

Oefeningen op hoofdstuk 5

GROUP BY HAVING

SYNTAX:

SELECT [DISTINCT] select_list
FROM table_name[,table_name,select]
WHERE rijcondities
GROUP BY groepskenmerken
HAVING conditie
ORDER BY kolomnamen

1. Geef per project het aantal medewerkers dat zich met dat project bezighoudt.

PROJ_NR	AANTAL
1	3
30	3
2	3
20	3
3	2
10	3

6 rows selected
select proj_nr, count(*) aantal from opdrachten
group by proj_nr;

2. Ga na hoeveel medewerkers er aan elk project werken en bereken van die aantallen het gemiddelde.

gem. aantal med/project

3

select round(avg(count(*))) "gem.aantal med/project" from
opdrachten group by proj_nr;

3. Geef per afdeling het aantal medewerkers dat uit de provincie Limburg (LI) komt .

AFD_NR	aantal medewerkers
3	2

select afd_nr, count(*) "aantal medewerkers" from
medewerkers

```
where upper(provincie)='LI'
group by afd_nr ;
```

4. Geef per manager het aantal ondergeschikten.

MGR_SOFI_NR	AANTAL
999444444	3
999666666	2
999555555	2

```
select mgr_sofi_nr, count(*) aantal from
medewerkers where mgr_sofi_nr is not null
group by mgr_sofi_nr;
```

5. Hoeveel projecten ondersteunt een afdeling per locatie.

AFD_NR	LOCATIE	aantal projecten
7	Eindhoven	1
1	Eindhoven	1
7	Oegstgeest	1
7	Groningen	1
3	Maastricht	2

```
select afd_nr, locatie, count(*) "aantal projecten"
from projecten
group by afd_nr,locatie;
```

6.

Geef weer hoeveel zonen en hoeveel dochters een medewerker heeft. Los op in 1 instructie.

SOFI_NR	RELATIE	AANTAL
999444444	DOCHTER	1
999444444	ZOON	1
999111111	ZOON	1
999111111	DOCHTER	1

```
select sofi_nr, relatie, count(*) aantal
from gezinsleden
where upper(relatie) in ('DOCHTER','ZOON')
group by sofi_nr, relatie;
```

7. Geef per afdeling het aantal vrouwelijke medewerkers dat minder verdient dan 33000 én een parkeerplaats heeft.

AFD_NR	AANTAL
3	1
7	1

```
select afd_nr,count(*) aantal from medewerkers
where lower(geslacht)='v'
and salaris <33000
and PARKEERPLAATS is not null
group by afd_nr
order by 1;
```

8. Welke van beide oplossingen is het meest performant.

Oplossing 1:

```
SELECT locatie, COUNT(proj_nr) aantal_projecten
FROM projecten
WHERE UPPER(locatie) IN ('EINDHOVEN','OEGSTGEEST')
GROUP BY locatie;
```

Oplossing 2:

```
SELECT locatie, COUNT(proj_nr) aantal_projecten
FROM projecten
GROUP BY locatie
HAVING UPPER(locatie) IN ('EINDHOVEN','OEGSTGEEST');
```

The theory (by theory I mean [SQL Standard](#)) says that WHERE restricts the result set before returning rows and HAVING restricts the result set after bringing all the rows. So WHERE is faster. On SQL Standard compliant DBMSs in this regard, only use HAVING where you cannot put the condition on a WHERE (like computed columns in some RDBMSs.)

You can just see the execution plan for both and check for yourself, nothing will beat that (measurement for your specific query in your specific environment with your data.)