

Proyecto Integrador II

Script

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.

Evidencia:

Tabla de monitoreo

	A-Z Field ▼	A-Z Type ▼	A-Z Null ▼	A-Z Key ▼	A-Z Default ▼	A-Z Extra ▼
1	ID	int	NO	PRI	[NULL]	auto_increment
2	ProductID	int	YES		[NULL]	
3	ProductName	varchar(255)	YES		[NULL]	
4	Unidades_Vendidas	int	YES		[NULL]	
5	Fecha	datetime	YES		[NULL]	

Trigger de monitoreo de unidades vendidas

Propiedades

Trigger Name: monitoreo_unidades_vendidas
Timing: AFTER
Type: INSERT
Table: sales_company.sales
Client Charset: utf8mb4
SQL Mode: ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO
Trigger Description:

Source

```

CREATE DEFINER=`root`@`localhost` TRIGGER `monitoreo_unidades_vendidas` AFTER INSERT ON `sales` FOR EACH ROW BEGIN
    DECLARE total INT;
    SELECT SUM(Quantity) INTO total FROM sales WHERE ProductID = NEW.ProductID;

    IF total > 200000 AND NOT EXISTS (
        SELECT 1 FROM monitoreo WHERE ProductID = NEW.ProductID
    ) THEN
        INSERT INTO monitoreo (ProductID, ProductName, Unidades_Vendidas, Fecha)
        SELECT
            p.ProductID,
            p.ProductName,
            total,
            NOW()
        FROM products p
        WHERE p.ProductID = NEW.ProductID;
    END IF;
END

```

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.

Evidencia:

Nuevo registro en la tabla de sales

	123 SalesID	123 SalesPersonID	123 CustomerID	123 ProductID	123 Quantity	123 Discoun	123 TotalPrice	🕒 SalesDate	A-Z TransactionNumber
1	6,758,126	9	84	103	1,876	0	1,200	2025-06-07 12:00:51	TX-001

Nuevo registro en la tabla de monitoreo ya que el producto “cream of tartar” ha llegado a 200.002 unidades vendidas.

	123 ID	123 ProductID	A-Z ProductName	123 Unidades_Vendidas	🕒 Fecha
1	2	103	Cream Of Tartar	200,002	2025-06-07 12:00:51

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.

Evidencia:

Creo indice ix_sales_products_person

Propiedades

Index Name: ix_sales_products_person

Index Type: BTree

☐ Unique

Comment:

Column: ProductID, SalesPersonID, Quantity

Table: sales_company.sales

Cardinality: 453

Additional:

Name	Column	Ascending	Nullable	Extra
123 ProductID	123 ProductID	[v]	[v]	
123 Quantity	123 Quantity	[v]	[v]	
123 SalesPersonID	123 SalesPersonID	[v]	[v]	

Primera consulta del Proyecto integrador I antes de crear el índice.

WITH top5 AS (
 SELECT
 s.ProductID,
 p.ProductName,
 SUM(s.Quantity) AS total_vendida
 FROM sales s
 JOIN products p ON s.ProductID = p.ProductID
 GROUP BY s.ProductID, p.ProductName
 ORDER BY total_vendida DESC
 LIMIT 5
)
ventas_vendedor AS (
 SELECT

sales(+) 1

WITH top5 AS (SELECT s.ProductID, p.Produ Enter a SQL expression to filter results (use Ctrl+Space)

	123 ProductID	A-Z ProductName	123 total_vendida	123 SalesPersonID	A-Z CONCAT (e.LastName, ' ', e.FirstName)	123 cantidad_vendida
1	103	Cream Of Tartar	200,002	9	King Daphne	10,551
2	179	Yoghurt Tubes	199,724	9	King Daphne	10,285
3	161	Longos - Chicken Wings	199,659	10	Vang Jean	10,785
4	47	Thyme - Lemon; Fresh	198,567	21	Brewer Devon	11,050
5	280	Onion Powder	198,163	21	Brewer Devon	10,570

Renovar Save Cancel Exportar datos ... 200 5 5 row(s) fetched - 9.058s, on 2025-06-07 at 12:15:00

Tiempo total 9,058 segundos

Misma consulta pero con el índice creado.

```

WITH top5 AS (
  SELECT
    s.ProductID,
    p.ProductName,
    SUM(s.Quantity) AS total_venta
  FROM sales s
  JOIN products p ON s.ProductID = p.ProductID
  GROUP BY s.ProductID, p.ProductName
  ORDER BY total_venta DESC
  LIMIT 5
),
ventas_vendedor AS (
  SELECT
    s.ProductID,
    s.SalesPersonID,
    SUM(s.Quantity) AS cantidad_venta
  FROM sales s
  JOIN top5 t ON s.ProductID = t.ProductID
  GROUP BY s.SalesPersonID
)
SELECT
  t.ProductID,
  t.ProductName,
  t.total_venta,
  v.SalesPersonID,
  CONCAT(e.LastName, ' ', e.FirstName) AS vendedor,
  v.cantidad_venta
FROM top5 t
JOIN ventas_vendedor v ON t.ProductID = v.ProductID
JOIN employees e ON v.SalesPersonID = e.SalesPersonID

```

ProductID	ProductName	total_venta	SalesPersonID	CONCAT (e.LastName, ' ', e.FirstName)	cantidad_venta
103	Cream Of Tartar	200,002	9	King Daphne	10,551
179	Yoghurt Tubes	199,724	9	King Daphne	10,285
161	Longos - Chicken Wings	199,659	10	Vang Jean	10,785
47	Thyme - Lemon; Fresh	198,567	21	Brewer Devon	11,050
280	Onion Powder	198,163	21	Brewer Devon	10,570

5 row(s) fetched - 6.013s, on 2025-06-07 at 12:18:27

Tiempo total 6,013 segundos

Segunda consulta del Proyecto integrador I antes de crear el índice.

```

WITH top5 AS (
  SELECT
    s.ProductID,
    p.ProductName,
    SUM(s.Quantity) AS total_venta
  FROM sales s
  JOIN products p ON s.ProductID = p.ProductID
  GROUP BY s.ProductID, p.ProductName
  ORDER BY total_venta DESC
  LIMIT 5
),
ventas_vendedor AS (
  SELECT
    s.ProductID,
    s.SalesPersonID,
    SUM(s.Quantity) AS cantidad_venta
  FROM sales s
  JOIN top5 t ON s.ProductID = t.ProductID
  GROUP BY s.SalesPersonID
)
SELECT
  t.ProductID,
  t.ProductName,
  t.total_venta,
  v.SalesPersonID,
  CONCAT(e.LastName, ' ', e.FirstName) AS vendedor,
  v.cantidad_venta / t.total_venta * 100 AS porcentaje_vendedor
FROM top5 t
JOIN ventas_vendedor v ON t.ProductID = v.ProductID
JOIN employees e ON v.SalesPersonID = e.SalesPersonID

```

ProductID	ProductName	total_venta	SalesPersonID	CONCAT (e.LastName, ' ', e.FirstName)	porcentaje_vendedor
103	Cream Of Tartar	200,002	9	King Daphne	5.28
179	Yoghurt Tubes	199,724	9	King Daphne	5.15
161	Longos - Chicken Wings	199,659	10	Vang Jean	5.4
47	Thyme - Lemon; Fresh	198,567	21	Brewer Devon	5.56
280	Onion Powder	198,163	21	Brewer Devon	5.33

5 row(s) fetched - 8.724s, on 2025-06-07 at 12:16:05

Tiempo total 8,724 segundos

Misma consulta pero con el índice creado.

```

WITH top5 AS (
  SELECT
    s.ProductID,
    p.ProductName,
    SUM(s.Quantity) AS total_venta
  FROM sales s
  JOIN products p ON s.ProductID = p.ProductID
  GROUP BY s.ProductID, p.ProductName
  ORDER BY total_venta DESC
  LIMIT 5
),
ventas_vendedor AS (
  SELECT
    s.ProductID,
    s.SalesPersonID,
    SUM(s.Quantity) AS cantidad_venta
  FROM sales s
  GROUP BY s.ProductID, s.SalesPersonID
)

```

ProductID	ProductName	CONCAT (e.LastName, ' ', e.FirstName)	porcentaje_vendedor
103	Cream Of Tartar	King Daphne	5.28
179	Yoghurt Tubes	King Daphne	5.15
161	Longos - Chicken Wings	Vang Jean	5.4
47	Thyme - Lemon; Fresh	Brewer Devon	5.56
280	Onion Powder	Brewer Devon	5.33

5 row(s) fetched - 5.999s, on 2025-06-07 at 12:19:13

Tiempo total 5,999 segundos

Conclusiones:

Consulta	Con índice	Sin índice	% Diferencia
Consulta 1	6.013	9.058	33.61
Consulta 2	5.999	8.724	31.23

La implementación de índices mejoró notablemente el rendimiento de ambas consultas:

- Consulta 1 mejoró un 33.6%.
- Consulta 2 mejoró un 31.2%.