

# Bases de Datos

---

## 8.- Consultas agrupadas. Subconsultas



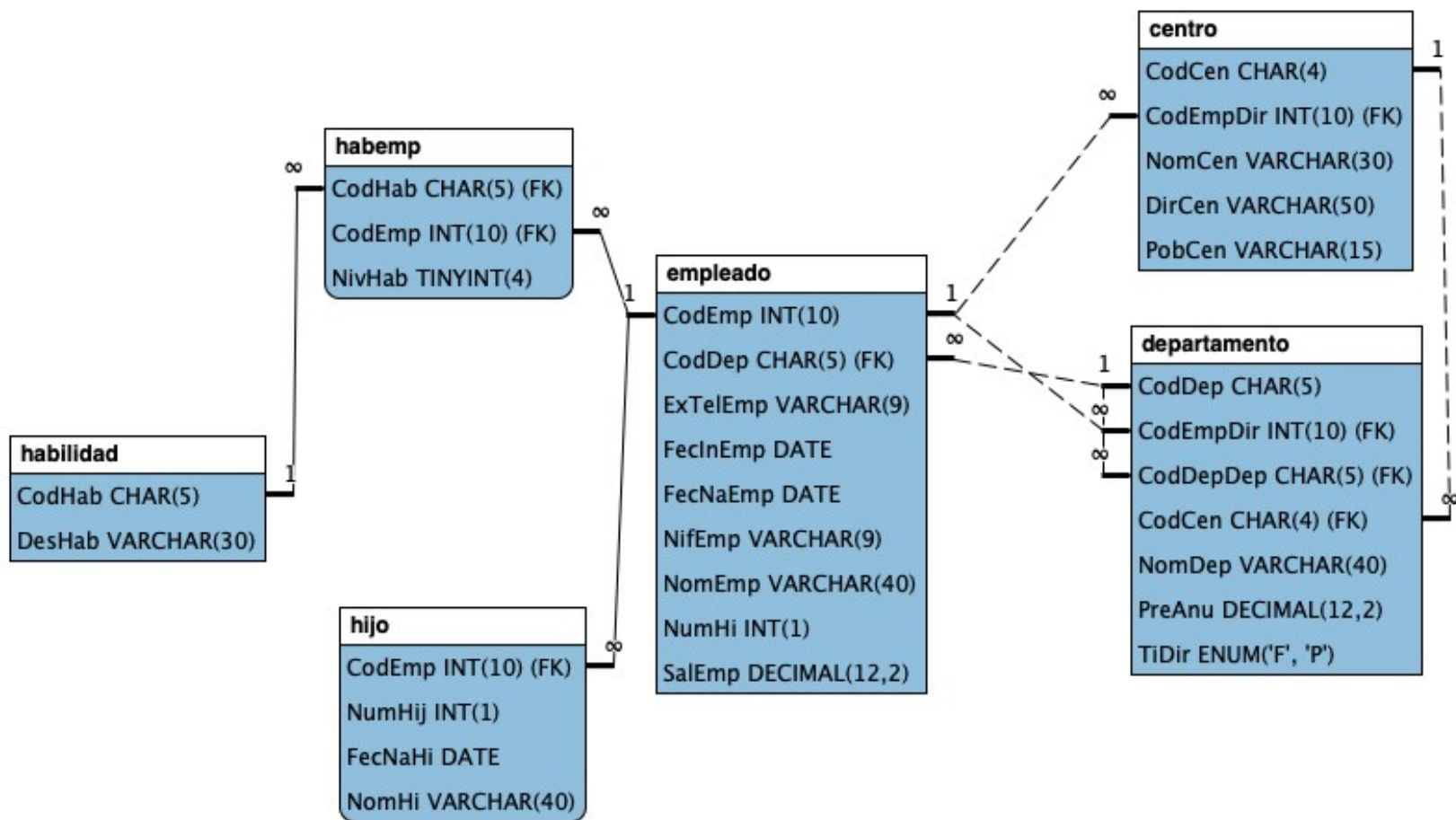
IES LA ENCANTÁ

# Índice



- Consultas agrupadas
  - Orden de las clausulas
- Subconsultas
  - Operadores IN, ANY, ALL, EXISTS
- Tablas derivadas

# BD empleados





# Select ampliado

**select** columna1, columna2 ...

**from** tabla1 **join** tabla 2 on ...

**where** condicion1 and/or condicion2...

**group by** columna1, columna2

**having** condicionAgrupada1,  
condicionAgrupada2, ...

**order by** orden1, orden2,...

**limit** cantidad;

# Consultas agrupadas



- Permiten realizar cálculos en vertical
- Pasos
  - 1) `select`: Indicar las columnas a agrupadas
  - 2) `select`: Indicar los cálculos mediante funciones agregadas (`count`, `sum`, `max`, `min`, `avg`...)
  - 3) `group by`: indicar las agrupaciones (deben coincidir al menos con las columnas a mostrar)

# group by I



- <https://mariadb.com/kb/en/group-by/>
- Nombre de departamento y presupuesto anual
  - `select nomdep, preanu from departamento;`
- Nombre de centro, departamento y presupuesto anual (de cada departamento)
  - `select c.nomcen, d.nomdep, d.preatu from departamento d join centro c on d.codcen=c.codcen;`

# group by II



- Nombre de centro y presupuesto (salen centro repetidos ya que muestra el presupuesto de cada departamento)
  - `select c.nomcen, d.preatnu from departamento d  
join centro c on d.codcen=c.codcen;`
- Solución: agrupar por centro
  - `select c.nomcen, sum(d.preatnu)  
from departamento d join centro c on  
d.codcen=c.codcen  
group by (c.nomcen);`

# having



- Permite filtrar tras realizar los cálculos de agrupación.
- Orden de ejecución de las cláusulas:
  - `where` que filtra las filas según las condiciones que pongamos.
  - `group by` que crea una tabla de grupos nueva.
  - `having` filtra los grupos.
  - `order by` que ordena o clasifica la salida.



# Ejemplo having



```
-- Centro y presupuesto anual de aquellos  
-- centros con más de 100.000.000 de presupuesto  
select c.nomcen, sum(d.preatu)  
from departamento d join centro c on d.codcen=c.codcen  
group by (c.nomcen)  
having sum(d.preatu) > 100000000;
```

```
-- Centro y presupuesto anual de aquellos  
-- centros con más de 100.000.000 de presupuesto  
-- y que sus departamentos tengan un presupuesto anual  
-- superior a 20.000.000  
select c.nomcen, sum(d.preatu)  
from departamento d join centro c on d.codcen=c.codcen  
where d.preatu > 20000000  
group by (c.nomcen)  
having sum(d.preatu) > 100000000;
```

# Subconsultas



- Consulta que aparece dentro de la cláusula WHERE o HAVING de otra sentencia SQL.
- La sentencia externa puede ser una SELECT, DELETE, INSERT o UPDATE.
- **Formato general más usado:**

sentencia  
externa

```
SELECT ... FROM ...  
WHERE expresión operador (  
    SELECT columna FROM tabla  
    WHERE condición)  
[ORDER BY col-1 [, col-2]..];
```

# Subconsultas: reglas



- La subconsulta se encierra entre paréntesis.
- La cláusula `order by` no puede ir en una subconsulta.
- A excepción del operador EXISTS, deben producir una sola columna de datos como resultado.
- Una subconsulta puede contener a otra subconsulta y ésta a otra y así n veces. Límite: según SGBDR.

# Operadores subconsultas



- El resultado de una subconsulta puede ser un único valor o varios valores.
- Para subconsulta de un solo valor:
  - Operadores de relación:  $=$ ,  $<$ ,  $>$ ,  $<>$ ,  $>=$ ,  $<=$
- Para subconsulta de ninguno o más valores:
  - `IN`, `ANY`, `ALL`, `EXISTS`.

# Ejemplos subconsultas



-- Cantidad de hijos máximo

```
select max(numhi) from empleado
```

-- Empleado con más hijos

```
select nomemp from empleado where numhi =  
    (select max(numhi) from empleado);
```

-- Empleados con menos hijos que 'Saladino Mandamás, Augusto'

```
select nomemp from empleado where numhi <  
    (select numhi from empleado where nomemp = "Saladino  
Mandamás, Augusto");
```

# Operadores IN, ANY



- [NOT] IN: Verdadero si el valor comparado coincide con alguno de los valores de la columna resultante de la subconsulta.
- ANY: Devuelve verdadero si la comparación se cumple para alguno de los valores resultantes de la subconsulta.

```
select nomdep from departamento where codcen in (  
    select codcen from centro where pobcen='Murcia');
```

-- Nombres y departamentos de los empleados cuyo sueldo  
-- es superior a alguno de sus compañeros de departamento.

```
select nomemp,coddep from empleado e  
where salem > any (  
    select salem from empleado where coddep=e.coddep);
```

# Operador ALL



- ALL: Devuelve verdadero si la comparación se cumple para todos los valores resultantes de la subconsulta.

```
-- Nombre, salario y código de departamento de los empleados que tienen el  
-- mayor sueldo de su departamento.
```

```
select nomemp,salemp,coddep from empleado e  
where salemp >= all  
    (select salemp from empleado where coddep=e.coddep);
```

```
-- Mismo resultado
```

```
select nomemp, salemp, coddep from empleado e  
where salemp = (select max(salemp) from empleado where coddep=e.coddep);
```

# Operador EXISTS



- [NOT] EXISTS: Devuelve verdadero si la subconsulta obtiene algún resultado (alguna fila).
  - No evalúa ninguna comparación
  - Hay que enlazar la consulta principal con la subconsulta

-- Nombre y departamento de los empleados para los que exista otro  
-- empleado que se haya incorporado a la empresa antes que ellos  
-- (e.fecinemp es la fecha de incorporación a la empresa del empleado  
-- que se está evaluando en la consulta principal o externa).

```
select nomemp, coddep from empleado e
where exists
  (select * from empleado where fecinemp < e.fecinemp);
```



# Tablas derivadas



- Colocar una subconsulta en vez de una tabla en la clausula `from`.
- La tabla derivada debe ir entre paréntesis y tener un alias para poder referenciarla.

-- Empleados que ganaban el máximo en cada departamento.

-- Con subconsulta

```
select nomemp, sal emp, coddep from empleado e
where sal emp = (select max(sal emp) from empleado
                 where coddep=e.coddep);
```

-- Con tabla derivada

```
select nomemp, sal emp, coddep from
empleado e, (select max(sal emp) max
              from empleado where coddep=e.coddep) empmax
where e.sal emp=empmax.max;
```



¿Alguna pregunta?