

Tarea 7_2

Investigar para las subconsultas (casos de uso, restricciones, ejemplos)

- **SELECT:** Dentro de la cláusula SELECT, el caso de uso más común es usar subconsultas para determinar los valores de las columnas que quieres que se desplieguen, de la forma:
`SELECT columna, (SELECT COUNT(*) FROM otra_tabla) AS total FROM mi_tabla;`
Es importante mencionar que las subconsultas en las cláusula SELECT deben devolver exactamente una columna, y que al trabajar con correlación se debe de tomar en cuenta el contexto de la consulta principal para adecuar correctamente a la subconsulta.
- **FROM:** Dentro de la cláusula FROM, el caso de uso más común es usar subconsultas para generar conjuntos de datos temporales que puedes usar en tu consulta principal. Esto se ve de la siguiente forma:
`SELECT columna FROM (SELECT columna FROM otra_tabla) AS subconsulta;`
Cabe mencionar que para al usar las subconsultas con FROM, el resultado debe regresar un número específico de columnas que coincida con la cantidad de columnas esperadas por la consulta principal, al igual que especificar correctamente el nombre de las columnas al trabajar con alias.
- **JOIN:** Dentro de la cláusula JOIN, el caso de uso más común es usar subconsultas para ubicar la información de otra tabla con la que quieres que se realicen las comparaciones. Esto se ve de la siguiente forma:
`SELECT columna FROM tabla1 JOIN (SELECT id, columna FROM tabla2 WHERE condicion) AS subconsulta ON tabla1.id = subconsulta.id;`
Cabe mencionar que JOIN puede ser sustituido por cualquiera de sus variantes (INNER JOIN, LEFT JOIN, RIGHT JOIN, SELF JOIN, etc).
Además de la anterior también es importante considerar los impactos en el rendimiento, y complejidad del código necesario para realizar las consultas.
- **WHERE:** Dentro de la cláusula WHERE, el caso de uso más común es usar subconsultas para filtrar resultados basados en valores calculados en otra consulta. Esto se ve la siguiente forma:
`SELECT columna FROM mi_tabla WHERE columna IN (SELECT columna FROM otra_tabla WHERE condicion);`
Cabe mencionar que las subconsultas que usen esta cláusula deberán regresar valores un solo valor o una lista de valores utilizando operadores aritméticos o la cláusula IN.
Igual vale la pena mencionar que se debe contemplar que se puede regresar el valor NULL como resultado de usar estos operadores.
- **HAVING:** Dentro de la cláusula HAVING el caso de uso más común, es usar subconsultas para aplicar condiciones a grupos de filas agregadas.
`SELECT columna, COUNT(*) FROM mi_tabla GROUP BY columna HAVING COUNT(*) > (SELECT valor FROM otra_tabla);`
Cabe mencionar que las subconsultas que usen esta cláusula deberán regresar valores un solo valor o una lista de valores utilizando operadores aritméticos o la cláusula IN.
Igual vale la pena mencionar que se debe contemplar que se puede regresar el valor NULL como resultado de usar estos operadores.

- **CORRELACIONADAS:** Este tipo de subconsulta es principalmente usada para filtrar resultados en función de condiciones en otra tabla, para hacer cálculos de agregación condicional en función de valores de la fila actual en la consulta principal, o para checar la existencia de registros en otra tabla. Ejemplo:

```
SELECT cliente_id,  
       (SELECT COUNT(*)  
        FROM ordenes  
        WHERE cliente_id = c.cliente_id AND fecha < '2023-01-01') AS  
num_ordenes_antes_2023  
FROM clientes c;
```

Como este tipo de subconsulta depende de la consulta principal, es importante considerar que espera recibir la consulta principal en ese apartado, para que de esta forma la adecuemos correctamente.

Bibliografía:

[1] "SQL Tutorial". W3Schools Online Web Tutorials. Accedido el 8 de noviembre de 2023. [En línea]. Disponible: <https://www.w3schools.com/sql/>

[2] "PostgreSQL: Documentation". PostgreSQL: The world's most advanced open source database. Accedido el 8 de noviembre de 2023. [En línea]. Disponible: <https://www.postgresql.org/docs/>