

# SQL El lenguaje de Consulta Estructurado

Subconsultas,  
Consultas Negativas

## Subconsultas

- Una **subconsulta** es una sentencia **SELECT** que aparece **dentro** de otra sentencia **SELECT** que llamaremos consulta principal.
- Se puede encontrar en la lista de selección, en la cláusula **WHERE** o en la cláusula **HAVING** de la consulta principal.
- Una subconsulta tiene la misma sintaxis que una sentencia **SELECT** normal exceptuando que aparece **encerrada entre paréntesis**, no puede contener la cláusula **ORDER BY** y tiene algunas restricciones en cuanto a número de columnas según el lugar donde aparece en la consulta principal.

## Subconsultas

- Listar el nombre de las asignaturas que tienen más créditos que la asignatura llamada 'Bases de Datos'.

```
SELECT Nombre  
FROM Asignaturas  
WHERE credits > (SELECT credits FROM Asignaturas  
WHERE upper(Nombre) = 'BASES DE DATOS');
```

NOMBRE
Prácticas en empresa

## Consultas Anidadas

- La subconsulta se ejecuta una vez por cada fila de la consulta principal.
- La subconsulta puede utilizar todos los atributos de las tablas que aparecen en el FROM de la consulta principal.
  - Imprescindible un alias de tabla.
- Las consultas anidadas imponen una cierta estructura (y orden de ejecución) a las consultas.
  - Ventaja: Suelen ser más fáciles de interpretar por el usuario, si el nivel de anidamiento es limitado.
  - Desventaja: más lentas de ejecutar.

## Consultas Anidadas

- Listar el nombre de las asignaturas que tienen mas créditos que las asignaturas de segundo curso.

```
SELECT Nombre  
FROM Asignaturas  
WHERE credits > (SELECT credits FROM Asignaturas  
WHERE curso = 2 ) ;
```

Error: single-row subquery  
returns more than one row

El resultado de la  
subconsulta devuelve  
más de una tupla

CREDITOS
6
6
6
4
4
6

## ANY / ALL

- Cuando las subconsultas que se utilizan en una expresión de comparación devuelven más de una tupla hay que combinar los operadores de comparación (<, >, =, !=,...) con ANY / ALL
- Operador **ANY** Subconsulta. Será verdad si algún valor del resultado de la subconsulta hace cierta la comparación.
- Operador **ALL** Subconsulta. Será verdad si todos los valores del resultado de la subconsulta hacen cierta la comparación.

## Consultas Anidadas

- Listar el nombre de las asignaturas que tienen mas créditos que **ALGUNA DE** las asignaturas de segundo curso.

```
SELECT Nombre  
FROM Asignaturas  
WHERE credits > ANY (SELECT credits FROM Asignaturas  
                      WHERE curso = 2 ) ;
```

NOMBRE
Prácticas en empresa
Programación Orientada a Objetos
Bases de Datos
Teoria de la señal
Logica Computacional
Matemática Discreta
Sistemas Operativos
Estructura de Computadores
Calculo Numerico
Dispositivos Electronicos
Modelos Computacionales
Ingeniería Web
Teoria de Automatas

## Consultas Anidadas

- Listar el nombre de las asignaturas que tienen mas créditos que **TODAS** las asignaturas de segundo curso.

```
SELECT Nombre  
FROM Asignaturas  
WHERE credits > ALL (SELECT credits FROM Asignaturas  
                     WHERE curso = 2 ) ;
```

NOMBRE
Bases de Datos
Programación Orientada a Objetos
Prácticas en empresa



## IN / EXISTS

- Los operadores IN y EXISTS se aplican a conjuntos de tuplas.
- *elem* **IN** *conjunto\_tuplas*
  - La expresión es cierta si el elemento pertenece al conjunto dado.
- **EXISTS** *conjunto\_tuplas*
  - La expresión es cierta si el conjunto de tuplas no está vacío. Es decir, existe al menos una tupla en el conjunto.
- Se pueden utilizar con el resultado de subconsultas para emular las reuniones.

## IN / EXISTS

- Mostrar el código de las asignaturas que imparte el profesor Manuel Enciso

```
SELECT I.Asignatura "Codigo Asignatura"  
FROM profesores P JOIN Impartir I ON  
      I.profesor = P.id  
WHERE upper(P.Nombre) = 'MANUEL'  
      AND upper(P.Apellido1) = 'ENCISO';
```

Codigo Asignatura
112
140

Se obtiene el id de  
Manuel Enciso

```
SELECT Asignatura "Codigo Asignatura"  
FROM Impartir  
WHERE profesor IN (SELECT id FROM Profesores  
      WHERE upper(Nombre) = 'MANUEL'  
      AND upper(Apellido1) = 'ENCISO') ;
```

Se obtienen los  
profesores con ese id

## IN / EXISTS

- Mostrar el código de las asignaturas que imparte el profesor Manuel Enciso

```
SELECT I.Asignatura "Codigo Asignatura"  
FROM profesores P, Impartir I  
WHERE I.profesor = P.id  
      AND upper(P.Nombre) = 'MANUEL'  
      AND upper(P.Apellido1) = 'ENCISO';
```

Codigo Asignatura
112
140

```
SELECT I.Asignatura "Codigo Asignatura"  
FROM Impartir I  
WHERE EXISTS (SELECT * FROM Profesores P  
              WHERE I.profesor = P.Id  
              AND upper(P.Nombre) = 'MANUEL'  
              AND upper(P.Apellido1) = 'ENCISO') ;
```

Se obtienen las asignaturas para las que existe un profesor llamado Manuel Enciso que la imparte (I.profesor = P.id)

## ¿Qué devuelven las siguientes consultas?

```
SELECT Asignatura "Codigo Asignatura"  
FROM Impartir  
WHERE EXISTS (SELECT *  
               FROM Profesores , Impartir  
               WHERE profesor = id  
               AND upper(nombre) = 'MANUEL'  
               AND upper(apellido1) = 'ENCISO') ;
```

```
SELECT Asignatura "Codigo Asignatura"  
FROM Impartir  
WHERE EXISTS (SELECT *  
               FROM Profesores , Impartir  
               WHERE profesor = id  
               AND upper(nombre) = 'PEDRO'  
               AND upper(apellido1) = 'ENCISO') ;
```

## Condición de reunión compuesta

- Mostrar el código de los alumnos que reciben clase del profesor de código C-34-TU-00

```
SELECT M.Alumno "Codigo Alumno"  
FROM Matricular M, Impartir I  
WHERE I.Asignatura=M.Asignatura  
      AND I.Grupo=M.Grupo  
      AND ICurso=M.Curso  
      AND I.Profesor = 'C-34-TU-00';
```

Codigo Alumno
12312342
25748955
54354355

Se obtienen los  
alumnos de esas  
asignaturas

```
SELECT Alumno "Codigo Alumno"  
FROM Matricular  
WHERE (Asignatura, Grupo, Curso) IN  
      (SELECT Asignatura, Grupo, Curso  
       FROM Impartir  
       WHERE Profesor = 'C-34-TU-00') ;
```

Se obtienen las  
asignaturas  
impartidas por el  
profesor

## Condición de reunión compuesta

- Mostrar el código de los alumnos que reciben clase del profesor de código C-34-TU-00

```
SELECT M.Alumno "Codigo Alumno"  
FROM Matricular M, Impartir I  
WHERE I.Asignatura=M.Asignatura  
      AND I.Grupo=M.Grupo  
      AND ICurso=M.Curso  
      AND I.Profesor = 'C-34-TU-00';
```

Codigo Alumno
12312342
25748955
54354355

```
SELECT Alumno "Codigo Alumno" FROM Matricular M  
WHERE EXISTS (SELECT *  
              FROM Impartir I  
              WHERE I.Asignatura=M.Asignatura  
                    AND I.Grupo=M.Grupo  
                    AND I.Curso=M.Curso  
                    AND I.Profesor = 'C-34-TU-00') ;
```

## Condición de reunión compuesta

- Mostrar el código de los alumnos que reciben clase del profesor de código C-34-TU-00

```
SELECT M.Alumno "Codigo Alumno"  
FROM Matricular M NATURAL JOIN Impartir I  
WHERE I.Profesor = 'C-34-TU-00';
```

Codigo Alumno
12312342
25748955
54354355

```
SELECT Alumno "Codigo Alumno" FROM Matricular M  
WHERE EXISTS (SELECT *  
               FROM Impartir I  
               WHERE I.Asignatura=M.Asignatura  
                     AND I.Grupo=M.Grupo  
                     AND ICurso=M.Curso  
                     AND I.Profesor = 'C-34-TU-00') ;
```

Alumnos matriculados en asignaturas en las que existe un profesor que las imparte  
(I.Asignatura=M.Asignatura & I.curso=M.Curso & I.Grupo=M.Grupo) con código dado

## Consultas Negativas

- El operador NOT en combinación con los operadores IN y EXISTS se puede utilizar para comprobar exhaustivamente que en todas las tuplas no se da una condición.
- *elem NOT IN conjunto\_tuplas*
  - La expresión es cierta si el elemento no pertenece al conjunto. Es decir, el elemento no es ninguno de los que aparecen en el conjunto.
- *NOT EXISTS conjunto\_tuplas*
  - La expresión es cierta si el conjunto de tupas está vacío. Es decir, no existe elemento en el conjunto de tuplas.



## Consultas Negativas

- Mostrar los datos de los profesores que no imparten la asignatura 112.

```
SELECT * FROM PROFESORES  
WHERE id IN (SELECT PROFESOR FROM IMPARTIR  
WHERE ASIGNATURA != 112);
```

```
SELECT profesor  
FROM impartir  
WHERE asignatura = 112;
```

PROFESOR
A-89-CEU-99
C-34-TU-00
G-34-TEU-96

ID	NOMBRE	APELLIDO1
A-89-CEU-99	Manuel	Enciso
C-34-TU-00	Enrique	Soler
C-67-CEU-98	Miguel	Ortiz
D-92-PC-92	Maria del Mar	Roldán
F-12-AY-02	Sergio	Gálvez
G-34-TEU-96	Ana	Jiménez
H-32-TU-99	Miguel	Hermoso
H-77-TU-99	Juana	Hernandez

¡¡ Salen profesores que no deberían salir!!  
Son aquellos que imparten otra asignatura  
además de la 112

- Condición positiva: una tupla tiene un valor diferente al dado.

## Consultas negativas

- Mostrar los datos de los profesores que no imparten la asignatura 112.

```
SELECT * FROM profesores  
WHERE Id NOT IN (SELECT Profesor FROM impartir  
                WHERE Asignatura = 112);
```

```
SELECT * FROM profesores P  
WHERE NOT EXISTS  
      (SELECT Profesor FROM impartir I  
       WHERE I.profesor = P.Id  
             AND I.Asignatura = 112);
```

Profesores para los que no existe un profesor que imparte clase en la asignatura 112 que tiene su mismo código (I.profesor = P.Id)

## Ejercicios

- Mostrar ...
  1. Código de las asignaturas que imparten los profesores que imparten la asignatura 112. Se deben eliminar las repeticiones.
  2. Nombre de los departamentos que no tienen ninguna asignatura con más de 6 créditos.
  3. Dni de alumnos matriculados en alguna asignatura impartida por el profesor de mayor antigüedad.
  4. Mostrar las parejas de profesores que no tienen ningún alumno en común.