

SQL

GROUP BY

Luc Vervoort

GROUP BY clause

-- VOORBEELD:

```
SELECT sum(salary)
FROM employees;
```

```
SELECT sum(salary)
FROM employees
WHERE department_id = 60;
```

```
SELECT sum(salary)
FROM employees
GROUP BY department_id;
```

```
SELECT department_id , sum(salary)
FROM employees
GROUP BY department_id;
```

GROUP BY: OPDRACHT 1

- maak een resultatenlijst met
 - het GEMIDDELDE loon per job-functie (job_id)
 - met alias voor de aggregerende functie
-
- 12 rows returned

job_id	gemiddeld loon
AC_ACCOUNT	8300.000000
AC_MGR	12000.000000
AD_ASST	4400.000000
AD PRES	24000.000000
AD_VP	17000.000000
IT_PROG	6400.000000
MK_MAN	13000.000000
MK_REP	6000.000000
SA_MAN	10500.000000
SA_REP	8866.666667
ST_CLERK	2925.000000
ST_MAN	5800.000000

GROUP BY: OPLOSSING 1

-- IS DIT EEN GOEDE OPLOSSING?

```
SELECT department_id, avg(salary)
FROM employees
GROUP BY job_id;
```

-- Dit genereert een **MISLEIDENDE** resultatenlijst !

-- Waarom is dit misleidend?

-- SQL-syntaxregel:

-- In combinatie met een aggregerende functie in de SELECT-clausule

-- is het NIET TOEGELATEN een kolomnaam op te nemen waarop niet gegroepeerd wordt !!!

-- MySQL zou dus eigenlijk een foutmelding (syntax-fout) moeten geven, maar doet het niet.

-- Oracle DB geeft wel een foutmelding.

GROUP BY: OPDRACHT 2

- maak een resultatenlijst met
- het LAAGSTE en het HOOGSTE loon
- binnen elke job-functie binnen elk departement
- 13 rows returned

	department_id	job_id	min_salary	max_salary
▶	NULL	SA_REP	6000	12000
	10	AD_ASST	3000	6000
	20	MK_MAN	9000	15000
	20	MK_REP	4000	9000
	50	ST_CLERK	2000	5000
	50	ST_MAN	5500	8500
	60	IT_PROG	4000	10000
	80	SA_MAN	10000	20000
	80	SA_REP	6000	12000
	90	AD PRES	20000	40000
	90	AD_VP	15000	30000
	110	AC_ACC...	4200	9000
	110	AC_MGR	8200	16000

GROUP BY: OPDRACHT 3

- maak een resultatenlijst met
- het AANTAL medewerkers dat is aangeworven
- per datum van aanwerving
- 19 rows returned

1987-06-17	1
1987-09-17	1
1989-09-21	1
1990-01-03	1
1991-05-21	1
1993-01-13	1
1994-06-07	2
1995-10-17	1
1996-02-17	1
1996-05-11	1
1997-01-29	1
1997-08-17	1
1998-03-15	1
1998-03-24	1
1998-07-09	1
1999-02-07	1
1999-05-24	1
1999-11-16	1
2000-01-29	1

GROUP BY: OPDRACHT 4

- maak een resultatenlijst met
- het AANTAL medewerkers dat is aangeworven
- per job-functie, per datum van aanwerving
- 20 rows returned

AC_ACCOUNT	1994-06-07	1
AC_MGR	1994-06-07	1
AD_ASST	1987-09-17	1
AD PRES	1987-06-17	1
AD_VP	1989-09-21	1
AD_VP	1993-01-13	1
IT_PROG	1990-01-03	1
IT_PROG	1991-05-21	1
IT_PROG	1999-02-07	1
MK_MAN	1996-02-17	1
MK_REP	1997-08-17	1
SA_MAN	2000-01-29	1
SA_REP	1996-05-11	1
SA_REP	1998-03-24	1
SA_REP	1999-05-24	1
ST_CLERK	1995-10-17	1
ST_CLERK	1997-01-29	1
ST_CLERK	1998-03-15	1
ST_CLERK	1998-07-09	1
ST_MAN	1999-11-16	1

HAVING: subclause van GROUP BY

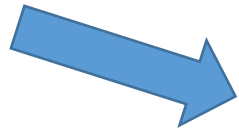
-- HAVING kan enkel samen met een GROUP BY gebruikt worden.

HAVING vs WHERE

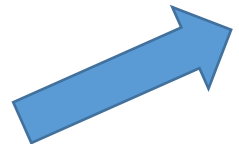
-- VOORBEELD: combinatie
-- van GROUP BY
-- met WHERE

```
SELECT job_id, avg(salary)  
FROM employees  
GROUP BY job_id;
```

```
SELECT *  
FROM employees  
WHERE department_id = 50;
```



```
SELECT job_id, avg(salary)  
FROM employees  
WHERE department_id = 50  
GROUP BY job_id;
```



- WHEN : voor voorwaarde op *record*-niveau
- HAVING : voor voorwaarde op *groep*-niveau

```
SELECT job_id , avg(salary)
```

```
FROM employees
```

```
WHERE department_id = 110
```

```
GROUP BY job_id;
```

-- “department_id=110” is een voorwaarde op record-niveau

```
SELECT job_id , avg(salary)
```

```
FROM employees
```

```
GROUP BY job_id
```

```
HAVING avg(salary) >= 9000 ;
```

-- “avg(salary) >= 9000” is een voorwaarde op groep-niveau

-- merk op:

```
SELECT job_id , avg(salary)
```

```
FROM employees
```

```
WHERE avg(salary) >= 9000
```

```
GROUP BY job_id;
```

-- **FOUT!** aggregerende functie in WHERE clause is niet toegelaten

-- combinatie:

```
SELECT job_id , avg(salary)
```

```
FROM employees
```

```
WHERE department_id = 110
```

```
GROUP BY job_id
```

```
HAVING avg(salary) >= 9000 ;
```

GROUP BY: OPDRACHT 5

- maak een resultatenlijst met
 - per departement het aantal medewerkers en de som van alle lonen,
 - voor zover de som van de lonen meer dan 10.000 bedraagt
-
- 6 rows returned

20	2	19000.00
50	5	17500.00
60	3	19200.00
80	3	30100.00
90	3	58000.00
110	2	20300.00

GROUP BY: OPDRACHT 6

- maak een resultatenlijst met
 - het aantal medewerkers per departement,
 - van elk departement met minstens 3 medewerkers
 - met uitzondering van departement 60,
 - gerangschikt volgens het aantal medewerkers
-
- 3 rows returned

80	3
90	3
50	5