

## Datagruppering GROUP BY - HAVING

**GROUP BY** klausulen bruges til at arrangere identiske data i grupper:

```
SELECT count(*)  
FROM tabel  
GROUP BY felt
```

Vil tælle hvor mange rækker der har samme værdi i det angivne felt for hver forskellig værdi.

```
SELECT Fornavn, count(*) from Navne GROUP BY Fornavn
```

Fornavn
Anders
Anders
Anders
Anders
Anders
Anders
Anna
Bente
Bente
Bente
Camilla
Charlotte
Charlotte
Charlotte
Charlotte

Resultat af GROUP BY

Fornavn	Antal
Anders	6
Anna	1
Bente	3
Camilla	1
Charlotte	4

Count(\*) tæller antallet af ens rækker.

Hvis vi henter fornavn ud af Navn i tabellen Navne med SUBSTRING:

```
SELECT substring(Navn,1,charindex(' ',Navn)) AS Fornavn,count(*) AS Antal FROM Navne  
GROUP BY substring(Navn,1,charindex(' ',Navn))
```

Tabellen Navne

Navn
Anders Andersen
Anders Christensen
Anders Jørgensen
Anders Kristensen
Anders Olsen
Anders Olsen
Anna Larsen
Bente Kristensen
Bente Kristensen

Udtræk grupperet på del af Navn

Fornavn	Antal
Anders	6
Anna	1
Bente	3
Camilla	1
Charlotte	4
Christian	2
Hans	1
Helle	2

## Datagruppering GROUP BY - HAVING

### Filtreringer med WHERE og HAVING

Vis gennemsnit salary for medarbejderne i hver afdeling:

```
SELECT deptno, avg(sal)
FROM employee
GROUP BY deptno
```

Vis gennemsnit salary for kvinderne i hver afdeling (der filtreres på række-niveau):

```
SELECT avg(sal), deptno
FROM employee
WHERE gender = 'F'
GROUP BY deptno
```

Vis antallet af kvinder i hver af de afdelinger hvor kvinderne i gennemsnit har salary over 1000 (der filtreres også på gruppe-niveau):

```
SELECT count(*),deptno
FROM employee
WHERE Gender = 'F'
GROUP BY deptno
HAVING avg(sal) > 1000
```

Hvis man gerne vil oprette en primary key på et felt i en tabel hvor der er indsat data, men dette ikke kan lade sig gøre fordi der er dupletter, kan disse dupletter findes på følgende måde, her i en tabel med Postnumre

```
SELECT Postnr, count(*) FROM Postnumre
GROUP BY Postnr
HAVING count(*) > 1
```

Endelig kan der sorteres med order BY på en eller flere af kolonnerne i outputtet(samme som ovenfor, men sorteret):

```
SELECT count(*) as Antal ,deptno
FROM employee
WHERE Gender = 'F'
GROUP BY deptno
HAVING avg(sal) > 1000
ORDER BY Antal desc
```

## Datagruppering GROUP BY - HAVING

Alle aggregate funktioner kan angives i samme SELECT, men af andre felter kun det/de felter, der grupperes på.

```
SELECT avg(sal), max(sal), min(sal), DeptNo
FROM Employee
GROUP BY DeptNo
```

### *Underforespørgsler eller Subqueries*

Hent alle oplysninger fra tabellen Postnumre, hvor Postnr indgår i tabelle Navne:

```
SELECT * FROM Postnumre
WHERE Postnr IN (SELECT Postnr FROM Navne)
```

Postnr	Bynavn
1110	København K
1200	København K
1324	København K
1350	København K
1532	København V

eller Postnr ikke indgår (NOT IN) :

```
SELECT * FROM Postnumre
WHERE Postnr NOT IN (SELECT Postnr FROM Navne)
```

```
SELECT felter FROM tabel-navne
[WHERE clause]
[GROUP BY clause]
[HAVING clause]
[ORDER BY clause]
```