

seccion-2-prueba-practica-sql

October 26, 2025

Dada la siguiente tabla empleados:

[]:	<code>id</code>	nombre	departamento	salario	fecha_contratacion
1	Ana	Ventas	5000	2020-01-15	
2	Luis	IT	6000	2019-08-21	
3	María	Ventas	5500	2021-05-10	
4	Juan	IT	7000	2018-11-03	

1. Escriba una consulta SQL para obtener el promedio de salario por departamento, excluyendo empleados contratados después del 1 de enero de 2020.

```
[ ]: SELECT departamento, avg(salario) as salario_prom
      FROM empleados
      WHERE fecha_contratacion <= '2020-01-01'
      GROUP BY 1;
```

Tienes las siguientes tablas en una base de datos:

```
[ ]: clientes
      id (INT, PRIMARY KEY)
      nombre (VARCHAR)
      apellido (VARCHAR)

      ventas
      id (INT, PRIMARY KEY)
      cliente_id (INT, FOREIGN KEY a clientes.id)
      producto (VARCHAR)
      fecha (DATE)
      monto (DECIMAL)
```

2. Escribe una consulta para obtener los 5 clientes con mayor monto total de ventas en los últimos 6 meses.

```
[ ]: SELECT c.id, CONCAT(c.nombre, " ", c.apellido) as nombre_completo, sum(v.monto) as total_ventas
      FROM clientes c
      INNER JOIN ventas v
      ON c.id = v.cliente_id
      WHERE v.fecha >= date_sub(current_date(), interval 6 month)
```

```
GROUP BY 1,2  
ORDER BY total_ventas desc  
LIMIT 5;
```

3. Escribe una consulta para calcular el ticket promedio de ventas por cliente en el último año.

```
[ ]: SELECT c.id, CONCAT(c.nombre, " ", c.apellido) as nombre_completo, ROUND(AVG(v.  
    monto), 2) as ticket_promedio  
FROM clientes c  
INNER JOIN ventas v  
ON c.id = v.cliente_id  
WHERE v.fecha >= date_sub(current_date(), interval 1 year)  
GROUP BY 1,2  
ORDER BY ticket_promedio desc;
```

4. Escribe una consulta para obtener el nombre completo de los clientes y su monto total de ventas.

```
[ ]: SELECT c.id, CONCAT(c.nombre, ' ', c.apellido) as nombre_completo, SUM(v.monto) as  
    total_ventas  
FROM clientes c  
LEFT JOIN ventas v  
ON c.id = v.cliente_id  
GROUP BY 1,2  
ORDER BY total_ventas desc;
```

5. Escribe una consulta para obtener el ingreso promedio de ventas por mes.

```
[ ]: SELECT EXTRACT(year FROM v.fecha) as año, EXTRACT(month FROM v.fecha) as mes,  
    ROUND(AVG(v.monto), 2) as ingreso_promedio  
FROM ventas v  
GROUP BY 1,2  
ORDER BY año desc, mes desc;
```

6. Escribe una consulta para calcular el ranking de clientes por ventas en el último año.

```
[ ]: SELECT c.id, CONCAT(c.nombre, ' ', c.apellido) as nombre_completo, SUM(v.monto) as  
    total_ventas,  
rank() over (order by sum(v.monto) desc) as ranking  
FROM clientes c  
INNER JOIN ventas v  
ON c.id = v.cliente_id  
WHERE v.fecha >= date_sub(current_date(), interval 1 year)  
GROUP BY 1,2  
ORDER BY total_ventas desc;
```

7. Escribe una consulta para calcular el total de ventas por cliente y luego selecciona solo los clientes cuyo total de ventas es superior al promedio general.

```
[ ]: WITH promedio_general AS (
    SELECT ROUND(AVG(monto), 2) as promedio_total
    FROM ventas
),
ventas_por_cliente AS (
    SELECT c.id, CONCAT(c.nombre, ' ', c.apellido) as nombre_completo, SUM(v.monto) as total_ventas_cliente
    FROM clientes c
    INNER JOIN ventas v
    ON c.id = v.cliente_id
    GROUP BY 1,2
)
SELECT vp.id, vp.nombre_completo, vp.total_ventas_cliente
FROM ventas_por_cliente vp
CROSS JOIN promedio_general pg
WHERE vp.total_ventas_cliente > pg.promedio_total
ORDER BY vp.total_ventas_cliente desc;
```