

5.2 Groepsfuncties



```
SELECT [DISTINCT] select_expressie  
FROM tabel_expressie  
[WHERE conditie]  
[GROUP BY expressie] [HAVING conditie]  
[ORDER BY {expressie} [ASC | DESC]]
```

5.2.1 Group by-component

- Regelmatig geïnteresseerd in **geaggregeerde informatie** (verzameling van rijen i.p.v. afzonderlijke rijen)
- Voorbeeld: **Aantal medewerkers per afdeling**

```
SELECT      afd      "afdeling"  
,          COUNT(mnr) "aantal_medewerkers"  
FROM        medewerkers  
GROUP BY    afd
```

AFDELING	AANTAL_MEDEWERKERS
10	3
20	5
30	6

AFD	NAAM
10	CLERCKX
10	DE KONING
10	WOUTERS
20	JACOBS
20	DE COOMAN
20	SLECHTEN
20	CASPERS
20	SWINNEN
30	DEFOUR
30	DEN RUYTER
30	ALLARD
30	JACOBS
30	BRIERS
30	MARTENS

Let op! Enkel in SELECT de kolom van de GROUP BY opnemen

5.2.1 Group by-component

- Voorbeeld: **Aantal medewerkers per geboortejaar**

```
col geboortejaar format a20;  
SELECT  
to_char (gbdatum, 'YYYY') "geboortejaar",  
count(mnr) "aantalMedewerkersInJaar"  
FROM medewerkers  
GROUP BY to_char(gbdatum, 'YYYY')
```

GEBOORTEJAAR	AANTALMEDEWERKERSINJAAR
1987	1
1976	1
1983	1
1982	2
1986	1
1985	2
1979	2
1989	1
1988	1
1972	1
1981	1
11 rows selected.	

5.2.1 Group by-component

- **Vb. Geef het aantal cursisten per cursus per begindatum**
(Ook toepasbaar op meerdere kolommen)

```
SELECT      i.cursus, i.begindatum
,           COUNT (i.cursist)
FROM        inschrijvingen i
GROUP BY    i.cursus, i.begindatum
```

CURS	BEGINDATU	COUNT(I.CURSIST)
---	-----	-----
WEB	17-DEC-15	5
WIN	04-FEB-16	2
SQL	08-OCT-15	3
SQL	17-DEC-15	2
ORG	10-AUG-15	3
...		

ORG	10-08-2015
ORG	10-08-2015
ORG	10-08-2015
ORG	27-09-2016
SQL	16-04-2015
SQL	16-04-2015
SQL	16-04-2015
SQL	16-04-2015
SQL	08-10-2015
SQL	08-10-2015
SQL	08-10-2015
SQL	17-12-2015
SQL	17-12-2015

WEB	17-12-2015
WEB	17-12-2015
WEB	17-12-2015
WEB	17-12-2015
WEB	17-12-2015

WEB	05-02-2016
WEB	05-02-2016
WEB	05-02-2016

WIN	04-02-2016
WIN	04-02-2016

5.2.2 Groepsfuncties

- COUNT () geeft aantal waarden alle datatypes
 - ▣ *Select count(mnr) from medewerkers; → 14*

- SUM () som van de waarden numeriek
 - ▣ *Select sum(mnr) from medewerkers; → 108172*

- AVG () gemiddelde waarden numeriek
 - ▣ *Select avg(mnr) from medewerkers; → 7726,57143 (=108172 / 14)*

5.2.2 Groepsfuncties

- MIN () minimumwaarde alle datatypes
 - ▣ *Select min(m.gbdatum) from medewerkers m; → 17/11/1972*

- MAX () maximumwaarde alle datatypes
 - ▣ *Select max(m.gbdatum) from medewerkers m; → 03/12/1989*

- STDEV () standaarddeviatie numeriek ⁽¹⁾

- VARIANCE () variantie numeriek ⁽²⁾

(1) De standaarddeviatie of standaardafwijking geeft de mate van spreiding aan in bepaalde data.
Het geeft aan hoezeer de geobserveerde waardes afwijken van het gemiddelde.

(2) De variantie is in de statistiek een maat voor de spreiding van een reeks waarden,
dat wil zeggen de mate waarin de waarden onderling verschillen.
Hoe groter de variantie, hoe meer de afzonderlijke waarden onderling verschillen,
en dus ook hoe meer de waarden van het "gemiddelde" afwijken.

5.2.2 Groepsfuncties

□ Hoeveel bedraagt het laagste maandsalaris?

- `SELECT MIN(maandsal) → 1600`
`FROM medewerkers`

□ Wie heeft het laagste salaris?

- `SELECT naam`
`FROM medewerkers`
`WHERE maandsal = (SELECT MIN(maandsal)`
`FROM medewerkers)`

of

- `SELECT m.naam, m.maandsal`
`FROM medewerkers m`
`INNER JOIN (SELECT MIN(maandsal) laagste FROM medewerkers) mm`
`ON m.maandsal = mm.laagste;`

NAAM	MAANDSAL	MIN(MAANDSAL)
-----	-----	-----
CASPERS	1800	1600
ALLARD	1600	
DEFOUR	2250	
JACOBS	4975	
MARTENS	2250	
BRIERS	5850	
CLERCKX	3450	
SWINNEN	4000	
DE KONING	7000	
DEN RUYTER	2500	
SLECHTEN	2700	
JACOBS	2800	
DE COOMAN	4000	
WOUTERS	2300	

NAAM

ALLARD

5.2.2 Groepsfuncties

- Hoeveel medewerkers telt de onderneming en in hoeveel verschillende afdelingen werken ze?

- ▣ `SELECT COUNT(mnr), COUNT(distinct afd)`
`FROM medewerkers`

<code>COUNT(MNR)</code>	<code>COUNT(DISTINCT AFD)</code>
-----	-----
14	3

- COUNT-functie accepteert naast kolomnamen ook een asterisk(*) → telt niet het aantal waarden, maar aantal rijen
→ werkt in grotere databanken trager → best primary key gebruiken

5.2.2 Groepsfuncties

□ Hoeveel verschillende functies zijn er in elke afdeling?

□ SELECT afd, COUNT(funcitie), COUNT(distinct functie)
FROM medewerkers
GROUP BY afd
ORDER BY afd

AFD	AANTAL_FUNCTIES	VERSCHILLEND_AANTAL
10	3	3
20	5	2
30	6	3

AFD	FUNCTIE
10	BOEKHOUDER
10	DIRECTEUR
10	MANAGER
20	MANAGER
20	TRAINER
20	TRAINER
20	TRAINER
20	TRAINER
30	BOEKHOUDER
30	MANAGER
30	VERKOPER
30	VERKOPER
30	VERKOPER
30	VERKOPER

5.2.2 Groepsfuncties

- Hoeveel personen ontvangen een commissie en hoeveel bedraagt het gemiddelde?
 - ▣ `SELECT SUM(comm), COUNT(comm), AVG(comm), AVG(nvl(comm,0))`
`FROM medewerkers;`

`SUM(COMM) COUNT(COMM) AVG(COMM) AVG(NVL(COMM,0))`

11400

4

2850

814,285714

NAAM	COMM
-----	-----
CASPERS	
ALLARD	3000
DEFOR	5000
JACOBS	
MARTENS	3400
BRIERS	
CLERCKX	
SWINNEN	
DE KONING	
DEN RUYTER	0
SLECHTEN	
JACOBS	
DE COOMAN	
WOUTERS	

Gemiddelde voor
iedereen die een
commissie heeft
(11400 / 4)

Gemiddelde voor
iedereen
(11400 / 14)

5.2.3 De Having-component

- WHERE-component: restrictie-operator voor de rijen
- HAVING-component: restricties op groepsniveau leggen

- **In welke afdeling zijn er meer dan 4 medewerkers?**

```
□ SELECT      afd, COUNT(mnr)
   FROM      medewerkers
   GROUP BY   afd
   HAVING     COUNT(mnr) > 4
```

AFD	COUNT(MNR)
30	6
20	5

5.2.3 De Having-component

- Welke functies in afdeling 10 of 20 hebben een totaal inkomen groter dan 5000?

SELECT functie, SUM(maandsal)
FROM medewerkers
WHERE afd in (10,20)
GROUP BY functie
HAVING SUM(maandsal) > 5000
ORDER BY SUM (maandsal)

NAAM	FUNCTIE	MAANDSAL	AFD
CLERCKX	MANAGER	3450	10
DE KONING	DIRECTEUR	7000	10
WOUTERS	BOEKHOUDER	2300	10
JACOBS	MANAGER	4975	20
DE COOMAN	TRAINER	4000	20
SLECHTEN	TRAINER	2700	20
CASPERS	TRAINER	1800	20
SWINNEN	TRAINER	4000	20
DEFOUR	VERKOPER	2250	30
DEN RUYTER	VERKOPER	2500	30
ALLARD	VERKOPER	1600	30
JACOBS	BOEKHOUDER	2800	30
BRIERS	MANAGER	5850	30
MARTENS	VERKOPER	2250	30

FUNCTIE	SUM(MAANDSAL)
-----	-----
DIRECTEUR	7000
MANAGER	8425
TRAINER	12500

5.2.4 BREAK en COMPUTE commando

SQL> select afd, functie, mnr, naam
from medewerkers
order by afd, functie;

AFD	FUNCTIE	MNR	NAAM
10	BOEKHOUDER	7934	WOUTERS
10	DIRECTEUR	7839	DE KONING
10	MANAGER	7782	CLERCKX
20	MANAGER	7566	JACOBS
20	TRAINER	7369	CASPERS
20	TRAINER	7902	DE COOMAN
20	TRAINER	7876	SLECHTEN
20	TRAINER	7788	SWINNEN
30	BOEKHOUDER	7900	JACOBS
30	MANAGER	7698	BRIERS
30	VERKOPER	7521	DEFOUR
30	VERKOPER	7844	DEN RUYTER
30	VERKOPER	7499	ALLARD
30	VERKOPER	7654	MARTENS

□ SQL> break on afd

AFD	FUNCTIE	MNR	NAAM
10	BOEKHOUDER	7934	WOUTERS
	DIRECTEUR	7839	DE KONING
	MANAGER	7782	CLERCKX
20	MANAGER	7566	JACOBS
	TRAINER	7369	CASPERS
	TRAINER	7902	DE COOMAN
	TRAINER	7876	SLECHTEN
	TRAINER	7788	SWINNEN
30	BOEKHOUDER	7900	JACOBS
	MANAGER	7698	BRIERS
	VERKOPER	7521	DEFOUR
	VERKOPER	7844	DEN RUYTER
	VERKOPER	7499	ALLARD
	VERKOPER	7654	MARTENS

5.2.4 BREAK en COMPUTE commando

SQL> select afd, functie, mnr, naam, maandsal
from medewerkers
order by afd, functie;

AFD	FUNCTIE	MNR	NAAM	MAANDSAL
10	BOEKHOUDER	7934	WOUTERS	2300
	DIRECTEUR	7839	DE KONING	7000
	MANAGER	7782	CLERCKX	3450
*****				-----
TOTAAL				12750
20	MANAGER	7566	JACOBS	4975
	TRAINER	7369	CASPERS	1800
	TRAINER	7902	DE COOMAN	4000
	TRAINER	7876	SLECHTEN	2700
	TRAINER	7788	SWINNEN	4000
*****				-----
TOTAAL				17475
30	BOEKHOUDER	7900	JACOBS	2800
	MANAGER	7698	BRIERS	5850
	VERKOPER	7521	DEFOUR	2250
	VERKOPER	7844	DEN RUYTER	2500
	VERKOPER	7499	ALLARD	1600
	VERKOPER	7654	MARTENS	2250
*****				-----
TOTAAL				17250

SQL> break on afd skip 1

SQL> COMPUTE sum LABEL totaal of maandsal on afd

! COMPUTE altijd samen met BREAK ON

5.2.4 BREAK en COMPUTE commando

SQL> select afd, functie, mnr, naam, maandsal
from medewerkers
order by afd, functie;

AFD	FUNCTIE	MNR	NAAM	MAANDSAL
10	BOEKHOUDER	7934	WOUTERS	2300
	DIRECTEUR	7839	DE KONING	7000
	MANAGER	7782	CLERCKX	3450

AANTAL MED		3		
TOTAAL				12750
20	MANAGER	7566	JACOBS	4975
	TRAINER	7369	CASPERS	1800
	TRAINER	7902	DE COOMAN	4000
	TRAINER	7876	SLECHTEN	2700
	TRAINER	7788	SWINNEN	4000

AANTAL MED		5		
TOTAAL				17475
30	BOEKHOUDER	7900	JACOBS	2800
	MANAGER	7698	BRIERS	5850
	VERKOPER	7521	DEFOUR	2250
	VERKOPER	7844	DEN RUYTER	2500
	VERKOPER	7499	ALLARD	1600
	VERKOPER	7654	MARTENS	2250

AANTAL MED		6		
TOTAAL				17250

SQL> break on afd skip 1

SQL> COMPUTE number label "AANTAL MEDEWERKERS" of functie on afd

5.2.4 BREAK en COMPUTE commando

- De toegestane functies van COMPUTE zijn:

AVG	gemiddelde
COUNT	het aantal not null-waarden in een kolom
MAX	maximum
MIN	minimum
NUMBER	aantal rijen
STD	standaardafwijking
SUM	som
VAR	variantie

- Compute-instellingen verwijderen.

SQL> Clear Computes

5.2.4 BREAK en COMPUTE commando

- Met behulp van de **ROLLUP**-operator kan je per groep ook een overzicht krijgen.

```
SQL> break on afd skip 1
```

```
SQL> select  afd, functie  
            , count (mnr) aantal  
from    medewerkers  
group by ROLLUP(afd, functie);
```

AFD	FUNCTIE	AANTAL
10	MANAGER	1
	DIRECTEUR	1
	BOEKHOUDER	1
		3
20	MANAGER	1
	TRAINER	4
		5
30	MANAGER	1
	VERKOPER	4
	BOEKHOUDER	1
		6
		14

Valkuilen

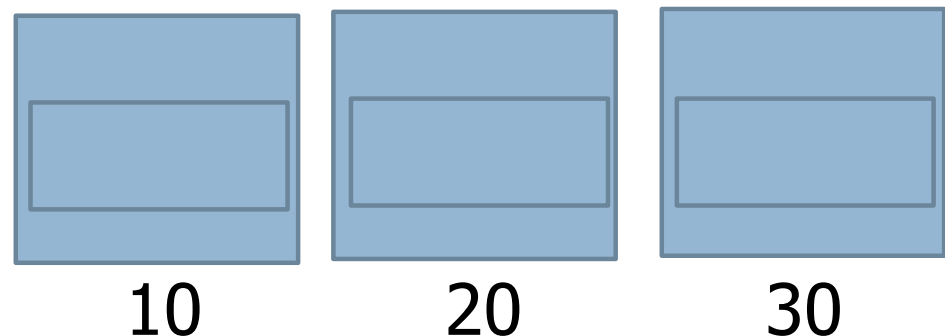
Zoek de veel voorkomende fouten en verbeter.

- ❑ `select afd, count(mnr)`
`from medewerkers;`

group by afd vergeten!

- ❑ `select m.afd, a.naam, count(mnr)`
`from medewerkers m`
`inner join afdelingen a`
`on m.afd = a.anr`
`group by m.afd, a.naam;`

group by m.afd,a.naam



Valkuilen

Zoek de veel voorkomende fouten en verbeter.

- `select a.anr, count(*)`
`from medewerkers m`
`right outer join afdelingen a`
`on m.afd = a.anr`
`group by a.anr;`

ANR	COUNT(*)
10	3
20	5
30	6
40	1

Geeft voor afdeling 40 ook 1!! Dit is niet correct !
Count(*) telt het aantal RIJEN!!!

`select a.anr, m.mnr`
`from medewerkers m`
`right outer join afdelingen a`
`on m.afd = a.anr`

ANR	MNR
20	7369
30	7499
30	7521
20	7566
30	7654
30	7698
10	7782
20	7788
10	7839
30	7844
20	7876
30	7900
20	7902
10	7934
40	

- `select a.anr, count(mnr)`
`from medewerkers m`
`right outer join afdelingen a`
`on m.afd = a.anr`
`group by a.anr;`

ANR	COUNT(MNR)
10	3
20	5
30	6
40	0

Geeft voor afdeling 40 0 Dit is correct !
Count(mnr) telt het aantal ingevulde velden