

Taller SQL: El ultimo Salto

Bienvenida/o al taller SQL práctico de "El Último Salto", una tienda especializada en paracaidismo y deportes extremos ubicada en Pasto, Nariño.

Productos principales: Paracaídas (militares, deportivos, emergencia) y accesorios de seguridad (cascos, altímetros, trajes de vuelo, gafas).

Lógica especial: Existe un cupón llamado "ULTIMO_SUSPIRO" que ofrece un **30% de descuento** exclusivamente a clientes mayores de 60 años (nuestros "saltadores de oro").

Base de datos: tienda.db contiene 4 tablas normalizadas con datos sintéticos de aproximadamente 1,000+ transacciones de venta en los últimos 2 años.

Requisitos Generales

La solución del taller debe ser entregada en un repositorio de github público. (debes crear un repositorio y ahí guardar la solución de los ejercicios) Ejemplo: <https://youtu.be/3WoVPWubrnY?si=Vt-bh9ZLySaKgHlR>

Organización de carpetas: Crea un repositorio Git con la estructura de carpetas y archivos que mejor te parezca. Busca mostrar aquí una habilidad de organización, orden y calidad en el entregable. Se profesional.



Por cada ejercicio:

- Crea un archivo XX_ejercicio.sql donde XX es el número (01, 02, etc.)
- Asegúrate de que el SQL es ejecutable directamente contra tienda.db
- Incluye comentarios explicativos si tu solución es compleja
- El resultado debe responder la pregunta de negocio

Documentación:

En tu README.md, explica brevemente cómo ejecutar las soluciones. Menciona cualquier supuesto o interpretación que hayas hecho. **Deja tus datos visibles para que sepa de quien es la solución.**

NOTA: Envía tu repositorio de solución de este taller a andresinsuastyd10@gmail.com

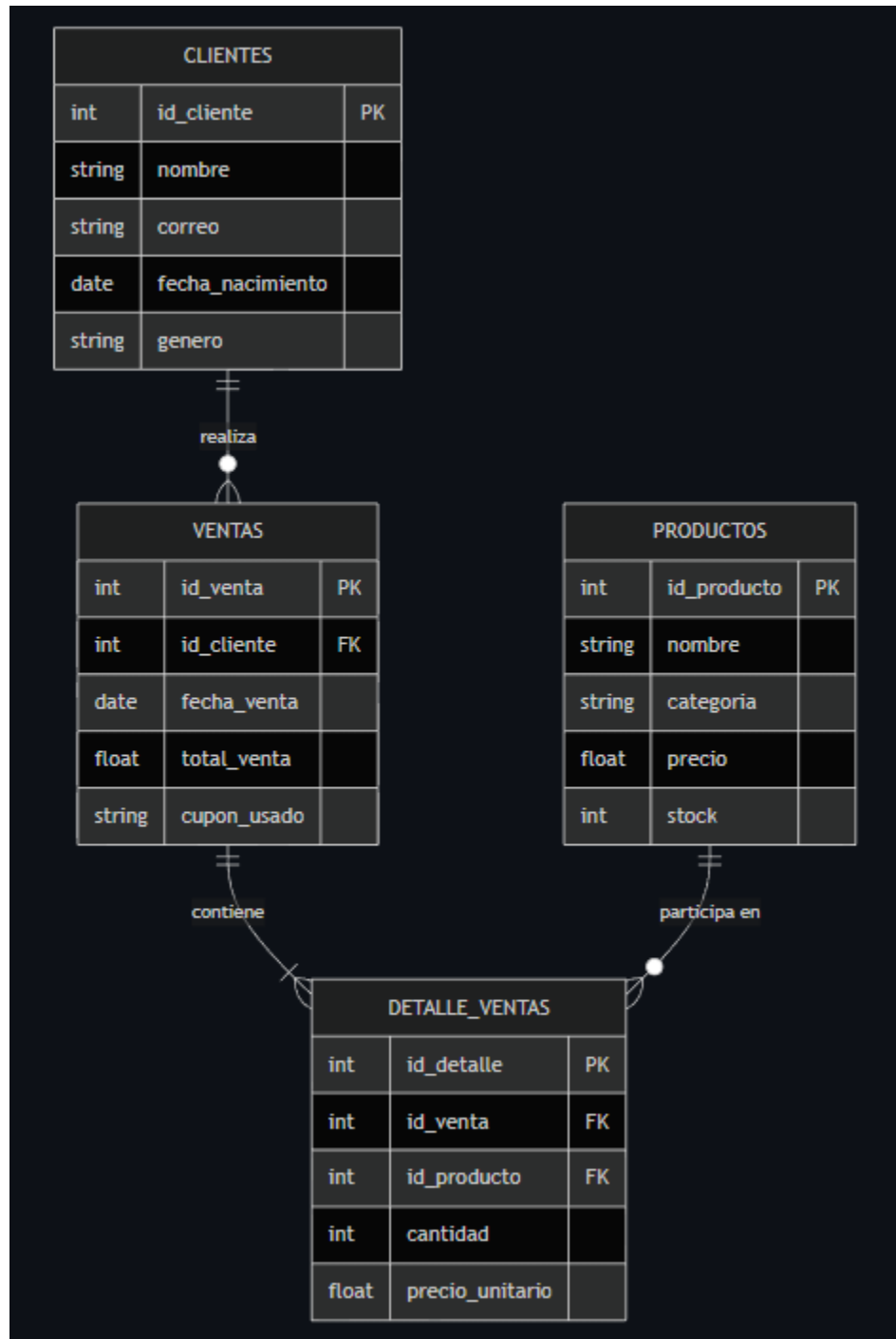
Con asunto: **CESMAG-ENTREGA 1.**

Se entrega de forma individual

Fecha límite: 11:59pm del 26 de Febrero del 2026. (10 días de plazo)

Diagrama Entidad Relación:

Te recomiendo hacer una exploración inicial de la base de datos, de las tablas y como se ven los registros, trata de conocer los datos antes de irte de lleno a solucionar los ejercicios.



- **Ejercicio 1** ¿Cuántos clientes tenemos registrados en total?
Pista: COUNT()*
- **Ejercicio 2** Enlista todos los productos de la categoría "Paracaídas" con sus precios, ordenados de mayor a menor precio.
Pista: Filtra por categoría y ordena descendente.
- **Ejercicio 3** ¿Cuál es el paracaídas más caro que tenemos?
Pista: MAX() + WHERE para filtrar categoría.
- **Ejercicio 4** ¿Cuántas transacciones de venta hemos registrado?
Pista: COUNT() sobre la tabla ventas.
- **Ejercicio 5** Muestra el nombre, correo y fecha de nacimiento de todos los clientes, ordenados alfabéticamente por nombre.
Pista: SELECT simple con ORDER BY.
- **Ejercicio 6** ¿Cuál fue el monto total de ventas en toda la historia?
Pista: SUM(total_venta) desde la tabla ventas.
- **Ejercicio 7** Enlista todos los productos de "Accesorios" con precio menor a \$500.
Pista: WHERE con múltiples condiciones.
- **Ejercicio 8** ¿Cuántas ventas utilizaron el cupón "ULTIMO_SUSPIRO"?
Pista: COUNT() con WHERE para el cupón específico.
- **Ejercicio 9** ¿Cuál fue la venta con mayor monto en toda la historia?
Pista: MAX(total_venta).
- **Ejercicio 10** Muestra todos los clientes de género "F" (femenino) ordenados por fecha de nacimiento (más viejitas primero).
Pista: WHERE genero = 'F' y ORDER BY fecha_nacimiento ASC.
- **Ejercicio 11** ¿Número total de líneas (detalles) de venta registradas?
Pista: COUNT() sobre detalle_ventas.*
- **Ejercicio 12** ¿Cuál es el precio promedio de los productos en stock?
Pista: AVG(precio) desde productos.
- **Ejercicio 13** ¿Cuántos productos tenemos en la categoría "Paracaídas"?
Pista: COUNT() + WHERE categoria.
- **Ejercicio 14** Muestra todas las ventas realizadas en el año 2025, ordenadas por fecha (más recientes primero).
Pista: WHERE + YEAR(), ORDER BY DESC.
- **Ejercicio 15** ¿Cuál es el accesorio más barato disponible?
Pista: MIN(precio) + WHERE categoria = 'Accesorios'.
- **Ejercicio 16** ¿Cuál es el cliente que más dinero ha gastado en total? Muestra su nombre y monto total.
Pista: SUM(total_venta), GROUP BY id_cliente, JOIN con clientes, ORDER BY y LIMIT.
- **Ejercicio 17** ¿Cuánto dinero hemos dejado de ingresar por el cupón "ULTIMO_SUSPIRO"? (Calcula la diferencia entre lo que hubiera sido sin descuento y lo que fue con descuento).
Pista: Calcula el total con descuento y sin descuento. Usa subconsultas o CTEs.
Consideración: Si el descuento es 30%, significa 70% del precio original. Invierte: monto_actual / 0.70 - monto_actual.
- **Ejercicio 18** Enlista todos los clientes mayores de 60 años que han hecho compras, mostrando nombre, edad aproximada y total gastado.
Pista: DATEDIFF o CAST(strftime() ...) para calcular edad. JOIN con ventas, GROUP BY, HAVING.

- **Ejercicio 19** ¿Cuál es el producto más vendido en cantidad (en términos de unidades, no dinero)?
Pista: SUM(cantidad) sobre detalle_ventas, GROUP BY producto, JOIN con productos.
- **Ejercicio 20** Para cada categoría de producto, calcula el ingreso total, cantidad de unidades vendidas y ticket promedio por venta.
Pista: GROUP BY categoría, SUM(total_venta), COUNT(), AVG().*
- **Ejercicio 21** ¿Cuáles son los clientes que han usado el cupón "ULTIMO_SUSPIRO" al menos 3 veces?
Pista: COUNT() con HAVING para filtrar grupos.
- **Ejercicio 22** Crea una clasificación de clientes por nivel de gasto:
 - "Alto Valor" si gastó más de \$5,000
 - "Medio Valor" si gastó entre \$2,000 y \$5,000
 - "Bajo Valor" si gastó menos de \$2,000
 - Muestra nombre, categoría de gasto y total gastado.*Pista: GROUP BY + CASE WHEN para clasificación.*
- **Ejercicio 23** ¿Cuáles son los 5 productos con mayor ingresos acumulado?
*Pista: JOIN detalle_ventas con productos, SUM(cantidad * precio_unitario), GROUP BY, ORDER BY DESC, LIMIT 5.*
- **Ejercicio 24** Para cada mes en 2025, calcula: total de ventas, número de transacciones, y ticket promedio.
Pista: strftime('%m', fecha_venta) + GROUP BY mes, AVG(total_venta).
- **Ejercicio 25** ¿Cuál es el cliente que compró la mayor variedad de productos diferentes (no cantidad, sino tipos distintos)?
Pista: COUNT(DISTINCT id_producto), GROUP BY id_cliente, ORDER BY DESC, LIMIT 1.
- **Ejercicio 26** Identifica clientes que compraron SOLO paracaídas (nunca compraron accesorios). Muestra nombre y total gastado.
Pista: Subconsulta o CTE para filtrar clientes cuyas compras contienen SOLO la categoría "Paracaídas".
- **Ejercicio 27** ¿Cuál es la diferencia en ingresos entre el mes con mayor venta y el mes con menor venta en 2025?
Pista: CTE para calcular ingresos por mes, luego MAX() - MIN().
- **Ejercicio 28** Crea un reporte de "clientes de riesgo": aquellos cuya última compra fue hace más de 180 días a partir de hoy. Muestra nombre, correo y fecha de última compra.
Pista: MAX(fecha_venta) por cliente, HAVING MAX(fecha_venta) < DATE('now', '-180 days').
- **Ejercicio 29** ¿Cuál es el porcentaje de ventas (por ingresos) que provienen del cupón "ULTIMO_SUSPIRO"?
*Pista: SUM(total_venta) con y sin cupón, luego calcula (ventas_con_cupón / total) * 100.*
- **Ejercicio 30** Genera un análisis demográfico: para cada rango de edad (18-29, 30-39, ..., 70+), cuenta clientes únicos, número de transacciones e ingresos totales. Ordena por rango de edad.
Pista: CASE WHEN para rangos de edad, GROUP BY rango, múltiples agregaciones.

Consejos y Buenas Prácticas

1. **Lee bien la pregunta:** Asegúrate de entender qué se pide antes de escribir SQL.
2. **Visualiza el flujo:** Dibuja (mentalmente o en papel) cómo se relacionan las tablas que vas a trabajar
3. **Comenta código complejo:** Si tu SQL tiene varias condiciones, explica por qué.
4. **Valida resultados:** ¿Tiene sentido el resultado? ¿Son cifras razonables?

Rubrica de Evaluación (orientativa)

- **Compleitud:** TODOS los ejercicios deben tener una propuesta de solución.
- **Aleatoriedad:** Se elige de manera aleatoria 5 ejercicios y se evalúa la solución. En base a esa exploración se entrega un punto por cada ejercicio bien hecho.
- **Claridad:** ¿El código es legible con nombres sensatos?
- **Documentación:** ¿Incluyes comentarios si es necesario?
- **Organización Git:** ¿El repo está bien estructurado y los commits son coherentes?
- **Readme:** Esta bien documentado tu repositorio?

Preguntas Frecuentes

P: ¿Puedo modificar la base de datos?

R: No. Trabaja solo con SELECT. Los datos son de lectura.

P: ¿Necesito optimizar los SQL?

R: No es obligatorio.

P: ¿Solo existe una solución para cada ejercicio?

R: No, lo más seguro es que haya varias formas de ser resuelto cada ejercicio.

P: ¿Hay restricciones de herramientas?

R: No, puedes apoyarte de herramientas que creas te puedan ayudar a estudiar y solucionar el taller.

P: ¿El taller se hace en una hora?

R: No. Tomará más tiempo. Recomendando no dejarlo para el final de la fecha límite. Es un taller digno de un futuro especialista.

Muchos éxitos!!