

FUNCIONS D'AGRUPACIÓ

FUNCIONS DE GRUP

- **MIN, MAX:** amb qualsevol tipus de dades
- **AVG, SUM, STDEV, VAR:** amb dades numèriques
- **DISTINCT** amb funcions de grup: SUM(DISTINCT SALARY)
- **COUNT**
 - ✓ **COUNT(*):** Nombre de files del resultat
 - ✓ **COUNT(expressió):** Nombre de files del resultat NO NULL
 - ✓ **COUNT(DISTINCT expressió):** Nombre de files del resultat NO NULL sense duplicats

LES FUNCIONS DE GRUP

- Ignoren els valors NULL
- No es poden utilitzar al WHERE

AGRUPACIÓ DE FILES

- **GROUP BY**
- **HAVING:** Condicions de grup

FUNCIONS “NUIADES”

QUÈ SÓN LES FUNCIONS DE GRUP?

Les funcions d'una fila (o de fila única) s'apliquen a les dades de cada fila, produint un resultat per fila.

Les **FUNCIONS DE GRUP** (*funcions d'agrupació o funciones de agregado*) s'apliquen a totes les files, o a un grup específic de files, i produeixen un únic resultat per grup.

EXEMPLE

| | | | |
|--------------------------------------|---|-------------|-------|
| Maximum salary in EMPLOYEES table | <table><tr><td>MAX(SALARY)</td></tr><tr><td>24000</td></tr></table> | MAX(SALARY) | 24000 |
| MAX(SALARY) | | | |
| 24000 | | | |

Les funcions de grup **ignoren els valors NULL !!**

7 FUNCIONS DE GRUP

MIN, MAX, COUNT, SUM, COUNT, AVG, STDDEV, VARIANCE

MIN i MAX

MIN

Retorna el **valor mínim** d'un conjunt de dades (numèriques, caràcter o data).

MAX

Retorna el **valor màxim** d'un conjunt de dades (numèriques, caràcter o data).

EXEMPLE

```
SELECT MIN(SALARY), MAX(HIRE_DATE)
FROM EMPLOYEES
```

| MIN(SALARY) | MAX(HIRE_DATE) |
|--|----------------|
| 2500 | 29/01/00 |
| 1 filas devueltas en 0,03 segundos Descargar | |

AVG i SUM

AVG

Retorna el **promig** d'un conjunt de dades numèriques. No té en compte els valors NULL.

SUM

Retorna la **suma** d'un conjunt de dades numèriques. No té en compte els valors NULL.

EXEMPLE

```
SELECT SUM(COMMISSION_PCT), ROUND(AVG(BONUS),2)
FROM EMPLOYEES
```

| SUM(COMMISSION_PCT) | ROUND(AVG(BONUS),2) |
|--|---------------------|
| ,85 | 1483,33 |
| 1 filas devueltas en 0,00 segundos Descargar | |

STDEV i VAR

STDEV

Desviació estàndard (*Standard Deviation*) d'un conjunt de dades numèriques.
No té en compte els valors NULL.

VAR

Variància d'un conjunt de dades numèriques.
No té en compte els valors NULL.

Són funcions estadístiques que no utilitzarem !!

DISTINCT AMB FUNCIONS DE GRUP

DISTINCT es pot utilitzar amb totes les funcions de grup per a que no operin amb els valors duplicats.

Salari dels empleats del departament 90:

| LAST_NAME | SALARY | DEPARTMENT_ID |
|-----------|--------|---------------|
| King | 24000 | 90 |
| Kochhar | 17000 | 90 |
| De Haan | 17000 | 90 |

3 filas devueltas en 0,03 segundos [Descargar](#)

Suma dels salaris dels empleats del departament 90:

```
SELECT SUM(SALARY)
FROM EMPLOYEES
WHERE DEPARTMENT_ID=90
```

| SUM(SALARY) |
|-------------|
| 58000 |

1 filas devueltas en 0,02 segundos [Descargar](#)

Suma dels salaris **DIFERENTS** dels empleats del departament 90:

```
SELECT SUM(DISTINCT SALARY)
FROM EMPLOYEES
WHERE DEPARTMENT_ID=90
```

| SUM(DISTINCTSALARY) |
|---------------------|
| 41000 |

1 filas devueltas en 0,01 segundos [Descargar](#)

**Només compta
un 17000 !!**

COUNT(*)

Retorna el **número de files** del resultat (inclosos **NULL** i duplicats).

EXEMPLES

Nombre empleats:

```
SELECT COUNT(*)  
FROM EMPLOYEES
```

| COUNT(*) |
|--|
| 20 |
| 1 filas devueltas en 0,00 segundos Descargar |

Nombre d'empleats del departament 50:

```
SELECT COUNT(*)  
FROM EMPLOYEES  
WHERE DEPARTMENT_ID=50;
```

| COUNT(*) |
|--|
| 5 |
| 1 filas devueltas en 0,06 segundos Descargar |

COUNT(*)

Retorna el **número de files** del resultat (inclosos **NULL** i duplicats).

EXAMPLES

Nombre d'empleats contractats abans del 1996:

ORACLE

```
SELECT COUNT(*)  
FROM EMPLOYEES  
WHERE EXTRACT(YEAR FROM HIRE_DATE) < 1996
```

| COUNT(*) |
|----------|
| 9 |

MariaDB

```
SELECT COUNT(*)  
FROM EMPLOYEES  
WHERE YEAR(HIRE_DATE) < 1996
```

| COUNT(*) |
|----------|
| 9 |

COUNT(expressió)

Retorna el número de files amb valors **NO NULL** per a expressió (inclosos duplicats)

EXEMPLE

Nombre empleats que tenen comissió:

```
SELECT COUNT(COMMISSION_PCT)  
FROM EMPLOYEES
```

| COUNT(COMMISSION_PCT) |
|--|
| 4 |
| 1 filas devueltas en 0,00 segundos Descargar |

COUNT(DISTINCT expressió)

Retorna el nombre de files **no NULL** del resultat sense comptar els duplicats.

EXAMPLE

Quants treballs (diferents) tenen els empleats ?

```
SELECT COUNT(DISTINCT JOB_ID)  
FROM EMPLOYEES
```

| COUNT(DISTINCTJOB_ID) |
|--|
| 12 |
| 1 filas devueltas en 0,01 segundos Descargar |

A quants departaments (diferents) hi treballen empleats ?

```
SELECT COUNT(DISTINCT DEPARTMENT_ID)  
FROM EMPLOYEES
```

| COUNT(DISTINCTDEPARTMENT_ID) |
|--|
| 7 |
| 1 filas devueltas en 0,01 segundos Descargar |

OPERAR AMB VALORS NULL

Les funcions d'agrupació ignoren els valors NULL.

EXAMPLE

```
SELECT AVG(COMMISSION_PCT)
FROM EMPLOYEES;
```

| AVG(COMMISSION_PCT) |
|---------------------|
| .2125 |

Només calcula la mitjana dels valors que no són NULL.

Si es volen tenir en compte els valors NULL, utilitzar una “**funció NULL**” per convertir els valors NULL en el valor “*adequat*”.

```
SELECT AVG(COALESCE(COMMISSION_PCT,0))
FROM EMPLOYEES;
```

| AVG(COALESCE(COMMISSION_PCT,0)) |
|---------------------------------|
| .0425 |

AGRUPACIÓ DE FILES: GROUP BY

Agrupar les files del resultat, segons una o més columnes, per **calcular resultats per grups** (enlloc de resultats totals).

SINTAXI GENERAL DEL SELECT

```
SELECT column, group_function(column)
FROM table
[WHERE condition]
[GROUP BY group_by_expression]
[ORDER BY column];
```

AGRUPACIÓ DE FILES: GROUP BY

EXEMPLE

Calcular el salari mitjà de cada departament: agrupem per departament i calculem el salari mitjà.

| EMPLOYEES | | | |
|-----------|---------------|--------|-------|
| ID | DEPARTMENT_ID | SALARY | |
| 1 | 10 | 4400 | 4400 |
| 2 | 20 | 13000 | 9500 |
| 3 | 20 | 6000 | |
| 4 | 50 | 5800 | 3500 |
| 5 | 50 | 2500 | |
| 6 | 50 | 2600 | |
| 7 | 50 | 3100 | |
| 8 | 50 | 3500 | 6400 |
| 9 | 60 | 4200 | |
| 10 | 60 | 6000 | |
| 11 | 60 | 9000 | 10033 |
| 12 | 80 | 11000 | |
| 13 | 80 | 10500 | |
| 14 | 80 | 8600 | |
| ... | | | |
| 19 | 110 | 12000 | |
| 20 | (null) | 7000 | |

Average salary in
EMPLOYEES table for
each department

| ID | DEPARTMENT_ID | AVG(SALARY) |
|----|---------------|-----------------------|
| 1 | 10 | 4400 |
| 2 | 20 | 9500 |
| 3 | 50 | 3500 |
| 4 | 60 | 6400 |
| 5 | 80 | 10033.333333333333... |
| 6 | 90 | 19333.333333333333... |
| 7 | 110 | 10150 |
| 8 | (null) | 7000 |

AGRUPACIÓ DE FILES: GROUP BY

EXEMPLE

Calcular el salari mitjà de cada departament: agrupem per departament i calculem el salari mitjà.

```
SELECT DEPARTMENT_ID, AVG(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
ORDER BY DEPARTMENT_ID
```

| DEPARTMENT_ID | AVG(SALARY) |
|---------------|------------------------------------|
| 10 | 4400 |
| 20 | 9500 |
| 50 | 3500 |
| 60 | 6400 |
| 80 | 10033,3333333333333333333333333333 |
| 90 | 19333,3333333333333333333333333333 |
| 110 | 10150 |
| - | 7000 |

AGRUPACIÓ DE FILES: GROUP BY

Normalment es mostra al SELECT el camp pel que agrupem, però les columnes del GROUP BY no tenen perquè ser a la llista del SELECT.

Es pot utilitzar qualsevol columna de la taula per agrupar, estigui al SELECT o no.

```
SELECT AVG(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID;
```

[illegible]

AGRUPACIÓ DE FILES: GROUP BY

ORACLE

A la llista del SELECT només hi poden sortir o camps pels que agrupem (del GROUP BY) o funcions de grup.

Si hi ha algun altre camp genera un **error**.

EXAMPLE

```
SELECT DEPARTMENT_ID, LAST_NAME, MAX(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
ORDER BY DEPARTMENT_ID
```



ORA-00979: not a GROUP BY expression

AGRUPACIÓ DE FILES: GROUP BY

MySQL no genera error.

EXEMPLE

```
SELECT DEPARTMENT_ID, LAST_NAME, MAX(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
ORDER BY DEPARTMENT_ID
```

| DEPARTMENT_ID | LAST_NAME | MAX(SALARY) |
|---------------|-----------|-------------|
| NULL | Grant | 7000.00 |
| 10 | Whalen | 4400.00 |
| 20 | Hartstein | 13000.00 |
| 50 | Mourgos | 5800.00 |
| 60 | Hunold | 9000.00 |
| 80 | Zlotkey | 11000.00 |
| 90 | King | 24000.00 |
| 110 | Higgins | 12000.00 |

Per cada departament, mostra el cognom del darrer empleat.

Agrupant per departament **no té sentit** mostrar el camp cognom de l'empleat !!

AGRUPAR PER MÉS D'UNA COLUMNA

GRUPS DINTRE DE GRUPS

EXEMPLE

Nombre de treballadors de cada departament: agrupar per departament i COUNT

```
SELECT DEPARTMENT_ID, COUNT(*) AS "Empleats"  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID  
ORDER BY DEPARTMENT_ID
```

| DEPARTMENT_ID | Empleats |
|---------------|----------|
| 10 | 1 |
| 20 | 2 |
| 50 | 5 |
| 60 | 3 |
| 80 | 3 |
| 90 | 3 |
| 110 | 2 |
| - | 1 |

8 rows returned in 0.02 seconds [Download](#)

AGRUPAR PER MÉS D'UNA COLUMNA

GRUPS DINTRE DE GRUPS

EXAMPLE

Nombre de treballadors de cada departament que tenen la mateixa feina: agrupar per departament i JOD_ID i COUNT.

```
SELECT DEPARTMENT_ID, JOB_ID, COUNT(*) AS "Empleats"  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID, JOB_ID  
ORDER BY DEPARTMENT_ID
```

Departament 20 tenia
2 treballadors

Departament 50 en
tenia 5

| DEPARTMENT_ID | JOB_ID | Empleats per depart. i job |
|---------------|----------|----------------------------|
| 10 | AD_ASST | 1 |
| 20 | MK_MAN | 1 |
| 20 | MK_REP | 1 |
| 50 | ST_CLERK | 4 |
| 50 | ST_MAN | 1 |
| 60 | IT_PROG | 3 |
| 80 | SA_MAN | 1 |
| 80 | SA_REP | 2 |
| 90 | AD_PRES | 1 |

WHERE AMB GROUP BY

Seleccionem les files amb WHERE abans d'agrupar.

EXAMPLE

Nombre d'empleats de cada departament dels departaments entre 50 i 90:

```
SELECT DEPARTMENT_ID, COUNT(*) AS "Nombre d'empleats"  
FROM employees  
WHERE DEPARTMENT_ID BETWEEN 50 AND 90  
GROUP BY DEPARTMENT_ID  
ORDER BY DEPARTMENT_ID
```

| DEPARTMENT_ID | Nombre d'empleats |
|--|-------------------|
| 50 | 5 |
| 60 | 3 |
| 80 | 3 |
| 90 | 3 |
| 4 filas devueltas en 0,00 segundos Descargar | |

1. Selecciona els registres que compleixen la condició
2. Agrupa
3. Compta

CONDICIONS SOBRE GRUPS: HAVING

- WHERE indica condicions sobre les files seleccionades.
- **HAVING** indica condicions sobre els grups seleccionats, amb funcions d'agrupació.

Si una consulta té **GROUP BY** i **HAVING**:

1. S'agrupen les files que compleixin la condició del WHERE, si n'hi ha.
2. S'apliquen les funcions de grup sobre les files agrupades.
3. Es mostren només els grups que compleixen la clàusula del HAVING.

CONDICIONES SOBRE GRUPOS: HAVING

SINTAXI GENERAL DEL SELECT

```
SELECT column, group_function  
FROM table  
[WHERE condition]  
[GROUP BY group by expression]  
[HAVING group condition]  
[ORDER BY column]
```

CONDICIONS SOBRE GRUPS: HAVING

EXAMPLE

Salari més alt dels departaments que tenen 3 o més empleats:

```
SELECT DEPARTMENT_ID, MAX(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
HAVING COUNT(*) >= 3
ORDER BY DEPARTMENT_ID
```

| DEPARTMENT_ID | COUNT(*) |
|---------------|----------|
| 10 | 1 |
| 20 | 2 |
| 50 | 5 |
| 60 | 3 |
| 80 | 3 |
| 90 | 3 |
| 110 | 2 |
| - | 1 |

8 filas devueltas en 0,01 segundos [Descargar](#)



| DEPARTMENT_ID | MAX(SALARY) |
|---------------|-------------|
| 50 | 5800 |
| 60 | 9000 |
| 80 | 11000 |
| 90 | 24000 |

4 filas devueltas en 0,01 segundos [Descargar](#)

CONDICIONS SOBRE GRUPS: HAVING

EXEMPLE

Salari més alt dels departaments que tenen el salari màxim més gran que 10000.

```
SELECT DEPARTMENT_ID, MAX(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
HAVING MAX(SALARY)>10000
ORDER BY DEPARTMENT_ID
```

| DEPARTMENT_ID | MAX(SALARY) |
|---------------|-------------|
| 10 | 4400 |
| 20 | 13000 |
| 50 | 5800 |
| 60 | 9000 |
| 80 | 11000 |
| 90 | 24000 |
| 110 | 12000 |
| - | 7000 |

8 rows returned in 0.01 seconds [Download](#)

| DEPARTMENT_ID | MAX(SALARY) |
|---------------|-------------|
| 20 | 13000 |
| 80 | 11000 |
| 90 | 24000 |
| 110 | 12000 |

4 filas devueltas en 0,04 segundos [Descargar](#)

DIFERÈNCIES ENTRE WHERE I HAVING

WHERE

- S'utilitza per a **filtrar** els registres que formaran part del resultat, **abans de fer els GRUPS**.
- La **visibilitat** del WHERE és **d'una única fila**, per això **no pot avaluar funcions de grup**.
- Les condicions del WHERE poden ser sobre qualsevol camp de la taula.

DIFERÈNCIES ENTRE WHERE I HAVING

HAVING

- S'utilitza per a filtrar **valors d'un grup després d'agrupar**.
- Com els grups ja estan fets, **pot utilitzar funcions** de grup en les seves condicions.
- Només podem utilitzar en les **condicions** del HAVING o **funcions de grup** o condicions en les que intervinguin només els **camp seleccionats** en la consulta. La resta de camps de la taula original ja no formen part del resultat quan s'aplica el HAVING.

DIFERÈNCIES ENTRE WHERE I HAVING

“COM A NORMA”

- Utilitzar **WHERE** per posar condicions sobre els camps de la taula per a seleccionar les files del resultat.
- Utilitzar **HAVING** per posar condicions que han de complir els grups per ser seleccionats (utilitzen funcions de grup).

DIFERÈNCIES ENTRE WHERE I HAVING

EXEMPLE

Mostrar quants treballadors hi ha de cada feina:

```
SELECT JOB_ID, COUNT(*)  
FROM EMPLOYEES  
GROUP BY JOB_ID
```

DIFERÈNCIES ENTRE WHERE I HAVING

WHERE

Podem posar qualsevol condició sobre qualsevol camp de la taula, sempre que no utilitzi funcions de grup:

```
SELECT JOB_ID, COUNT(*)  
FROM EMPLOYEES  
WHERE COUNT(*)>1  
GROUP BY JOB_ID
```



ORA-00934: función de grupo no permitida aquí

DIFERÈNCIES ENTRE WHERE I HAVING

HAVING

Condicions només sobre els camps de SELECT o que utilitzin funcions de grup.

```
SELECT JOB_ID, COUNT(*)  
FROM EMPLOYEES  
GROUP BY JOB_ID  
HAVING EMPLOYEE_ID<200
```



ORA-00979: no es una expresión GROUP BY

```
SELECT JOB_ID, COUNT(*)  
FROM EMPLOYEES  
GROUP BY JOB_ID  
HAVING UPPER(JOB_ID) LIKE 'S%'
```

JOB_ID és al SELECT

```
SELECT JOB_ID, COUNT(*)  
FROM EMPLOYEES  
GROUP BY JOB_ID  
HAVING COUNT(*)>1
```

DIFERÈNCIES ENTRE WHERE I HAVING

Millor posar condicions simples al **WHERE**, i deixar condicions de grup al **GROUP BY** !!

PER NO LIAR 😊

FUNCIONS NIUADES - ORACLE

Aplicar una funció de grup al resultat d'una funció de grup.

EXAMPLE

Mostrar el salari mitjà més petit de tots els departaments.

1. Busquem quin és el salari mitjà de cada departament.

```
SELECT DEPARTMENT_ID, AVG(SALARY)
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
ORDER BY 2
```

[illegible]

FUNCIONS NIUADES - ORACLE

2. Per trobar el salari més petit dels salaris promig, apliquem la funció **MIN** al resultat de la funció AVG:

```
SELECT MIN(AVG(SALARY))  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID
```

| MIN(AVG(SALARY)) | |
|--|--|
| 3500 | |
| 1 rows returned in 0.03 seconds Download | |

FUNCIONS NIUADES - MySQL

EXAMPLE

Mostrar el salari mitjà més petit de tots els departaments.

La mateixa sentència SQL executada en MariaDB:

```
SELECT MIN(AVG(SALARY))  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID
```

Error

Consulta SQL: [Copia](#) ⓘ

```
SELECT MIN(AVG(SALARY))  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID LIMIT 0, 25
```

MySQL diu: ⓘ

#1111 - Invalid use of group function

MySQL no permet que una funció de grup cridi a una altra funció.

Veurem la solució en l'apartat de subconsultes.