

Base de Datos (SQL)

Problema 1:

- I. De las 3 tablas anexas, escribe una consulta en SQL (Cualquier intérprete) que te dé los nombres de aquellos estudiantes cuyos amigos perciben un ingreso mayor que ellos. Los nombres deben de estar ordenados por el salario que ganan sus amigos.

```
Query  Query History
1  SELECT
2      E.Nombre AS Nombre_Estudiante,
3      S1.Salario AS Salario_Estudiante,
4      S2.Salario AS Salario_Amigo
5  FROM
6      Estudiantes E
7  JOIN
8      Salario S1 ON E.ID = S1.ID
9  JOIN
10     Amigos A ON E.ID = A.ID
11 JOIN
12     Salario S2 ON A.Amigos_ID = S2.ID
13 WHERE
14     S2.Salario > S1.Salario
15 ORDER BY
16     S2.Salario;
17 |
```

- E = tabla Estudiantes
- S1 = salario del estudiante (tabla Salario)
- S2 = salario del amigo (otra vez la tabla Salario, pero conectada al amigo)

Lógica paso a paso de la consulta:

1. Relacionar cada estudiante con su salario, estudiantes se une con Salario usando su ID (eso es S1).
2. Conectar al estudiante con sus amigos, esto usando la tabla Amigos, que dice quién es amigo de quién
3. Obtener el salario del amigo, se vuelve a usar la tabla Salario pero ahora con el Amigos_ID (eso es S2).
4. Comparar ambos salarios, se filtran los casos donde $S2.Salario > S1.Salario$, o sea, donde el amigo gana más que el estudiante.
5. Ordenar por salario del amigo.

- II. De las 3 tablas anexas, escribe una consulta que te dé el nombre del estudiante y el promedio del salario de sus amigos, pero solo para aquellos estudiantes cuyo promedio de salario de amigos sea mayor que el promedio de salario de todos los estudiantes. Los resultados deben estar ordenados por el promedio del salario de sus amigos en orden descendente.

```
Query Query History
1  -- calcula el promedio general de salario de todos los estudiantes
2  WITH PromedioGeneral AS (
3      SELECT AVG(Salario) AS Promedio
4      FROM Salario
5  ),
6
7  -- calcula el promedio del salario de los amigos por cada estudiante
8  PromedioSalarioAmigos AS (
9      SELECT
10         Estudiantes.ID,
11         Estudiantes.Nombre,
12         AVG(SalarioAmigo.Salario) AS Promedio_Amigos
13      FROM Estudiantes
14      JOIN Amigos ON Estudiantes.ID = Amigos.ID
15      JOIN Salario AS SalarioAmigo ON Amigos.Amigos_ID = SalarioAmigo.ID
16      GROUP BY Estudiantes.ID, Estudiantes.Nombre
17  )
18  |
19  -- filtra los estudiantes cuyo promedio de amigos > promedio general
20  SELECT
21      PromedioSalarioAmigos.Nombre,
22      PromedioSalarioAmigos.Promedio_Amigos
23  FROM
24      PromedioSalarioAmigos,
25      PromedioGeneral
26  WHERE
27      PromedioSalarioAmigos.Promedio_Amigos > PromedioGeneral.Promedio
28  ORDER BY
29      PromedioSalarioAmigos.Promedio_Amigos DESC;
30
```

Lógica paso a paso de la consulta:

1. Calcular el promedio de salario de todos los estudiantes, para así compararlo después con los promedios de los amigos.
2. Sacar el promedio del salario de los amigos de cada estudiante.
3. Comparar ambos promedios, para solo quedarnos con los estudiantes cuyo promedio de salario de amigos sea mayor que el promedio general.
4. Ordenar los resultados de mayor a menor. Mostramos primero a los que tienen amigos con salarios más altos.

Para resolver ambos problemas, fue necesario tener algo de contexto sobre consultas SQL básicas y cómo funcionan los joins, especialmente cuando se relacionan varias tablas entre sí. También consultas como AVG() para promedios, el uso de alias (como S1, S2 o SalarioAmigo), y cómo se pueden usar subconsultas o CTEs (WITH) para organizar mejor los pasos y hacer comparaciones.