

Font: **materials IOC** a la plataforma actual: activitats **apartat 7.1 Consultes avançades SQL amb funcions sobre operacions d'agrupació**

WHERE/HAVING, funcions d'agregat i GROUP BY

- WHERE/HAVING, totes dues clàusules permeten referir condicions a avaluar pel SGBD.
- Funcions d'agregat, com el seu propi nom indica, són càlculs que s'apliquen a agregats, a un conjunt de fileres, a una agregat de fileres. Per fer-les servir ha de ser indicat de forma adient sobre quin conjunt de fileres s'ha de fer el càlcul.
- GROUP BY, permet fer agrupacions/conjunts de fileres sobre camps d'una taula. Es pot fer servir per agrupar i prou, potser amb una ordenació posterior, de cara a mostrar dades, fer un llistat de dades agrupades. Es pot fer servir per indicar un conjunt de registres (agrupació) sobre el que s'ha d'aplicar una funció d'agregat.
- WHERE/HAVING/GROUP BY. HAVING estableix condicions sobre GROUP BY com WHERE ho fa amb SELECT. El que canvia és que el que hi ha al WHERE s'aplica abans de fer les agrupacions, si hi han, i el que hi ha al HAVING s'aplica després. En tot cas, WHERE s'aplica al SELECT i, quan això, està fet, llavors s'aplica HAVING. Per això HAVING i WHERE poden contenir sintaxi similar, condicions al cap i a la fi, amb la diferència que HAVING pot contenir funcions d'agregat.

Anàlisi de les activitats: jugant amb la solució es proposen variacions i la seua validesa

1.- Visualitzeu les oficines en les quals el salari mitjà és superior o igual que la mitjana entre tots els salaris.

```
SELECT officecode, AVG(salary) FROM 'RRHH'.employees
GROUP BY officecode
HAVING AVG(salary) >=(SELECT AVG(salary) FROM 'RRHH'.employees);
```

Aquesta consulta també és vàlida, doncs la clàusula HAVING el que fa és de WHERE sobre el conjunt de fileres de l'agrupació feta, si no està, doncs simplement es fa una agrupació que permet determinar els grups per aplicar la funció d'agregat.

```
SELECT officecode, AVG(salary) FROM 'RRHH'.employees
GROUP BY officecode
```

() Aquesta consulta no és vàlida, doncs es vol mostrar un càlcul, una funció d'agregat i a la vegada una dada: selecciona tots els codis d'oficina i mostra'n la mitjana del salary... com fer per mostrar cada codi d'oficina acompanyat d'un valor calculat?...es necessari saber sobre quines fileres s'ha de fer el càlcul... no es pot. Per això a l'errada que surt si s'executa la consulta el postgresQL us recomanarà que afegiu una clàusula GROUP BY sobre el camp del codi d'oficina.*

```
SELECT officecode, AVG(salary) FROM 'RRHH'.employees
```

2.- A partir de la taula **employees**, visualitzeu el nombre de representants de vendes de l'oficina de Paris.

```
SELECT count(*)
FROM 'RRHH'.employees
WHERE officecode IN (SELECT officecode
                     FROM 'RRHH'.offices
                     WHERE city='Paris')
AND jobtitle='Sales Rep'
GROUP BY jobtitle;
```

Aquesta consulta també és vàlida: s'aplica la funció d'agregat a una agrupació, a un conjunt de fileres feta, en aquest cas, sobre totes les fileres de la taula d'empleats, sense filtre de condició prèvi, sense WHERE.

```
SELECT count(*)
FROM 'RRHH'.employees
GROUP BY jobtitle;
```

() (**) Aquesta consulta també és vàlida: s'aplica la funció d'agregat a un conjunt de fileres feta, en aquest cas, sobre totes les fileres de la taula d'empleats, sense filtre de condició prèvi, sense WHERE i sense agrupació.*

```
SELECT count(*)
FROM 'RRHH'.employees
```

3.- Partint de la taula **employees**, visualitzeu per a cada ofici dels empleats de l'oficina de San Francisco la suma dels seus salaris.

```
SELECT jobtitle, SUM(salary)
FROM 'RRHH'.employees
WHERE officecode IN (SELECT officecode
                     FROM 'RRHH'.offices
                     WHERE city LIKE 'San Francisco')
GROUP BY jobtitle;
```

4.- Seleccionen els cognoms de la taula **employees** amb un salari major o igual a la mitjana del salari de la seva oficina.

```
SELECT lastname, salary
FROM 'RRHH'.employees e
WHERE salary >= (SELECT avg(salary)
                 FROM 'RRHH'.employees
                 WHERE officecode=e.officecode);
```

Aquesta consulta també és vàlida: a la subconsulta es deixa de fer la comprovació que els empleats siguin de la mateixa oficina, llavors seria un detall de la lògica de la consulta, però la sintaxi és correcta.

```
SELECT lastname, salary
FROM 'RRHH'.employees e
WHERE salary >= (SELECT avg(salary)
                 FROM 'RRHH'.employees);
```

() Aquesta consulta no és vàlida: les funcions d'agregat han de ser aplicades a un grup de fileres. És el cas ja comentat abans, però aquí apareix al WHERE: es vol que es mostrin el cognom i el salary dels salaris majors i iguals a la mitjana... la mitjana de què, sobre quines fileres s'ha de fer el càlcul.*

```
SELECT lastname, salary
FROM 'RRHH'.employees e
WHERE salary >= avg(salary)
```

5.- A partir de la taula **employees** visualitzeu el nombre d'empleats de cada oficina que tinguin l'ofici 'Sales Rep'.

```
SELECT officecode, COUNT(*) "Noemple"
FROM 'RRHH'.EMPLOYEES
WHERE jobtitle LIKE 'Sales Rep'
GROUP BY officecode;
```

6.- Busqueu les oficines que tenen més de dues persones treballant en la mateixa professió.

```
SELECT officecode, COUNT(*)
FROM 'RRHH'.employees
GROUP BY officecode, jobtitle
HAVING COUNT(*) > 2;
```

() Aquesta consulta no és vàlida: les funcions d'agregat han de ser aplicades a un grup de fileres i no es pot mostrar un càlcul i un camp sense que hi hagi una agrupació pel camp al que no s'aplica la funció d'agregat: manca el GROUP BY*

```
SELECT officecode, COUNT(*)
FROM 'RRHH'.employees
HAVING COUNT(*) > 2;
```

*(**) Aquesta consulta és vàlida: aquí es veu com el HAVING és com un WHERE, serveix per referir condicions, simplement s'aplica després d'haver-se fet la selecció de camps del SELECT. Potser la consulta no té sentit, de fet, es extrany trobar consultes on hi hagi un HAVING amb sentit sense una consulta prèvia de filtratge completa (SELECT WHERE) o sense un GROUP BY, però és computable: es demana fer un recompte sobre tots el empleats i d'aquell resultat obtingut, es comprova la condició que el càlcul fet sigui major que dos.*

```
SELECT COUNT(*)
FROM 'RRHH'.employees
HAVING COUNT(*) > 2;
```