

1. Subconsulta simple en WHERE (valor único)

Ejercicio: Mostrar a los empleados cuyo salario supere el salario promedio de todos los empleados.

1. Análisis:

- Primero, necesitamos calcular el salario promedio de todos los empleados.
- Ese valor (digamos 2,200 USD) se usará para filtrar a quienes tengan un salario mayor.

2. Subconsulta:

- Para obtener el promedio: `SELECT AVG(Salario) FROM Empleados`.
- Esa subconsulta devuelve un solo valor (ej. 2,200).

3. Consulta principal:

- Se filtra `WHERE Salario > (subconsulta)`.

SQL final:

```
SELECT EmpleadoID, Nombre, Salario
FROM Empleados
WHERE Salario > (
    SELECT AVG(Salario)
    FROM Empleados
);
```

4. Explicación:

- Paso 1: Evalúa la subconsulta `SELECT AVG(Salario) FROM Empleados`, obteniendo el promedio (por ejemplo, 2200).
 - Paso 2: La consulta principal filtra a los empleados con `Salario > 2200`.
-

2. Subconsulta con IN en WHERE (lista de valores)

Ejercicio: Mostrar a todos los empleados cuyos departamentos estén en las sucursales denominadas "Central" o "Norte". Supongamos que la tabla Departamentos tiene una columna Sucursal (con valores "Central", "Norte", "Sur", etc.).

1. Análisis:

- Necesitamos los DepartamentoID de todos los departamentos donde Sucursal = 'Central' o 'Norte'.
- Luego filtraremos Empleados que pertenezcan a dichos departamentos.

2. Subconsulta:

- `SELECT DepartamentoID FROM Departamentos WHERE Sucursal IN ('Central', 'Norte')`

3. Consulta principal:

- `SELECT ... FROM Empleados WHERE DepartamentoID IN (subconsulta).`

SQL final:

```
SELECT EmpleadoID, Nombre, DepartamentoID
FROM Empleados
WHERE DepartamentoID IN (
    SELECT DepartamentoID
    FROM Departamentos
    WHERE Sucursal IN ('Central', 'Norte')
);
```

4. Explicación:

- Paso 1: La subconsulta encuentra todos los DepartamentoID cuyas sucursales estén en "Central" o "Norte".
- Paso 2: La consulta externa filtra a los empleados que tengan DepartamentoID en esa lista devuelta.

3. Subconsulta en la cláusula SELECT (columna calculada)

Ejercicio: Para cada empleado, mostrar el total de ventas que ha realizado (asumiendo tabla Ventas con EmpleadoID, Cantidad, PrecioUnitario).

1. Análisis:

- Cada fila de la tabla principal (Empleados) quiere mostrar una columna adicional: la suma de (Cantidad * PrecioUnitario) de sus ventas en Ventas.

2. Subconsulta:

- `SELECT SUM(Cantidad * PrecioUnitario) FROM Ventas WHERE Ventas.EmpleadoID = Empleados.EmpleadoID`

3. Consulta principal:

- `SELECT EmpleadoID, (subconsulta) AS TotalVentas FROM Empleados;`

SQL final:

```
SELECT
  E.EmpleadoID,
  E.Nombre,
  (SELECT SUM(V.Cantidad * V.PrecioUnitario)
   FROM Ventas V
   WHERE V.EmpleadoID = E.EmpleadoID
  ) AS TotalVentas
FROM Empleados E;
```

4. Explicación:

- Por cada empleado en la salida, se ejecuta la subconsulta que suma las ventas de ese empleado.
 - El resultado se incluye como la columna TotalVentas.
-

4. Subconsulta en la cláusula FROM (tabla derivada)

Ejercicio: Listar los departamentos que tengan más de 5 empleados. Queremos filtrar después de agrupar, pero sin usar HAVING.

1. Análisis:

- Para saber cuántos empleados hay por departamento, normalmente haríamos GROUP BY DepartamentoID.
- Queremos que el resultado de ese agrupamiento sea tratado como si fuera una "tabla".

Subconsulta (como "tabla derivada"):

```
(  
  SELECT DepartamentoID, COUNT(*) AS NumEmpleados  
  FROM Empleados  
  GROUP BY DepartamentoID  
) AS DeptosCount
```

Consulta principal:

```
SELECT  
  DeptosCount.DepartamentoID,  
  DeptosCount.NumEmpleados  
FROM (  
  SELECT DepartamentoID, COUNT(*) AS NumEmpleados  
  FROM Empleados  
  GROUP BY DepartamentoID  
) AS DeptosCount  
WHERE DeptosCount.NumEmpleados > 5;
```

2. Explicación:

- Paso 1: la subconsulta agrupa los empleados y obtiene el conteo por departamento.

- Paso 2: se filtra el resultado (WHERE NumEmpleados > 5) en la consulta externa.
-

5. Subconsulta correlacionada en WHERE

Ejercicio: Mostrar los productos cuyo precio supere el promedio de precios de su propia categoría. Suponemos que la tabla Productos tiene las columnas ProductoID, Precio, CategoricalID.

1. Análisis:

- Cada producto se compara con el promedio de su categoría.
- Para ello, la subconsulta hace referencia a la fila externa con CategoricalID.

Subconsulta (correlacionada):

```
SELECT AVG(P2.Precio)
FROM Productos P2
WHERE P2.CategoricalID = P.CategoricalID
```

- Observa que P.CategoricalID es la columna de la consulta externa.

Consulta principal:

```
SELECT
  P.ProductoID,
  P.NombreProducto,
  P.Precio
FROM Productos P
WHERE P.Precio > (
  SELECT AVG(P2.Precio)
  FROM Productos P2
  WHERE P2.CategoricalID = P.CategoricalID
);
```

2. Explicación:

- La subconsulta se evalúa para cada producto, calculando el promedio de la categoría correspondiente.
 - Se devuelven solo los productos con precio superior a ese promedio.
-

6. Subconsulta correlacionada en SELECT

Ejercicio: Para cada empleado, mostrar el salario y el salario mínimo de su departamento. Se asume DepartamentoID en Empleados.

1. Análisis:

- Queremos en cada fila, además del salario del empleado, el salario mínimo (o cualquier otro agregado) del mismo departamento.

Subconsulta:

```
(SELECT MIN(E2.Salario)
FROM Empleados E2
WHERE E2.DepartamentoID = E.DepartamentoID)
```

Consulta principal:

```
SELECT
  E.EmpleadoID,
  E.Nombre,
  E.Salario,
  (SELECT MIN(E2.Salario)
   FROM Empleados E2
   WHERE E2.DepartamentoID = E.DepartamentoID
  ) AS MinSalarioDepto
FROM Empleados E;
```

2. Explicación:

- Al llamarse “correlacionada”, la subconsulta depende de E.DepartamentoID.
- Para cada fila (empleado), se busca el salario mínimo del departamento de ese empleado.

7. Subconsulta en HAVING

Ejercicio: Mostrar los departamentos que tengan un total de salario mayor que el promedio de los totales de salario de todos los departamentos.

1. Análisis:

- Primero agrupamos a los empleados por departamento para sumar salarios (ej. SUM(Salario)).
- Luego, queremos comparar cada suma con el promedio de esas sumas (esto requiere otra consulta).

2. Subconsulta:

Podría calcular el promedio de la suma de salarios por departamento:

```
SELECT AVG(SalTotal)
FROM (
  SELECT SUM(Salario) AS SalTotal
  FROM Empleados
  GROUP BY DepartamentoID
) AS T
```

Consulta principal:

```
SELECT DepartamentoID, SUM(Salario) AS TotalSalario
FROM Empleados
GROUP BY DepartamentoID
HAVING SUM(Salario) > (
    SELECT AVG(SalTotal)
    FROM (
        SELECT SUM(Salario) AS SalTotal
        FROM Empleados
        GROUP BY DepartamentoID
    ) AS T
);
```

3. Explicación:

- Paso 1: Agrupas y sumas por departamento.
- Paso 2: En el HAVING, filtras los grupos que superen el promedio de sumas de los departamentos.
- La subconsulta en HAVING vuelve a calcular el promedio de sumas, usando una sub-tabla derivada (AS T).

8. Subconsulta más compleja: Filtrar empleados que vendieron más que su propio promedio mensual

Escenario: Queremos ver empleados cuyo total de ventas en un mes supere el promedio de ese mismo mes.

- Tabla Ventas: (VentaID, EmpleadoID, FechaVenta, TotalVenta).
- Tabla Empleados: (EmpleadoID, Nombre, DepartamentoID, ...).

Paso 1: Subconsulta por mes y empleado

```
SELECT
    EmpleadoID,
    MONTH(FechaVenta) AS Mes,
    SUM(Cantidad * PrecioUnitario) AS VentasMes
FROM Ventas
GROUP BY EmpleadoID, MONTH(FechaVenta)
```

- Obtiene la venta total de cada empleado por cada mes.

Paso 2: Filtrar usando tabla derivada y subconsulta correlacionada (opcional)

Podríamos:

1. Crear una tabla derivada T con (EmpleadoID, Mes, VentasMes).
2. Usar otra subconsulta para encontrar el promedio de VentasMes dentro de ese mismo mes.

```
SELECT
    T.EmpleadoID,
    T.Mes,
    T.VentasMes
FROM (
    SELECT
        EmpleadoID,
        MONTH(FechaVenta) AS Mes,
        SUM(Cantidad * PrecioUnitario) AS VentasMes
    FROM Ventas
    GROUP BY EmpleadoID, MONTH(FechaVenta)
) AS T
WHERE T.VentasMes > (
    SELECT AVG(T2.VentasMes)
    FROM (
        SELECT
            EmpleadoID,
            MONTH(FechaVenta) AS Mes,
```

```
SUM(Cantidad * PrecioUnitario) AS VentasMes
FROM Ventas
GROUP BY EmpleadoID, MONTH(FechaVenta)
) AS T2
WHERE T2.Mes = T.Mes
);
```

- Explicación:
 - La subconsulta interna (T2) calcula el promedio de VentasMes para todos los empleados de ese Mes.
 - Se filtra WHERE T2.Mes = T.Mes, correlacionándolo con el mes de la tabla derivada externa (T).
 - Finalmente, seleccionamos solo quienes vendieron más que ese promedio mensual.