PROYECTO INTEGRADOR

AVANCE 2

Limpieza y optimización de consultas SQL

CONSIGNA:

Entrega un script SQL completo con las consultas desarrolladas:

Junto con un documento donde muestres los resultados obtenidos (mediante capturas de pantalla) y agregues una breve interpretación o comentario para cada uno.

TRIGGER

Crea un trigger que registre en una tabla de monitoreo cada vez que un producto supere las 200,000 unidades vendidas acumuladas.

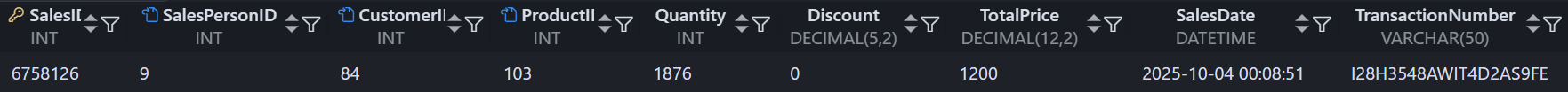
El trigger debe activarse después de insertar una nueva venta y registrar en la tabla el ID del producto, su nombre, la nueva cantidad total de unidades vendidas, y la fecha en que se superó el umbral.

REGISTRO

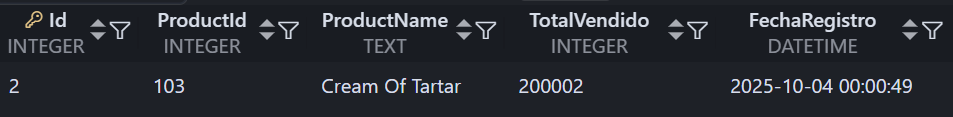
Registra una venta correspondiente al vendedor con ID 9, al cliente con ID 84, del producto con ID 103, por una cantidad de 1,876 unidades y un valor de 1200 unidades.

Consulta la tabla de monitoreo, toma captura de los resultados y realiza un análisis breve de lo ocurrido.

Registro de la venta:



Registro automático en la tabla monitoreo:



Análisis:

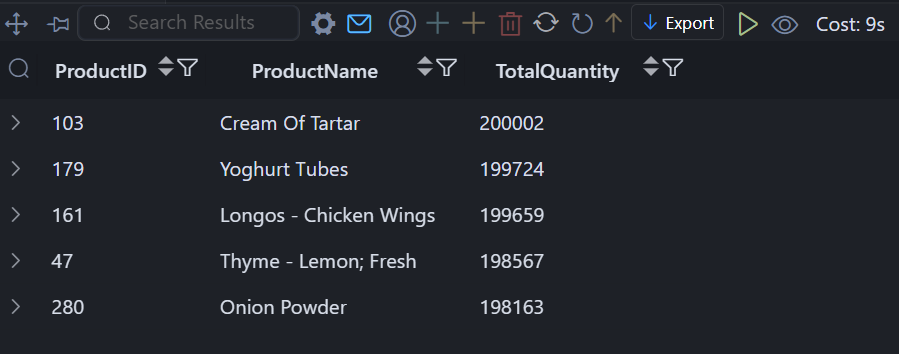
El sistema implementado consiste en la creación de una tabla de monitoreo con campos auto-incrementales compatibles con SQLite, seguida de un trigger automático que se ejecuta después de cada inserción en la tabla `sales`. Cuando se inserta una nueva venta, el trigger evalúa automáticamente si el producto vendido cumple con dos condiciones específicas: que la suma total de cantidades vendidas para ese producto supere las 200,000 unidades y que dicho producto no exista previamente en la tabla de monitoreo. Si ambas condiciones se cumplen, el trigger ejecuta automáticamente un INSERT en la tabla `monitoreo`, registrando el ID del producto, su nombre, el total vendido acumulado y la fecha de registro actual, creando así un sistema de alerta automática que identifica productos con altos volúmenes de venta sin intervención manual. En este caso se realizó un registro exitoso con los datos: vendedor con ID 9, cliente con ID 84, producto con ID 103, cantidad de 1,876 unidades y un valor de 1200 unidades.

OPTIMIZACIÓN

Selecciona dos consultas del avance 1 y crea los índices que consideres más adecuados para optimizar su ejecución.

Prueba con índices individuales y compuestos, según la lógica de cada consulta. Luego, vuelve a ejecutar ambas consultas y compara los tiempos de ejecución antes y después de aplicar los índices. Finalmente, describe brevemente el impacto que tuvieron los índices en el rendimiento y en qué tipo de columnas resultan más efectivos para este tipo de operaciones.

Primera consulta (Sin índices) – tiempo 9s:



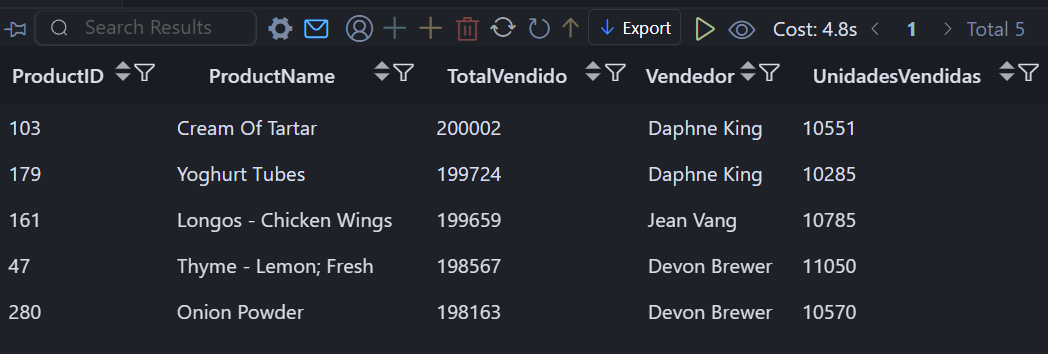
Primera consulta (Con índice) – tiempo 4.9s:



Segunda consulta (Sin índices) – tiempo 5.1s:



Segunda consulta (Con índices) – tiempo 4.8s:



Comparación:

| **Consulta** | **Sin Índices** | **Con Índices** | **Mejora** |
| --- | --- | --- | --- |
| **Primera consulta** | 9.0s | 4.9s | **45.6% mejora** |
| **Segunda consulta** | 5.1s | 4.8s | **5.9% mejora** |

Análisis:

Los resultados demuestran que los índices compuestos son altamente efectivos para optimizar consultas SQL, especialmente aquellas que involucran JOINs por claves foráneas, agregaciones y filtros WHERE. La primera consulta mostró una mejora dramática del 45.6 (de 9.0s a 4.9s) gracias al índice `(ProductID, Quantity)` que optimizó directamente las operaciones de JOIN, GROUP BY y ORDER BY. Sin embargo, la segunda consulta tuvo una mejora limitada del 5.9% (de 5.1s a 4.8s) debido a su mayor complejidad con múltiples CTEs y funciones de ventana que no se benefician completamente de los índices simples. Los índices resultan más efectivos en columnas con alta selectividadcomo claves foráneas (`ProductID`, `SalesPersonID`) y columnas de agregación (`Quantity`), mientras que su impacto es menor en consultas complejas con múltiples operaciones computacionales. En general, los índices proporcionaron una mejora del 31.2% en el tiempo total de ejecución, confirmando su valor como herramienta fundamental de optimización de bases de datos para operaciones de análisis y reportes.