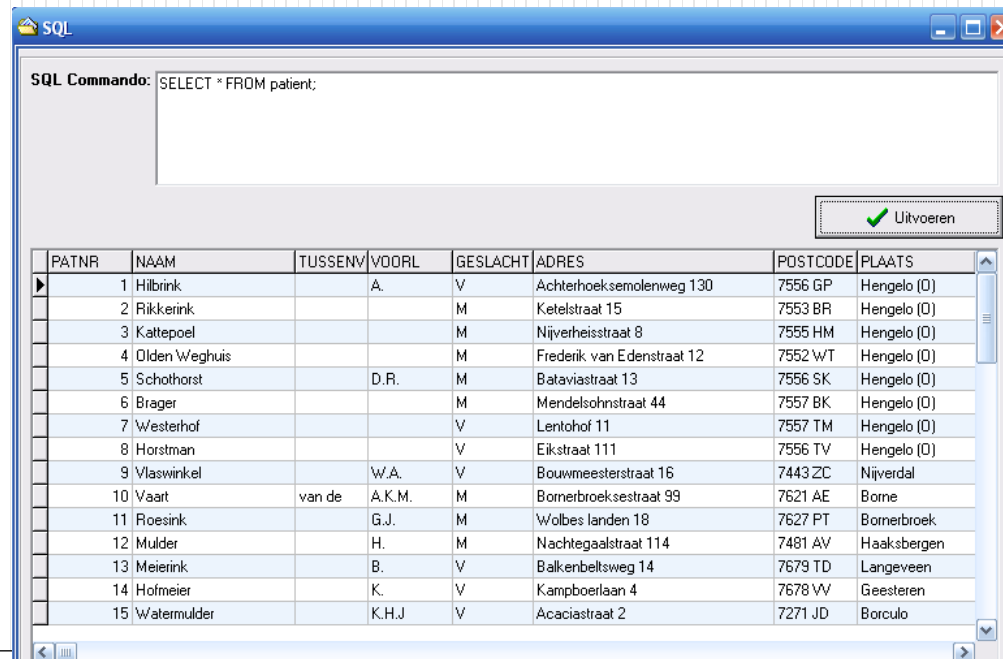
* gebruik in een statement

SELECT * FROM ...

Alles wordt uit een tabel
geselecteerd en getoond

Voorbeeld:

```
SELECT * FROM patient;
```



The screenshot shows a window titled "SQL" with a text area containing the command "SELECT * FROM patient;". Below the text area is a green checkmark icon and the label "Uitvoeren". The results are displayed in a table with the following columns: PATNR, NAAM, TUSSEN, VOORL, GESLACHT, ADRES, POSTCODE, and PLAATS.

PATNR	NAAM	TUSSEN	VOORL	GESLACHT	ADRES	POSTCODE	PLAATS
1	Hilbrink		A.	V	Achterhoeksemolenweg 130	7556 GP	Hengelo (D)
2	Rikkerink			M	Ketelstraat 15	7553 BR	Hengelo (D)
3	Kattepoel			M	Nijverheisstraat 8	7555 HM	Hengelo (D)
4	Olden Weghuis			M	Frederik van Edenstraat 12	7552 WT	Hengelo (D)
5	Schothorst		D.R.	M	Bataviastraat 13	7556 SK	Hengelo (D)
6	Brager			M	Mendelsohnstraat 44	7557 BK	Hengelo (D)
7	Westerhof			V	Lentohof 11	7557 TM	Hengelo (D)
8	Horstman			V	Eikstraat 111	7556 TV	Hengelo (D)
9	Vlaswinkel		W.A.	V	Bouwmeesterstraat 16	7443 ZC	Nijverdal
10	Vaart	van de	A.K.M.	M	Bornerbroeksestraat 99	7621 AE	Borne
11	Roesink		G.J.	M	Wolbes landen 18	7627 PT	Bornerbroek
12	Mulder		H.	M	Nachtegaalstraat 114	7481 AV	Haaksbergen
13	Meierink		B.	V	Balkenbeltsweg 14	7679 TD	Langeveen
14	Hofmeier		K.	V	Kampboerlaan 4	7678 VV	Geesteren
15	Watermulder		K.H.J	V	Acaciastraat 2	7271 JD	Borculo

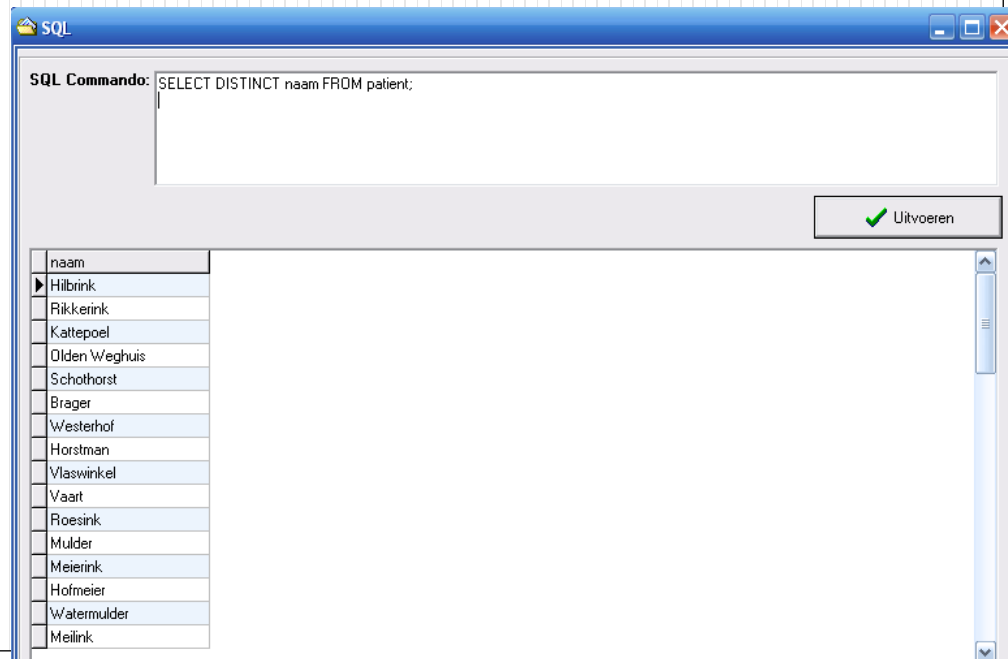
DISTINCT gebruik in een statement

SELECT DISTINCT ... FROM ...

Het attribuut wordt uit een tabel geselecteerd en is uniek als resultaat

Voorbeeld:

```
SELECT DISTINCT naam  
FROM patient;
```



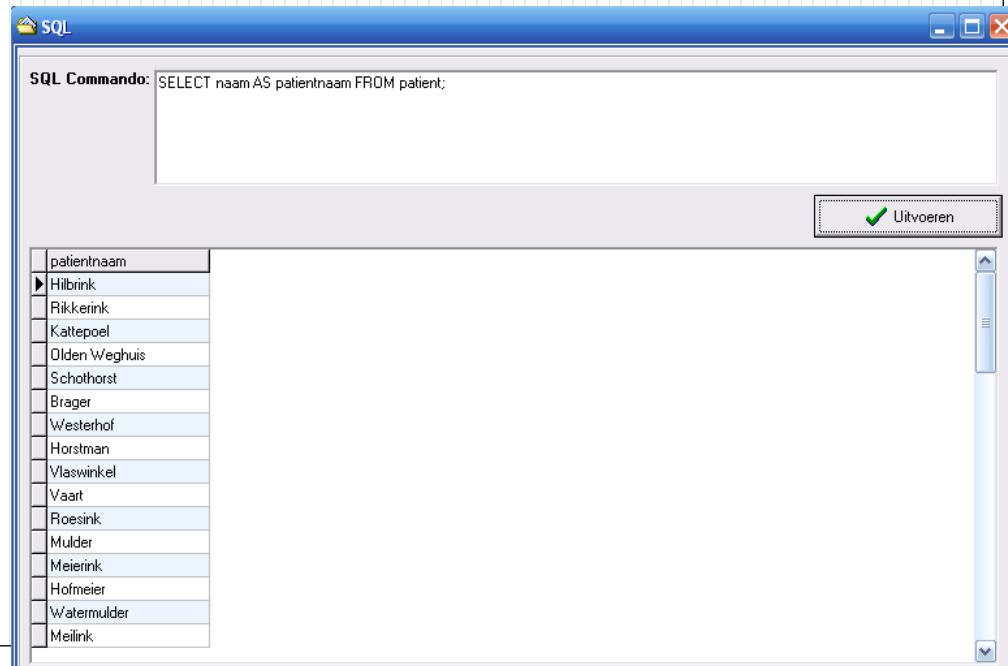
AS gebruik in een statement

SELECT ... AS ... FROM ...

Het attribuut wordt uit een tabel geselecteerd en krijgt bepaald label

Voorbeeld:

```
SELECT naam AS patientnaam  
FROM patient;
```



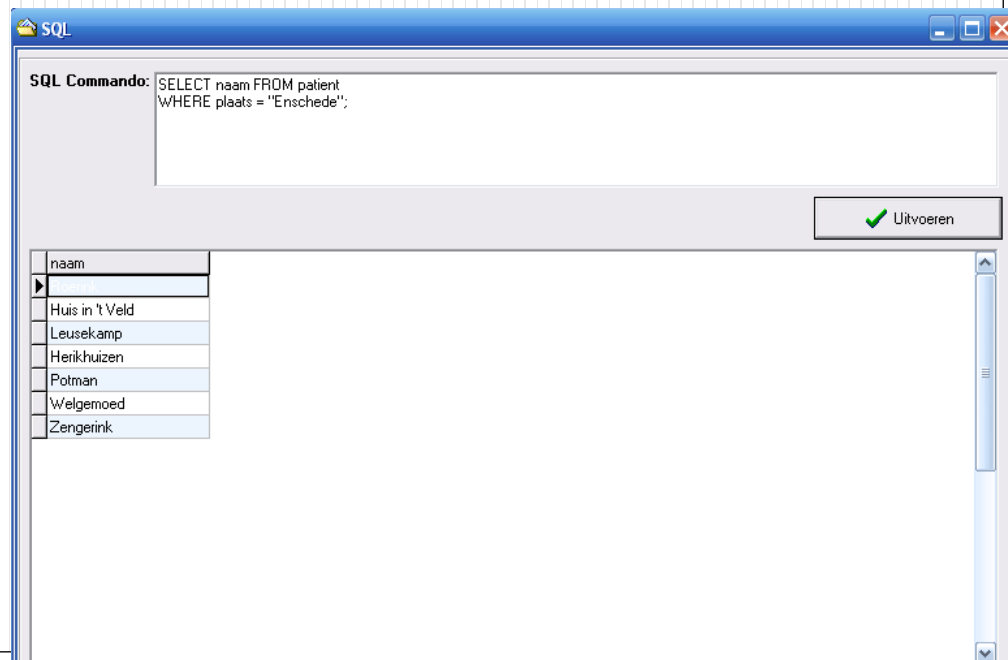
WHERE en = “...” criteria

SELECT ... FROM ...
WHERE ... = “...”

Het attribuut wordt uit een
tabel geselecteerd waarvoor
geldt dat het gelijk is aan iets

Voorbeeld:

```
SELECT naam FROM patient  
WHERE plaats = “Enschede”;
```



ORDER BY

SELECT ... FROM ...
ORDER BY ... (DESC of ASC)

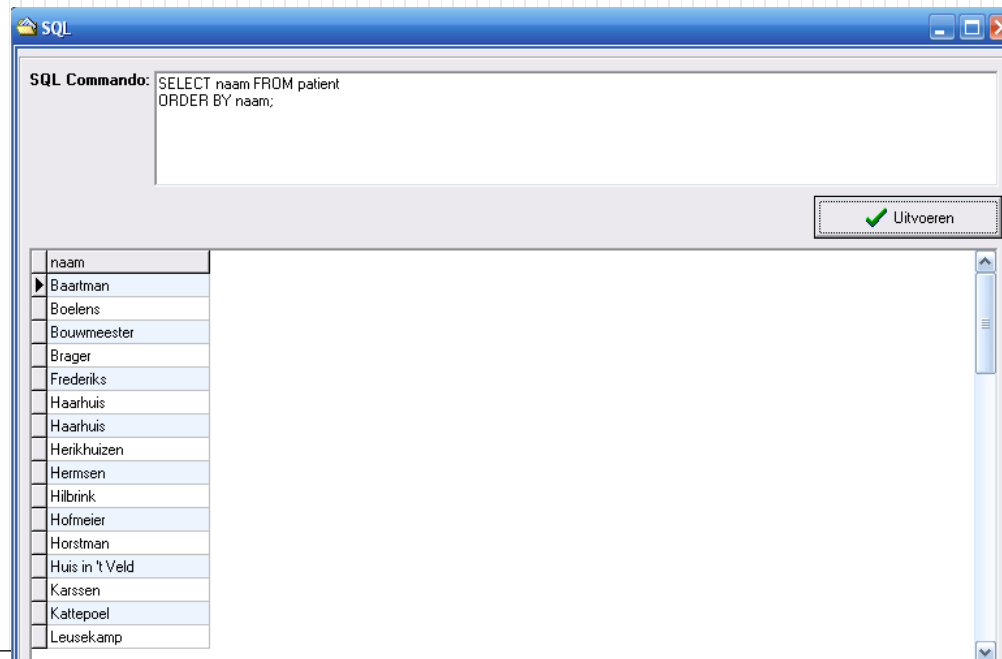
Het attribuut wordt gesortereert.

Standaard oplopend (ASC)

DESC is aflopend

Voorbeeld:

```
SELECT naam FROM patient  
WHERE plaats = "Enschede"  
ORDER BY naam DESC;
```



A thick horizontal bar with a solid orange color, spanning the width of the slide. It is positioned in the upper half of the slide, below a thin light pink line and above a thin grey line.

Maak de 30 vragen die bij les 1 horen

SQL

Les 2

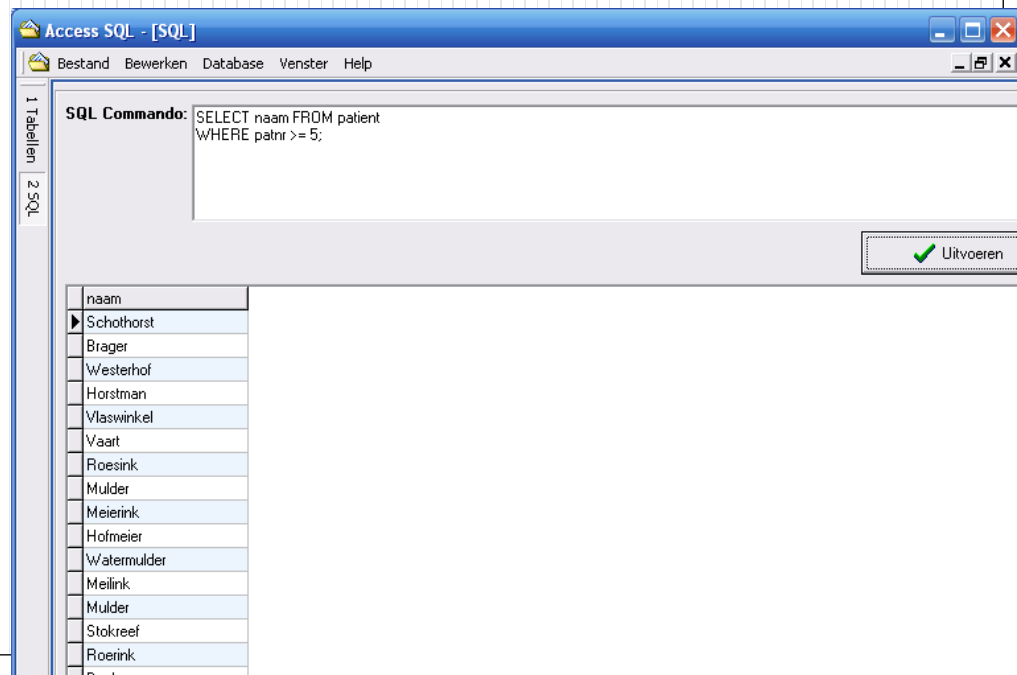
*Logische vergelijkingen, <, >, =, >=, <=, <>, NOT, IS
NULL, AND, OR, #...#, LIKE, BETWEEN, YEAR(),
MONTH(), DAY(), DATE(), -, +*

Veel voorkomende vergelijkingen

- > Groter dan
- < Kleiner dan
- >= Groter of gelijk aan
- <= Kleiner of gelijk aan
- <> Ongelijk aan

Voorbeeld:

```
SELECT naam FROM patient  
WHERE patnr >= 5;
```



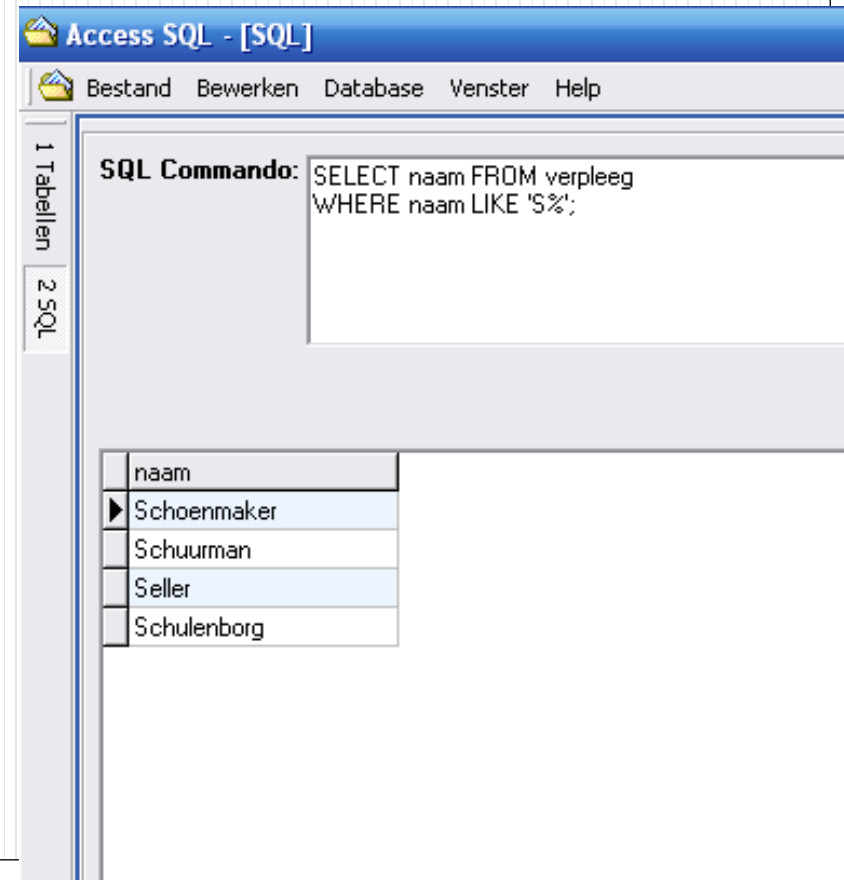
LIKE

... LIKE '...' (jokerteken %)

Wordt gebruikt om naar een speciale waarde te kijken. Zoals begint met een S

Voorbeeld:

```
SELECT naam FROM verpleeg  
WHERE plaats LIKE '%dam%';
```



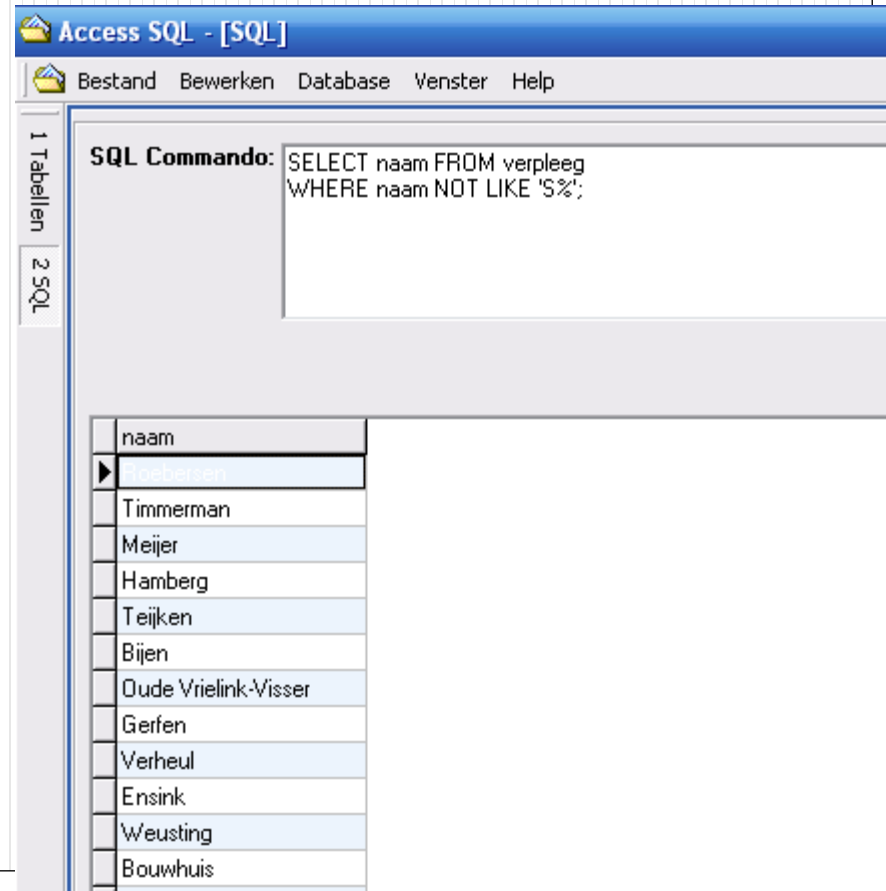
NOT

NOT ... (bijv. IS NOT NULL
of NOT LIKE
of NOT BETWEEN)

Inventeert de functie die erachter staat
(omdraaien)

Voorbeeld:

```
SELECT naam FROM verpleeg  
WHERE naam NOT LIKE 'S%';
```



AND, OR

AND ... EN

OR ... OF

Zijn operators die een conditie opleggen

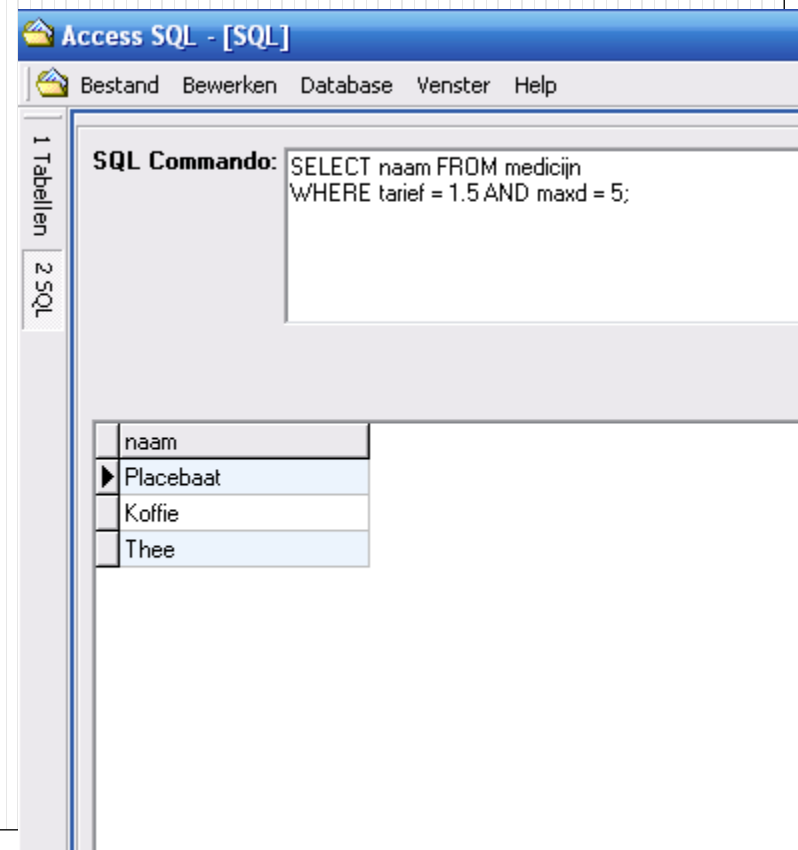
Conditie 1 AND conditie 2

Conditie 1 OR conditie 2

Voorbeeld:

```
SELECT naam FROM medicijn
```

```
WHERE tarief = 1.5 OR maxd = 5;
```



BETWEEN

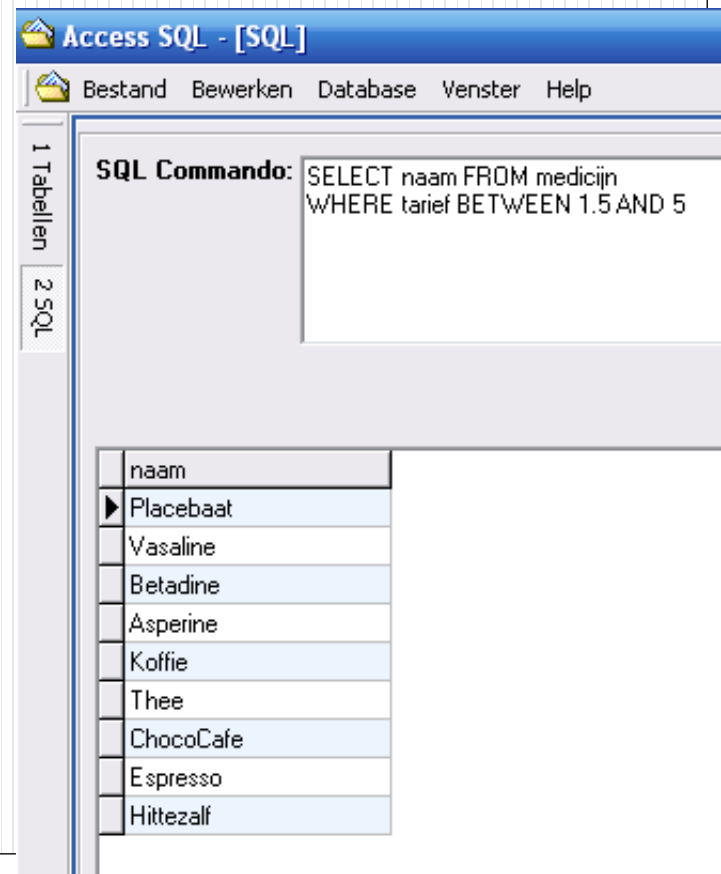
... BETWEEN ... AND ...

Controleert of een attribuut tussen

Bijv. 2 getallen zit.

Voorbeeld:

```
SELECT naam FROM medicijn  
WHERE tarief BETWEEN 1.5 AND 5
```



SQL

Les 3

Average, Null, Update, Insert, Delete

AVERAGE

AVG(.....)

Berekent een gemiddelde uit van alle records onder een attribuut.

Voorbeeld:

```
SELECT AVG(tarief) FROM medicijn
```


NULL

IS NULL

Null laat lege waardes zien in een attribuut.

Voorbeeld:

```
SELECT naam FROM medicijn WHERE naam IS NULL
```

```
SELECT naam FROM medicijn WHERE naam IS NOT NULL
```

Update

We gebruiken update om gegevens in een tabel te wijzigen

```
UPDATE klant
```

```
SET straat= "trompstraat 21", "postcode= 1188LN"
```

```
WHERE klant.id = 5;
```

Insert

Met insert kan je nieuwe rijen toevoegen

```
INSERT INTO klant (voornaam, achternaam)  
VALUES ("Jeff", "Zwijzen")
```

Delete

Delete verwijdert rijen

```
DELETE FROM Klant
```

```
WHERE Klant.voornaam = "Jeff"
```

Een berekende waarde vergelijken met een andere waarde

Je kan door gebruik te maken van Haakjes (...) in je WHERE een nieuwe SELECT statement maken.

Bijv: `SELECT AVG prijs from artikelen WHERE avg prijs < (SELECT naam FROM prijs WHERE schoen = "Nike");`

Maak de 20 vragen die bij les 2 horen

SQL

Les 4

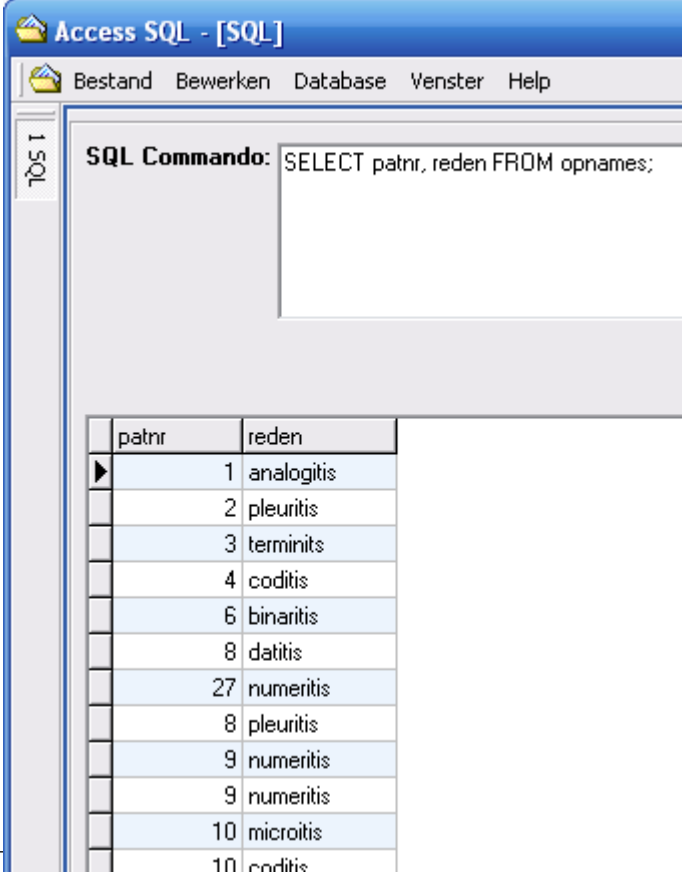
INNER JOINS en complexe INNER JOINS

Gegevens uit meerdere tabellen

SELECT patnr, reden FROM opnames;

Wie is patient 27?

Hoe kan ik daar achter komen?



The screenshot shows the 'Access SQL - [SQL]' window. The menu bar includes 'Bestand', 'Bewerken', 'Database', 'Venster', and 'Help'. The 'SQL Commando:' field contains the query 'SELECT patnr, reden FROM opnames;'. Below the command field, a table displays the results of the query. The table has two columns: 'patnr' and 'reden'. The data rows are as follows:

patnr	reden
1	analogitis
2	pleuritis
3	terminitis
4	coditis
6	binaritis
8	datitis
27	numeritis
8	pleuritis
9	numeritis
9	numeritis
10	microitis
10	coditis

Gegevens uit meerdere tabellen

Voor een attribuut zet je altijd de tabelnaam

Dan weet SQL uit welke tabel het attribuut komt

Bijv.

```
SELECT patient.naam, special.naam FROM ...
```

INNER JOIN

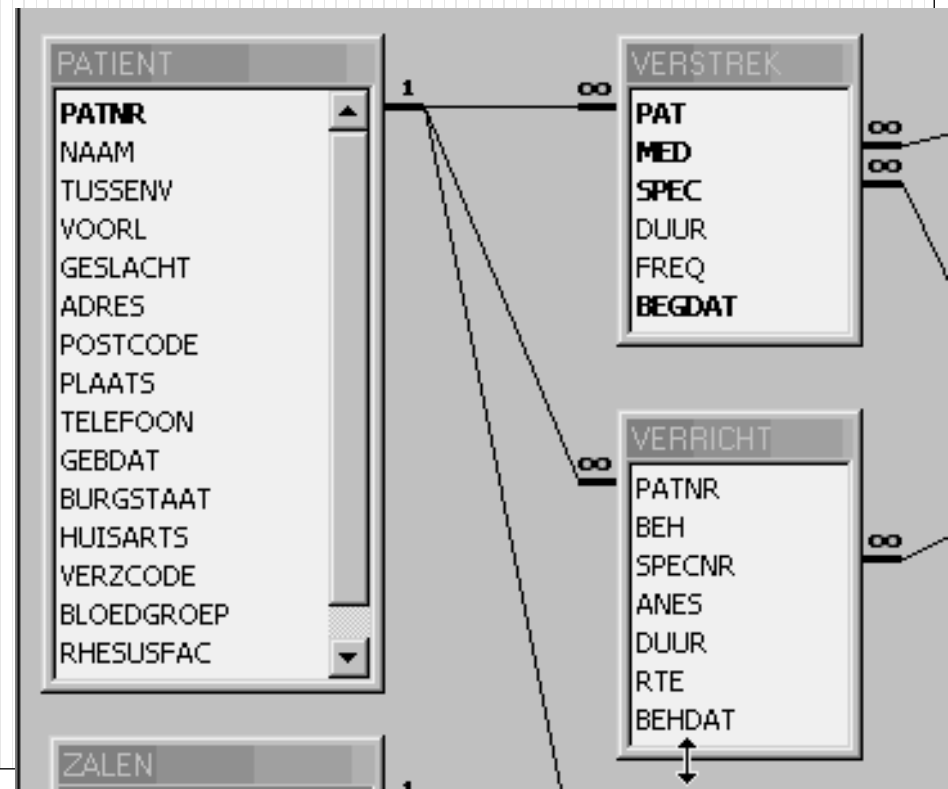
... INNER JOIN ... ON ...

Koppeling tussen 2 tabellen

Op basis van de overeenkomende sleutels.

Zie

SQL OverzichtTabellenRelaties.doc

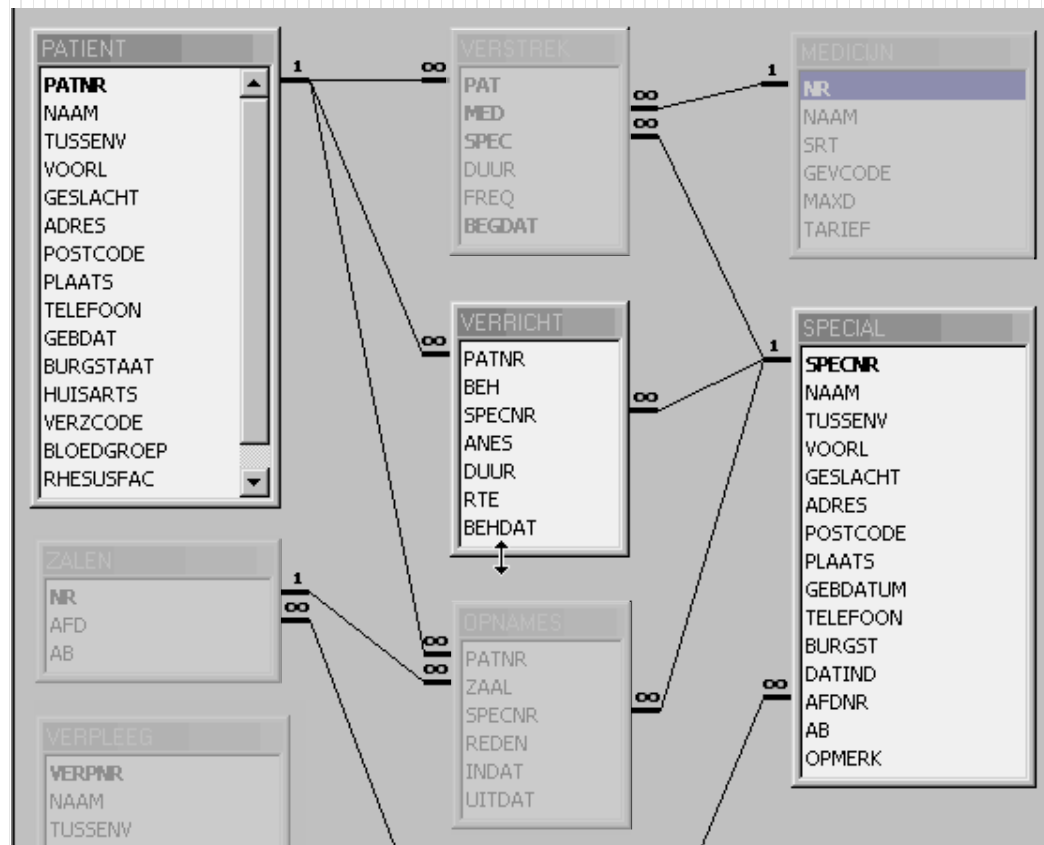


INNER JOIN

```
SELECT patient.naam, opnames.reden FROM patient  
INNER JOIN opnames ON patient.patnr = opnames.patnr;
```

naam	reden
▶ Hilbrink	analogitis
Rikkerink	pleuritis
Kattepoel	terminitis
Olden Weghuis	coditis
Schothorst	Magneitis
Brager	binaritis
Horstman	datitis
Horstman	pleuritis
Vlaswinkel	numeritis
Vlaswinkel	numeritis
Vaart	microitis

Complexere INNER JOINS



Complexere INNER JOINS

Meerdere INNER JOINS gebruiken? Zet het tussen ()

```
SELECT patient.naam, opnames.reden, special.naam FROM (patient  
INNER JOIN opnames ON patient.patnr = opnames.patnr)  
INNER JOIN special ON opnames.specnr = special.specnr;
```

	patient.naam	reden	special.naam
►	Hilbrink	analogitis	Delfgauw
	Rikkerink	pleuritis	Braak
	Kattepoel	terminits	Hekhuis
	Olden Weghuis	coditis	Braak
	Schothorst	Magneitis	Ruttermann
	Brager	binaritis	Weustink
	Horstman	datitis	Bekhuis

Complexere INNER JOINS

Bijv. met WHERE etc.

```
SELECT patient.naam, opnames.reden, special.naam FROM (patient
INNER JOIN opnames ON patient.patnr = opnames.patnr)
INNER JOIN special ON opnames.specnr = special.specnr
WHERE special.plaats = "Enschede"
ORDER BY patient.naam;
```

	patient.naam	reden	special.naam
▶	Haarhuis	numeritis	Bekhuis
	Horstman	pleuritis	Bekhuis
	Horstman	datitis	Bekhuis
	Huis in 't Veld	banaritis	Bekhuis
	Watermulder	binaritis	Bekhuis

Maak de 20 vragen die bij les 3 horen

Rechten en users

- Het maken van users aan en het toekennen van rechten
- `CREATE USER 'username'@'%' IDENTIFIED BY 'PASSWORD';`
- `GRANT ALL PRIVILEGES ON dbname.tablename TO 'username'@'%' WITH GRANT OPTION;`
- `FLUSH PRIVILEGES;`