
7.- SUBCONSULTAS MULTI_COLUMNNA

- 7.1.- Definición.
- 7.2.- Comparación entre columnas.
- ~~7.3-~~ Correspondencia entre JOINS y subconsultas.

7.1.- DEFINICIÓN

Hasta ahora se han creado subconsultas mono_registro y multi_registro, en las que sólo una columna era comparada en la cláusula WHERE o cláusula HAVING de la sentencia SELECT. Si se quiere comparar dos o más columnas habrá que escribir en la cláusula WHERE compuesta, la cual utiliza operadores lógicos. Las subconsultas multi_columna, permitirán transformar diferentes condiciones WHERE en una única cláusula WHERE.

Sintaxis:

```
SELECT column1, column2,...  
FROM tabla  
WHERE (column1, column2 ... ) IN ( SELECT column1, column2,...  
                                FROM tabla  
                                WHERE condición);
```

Ejemplo:

Visualizar el nombre, el número de departamento, el salario y la comisión de cualquier empleado cuyo salario y comisión se corresponden ambos con el salario y la comisión de cualquier empleado del departamento 30.

```
SELECT ENAME,DEPTNO, SAL, COMM  
FROM EMP  
WHERE (SAL, NVL(COMM,0)) IN (SELECT SAL, NVL(COMM,0)  
                           FROM EMP  
                           WHERE DEPTNO = 30);
```

Las subconsultas ignoran los nulos pues, no tienen con qué compararlos.

7.2.- COMPARACIÓN ENTRE COLUMNAS.

- **PAIRWISE:** consiste en comparar simultáneamente en la WHERE las columnas que están antes del operador con las que están en la subconsulta.

Ejemplo:

```
SELECT ENAME,DEPTNO, SAL, COMM
FROM EMP
WHERE (SAL, JOB) IN (SELECT SAL, JOB
                     FROM EMP
                     WHERE DEPTNO = 30);
```

- **NONPAIRWISE:** compara una columna que está antes del operador con otra que está dentro de la SUBQUERY y así sucesivamente.

Ejemplo:

```
SELECT ENAME,DEPTNO, SAL, COMM
FROM EMP
WHERE SAL IN          (SELECT SAL
                       FROM EMP
                       WHERE DEPTNO = 30)

AND

JOB IN               (SELECT JOB
                     FROM EMP
                     WHERE DEPTNO = 30);
```

7.3- CORRESPONDENCIA ENTRE JOINS Y SUBCONSULTAS

Normalmente cualquier consulta entre varias tablas se puede obtener de dos formas, mediante un JOIN o mediante una subconsulta. Para que esta correspondencia pueda darse es necesario que exista el tipo de JOIN llamado EQUIJOIN, es decir, que las clave principal de una tabla sea clave ajena/s de otra/s tablas.

Se puede decir entonces que si existe una condición de EQUIJOIN, cualquier JOINS se puede expresar como una subconsulta y se obtendrá el mismo resultado en ambos casos.

Ejemplo:

A los departamentos que son directores en funciones se asignará una gratificación de 5% del salario. Hallar por orden alfabético los nombres de estos empleados y la gratificación correspondiente a cada uno de ellos.

Como subconsulta:

```
SELECT NOMEM, 0.05*SALAR
FROM TEMPLE
WHERE NUMEM IN (SELECT DIREC
                FROM TDEPTO
                WHERE TIDIR = 'F')
ORDER BY NOMEM;
```

Como join:

```
SELECT NOMEM, 0.05* SALAR
FROM TDEPTO D, TEMPLE E
WHERE D.DIREC= E.NUMEM AND TIDIR = 'F'
ORDER BY 1;
```