

Relationele databases

```
43 <body <?php body_class()>
44 <div id="fb-root"></div>
45 <script>(function(d, s, id) {
46     var js, fjs = d.getElementsByTagName(s)[0];
47     if (d.getElementById(id)) return;
48     js = d.createElement(s); js.id = id;
49     js.src = "//connect.facebook.net/en_US/sdk.js#xfbml=1&version=v2.6&appId=29904883629212";
50     fjs.parentNode.insertBefore(js, fjs);
51 })(document, 'script', 'facebook-jssdk');</script>
52 <div id="page" class="site">
53     <a class="skip-link screen-reader-text" href="#content"><?php esc_html_e( 'Skip to content', 'urduTube' );></a>
54     <header id="masthead" class="site-header" role="banner">
55         <div class="site-branding">
56             <div class="navBtn pull-left">
57                 <?php if ( is_home() && $xpanel['homepage-style'] == 1 ) { ?>
58                     <a href="#" id="openMenu"><?php echo esc_html( $xpanel['fa-home'] );></a>
59                 <?php } else { ?>
60                     <a href="#" id="openMenu2"><?php echo esc_html( $xpanel['fa-bars'] );></a>
61                 <?php } ?>
62             </div>
63             <div class="logo pull-left">
64                 <a href="<?php echo esc_url( home_url() ); ?>">
65                     
67             </div>
68             <div class="search-box hidden-xs hidden-sm pull-left ml-10">
69                 <?php get_search_form(); ?>
70             </div>
71             <div class="submit-btn hidden-xs hidden-sm pull-left ml-10">
72                 <a href="<?php echo get_page_link( $xpanel['submit-link'] ); ?>" class="header-submit-btn">
73             </div>
74             <div class="user-info pull-right mr-10">
75                 <?php
76                 if ( is_user_logged_in() ) {
77                     <?php echo esc_html( $current_user->display_name );>
78                 }
79             </div>
80         </div>
81     </div>
82 </div>
83 </body></div>
```

Inhoud

1. Intro relationele databases en eenvoudige queries
2. Geavanceerde queries: geaggregeerde functies en groepen
3. Intro normaliseren m.b.v. relaties tussen tabellen
4. Tabellen uitbreiden met behulp van JOIN
5. Ontwerpen van genormaliseerde databases
6. Het gebruik van subqueries
7. Optionele velden, NULL en LEFT OUTER JOINS

Intro relationele databases

Lesdoelen

- Je kent de kenmerken van een relationele database.
- Je maakt kennis met strokendiagrammen.
- Je raakt vertrouwt met eenvoudige SELECT queries.

Voorbeelden van gegevensbestanden

geef voorbeelden...

Voorbeelden van gegevensbestanden

Relationele databases

- Informatie is geordend in tabellen

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

Relationele databases

- Informatie is geordend in tabellen

veld

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

record

Relationele databases

- Alle opgevraagde informatie komt als tabel

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

“geef van alle records de velden voorletters,
tussenvoegsel en achternaam”

voorletters	tussenvoegsel	achternaam
T.Y.	van	Dam
L.A.P.		Zwans
R.	de	Groot

Relationele databases

- **Alle** opgevraagde informatie komt als tabel, ook als het één waarde is.

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

“geef van alle records het leerlingnummer van het record dat het hoogste leerlingnummer heeft”

leerlingnummer

1548960

Relationele databases

- **Alle** opgevraagde informatie komt als tabel, ook als het één waarde is.

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

“geef het hoogste leerlingnummer”

leerlingnummer

1548960

Relationele databases

- **Alle** opgevraagde informatie komt als tabel, ook als het één waarde is.

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

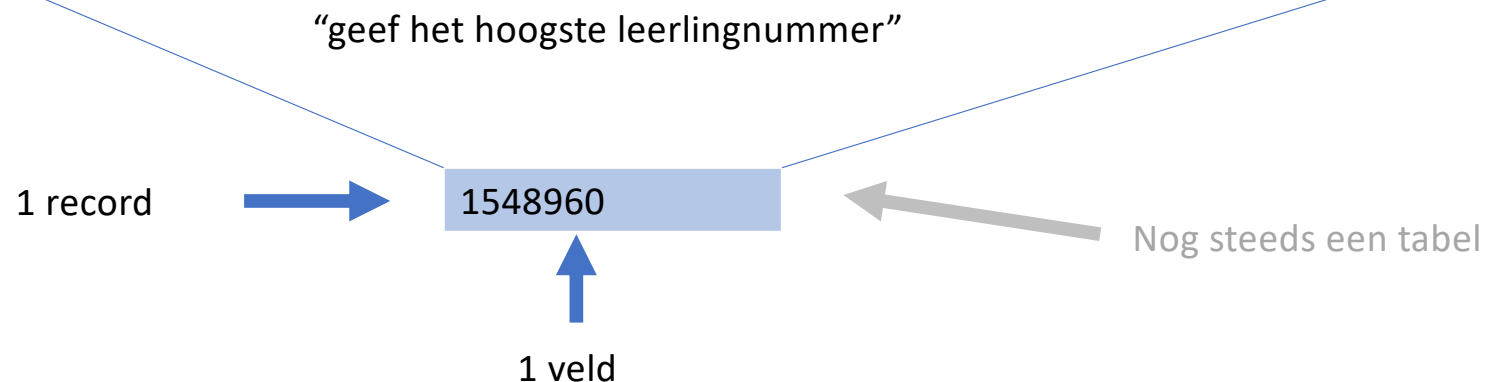
“geef het hoogste leerlingnummer”

1548960

Relationele databases

- **Alle** opgevraagde informatie komt als tabel, ook als het één waarde is.

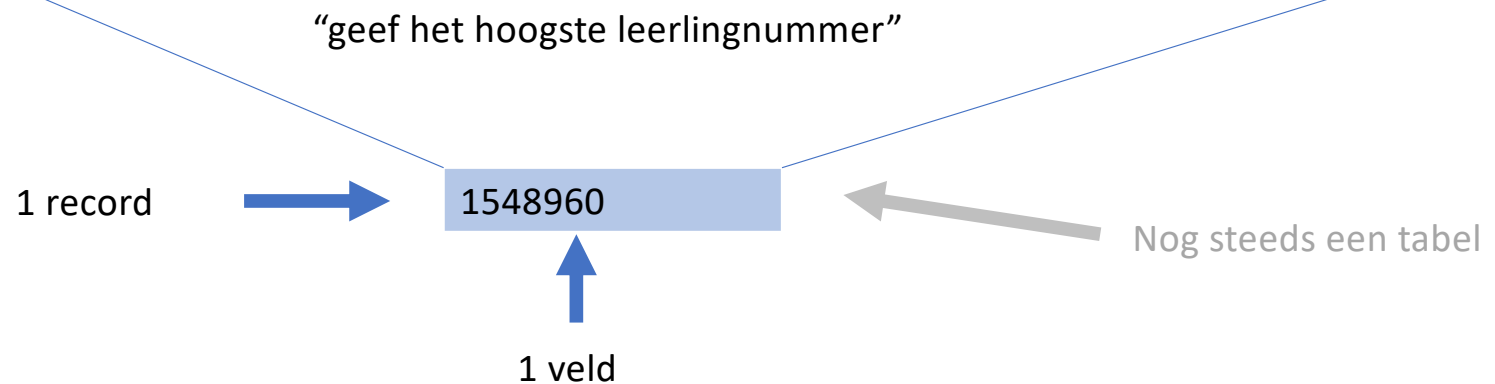
leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com



Relationele databases

- **Alle** opgevraagde informatie komt als tabel, ook als het één waarde is.

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com



Strokendiagrammen

- Je wilt vaak weergeven hoe een database in elkaar zit
- Je gaat nooit alle tabellen helemaal weergeven

leerlingen

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
1102359	T.Y.	van	Dam	thea.van.dam@hotmail.com
1548960	L.A.P.		Zwans	leoleoleo44@gmail.com
1028523	R.	de	Groot	rudolv@yahoo.com

personeel

afkorting	voorletters	tussenvoegsel	achternaam	email
cam	A.P.		Cammeraat	cam@emmauscollege.nl
gee	S.J.H.	van	Geest	gee@emmauscollege.nl
per	J.		Perdaan	per@emmauscollege.nl

vakken

afkorting	naam
ak	aardijkkunde
bi	biologie
in	informatica

Strokendiagrammen

- Je wilt vaak weergeven hoe een database in elkaar zit
- Je gaat nooit alle tabellen helemaal weergeven
- Oplossing: alleen de tabelkoppen + extra informatie

leerlingen

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
----------------	-------------	---------------	------------	-------

personeel

afkorting	voorletters	tussenvoegsel	achternaam	email
-----------	-------------	---------------	------------	-------

vakken

afkorting	naam
-----------	------

Structured Query Language

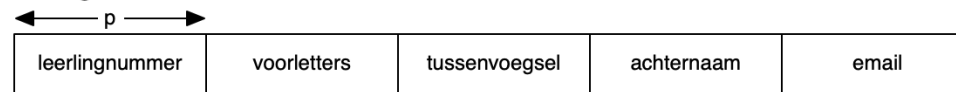
- Dé taal voor het werken met relationele databases.
- Aantal SQL-dialecten, maar de basis is hetzelfde.
 - SELECT -> opvragen van informatie
 - UPDATE -> wijzigen van velden
 - DELETE -> verwijderen van records

SELECT - eenvoudig

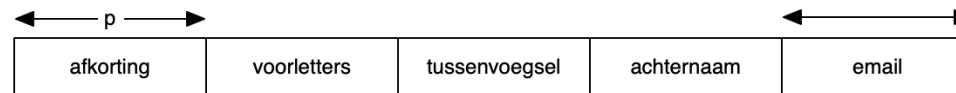
SELECT <velden> FROM <tabel>

```
SELECT leerlingnummer FROM leerlingen;  
SELECT afkorting, achternaam FROM personeel;  
SELECT * FROM vakken;
```

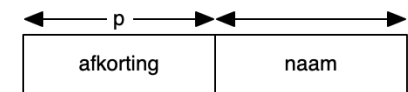
Leerlingen



Personeel



Vakken



SELECT

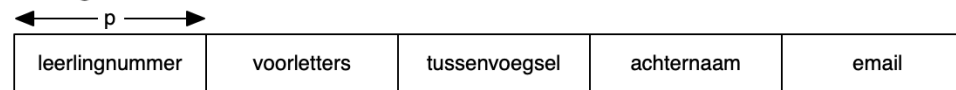
SELECT <velden> FROM <tabel> WHERE <veld> = <waarde>

```
SELECT * FROM leerlingen WHERE leerlingnummer > 1100000;
```

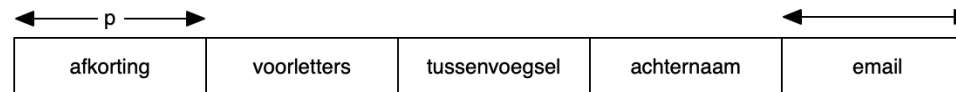
```
SELECT afkorting FROM personeel
```

```
WHERE voorletters = "S.G.J." AND achternaam = "Geest"
```

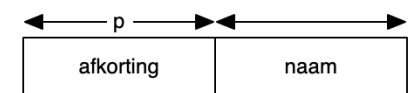
Leerlingen



Personeel



Vakken



Geaggregeerde gegevens

Inhoud

1. Intro relationele databases en eenvoudige queries
2. Geavanceerde queries: geaggregeerde functies
3. Intro normaliseren m.b.v. relaties tussen tabellen
4. Tabellen uitbreiden met behulp van JOIN
5. Ontwerpen van genormaliseerde databases
6. ...

Lesdoelen

- Je begrijpt hoe je geaggregeerde functies kunt gebruiken
- Je begrijpt hoe je resultaten kunt groeperen
- Je begrijpt hoe je met HAVING groepen kunt filteren

Geaggregeerde functies

- Soms wil je niet alleen rauwe gegevens uit een database, maar informatie die met behulp van de losse gegevens kan worden verkregen:
 - het artikel met de hoogste prijs
 - het aantal leerlingen in 5V
 - het gemiddelde cijfer voor het PWS
 - de reiskosten in maart
- SQL heeft hiervoor speciale functies die je in een query kunt gebruiken: COUNT, MIN, MAX, SUM, AVG

Geaggregeerde functies

Artikel				
← p →	id	naam	voorraad	prijs

```
SELECT MIN(prijs) FROM Artikel;
```

```
SELECT naam FROM Artikel WHERE prijs = MIN(prijs);
```

```
SELECT AVG(prijs) as gemiddelde_prijs FROM Artikel;
```

Vraag: Wat is de query voor het totaal aantal artikelen in voorraad?

Vraag: Wat is de naam van het artikel waar het meeste voorraad van is?

Geaggregeerde functies

```
SELECT COUNT(*) FROM leerlingen;
```

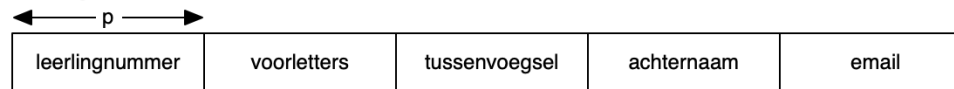
```
SELECT COUNT(*) as totaal_aantal_leerlingen FROM leerlingen;
```

```
SELECT COUNT(*) as aantal_martens FROM personeel
```

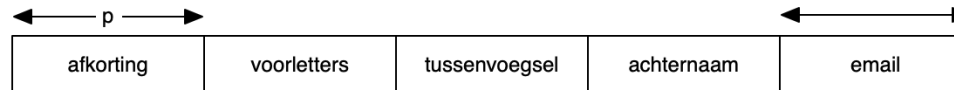
```
WHERE achternaam = "Martens";
```

```
SELECT COUNT(*) as aantal_afk_c FROM personeel WHERE afkorting LIKE "C%";
```

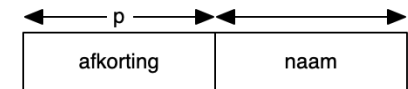
Leerlingen



Personeel



Vakken



Geavanceerde filters

Geavanceerd filteren

- Je kunt met behulp van AND en OR complexe filters maken:
- “De naam van alle artikelen met **een prijs groter dan 2,00 EN een prijs kleiner dan 4,00**”
- “Het id en de voorraad van alle artikelen met **een prijs kleiner dan 0,20 OF een prijs groter dan 400,00**”
- “Het id van alle artikelen met **een voorraad kleiner dan 15 OF (met als categorie ‘badlinnen’ EN een voorraad kleiner dan 25)**”

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Geavanceerd filteren

- “De naam van de artikelen met een prijs groter of gelijk aan 5 maal de prijs van het goedkoopste artikel”
- `SELECT naam FROM Artikel WHERE prijs > 5 * MIN(prijs);`
- “Alle velden van de artikelen met een voorraad kleiner dan 10 minder dan de gemiddelde voorraad”
- `SELECT * FROM Artikel WHERE voorraad < AVG(voorraad) - 10`

← p →					Artikel
id	naam	voorraad	categorie	prijs	
id	naam	voorraad	categorie	prijs	
1	washandje	19	badlinnen	1,99	
2	handdoek	23	badlinnen	4,99	
3	wok 23cm	12	kookgerei	29,99	
4	USB-kabel 2m	34	elektronica	3,99	
...	

Geavanceerd filteren

- “De naam van alle artikelen met **een prijs groter dan 2,00 EN een prijs kleiner dan 4,00**”
- `SELECT naam FROM Artikel WHERE prijs > 2.00 AND prijs < 4.00;`

Artikel

← p →				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
4	USB-kabel 2m	34	elektronica	3,99

Geavanceerd filteren

- “De naam van alle artikelen met **een prijs groter dan 2,00 EN een prijs kleiner dan 4,00**”
- `SELECT naam FROM Artikel WHERE prijs > 2.00 AND prijs < 4.00;`

← p →

Artikel				
id	naam	voorraad	categorie	prijs

naam

USB-kabel 2m

Geavanceerd filteren

- “Het id en de voorraad van alle artikelen met **een prijs kleiner dan 0,20 OF een prijs groter dan 400,00**”
- SELECT id, voorraad FROM Artikel WHERE prijs < 0.20 OR prijs > 400.00;

← p →				Artikel
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Geavanceerd filteren

- “Het id van alle artikelen met **een voorraad kleiner dan 15 OF (met als categorie ‘badlinnen’ EN een voorraad kleiner dan 25)**”
- `SELECT id FROM Artikel
WHERE voorraad < 15 OR
(categorie = ‘badlinnen’ AND voorraad < 25)`

← p →				Artikel
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Geavanceerd filteren

- “Het id van alle artikelen met **een voorraad kleiner dan 15 OF (met als categorie ‘badlinnen’ EN een voorraad kleiner dan 25)**
- SELECT id FROM Artikel
WHERE **voorraad < 15 OR**
(categorie = ‘badlinnen’ AND voorraad < 25)

← p →				Artikel
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Geavanceerd filteren

- “Het id van alle artikelen met **een voorraad kleiner dan 15 OF (met als categorie ‘badlinnen’ EN een voorraad kleiner dan 25)**”
- `SELECT id FROM Artikel
WHERE voorraad < 15 OR
(categorie = ‘badlinnen’ AND voorraad < 25)`

← p →				Artikel
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Geavanceerd filteren

- “Het id van alle artikelen met **een voorraad kleiner dan 15 OF (met als categorie ‘badlinnen’ EN een voorraad kleiner dan 25)**”
- `SELECT id FROM Artikel
WHERE voorraad < 15 OR
(categorie = ‘badlinnen’ AND voorraad < 25)`

Artikel

← p →				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99

Geavanceerd filteren

- “Het id van alle artikelen met **een voorraad kleiner dan 15 OF (met als categorie ‘badlinnen’ EN een voorraad kleiner dan 25)**”
- `SELECT id FROM Artikel
WHERE voorraad < 15 OR
(categorie = ‘badlinnen’ AND voorraad < 25)`

← p →				Artikel
id	naam	voorraad	categorie	prijs

id
1
2
3

SELECT

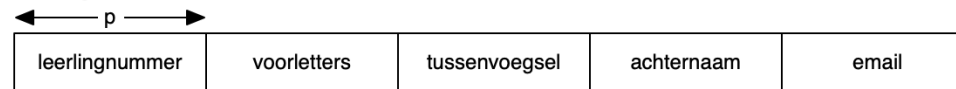
SELECT <velden> FROM <tabel> WHERE <veld> = <waarde>

```
SELECT * FROM leerlingen WHERE leerlingnummer > 110000;
```

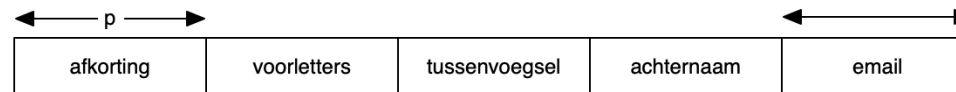
```
SELECT afkorting FROM personeel
```

```
WHERE voorletters = "S.G.J." AND achternaam = "Geest"
```

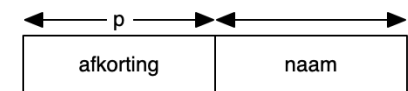
Leerlingen



Personeel



Vakken



Filteren met IN

- `SELECT * FROM Artikel WHERE categorie = "badlinnen" OR categorie = "kookgerei" OR categorie = "kleding"`
- `SELECT * FROM Artikel WHERE categorie IN ("badlinnen", "kookgerei", "kleding")`

Artikel

← p →				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- `SELECT * FROM Artikel WHERE categorie IN ("badlinnen", "kookgerei", "kleding")`

badlinnen
kookgerei
kleding

Je kunt dit ook opvatten als een tabel met 1 kolom

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- `SELECT * FROM Artikel WHERE categorie IN ("badlinnen", "kookgerei", "kleding")`

badlinnen
kookgerei
kleding

Je kunt dit ook opvatten als een tabel met 1 kolom
... zodat hier ook een SELECT query voor 1 veld kunt neerzetten
Een "query in een query" -> subquery

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- `SELECT * FROM Artikel WHERE categorie IN (SELECT categorie from Topcategory”)`

Topcategorie

id	categorie
1	badlinnen
2	kookgerei
3	kleding

Artikel

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- `SELECT * FROM Artikel WHERE categorie IN (SELECT categorie from Topcategorie)`

Topcategorie

id	categorie
1	badlinnen
2	kookgerei
3	kleding

Artikel

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- SELECT * FROM Artikel WHERE categorie IN ("badlinnen", "kookgerei", "kleding")

Topcategorie

id	categorie
1	badlinnen
2	kookgerei
3	kleding

Artikel

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- **SELECT * FROM Artikel WHERE categorie IN ("badlinnen", "kookgerei", "kleding")**

Topcategorie

id	categorie
1	badlinnen
2	kookgerei
3	kleding

Artikel

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Filteren met IN

- `SELECT * FROM Artikel WHERE categorie IN ("badlinnen", "kookgerei", "kleding")`

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99

SELECT m.b.v. IN

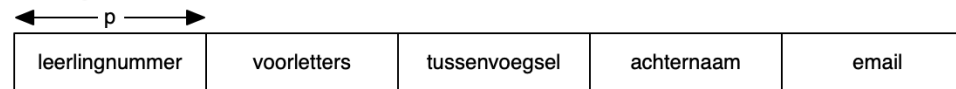
SELECT <velden> FROM <tabel> WHERE <veld> = <waarde>

```
SELECT * FROM leerlingen WHERE leerlingnummer > 110000;
```

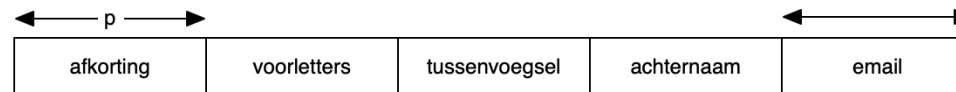
```
SELECT afkorting FROM personeel
```

```
WHERE voorletters = "S.G.J." AND achternaam = "Geest"
```

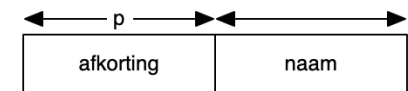
Leerlingen



Personeel



Vakken



Samengenomen (geaggregeerde) data

- Je kunt functies op rijen loslaten, zoals
 - de som `SUM(veldnaam)`
 - gemiddelde `AVG(veldnaam)`
 - maximum- / minimumwaarde `MAX(veldnaam) / MIN(veldnaam)`
 - het aantal rijen `COUNT(*)` of `COUNT(veldnaam)`

Artikel

← p →				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Samengenomen (geaggregeerde) data

- `SELECT COUNT(*) FROM Artikel WHERE voorraad > 20`
output: 2
- `SELECT MAX(prijs) FROM Artikel`
wat is de output?
- `SELECT MIN(prijs) FROM Artikel WHERE voorraad < 25`
wat is de output?

Artikel

← p →				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99

Samengenomen (geaggregeerde) data

- Bedenk de query voor:
de namen van de artikelen met een prijs onder het gemiddelde

← p →

Artikel				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99

Samengenomen (geaggregeerde) data

- Bedenk de query voor:
de namen van de artikelen met een prijs onder de gemiddelde prijs
- `SELECT naam FROM Artikel WHERE prijs < AVG(prijs);`

← p →

Artikel				
id	naam	voorraad	categorie	prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99

Groeperen

- Soms wil je geaggregeerde informatie niet over **alle** geselecteerde rijen, maar gegroepeerd op waarden van een bepaald veld

Artikel

← p →

id	naam	voorraad	categorie	prijs
----	------	----------	-----------	-------

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	wok 23cm	12	kookgerei	29,99
4	USB-kabel 2m	34	elektronica	3,99
...

Groeperen

Artikel

← p →				
id	naam	voorraad	categorie	prijs

```
SELECT COUNT(*) FROM Artikel;
```

COUNT(*)
4

```
SELECT categorie, COUNT(*) FROM Artikel GROUP BY
```

categorie	COUNT(*)
badlinnen	2
kookgerei	1
elektronica	1

Groeperen

- Met WHERE filter je individuele rijen:

```
SELECT naam, COUNT(*) FROM Artikel  
WHERE prijs > 2  
GROUP BY categorie;
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Groeperen

- Met WHERE filter je individuele rijen:

```
SELECT naam, COUNT(*) FROM Artikel
```

```
WHERE prijs > 2
```

```
GROUP BY categorie;
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Groeperen

- Met WHERE filter je individuele rijen:

```
SELECT naam, COUNT(*) FROM Artikel  
WHERE prijs > 2  
GROUP BY categorie;
```

id	naam	voorraad	categorie	prijs
1				
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Groeperen

- Met WHERE filter je individuele rijen:

```
SELECT naam, COUNT(*) FROM Artikel  
WHERE prijs > 2  
  
GROUP BY categorie;
```

id	naam	voorraad	categorie	prijs	COUNT(*)
1					
2	handdoek	23	badlinnen	4,99	2
3	gastendoek	33	badlinnen	2,49	
4	wok 23cm	12	kookgerei	29,99	1
5	USB-kabel 2m	34	elektronica	3,99	1
...	

Groeperen

- Met WHERE filter je individuele rijen:

```
SELECT naam, COUNT(*) FROM Artikel  
WHERE prijs > 2  
GROUP BY categorie;
```

naam	COUNT(*)
handdoek	2
wok 23cm	1
USB-kabel 2m	1

Groepen filteren

- Soms wil je geen rijen, maar groepen filteren.
 - zoals groepen waarin minder dan 2 artikelen voorkomen wegfilteren
- Hiervoor gebruik je HAVING

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Groeperen

- Met HAVING filter je groepen

```
SELECT categorie, COUNT(*) FROM Artikel  
GROUP BY categorie  
HAVING COUNT(*) >= 2
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99

Groeperen

- Met HAVING filter je groepen

```
SELECT categorie, COUNT(*) FROM Artikel
```

```
GROUP BY categorie
```

```
HAVING COUNT(*) >= 2
```

id	naam	voorraad	categorie	prijs	COUNT(*)
1	washandje	19	badlinnen	1,99	3
2	handdoek	23	badlinnen	4,99	
3	gastendoek	33	badlinnen	2,49	
4	wok 23cm	12	kookgerei	29,99	1
5	USB-kabel 2m	34	elektronica	3,99	1

Groeperen

- Met HAVING filter je groepen

```
SELECT categorie, COUNT(*) FROM Artikel  
GROUP BY categorie  
HAVING COUNT(*) >= 2
```

id	naam	voorraad	categorie	prijs	COUNT(*)
1	washandje	19	badlinnen	1,99	3
2	handdoek	23	badlinnen	4,99	
3	gastendoek	33	badlinnen	2,49	
4	wok 23cm	12	kookgerei	29,99	1
5	USB kabel 2m	34	elektronica	3,99	1

Groeperen

- Met HAVING filter je groepen

```
SELECT categorie, COUNT(*) FROM Artikel  
GROUP BY categorie  
HAVING COUNT(*) >= 2
```

id	naam	voorraad	categorie	prijs	COUNT(*)
1	washandje	19	badlinnen	1,99	3

Groeperen

- Met HAVING filter je groepen

```
SELECT categorie, COUNT(*) FROM Artikel  
GROUP BY categorie  
HAVING COUNT(*) >= 2
```

categorie	COUNT(*)
badlinnen	3

Vraag

- Geef de query voor: per categorie de gemiddelde prijs

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: per categorie de gemiddelde prijs

```
SELECT naam, AVG(prijs) FROM Artikel  
GROUP BY categorie;
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: per categorie de gemiddelde prijs, maar alleen van die categorieën die meer dan 50 stuks in voorraad hebben

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: per categorie de gemiddelde prijs, maar alleen van die categorieën die meer dan 50 stuks in voorraad hebben

```
SELECT naam, AVG(prijs) FROM Artikel  
GROUP BY categorie  
HAVING SUM(voorraad) > 50
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: per categorie de voorraad van alle artikelen met een prijs > 2

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: per categorie de voorraad van alle artikelen met een prijs > 2

```
SELECT categorie, SUM(voorraad) FROM Artikel  
WHERE prijs > 2  
GROUP BY categorie;
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: verkoopwaarde van alle artikelen in voorraad

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Vraag

- Geef de query voor: verkoopwaarde van alle artikelen in voorraad

```
SELECT SUM(voorraad * prijs) FROM Artikel
```

id	naam	voorraad	categorie	prijs
1	washandje	19	badlinnen	1,99
2	handdoek	23	badlinnen	4,99
3	gastendoek	33	badlinnen	2,49
4	wok 23cm	12	kookgerei	29,99
5	USB-kabel 2m	34	elektronica	3,99
...

Relaties tussen tabellen

Inhoud

1. Intro relationele databases en eenvoudige queries
2. Geavanceerde queries: geaggregeerde functies en groepen
3. Intro normaliseren m.b.v. relaties tussen tabellen
4. Tabellen uitbreiden met behulp van JOIN
5. Ontwerpen van genormaliseerde databases
6. ...

Lesdoelen

- Je begrijpt hoe je informatie met behulp van sleutels aan elkaar kunt verbinden
- Je kent het verschil tussen sleutels en verwijzende sleutels
- Je maakt kennis met normaliseren
- Je maakt kennis met JOIN

Vraag

- Maak strokendiagram voor het opslaan van jouw cijfers. Ieder cijfer krijgt als primary key de afkorting van het vak + de toetscode (bijv. ak-t01 of in-P03)

Schrijf hier je voorstel



Antwoord

- Maak strokendiagram voor het opslaan van jouw cijfers. Ieder cijfer krijgt als primary key de afkorting van het vak + de toetscode (bijv. ak-T01 of in-P03)



code	cijfers
ak-T01	8.4
in-P03	7.2
...	...

Vraag

- Maak strokendiagram voor het opslaan van de cijfers van alle leerlingen uit jouw klas. Van de leerling sla je het **leerlingnummer** en de **voornaam** ieder cijfer krijgt als primary key de afkorting van het vak + de toetscode (bijv. ak-t01 of in-P03)

Schrijf hier je voorstel



- Maak strokendiagram voor het opslaan van jouw cijfers. Ieder cijfer krijgt als primary key de afkorting van het vak + de toetscode (bijv. ak-T01 of in-P03)

cijfers

leerlingnummer	voornaam	cijfercode	cijfer
----------------	----------	------------	--------

leerlingnummer	voornaam	code	cijfers
101279	Joop	ak-T01	8.4
139002	Andrea	in-P03	7.2
139002	Andrea	bi-T02	5.1
101279	Joop	in-P03	4.8
099374	Karim	mo-T01	7.8
...

- Maak strokendiagram voor het opslaan van jouw cijfers. Ieder cijfer krijgt als primary key de afkorting van het vak + de toetscode (bijv. ak-T01 of in-P03)

cijfers

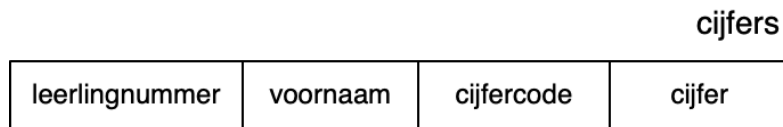
leerlingnummer	voornaam	cijfercode	cijfer
----------------	----------	------------	--------

wat is de primary key?!



leerlingnummer	voornaam	code	cijfers
101279	Joop	ak-T01	8.4
139002	Andrea	in-P03	7.2
139002	Andrea	bi-T02	5.1
101279	Joop	in-P03	4.8
099374	Karim	mo-T01	7.8
...

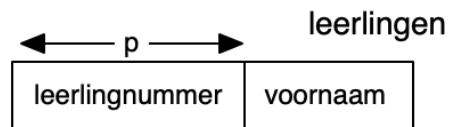
- Maak strokendiagram voor het opslaan van jouw cijfers. Ieder cijfer krijgt als primary key de afkorting van het vak + de toetscode (bijv. ak-T01 of in-P03)



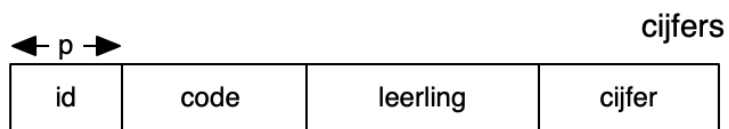
dubbele informatie
(redundantie)

wat is de primary key?!

leerlingnummer	voornaam	code	cijfers
101279	Joop	ak-T01	8.4
139002	Andrea	in-P03	7.2
139002	Andrea	bi-T02	5.1
101279	Joop	in-P03	4.8
099374	Karim	mo-T01	7.8
...

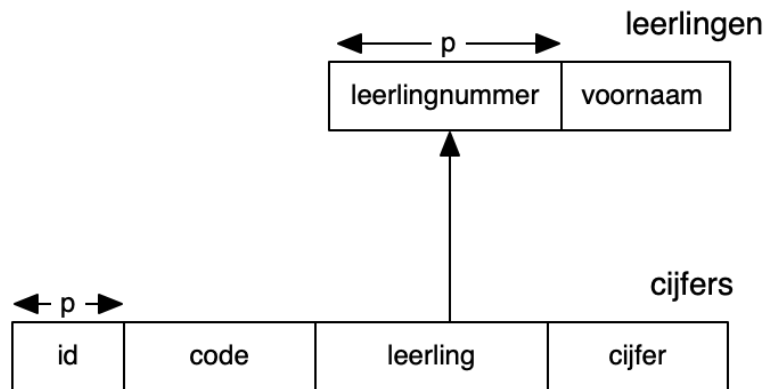


leerlingnummer	voornaam
101279	Joop
139002	Andrea
099374	Karim
...	...



id	code	leerlingnummer	cijfers
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8
5	mo-T01	099374	7.8
...

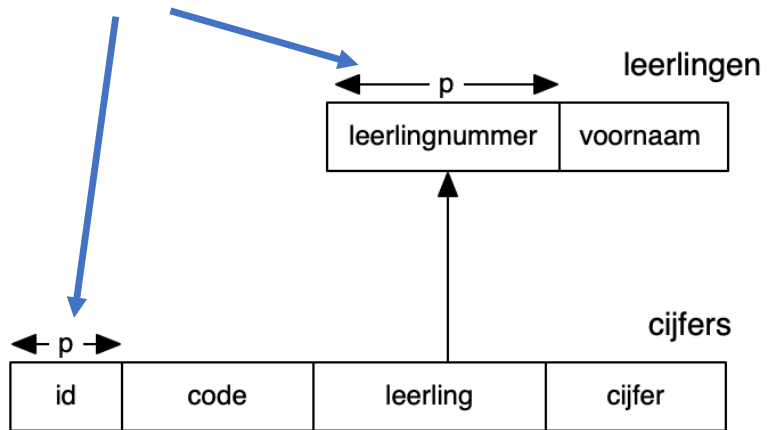
Antwoord



leerlingnummer	voornaam
101279	Joop
139002	Andrea
099374	Karim
...	...

id	code	leerlingnummer	cijfers
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8
5	mo-T01	099374	7.8
...

primaire sleutels



verwijzende sleutel

leerlingnummer	voornaam
101279	Joop
139002	Andrea
099374	Karim
...	...

code	leerlingnummer	cijfers
ak-T01	101279	8.4
in-P03	139002	7.2
bi-T02	139002	5.1
in-P03	101279	4.8
mo-T01	099374	7.8
...

Het gebruik van JOIN

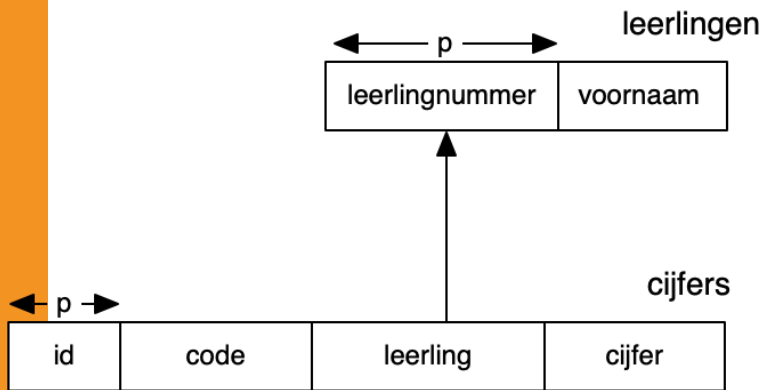
Inhoud

1. Intro relationele databases en eenvoudige queries
2. Geavanceerde queries: geaggregeerde functies en groepen
3. Intro normaliseren m.b.v. relaties tussen tabellen
4. Tabellen uitbreiden met behulp van JOIN
5. Ontwerpen van genormaliseerde databases

Lesdoelen

- Je begrijpt hoe je met JOIN tabel en tabel kunt uitbreiden met andere tabellen
- Je leert in welke volgorde onderdelen worden afgewerkt bij een eenvoudige query met JOINS.

Schooldatabase



leerlingnummer	voornaam
101279	Joop
139002	Andrea
099374	Karim
...	...

id	code	leerlingnummer	cijfers
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8
5	mo-T01	099374	7.8
...

Schooldatabase

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
		3	bi-T02	139002	5.1
		4	in-P03	101279	4.8

SELECT * FROM leerlingen JOIN cijfers

leerlingnummer	voornaam	code	leerling	cijfer
101279	Joop	ak-T01	101279	8.4
101279	Joop	in-P03	139002	7.2
101279	Joop	bi-T02	139002	5.1
101279	Joop	in-P03	101279	4.8
139002	Andrea	ak-T01	101279	8.4
139002	Andrea	in-P03	139002	7.2
139002	Andrea	bi-T02	139002	5.1
139002	Andrea	in-P03	101279	4.8

Schooldatabase

leerlingnummer	voornaam
101279	Joop
139002	Andrea

id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

uitgebreide tabel

SELECT * FROM **leerlingen** JOIN **cijfers**

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

leerlingnummer	voornaam
101279	Joop
139002	Andrea

id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

SELECT * FROM leerlingen JOIN cijfers

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

leerlingnummer	voornaam
101279	Joop
139002	Andrea

id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

SELECT * FROM leerlingen JOIN cijfers

leerlingnummer	voornaam	id	code	leerling	cijfers
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

leerlingnummer	voornaam
101279	Joop
139002	Andrea

id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

SELECT * FROM leerlingen JOIN cijfers

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

leerlingnummer	voornaam
101279	Joop
139002	Andrea

id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

SELECT * FROM leerlingen JOIN cijfers

leerlingnummer	voornaam	id	code	leerling	cijfers
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

Ik heb niets aan rijen met

SELECT * FROM leerlingen JOIN cijfers
 leerlingnummer, voornaam, id, code
 cijfers van Joop met
 cijfers van Andrea of andersom

leerlingnummer	voornaam
101279	Joop
139002	Andrea

id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

leerlingnummer	voornaam
101279	Joop
139002	Andrea

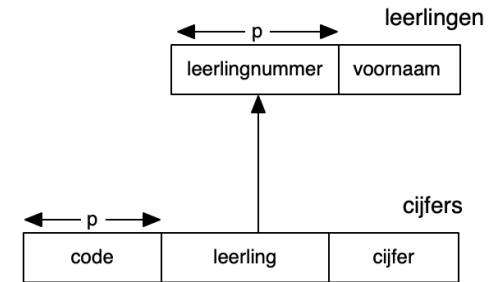
id	code	leerling	cijfer
1	ak-T01	101279	8.4
2	in-P03	139002	7.2
3	bi-T02	139002	5.1
4	in-P03	101279	4.8

SELECT * FROM leerlingen JOIN cijfers

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

SELECT * FROM leerlingen JOIN cijfers

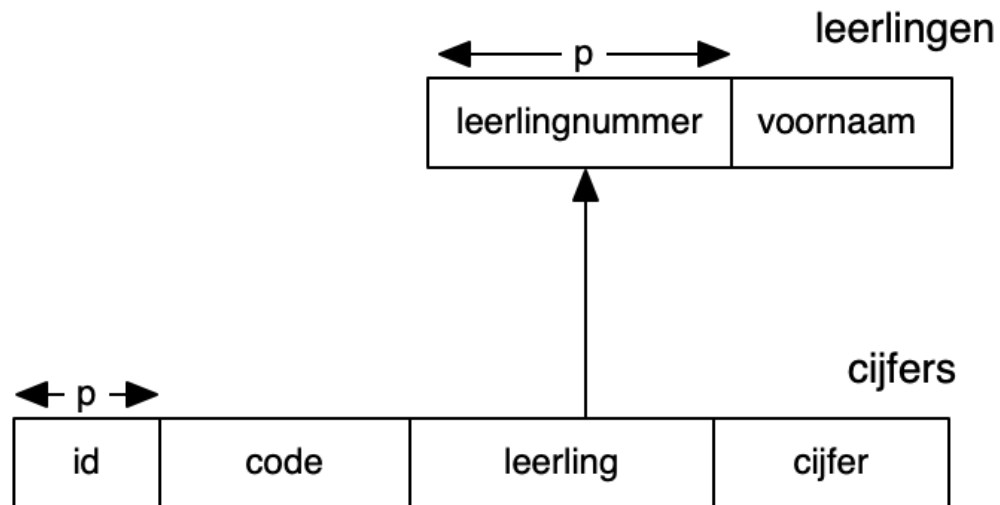


leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	139002	7.2
101279	Joop	3	bi-T02	139002	5.1
101279	Joop	4	in-P03	101279	4.8
139002	Andrea	1	ak-T01	101279	8.4
139002	Andrea	2	in-P03	139002	7.2
139002	Andrea	3	bi-T02	139002	5.1
139002	Andrea	4	in-P03	101279	4.8

Schooldatabase

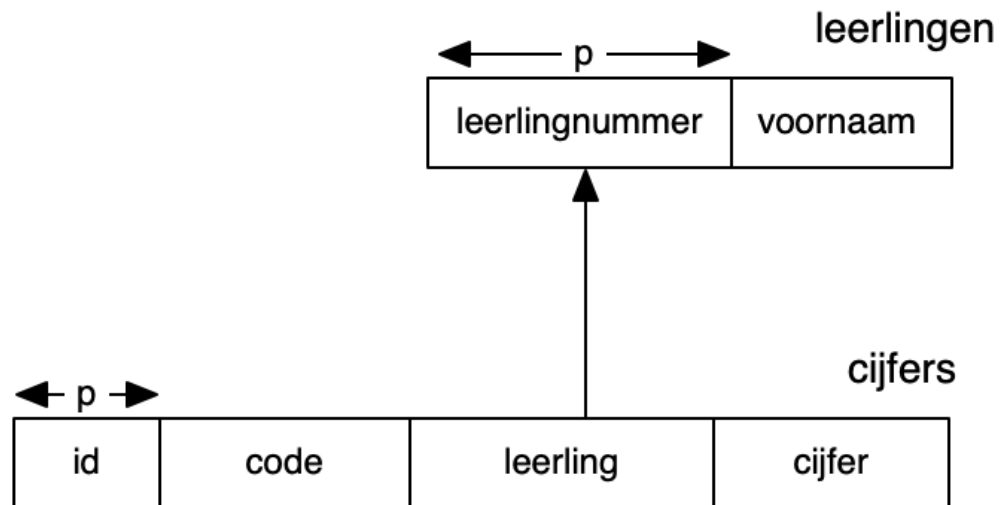
```
SELECT * FROM leerlingen JOIN cijfers
```

"alleen als cijfers.leerling hetzelfde is als leerlingen.leerlingnummer"



Schooldatabase

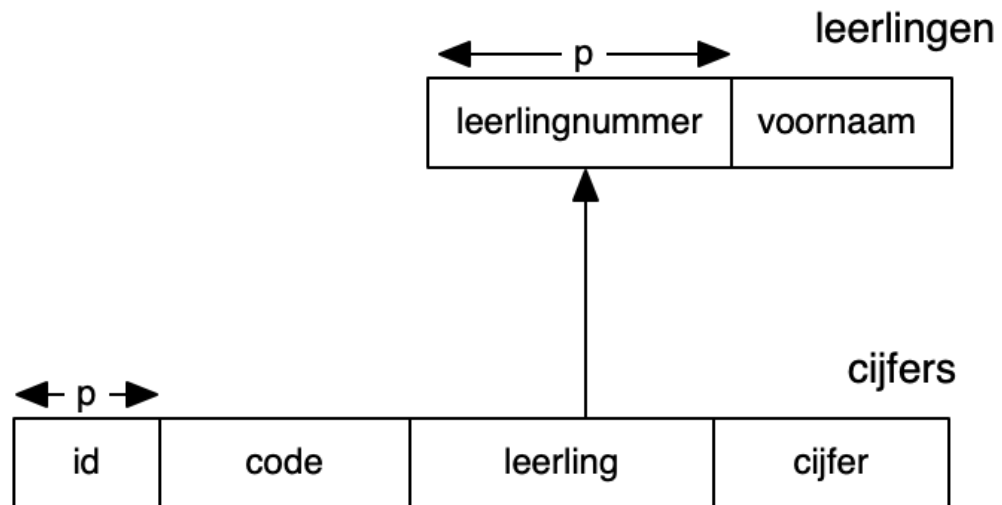
```
SELECT * FROM leerlingen JOIN cijfers ON cijfers.leerling = leerlingen.leerlingnummer
```



Schooldatabase

```
SELECT * FROM leerlingen
```

```
JOIN cijfers ON cijfers.leerling = leerlingen.leerlingnummer
```



Schooldatabase

```
SELECT * FROM leerlingen
JOIN cijfers ON leerlingen.leerlingnummer = cijfers.leerling
```

leerlingnummer	voornaam	code	leerling	cijfer
101279	Joop	ak-T01	101279	8.4
139002	Andrea	in-P03	139002	7.2
		bi-T02	139002	5.1
		in-P03	101279	4.8

leerlingnummer	voornaam	id	code	leerling	cijfer
101279	Joop	1	ak-T01	101279	8.4
101279	Joop	2	in-P03	101279	4.8
139002	Andrea	3	in-P03	139002	7.2
139002	Andrea	4	bi-T02	139002	5.1

Schooldatabase

SELECT voornaam, cijfer FROM leerlingen
JOIN cijfers ON leerlingen.leerlingnummer = cijfers.leerling

leerlingnummer	voornaam	code	leerling	cijfer
101279	Joop	ak-T01	101279	8.4
139002	Andrea	in-P03	139002	7.2
		bi-T02	139002	5.1
		in-P03	101279	4.8

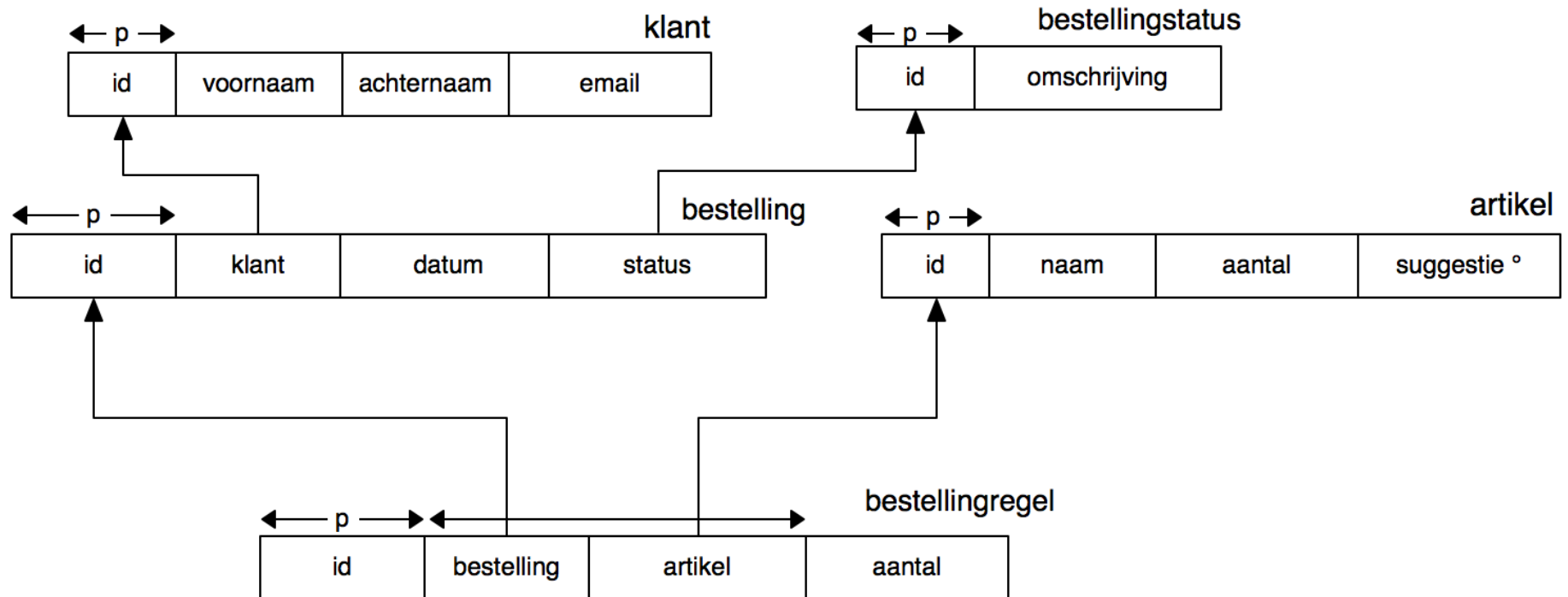
voornaam	cijfer
Joop	8.4
Joop	4.8
Andrea	7.2
Andrea	5.1

Stappen bij oplossen query-vragen:

- Uit welke tabel(len) moet ik de informatie halen?
Heb je informatie uit meerdere tabellen nodig -> gebruik JOIN
- ...
- Selecteer de kolommen die je nodig hebt.

Vraag: welke tabellen heb ik nodig?

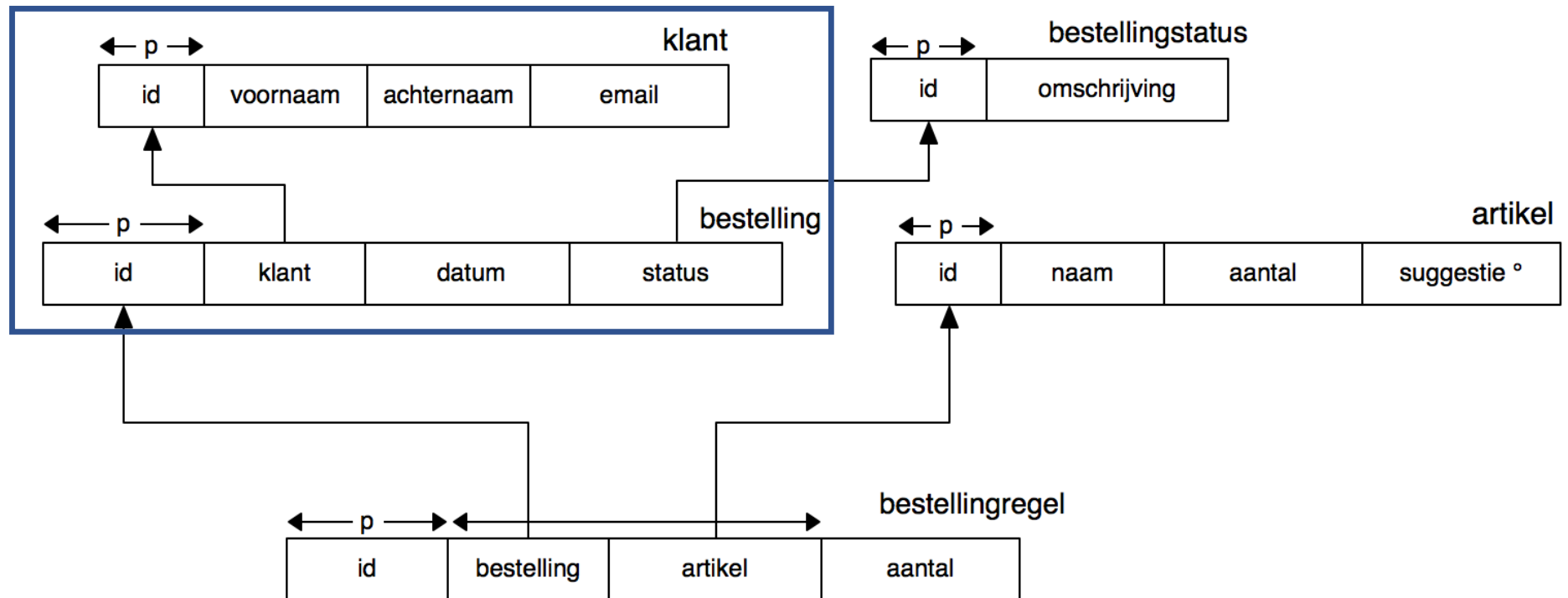
Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.



Vraag: wat komt er na ON?

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.

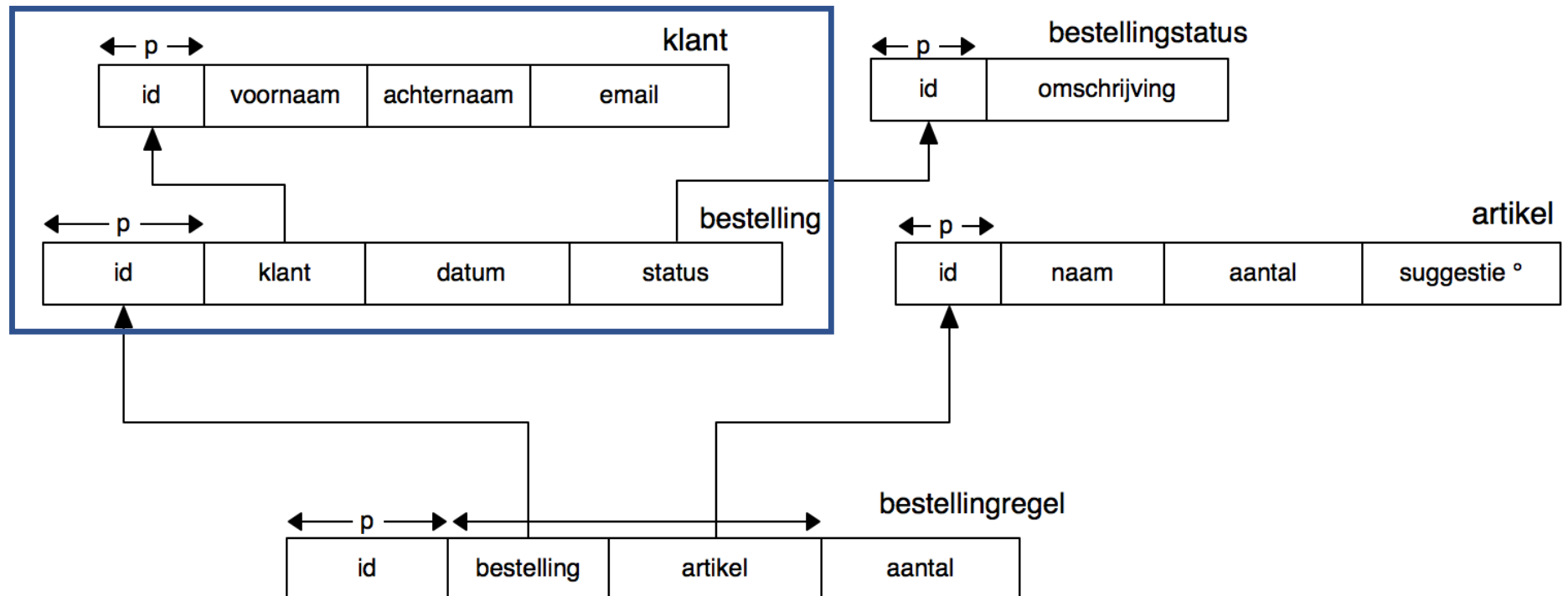
SELECT..... FROM bestelling JOIN klant ON



Vraag: wat komt er na ON?

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.

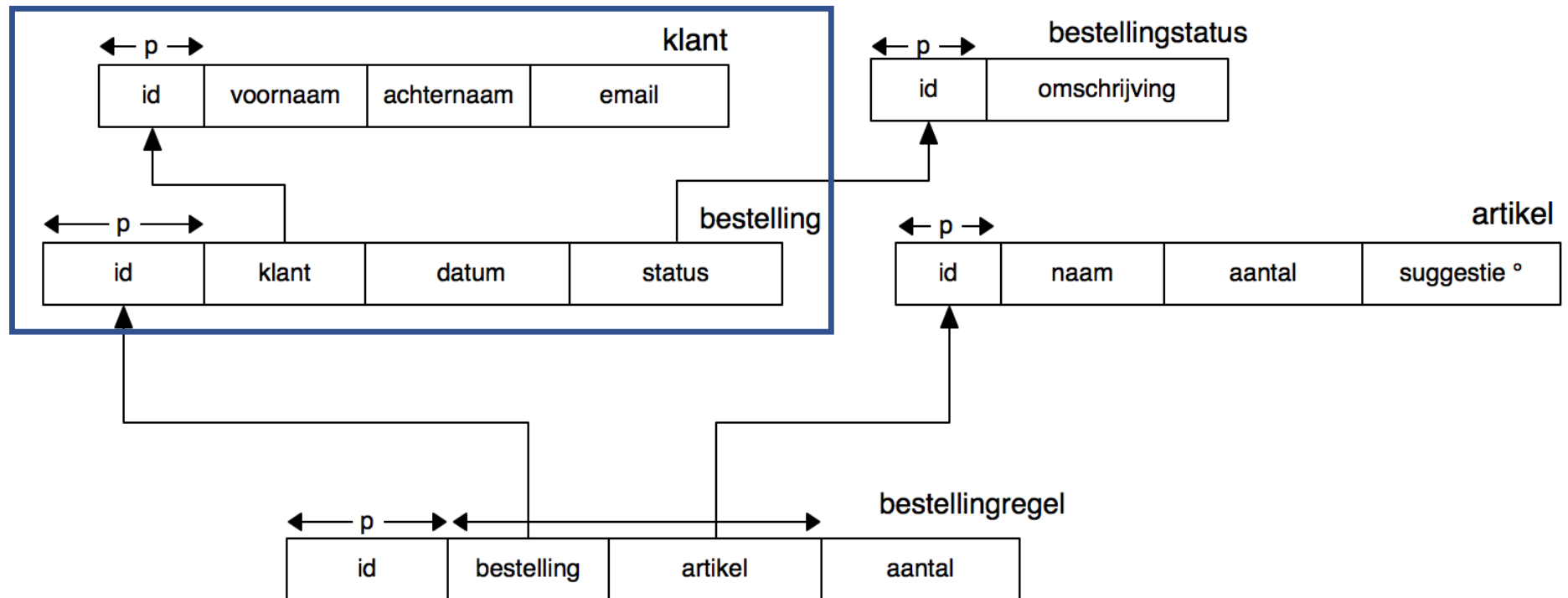
SELECT..... FROM bestelling JOIN klant ON "alleen die regels waar bestelling.klant en klant.id gelijk zijn"



Vraag: wat komt er na ON?

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.

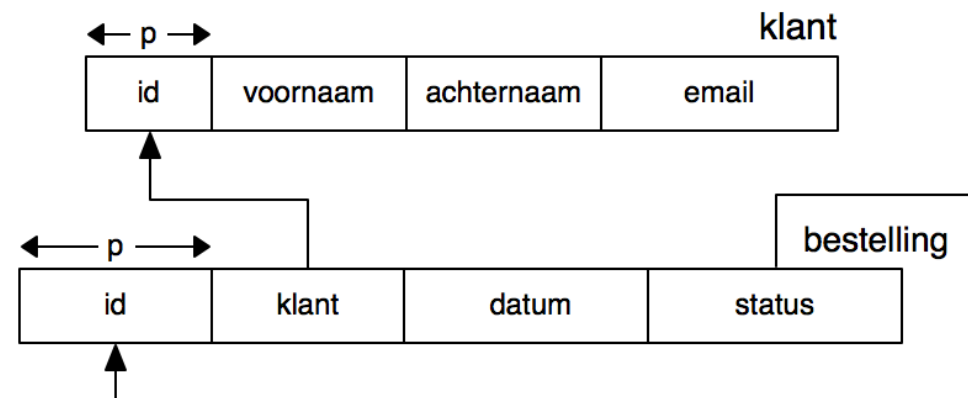
SELECT..... FROM bestelling JOIN klant ON bestelling.klant = klant.id



Vraag: welke kolommen selecteer je?

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.

SELECT..... FROM bestelling JOIN klant ON bestelling.klant = klant.id



id	klant	datum	status	id	voornaam	achternaam	email
1	1	29/09/2018	4	1	Jan	Fontein	j.fontein@hotmail.com
2	5	30/09/2018	4	5	Laila	d'Hiver	laila@gmail.com
3	2	01/01/2019	4	2	Kasim	Nuberres	knuberres@gmail.com
4	1	03/03/2020	1	1	Jan	Fontein	j.fontein@hotmail.com

Vraag: welke kolommen selecteer je?

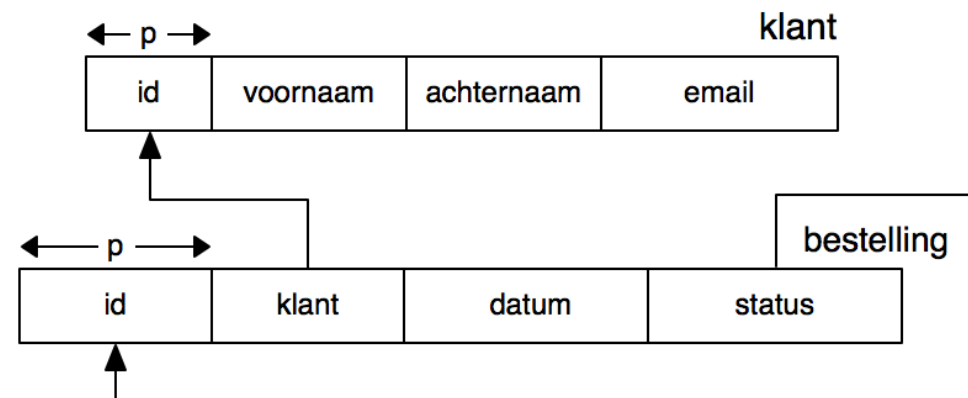
Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.

```
SELECT bestelling.id, voornaam, achternaam
FROM bestelling
JOIN klant ON bestelling.klant = klant.id
```

bestelling.id



klant.id



id	klant	datum	status	id	voornaam	achternaam	email
1	1	29/09/2018	4	1	Jan	Fontein	j.fontein@hotmail.com
2	5	30/09/2018	4	5	Laila	d'Hiver	laila@gmail.com
3	2	01/01/2019	4	2	Kasim	Nuberres	knuberres@gmail.com
4	1	03/03/2020	1	1	Jan	Fontein	j.fontein@hotmail.com

Vraag: wat komt er na ON?

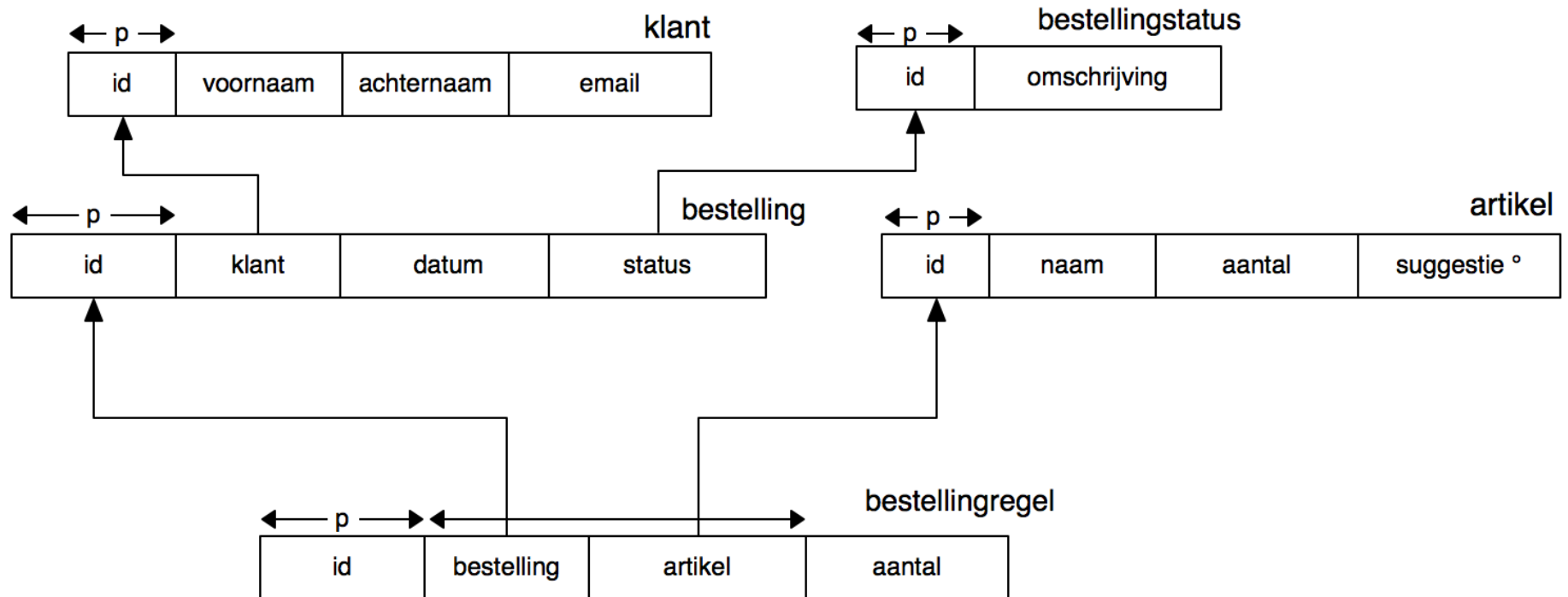
Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant.

```
SELECT bestelling.id, voornaam, achternaam  
FROM bestelling  
JOIN klant ON bestelling.klant = klant.id;
```

id	voornaam	achternaam
1	Jan	Fontein
2	Laila	d'Hiver
3	Kasim	Nuberres
4	Jan	Fontein

Vraag: welke tabellen heb ik nodig?

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.



Vraag: welke tabellen heb ik nodig?

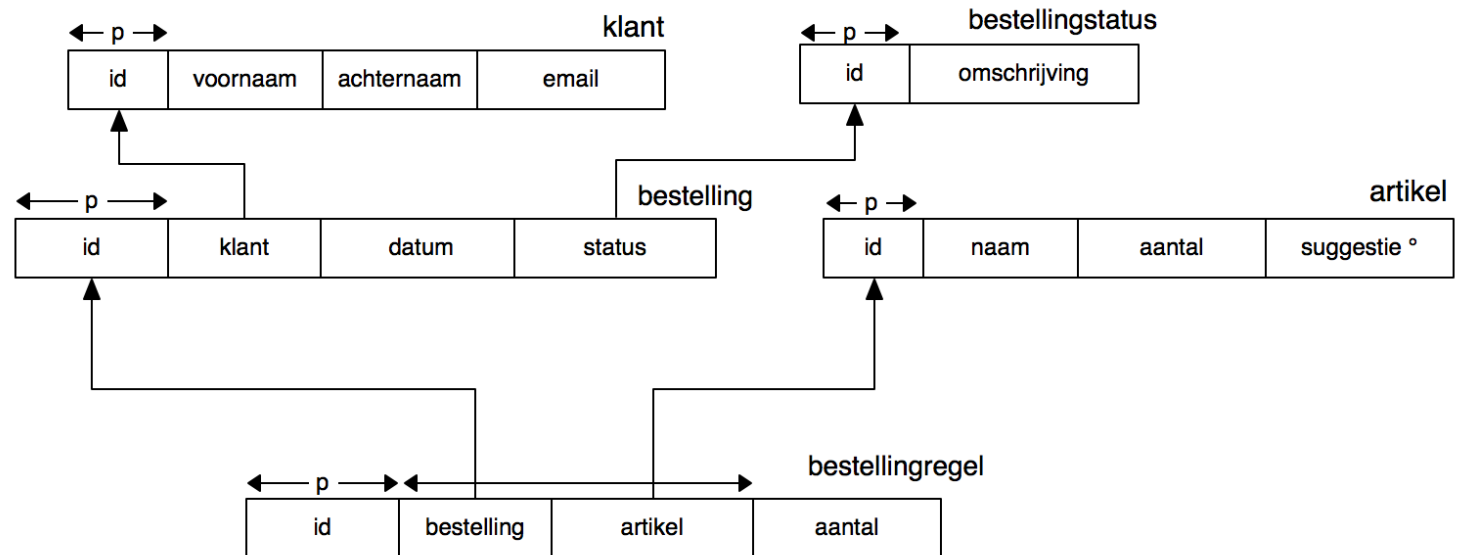
Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

SELECT ...

FROM bestelling

JOIN klant ON bestelling.klant = klant.id

JOIN bestellingstatus ON bestelling.status = status.id



'Tussentabel'

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

SELECT ...

FROM bestelling

JOIN klant ON bestelling.klant = klant.id

JOIN bestellingstatus ON bestelling.status = status.id

bestellingstatus

id	voornaam
1	verzonden
2	ingepakt
3	betaald
4	bezorgd

bestelling.id

klant.id

bestellingstatus.id

id	klant	datum	status	id	voornaam	achternaam	email	id	omschrijving
1	1	29/09/2018	4	1	Jan	Fontein	j.fontein@hotmail.com	4	bezorgd
2	5	30/09/2018	4	5	Laila	d'Hiver	laila@gmail.com	4	bezorgd
3	2	01/01/2019	4	2	Kasim	Nuberres	knuberres@gmail.com	4	bezorgd
4	1	03/03/2020	1	1	Jan	Fontein	j.fontein@hotmail.com	1	verzonden

'Tussentabel'

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

SELECT ...

FROM bestelling

JOIN klant ON bestelling.klant = klant.id

JOIN bestellingstatus ON bestelling.status = status.id

bestellingstatus

id	voornaam
1	verzonden
2	ingepakt
3	betaald
4	bezorgd

bestelling.id

klant.id

bestellingstatus.id

id	klant	datum	status	id	voornaam	achternaam	email	id	omschrijving
1	1	29/09/2018	4	1	Jan	Fontein	j.fontein@hotmail.com	4	bezorgd
2	5	30/09/2018	4	5	Laila	d'Hiver	laila@gmail.com	4	bezorgd
3	2	01/01/2019	4	2	Kasim	Nuberres	knuberres@gmail.com	4	bezorgd
4	1	03/03/2020	1	1	Jan	Fontein	j.fontein@hotmail.com	1	verzonden

Kolommen selecteren

Alle id's van de bestellingen met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

```
SELECT bestelling.id, voornaam, achternaam, omschrijving
FROM bestelling
JOIN klant ON bestelling.klant = klant.id
JOIN bestellingstatus ON bestelling.status = status.id;
```

id	voornaam	achternaam	omschrijving
1	Jan	Fontein	bezorgd
2	Laila	d'Hiver	bezorgd
3	Kasim	Nuberres	bezorgd
4	Jan	Fontein	verzonden

Maar nu...

Alle id's van **de bestellingen van Jan** met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

```
SELECT ...  
FROM bestelling  
JOIN klant ON bestelling.klant = klant.id  
JOIN bestellingstatus ON bestelling.status = status.id
```

id	klant	datum	status	id	voornaam	achternaam	email	id	omschrijving
1	1	29/09/2018	4	1	Jan	Fontein	j.fontein@hotmail.com	4	bezorgd
2	5	30/09/2018	4	5	Laila	d'Hiver	laila@gmail.com	4	bezorgd
3	2	01/01/2019	4	2	Kasim	Nuberres	knuberres@gmail.com	4	bezorgd
4	1	03/03/2020	1	1	Jan	Fontein	j.fontein@hotmail.com	1	verzonden

Maar nu...

Alle id's van **de bestellingen van Jan** met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

```
SELECT ...  
FROM bestelling  
JOIN klant ON bestelling.klant = klant.id  
JOIN bestellingstatus ON bestelling.status = status.id  
WHERE voornaam = "Jan";
```

id	klant	datum	status	id	voornaam	achternaam	email	id	omschrijving
1	1	29/09/2018	4	1	Jan	Fontein	j.fontein@hotmail.com	4	bezorgd
4	1	03/03/2020	1	1	Jan	Fontein	j.fontein@hotmail.com	1	verzonden

Maar nu...

Alle id's van **de bestellingen van Jan** met daarbij de voor- en achternaam van de klant en de omschrijving van de bestellingstatus.

```
SELECT bestelling.id, voornaam, achternaam, omschrijving
FROM bestelling
JOIN klant ON bestelling.klant = klant.id
JOIN bestellingstatus ON bestelling.status = status.id
WHERE voornaam = "Jan";
```

id	voornaam	achternaam	omschrijving
1	Jan	Fontein	bezorgd
4	Jan	Fontein	verzonden

Stappenplan Queries maken

1. **Bekijk de database**
 - A. bekijk de gegevens in alle tabellen,
 - B. lees de tekst die erbij staat zodat je snapt hoe de tabellen aan elkaar gerelateerd zijn.
2. **Welke velden willen we zien?**
 - A. Welke velden uit welke tabel?
 - B. Welke velden die ik moet berekenen?
 - C. Tip: gebruik sum, count enzovoort om te berekenen
 - D. Tip: vervang pas aan het einde SELECT * door SELECT <de velden die je nodig hebt>
3. **Welke tabellen heb ik nodig?**
 - A. Alle tabellen waaruit ik velden gebruik EN
 - B. Tabellen die nodig zijn om “te verbinden”
 - C. Tip: Gebruik JOIN en ON bij meer dan 1 tabel
4. **Moet ik rijen groeperen?**
 - A. Tip: vaak (niet altijd) nodig als er “per” in de vraag staat en als er berekende velden zijn
 - B. Gebruik GROUP BY
5. **Moet ik rijen filteren**
 - A. Gebruik HAVING (met GROUP BY) of WHERE
6. **Moet ik sorteren?**
 - A. Gebruik ORDER BY

Subqueries

Inhoud

1. Intro relationele databases en eenvoudige queries
2. Geavanceerde queries: geaggregeerde functies en groepen
3. Intro normaliseren m.b.v. relaties tussen tabellen
4. Tabellen uitbreiden met behulp van JOIN
5. Ontwerpen van genormaliseerde databases
6. Het gebruik van subqueries

Lesdoelen

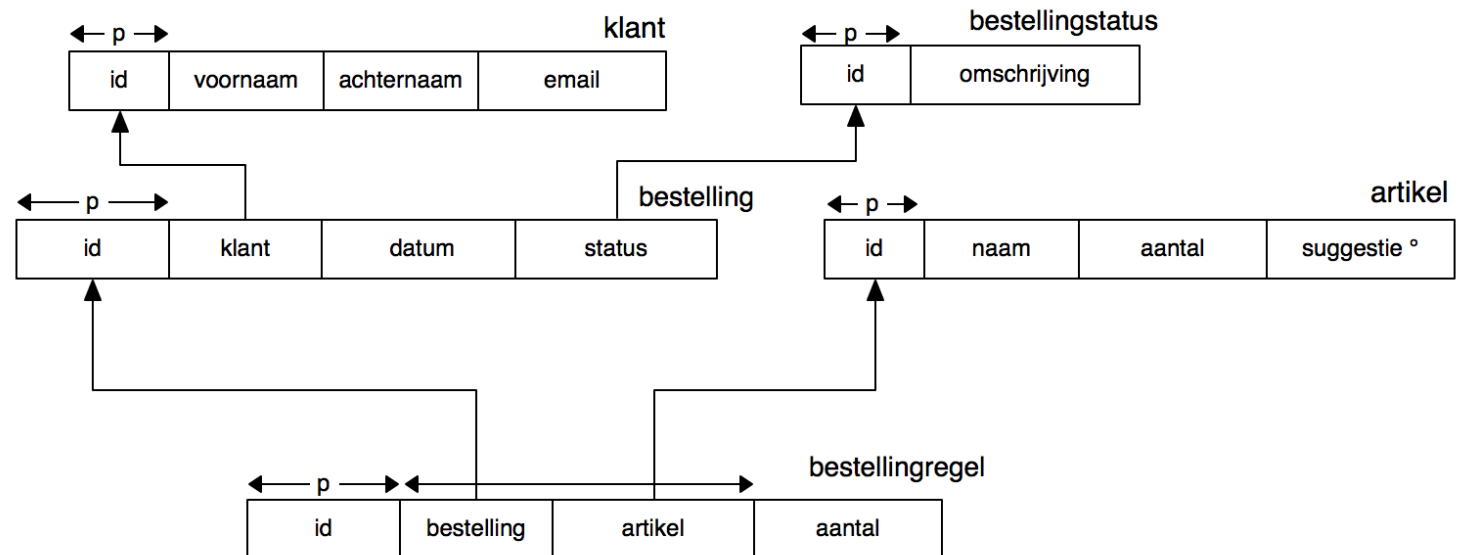
- Je begrijpt wat een subquery is.
- Je kunt een subquery toepassen.

Grootste aantal exemplaren van een artikel

```
SELECT MAX(aantal) FROM artikel;
```

MAX(aantal)

29

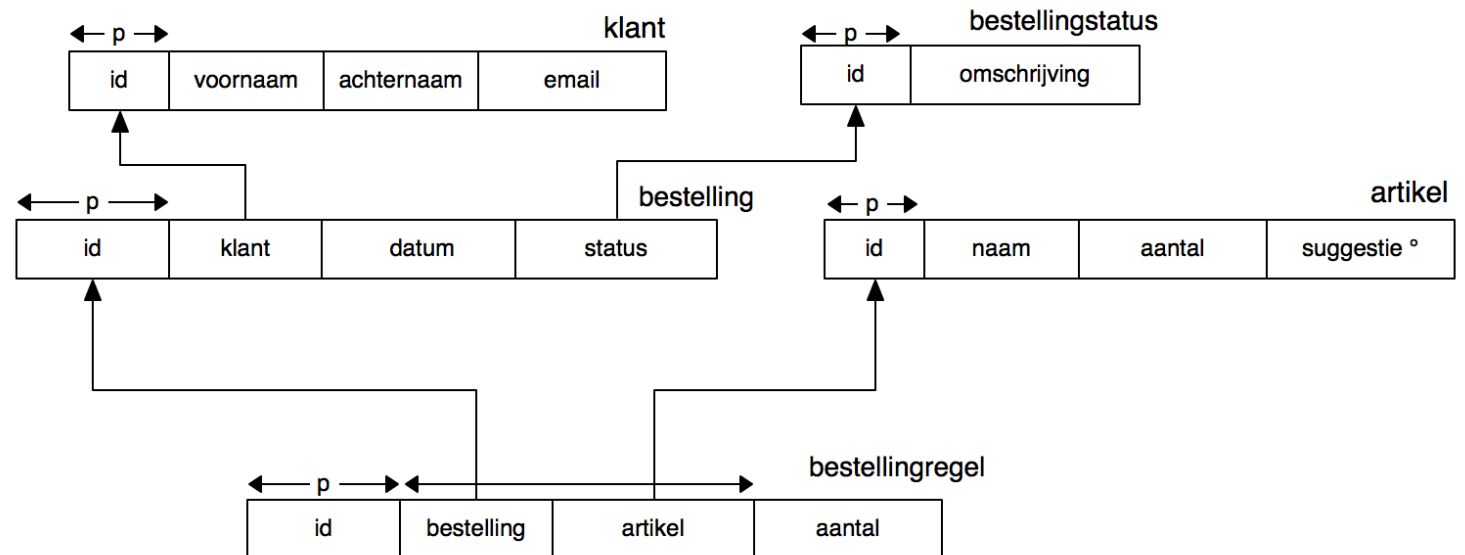


Grootste aantal exemplaren van een artikel

SELECT MAX(aantal) as "grootste hoeveelheid" FROM artikel;

**grootste
hoeveelheid**

29



Naam van artikel met de grootste hoeveelheid

```
SELECT naam FROM artikel ...
```

artikel

id	naam	aantal	suggestie
1	bout M4	12	72
2	moer M4	9	83
3	plaatschroef	29	NULL
4	spaanplaat 1m2	6	NULL

Naam van artikel met de grootste hoeveelheid

SELECT naam FROM artikel WHERE ...

rijen filteren



artikel

id	naam	aantal	suggestie
1	bout M4	12	72
2	moer M4	9	83
3	plaatschroef	29	NULL
4	spaanplaat 1m2	6	NULL
...

Naam van artikel met de grootste hoeveelheid

```
SELECT naam FROM artikel WHERE
```

artikel

id	naam	aantal	suggestie
1	bout M4	12	72
2	moer M4	9	83
3	plaatschroef	29	NULL
4	spaanplaat 1m2	6	NULL
...

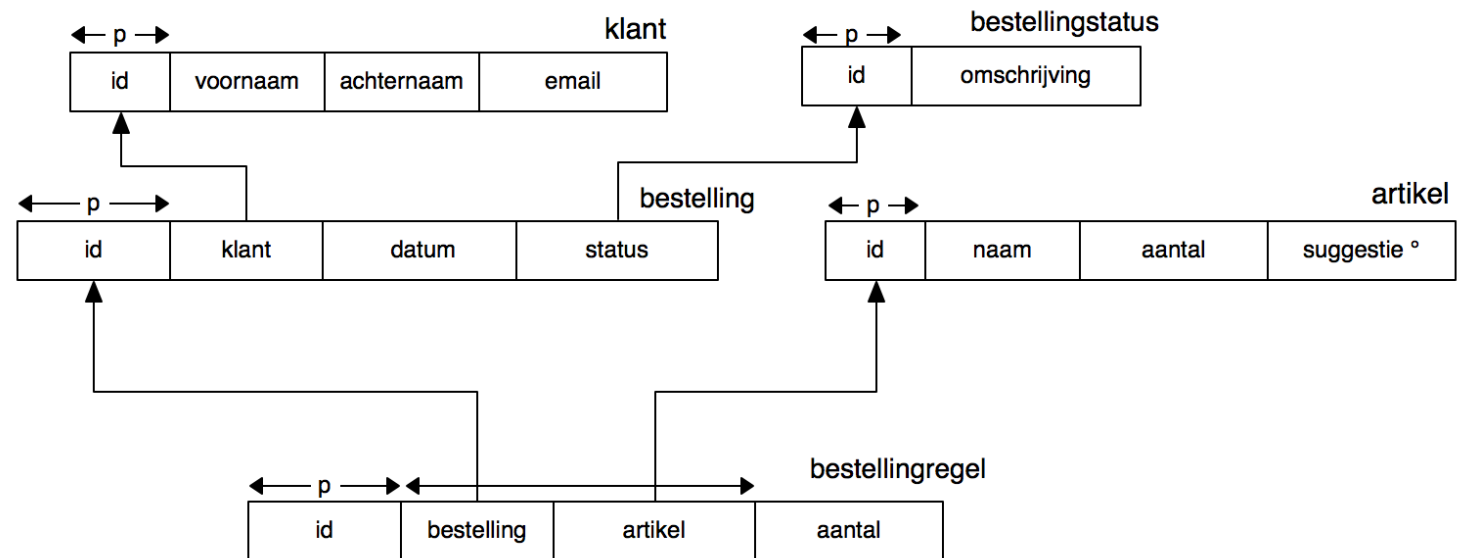
Query met IN

```
SELECT <velden> FROM <tabel>
```

```
WHERE <veld> = waarde1 OR veld = waarde2 OR veld = waarde3;
```

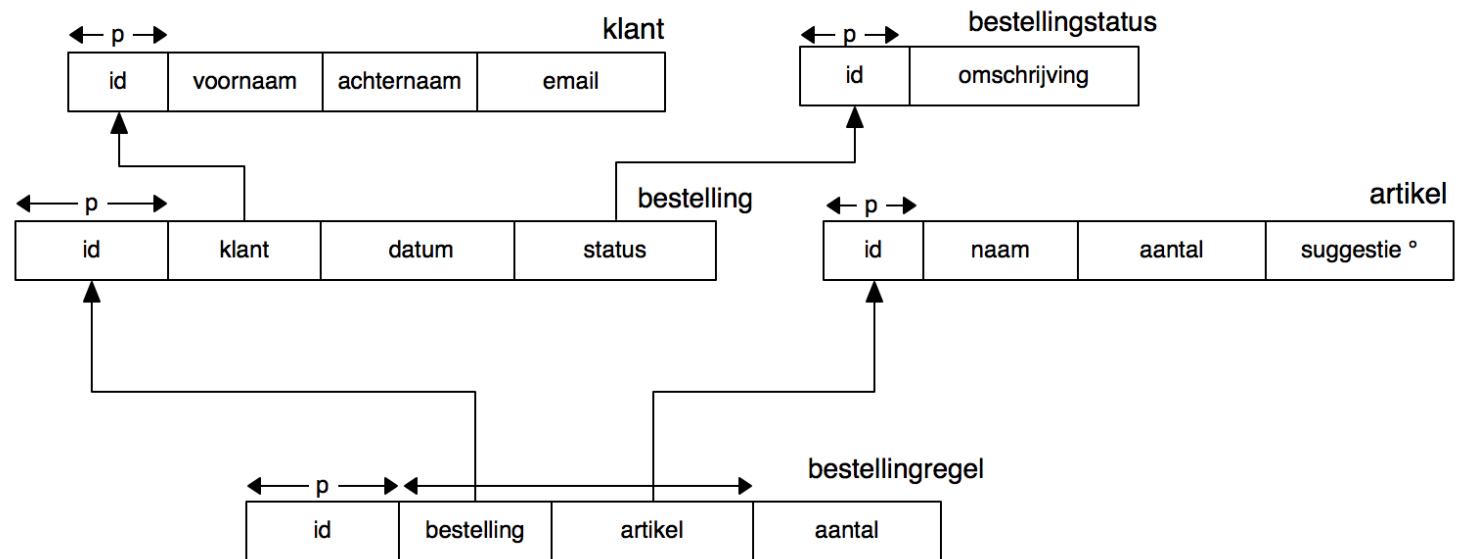
```
SELECT <velden> FROM <tabel>
```

```
WHERE <veld> IN (waarde1, waarde2, waarde3);
```



Query met IN

```
SELECT naam FROM artikel  
WHERE aantal = 3 OR aantal = 5 OR aantal = 7;  
SELECT naam FROM artikel  
WHERE aantal IN (3, 5, 7);
```



Query met IN

```
SELECT naam FROM artikel  
WHERE aantal IN (3, 5, 7);
```

(3, 5, 7) mag je ook aanleveren als een tabel:

```
SELECT aantal FROM artikel  
WHERE aantal = 3 OR aantal = 5 OR aantal = 7;
```

Beetje onzinnig hier

aantal
3
5
7

Optionele velden, NULL en LEFT OUTER JOINS

Inhoud

1. Intro relationele databases en eenvoudige queries
2. Geavanceerde queries: geaggregeerde functies en groepen
3. Intro normaliseren m.b.v. relaties tussen tabellen
4. Tabellen uitbreiden met behulp van JOIN
5. Ontwerpen van genormaliseerde databases
6. Het gebruik van subqueries
7. Optionele velden, NULL en LEFT OUTER JOINS

Lesdoelen

- Kennis opfrissen
- Je begrijpt wat een optioneel veld is en hoe je die weergeeft in een strokendiagram
- Je weet wat NULL is
- Je weet wanneer je een LEFT OUTER JOIN gebruikt.

Even opfrissen...

Een relationele database werkt met tabellen, vaak heel veel tabellen

id	voornaam	achternaam	omschrijving
1	Jan	Fontein	bezorgd
4	Jan	Fontein	verzonden

Even opfrissen...

SELECT <velden> FROM <tabel>

JOIN <tabel> ON ...

WHERE <voorwaarde>

GROUP BY <veld>

HAVING <voorwaarde>

ORDER BY <veld>

Informatie opvragen

Tabel uitbreiden met informatie uit andere tabel

Rijen filteren met behulp van een voorwaarde

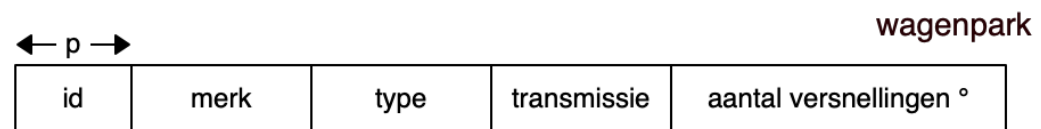
Bij geaggregeerde functies (zoals MAX(...)) groeperen van resultaten

Groepen filteren

Rijen sorteren op basis van een veld

Optionele velden

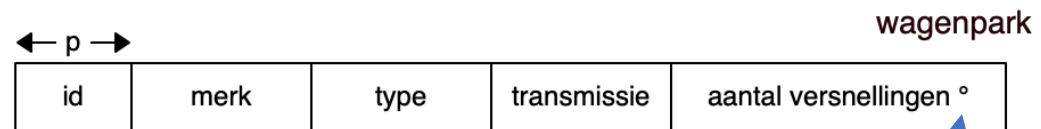
Soms ontkom je niet aan het feit dat een veld leeg blijft.



id	merk	type	transmissie	aantal versnellingen
1	Opel	Astra 1.6	handmatig	6
2	Opel	Zafira 1.8	handmatig	5
3	Tesla	Model 3	automaat	NULL
...

Optionele velden

Soms ontkom je niet aan het feit dat een veld leeg blijft.



id	merk	type	transmissie	aantal versnellingen
1	Opel	Astra 1.6	handmatig	6
2	Opel	Zafira 1.8	handmatig	5
3	Tesla	Model 3	automaat	NULL
...

Optioneel veld geef je aan met een °

NULL betekent “geen waarde” of “onbekend”

Is dus niet hetzelfde als 0 of "" (lege string)

Optionele velden

Soms ontkom je niet aan het feit dat een veld leeg blijft

wagenpark

id	merk	type	aantal versnellingen °
----	------	------	------------------------

id	merk	type	aantal versnellingen
1	Opel	handmatig	6
2	Opel Astra 1.8	handmatig	
3	Tesla Model 3	toerwagen	NULL
...

Is dit een goed genormaliseerde database?

Optionele velden

Geef suggesties voor het netter opslaan van deze gegevens.

wagenpark

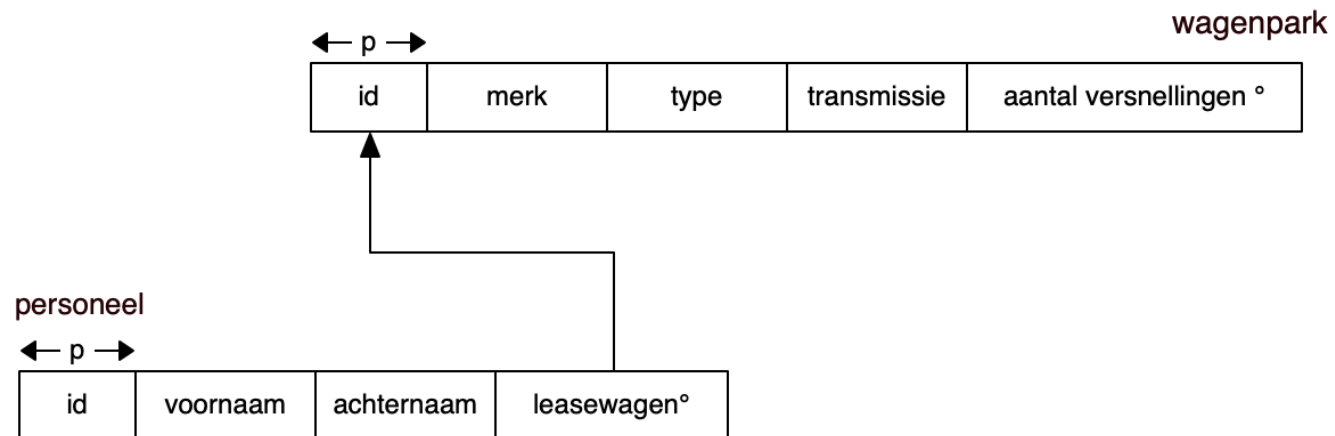
← p →

id	merk	type	transmissie	aantal versnellingen °
----	------	------	-------------	------------------------

id	merk	type	transmissie	aantal versnellingen
1	Opel	Astra 1.6	handmatig	6
2	Opel	Zafira 1.8	handmatig	5
3	Tesla	Model 3	automaat	NULL
...

Optionele velden

Ook velden met een verwijzende sleutel mogen optioneel zijn

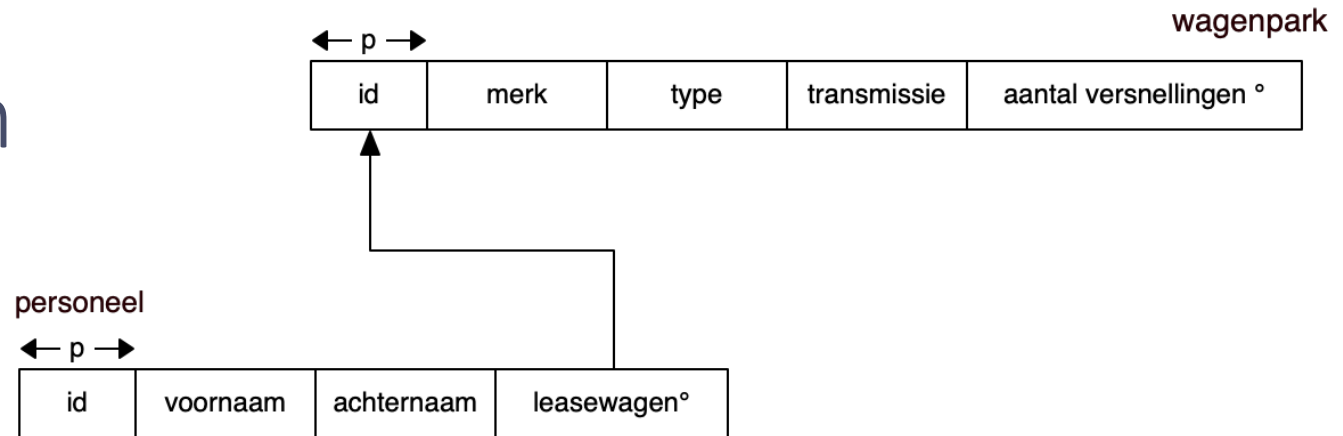


Optionele velden

Nu een JOIN:

```
SELECT * FROM personeel
```

```
JOIN wagenpark ON personeel.leasewagen = wagenpark.id
```



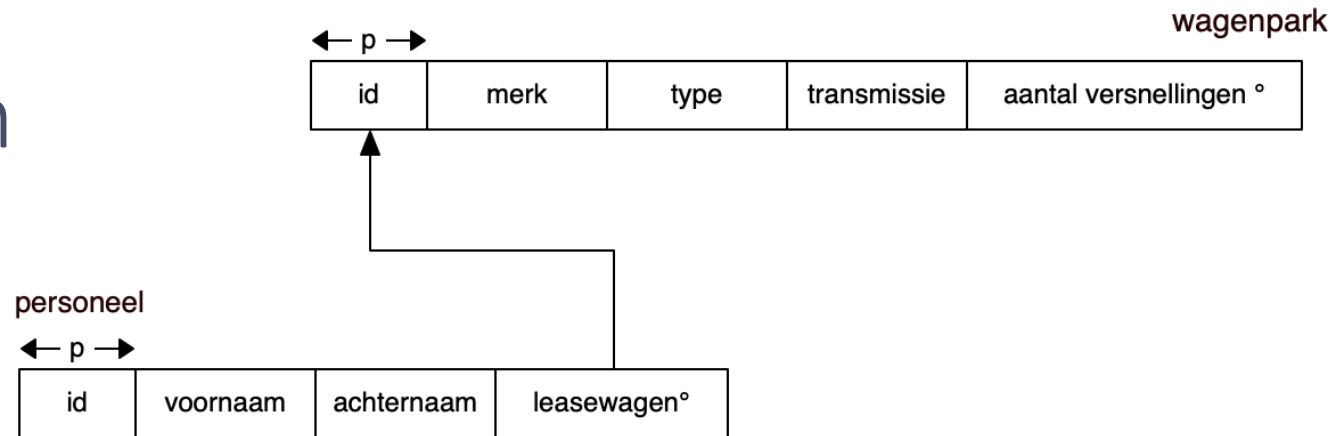
id	voornaam	achternaam	leasewagen	id	merk	type	transmissie	aantal versnellingen
1	Suus	Pletter	2	2	Opel	Zafira 1.8	handmatig	5
2	Joop	Graafland	NULL	NULL	NULL	NULL	NULL	NULL
3	Marina	Grootland	1	1	Opel	Astra 1.6	handmatig	6
...

Optionele velden

Nu een JOIN:

```
SELECT * FROM personeel
```

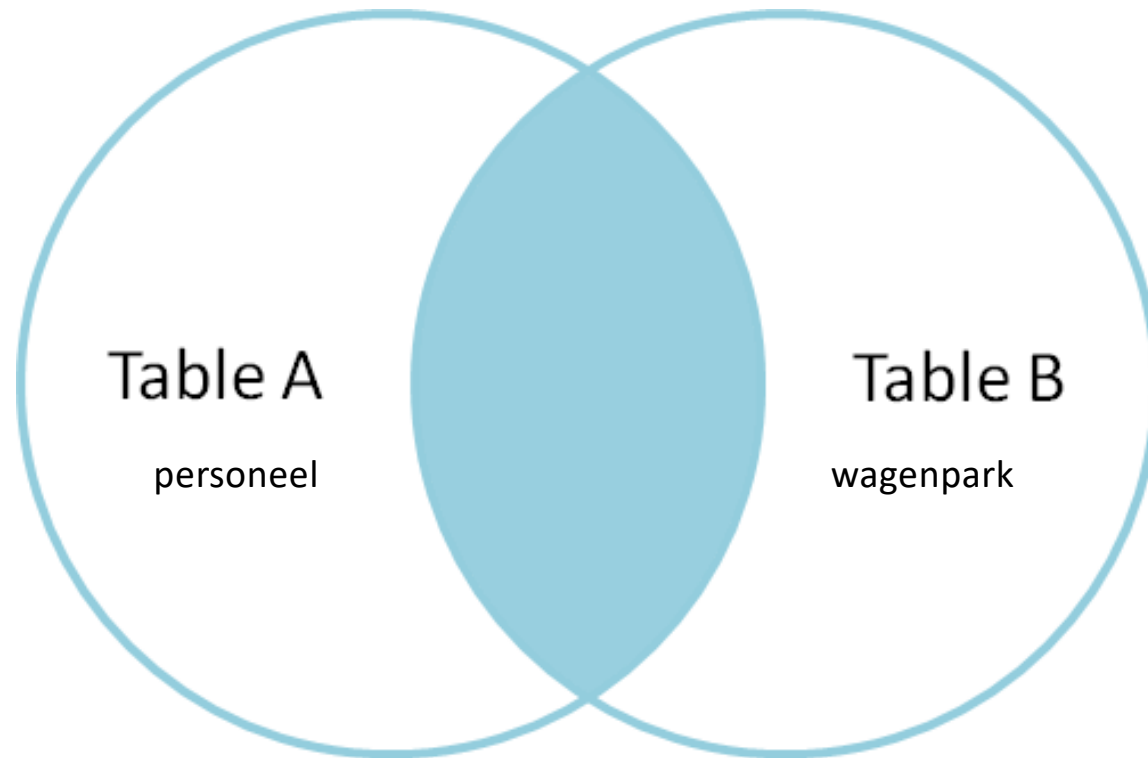
```
JOIN wagenpark ON personeel.leasewagen = wagenpark.id
```



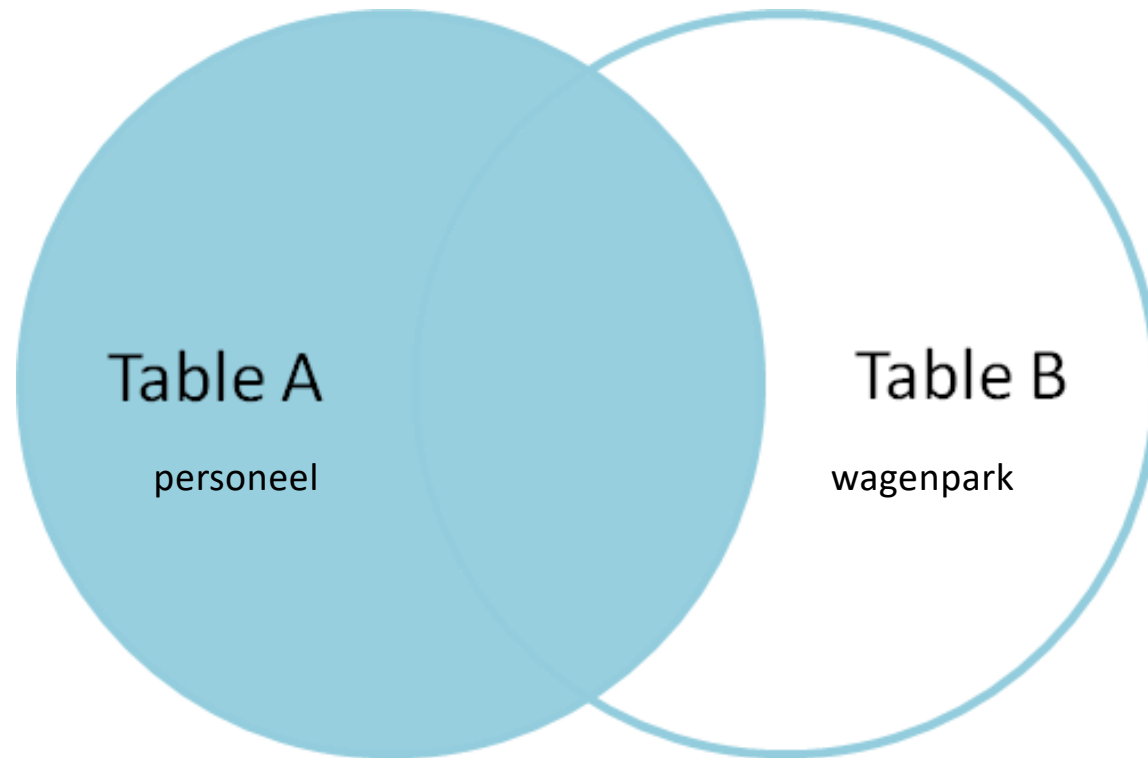
Rijen die niet uitgebreid kunnen worden, worden gefilterd bij een (inner) JOIN

id	voornaam	achternaam	leasewagen	id	merk	type	transmissie	aantal versnellingen
1	Suus	Pletter	2	2	Opel	Zafira 1.8	handmatig	5
3	Marina	Grootland	1	1	Opel	Astra 1.6	handmatig	6
...

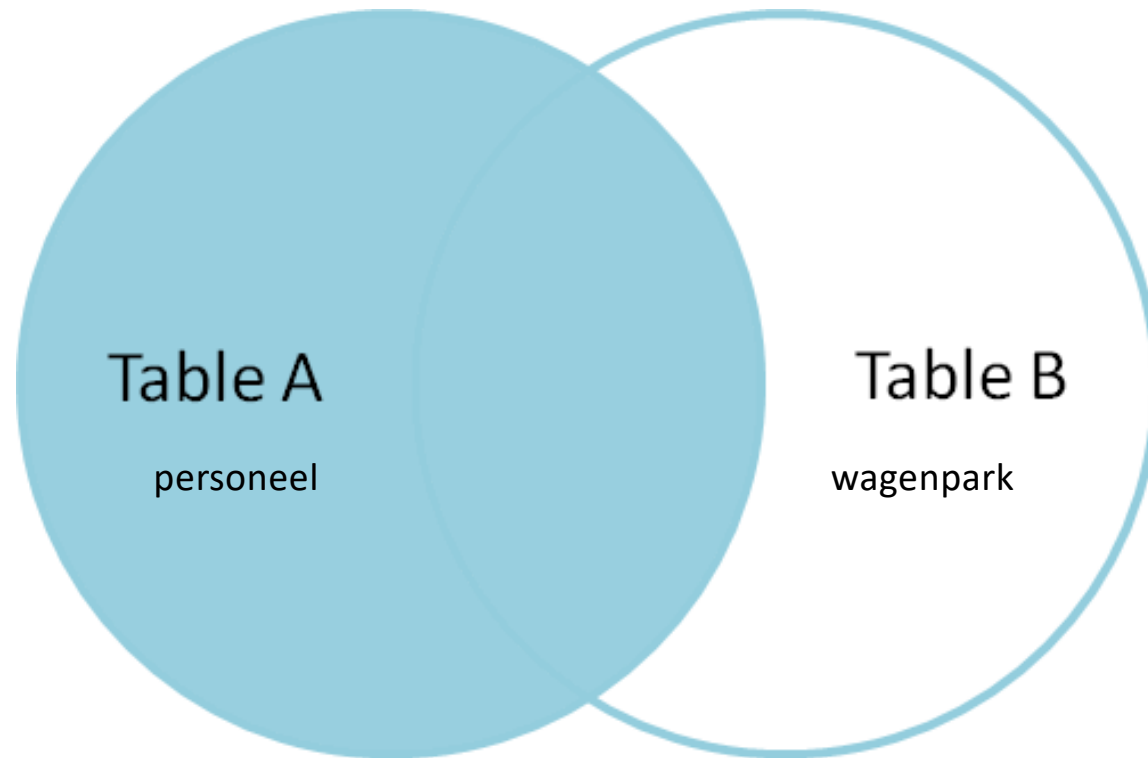
Schematisch weergegeven (INNER) JOIN



Maar je wilt dit

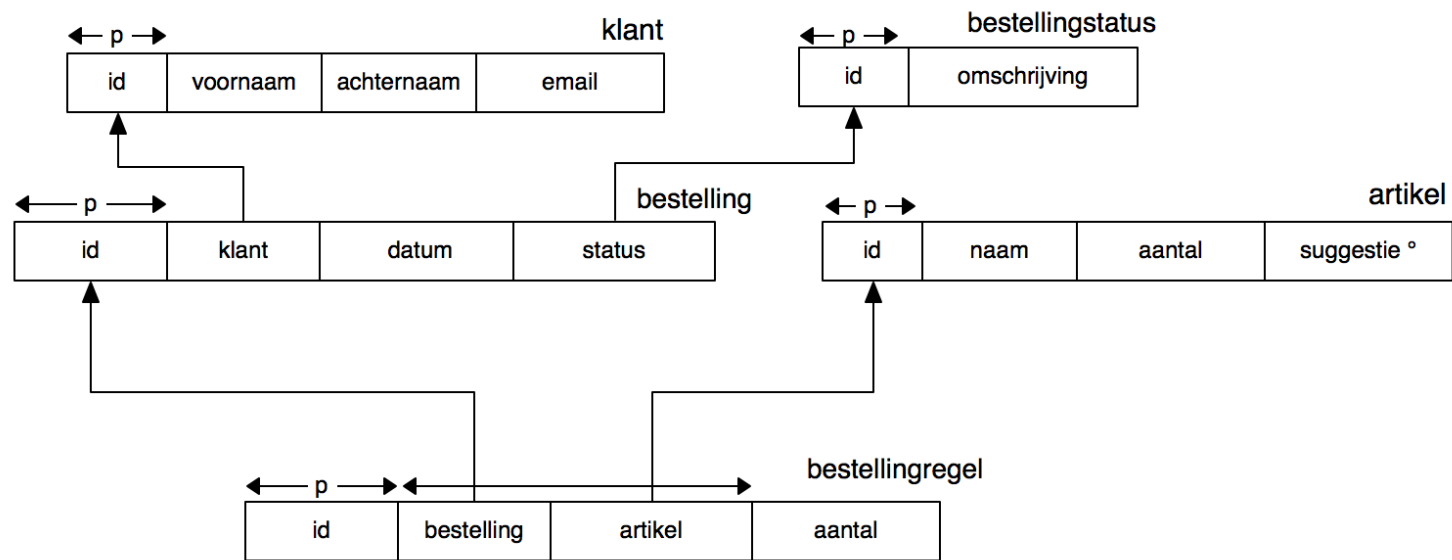


LEFT OUTER JOIN

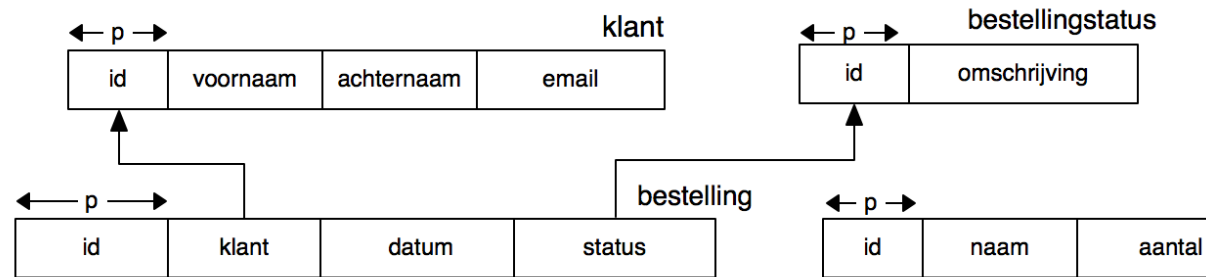


Voorbeeld

Ik wil zien hoe vaak de 4 verschillende statuses voorkomen



Voorbeeld

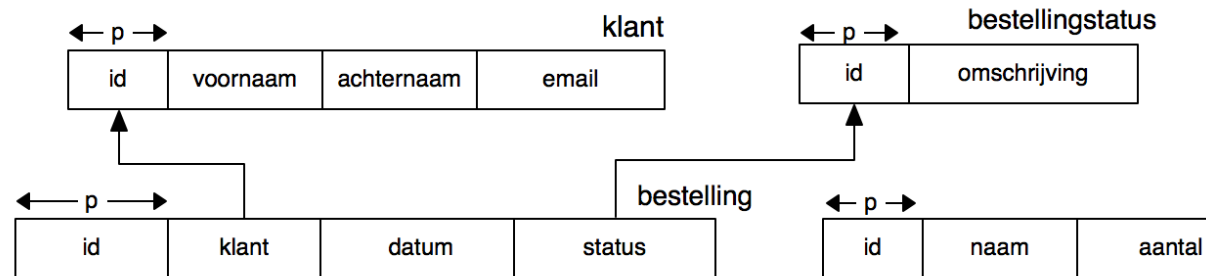


Het begin

SELECT * FROM ...

- A) bestelling
- B) bestellingstatus
- C) maakt niet uit

Voorbeeld

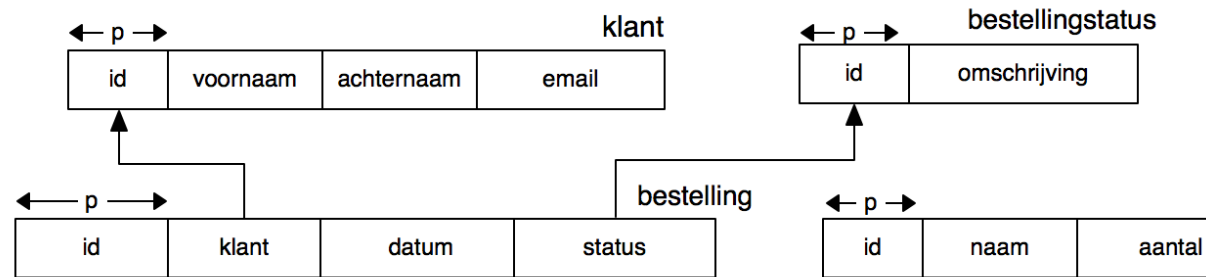


Het begin

```
SELECT * FROM bestellingstatus
```

Ook als een status niet voorkomt, wil je 'm wel in de uitvoertabel zien. Dat is een OUTER JOIN. Aangezien er alleen een LEFT OUTER JOIN is, moet bestellingstatus dus aan de linkerkant van de JOIN staan.

Voorbeeld

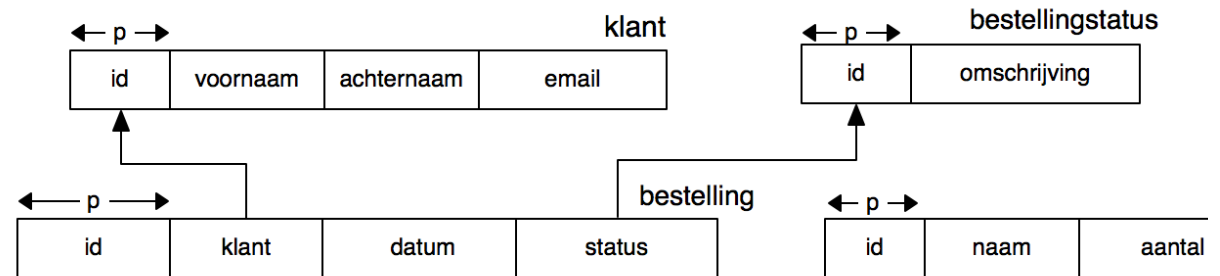


Het begin

```
SELECT * FROM bestellingstatus
JOIN bestelling ON ...
```

- A) bestelling.id = bestellingstatus.id
- B) bestelling.status = bestellingstatus.id
- C) bestellingstatus.id = bestelling.id
- D) bestelling.id = bestelling.status

Voorbeeld

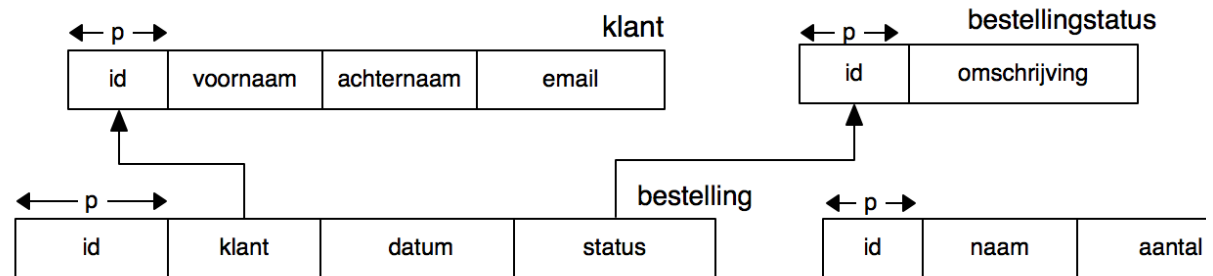


Het begin

```
SELECT * FROM bestellingstatus
```

```
JOIN bestelling ON bestellingstatus.id = bestelling.status
```

Voorbeeld

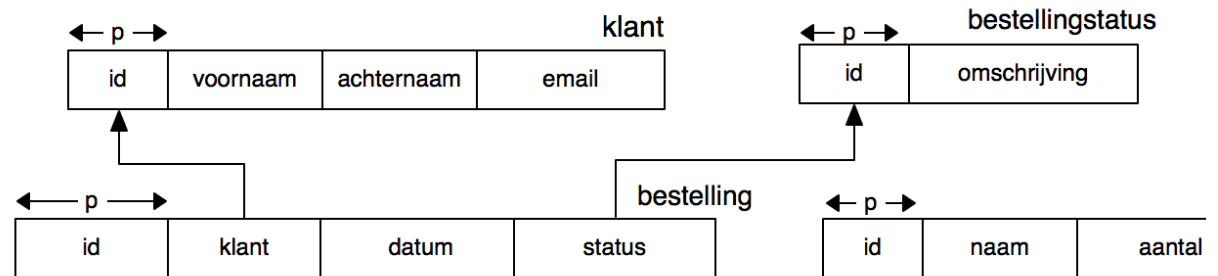


Het begin

```
SELECT omschrijving, count(*) FROM bestellingstatus
JOIN bestelling ON bestellingstatus.id = bestelling.status
GROUP BY ...
```

- A) bestelling.id
- B) bestelling.status
- C) bestellingstatus.id
- D) bestellingstatus.omschrijving

Voorbeeld



Het begin

```
SELECT omschrijving, count(*) FROM bestellingstatus
JOIN bestelling ON bestellingstatus.id = bestelling.status
GROUP BY bestellingstatus.id
```

Multipliciteitendiagram en strokendiagram

Database ontwerpen

Menselijke tekst

- Nederlands (of Engels)



Multipliciteiten- diagram

- Tabellen
- Relaties tussen tabellen



Stroken-diagram

- Tabellen
- Velden



SQL

- Tabellen met velden en datatype

Lesdoelen

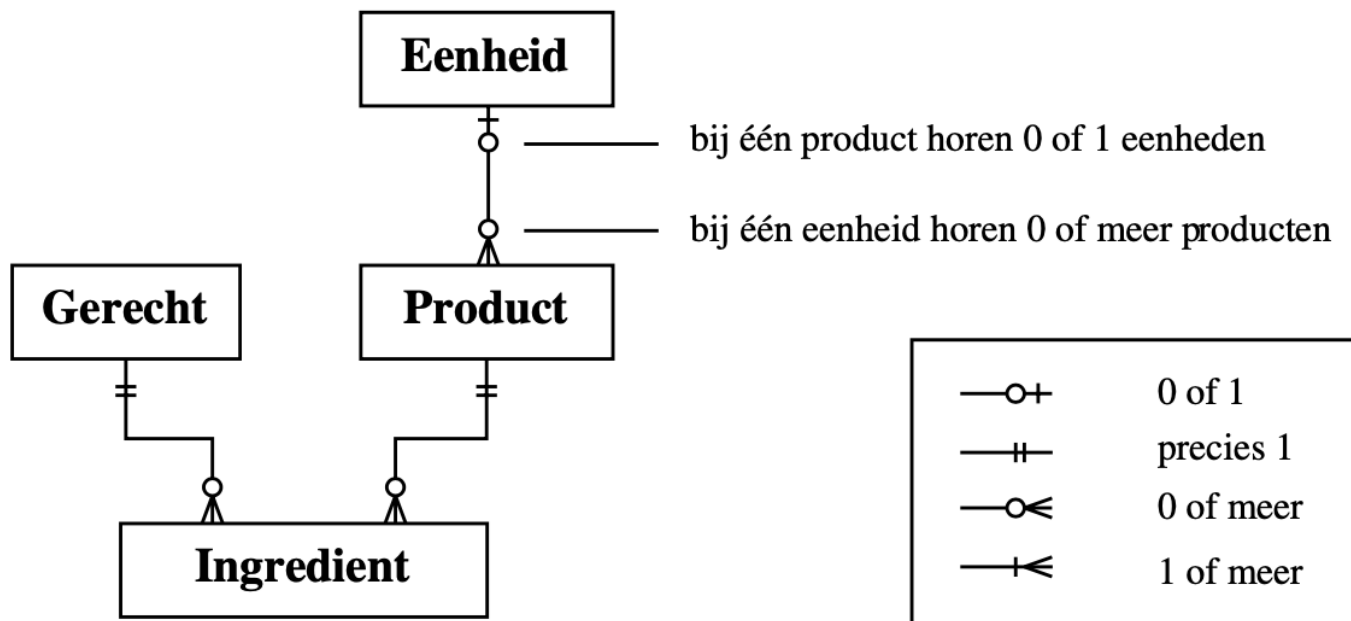
- Je kunt uitleggen wat een multipliciteitendiagram is.
- Je kunt een multipliciteitendiagram opstellen op basis van een korte beschrijving van gegevens die je in een database wilt opslaan
- Je kunt een multipliciteitendiagram gebruiken om, samen met aanvullende informatie over gegevensvelden, een strokendiagram op te stellen

Kortom: je kunt de volgende stappen doorlopen:

1. Beschrijving van gegevens
2. multipliciteitendiagram (relaties vastleggen)
3. strokendiagram (tabellen en velden vastleggen)
4. SQL (database tabellen maken en vullen)

Multipliciteitendiagram

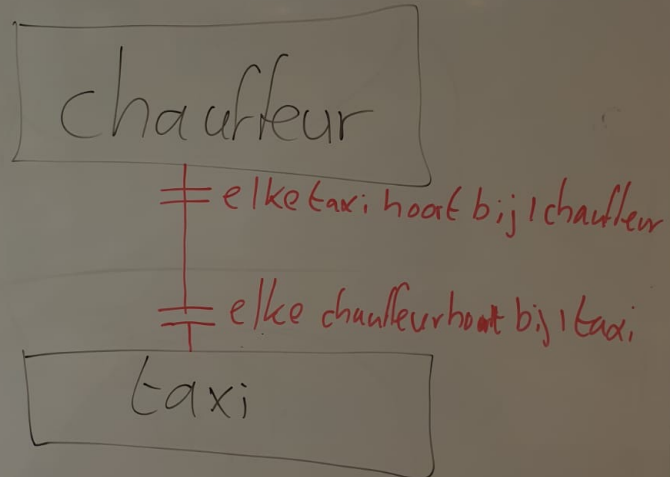
- In een multipliciteitendiagram leg je de relatie tussen gegevens vast.



FIGUUR 12.9 Multipliciteitendiagram van Reijnders' Toetjesboek

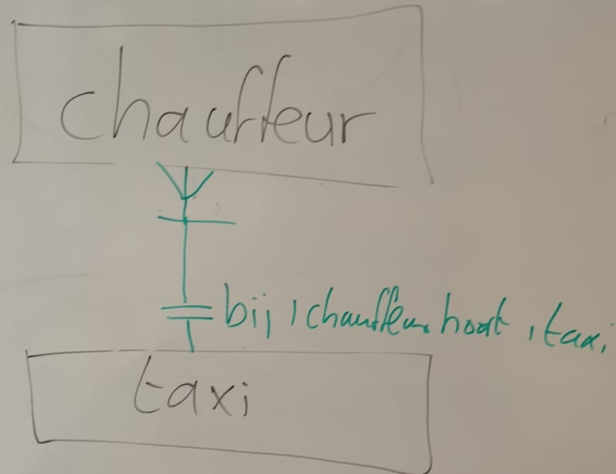
Voorbeeld

Taxibedrijf. DB met rooster
chauffeurs op taxis. Elke chauffeur
heeft zijn/haar eigen taxi.



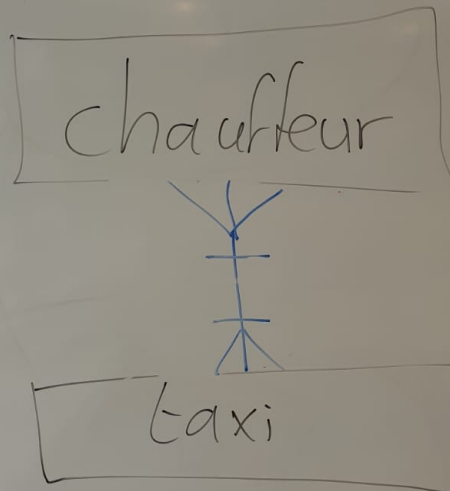
Voorbeeld

Taxibedrijf. DB met raster
chauffeurs op taxis. Elke
chauffeur rijdt steed in dezelfde
taxi



Voorbeeld

Taxibedrijf. DB met raster
chauffeurs op taxis. Elke
chauffeur pakt een willekeurige
taxi



Strokendiagrammen

- Strokendiagram: tabelkoppen + extra informatie

Leerlingen

← p →

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
----------------	-------------	---------------	------------	-------

Personeel

← p →

afkorting	voorletters	tussenvoegsel	achternaam	email
-----------	-------------	---------------	------------	-------

Vakken

← p →

afkorting	naam
-----------	------

Vraag: welk veld zou een extra sleutel kunnen zijn?

Leerlingen

← p →

leerlingnummer	voorletters	tussenvoegsel	achternaam	email
----------------	-------------	---------------	------------	-------

Personeel

← p →

afkorting	voorletters	tussenvoegsel	achternaam	email
-----------	-------------	---------------	------------	-------

Vakken

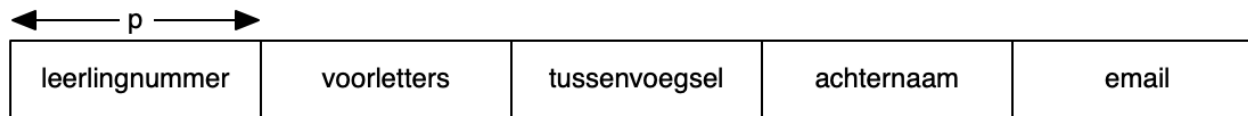
← p →

afkorting	naam
-----------	------

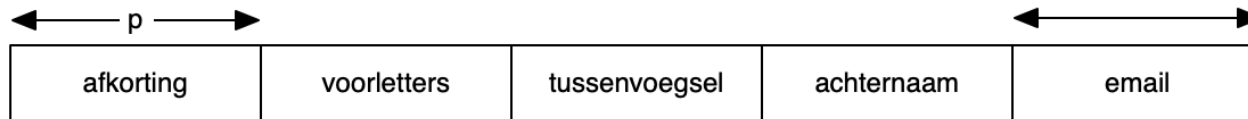
Strokendiagrammen

- Extra sleutelvelden

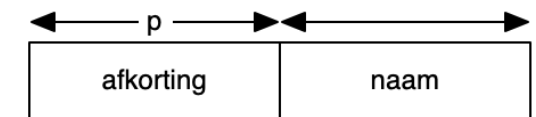
Leerlingen



Personeel

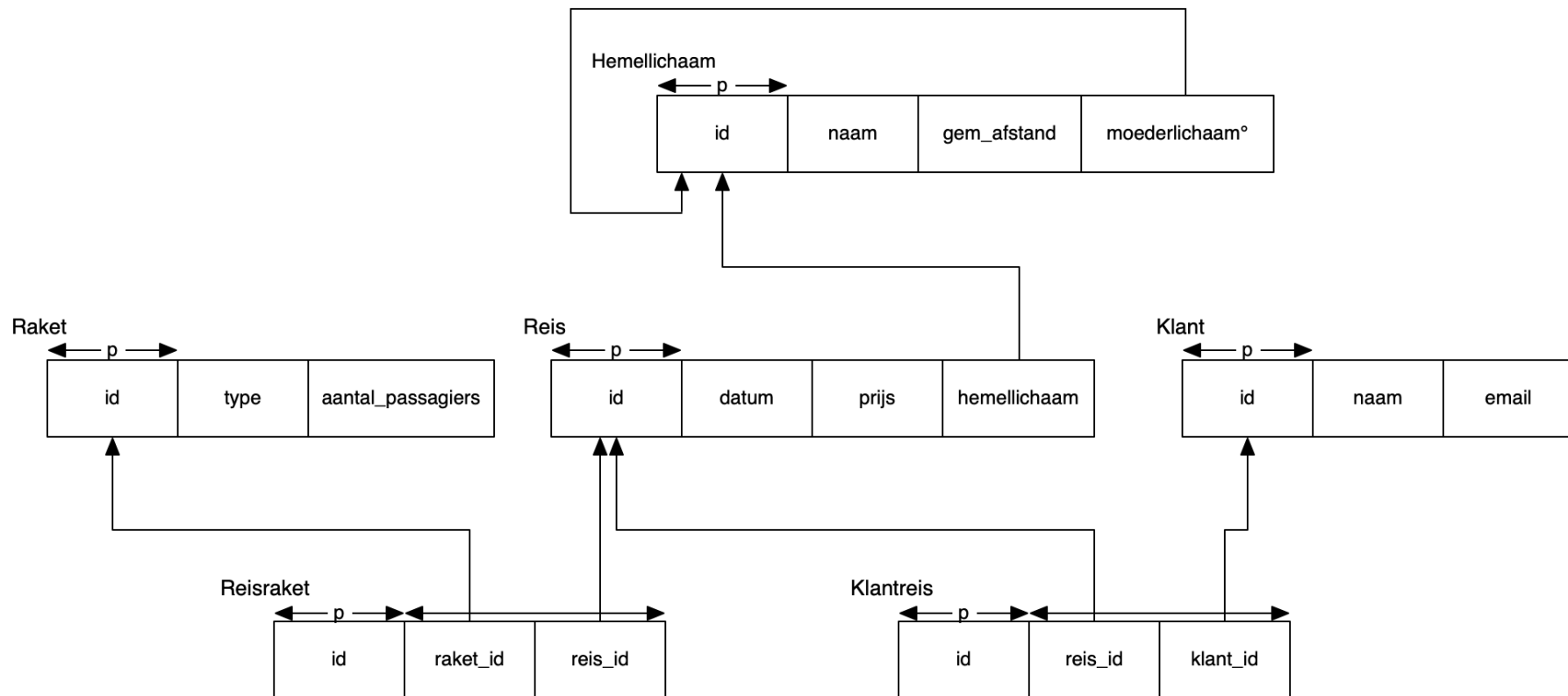


Vakken

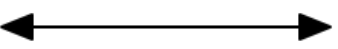
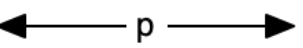


Strokendiagram van Ruimtereisbureau

Hoe zag het multipliciteitendiagram eruit?



Strokendiagrammen

- Iedere tabel heeft één of meerdere 'sleutels'. Wordt in strokendiagram aangegeven met  boven veld.
- Met een sleutel kun je **gegarandeerd** een specifiek record vinden.
- Een sleutel is per definitie dus ook **uniek**. Unieke waarden zijn automatisch sleutels.
- Primary key: de hoofdsleutel. Verplicht in iedere tabel. Aangegeven met  boven een veld. (Khan gebruik afkorting 'PK')
- Database management systeem (DBMS) gebruik primary keys om indexbestanden aan te maken waardoor het extra snel werkt.