

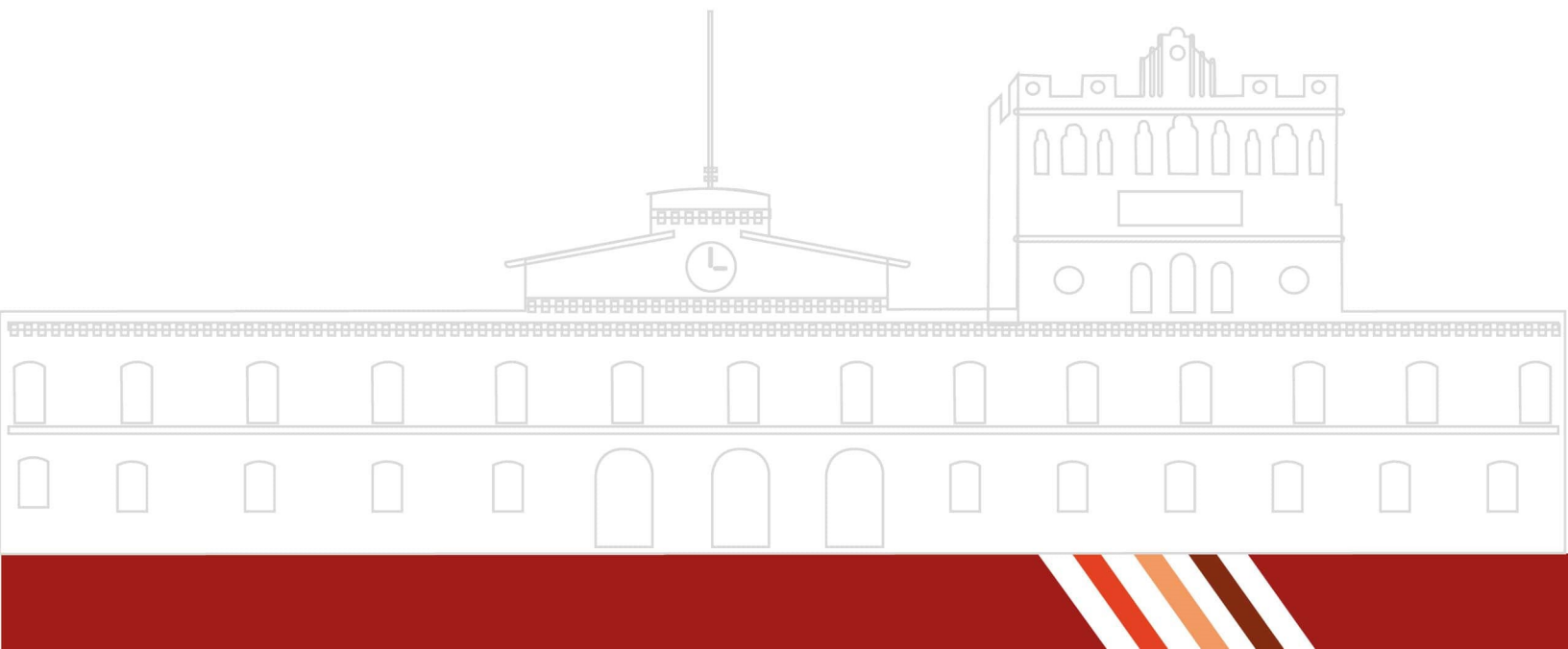
REPORTE DE PRÁCTICA NO. 3.1

NOMBRE DE LA PRÁCTICA: Consultas a la BD Distribuidora

ALUMNO: Hernández Sánchez Juan German

MATERIA: Base de Datos Distribuidas

DOCENTE: Eduardo Cornejo Velazquez



Introducción

Diseñar las sentencias de álgebra relacional y sentencias SQL para las siguientes consultas.

Reporte de compras del mes de enero que incluya el nombre del proveedor, el nombre de la herramienta, cantidad, precio unitario y precio total

ALGEBRA RELACIONAL

$ComprasEnEnero \leftarrow \sigma(mes(fecha_compra) = 1)(Compras)$
 $Resultado \leftarrow \pi(nombre_proveedor, nombre_herramienta, cantidad, precio_unitario, (cantidad * precio_unitario) \rightarrow precio_total)(ComprasEnEneroProveedoresHerramientas)$

SQL

SELECT s.name AS SupplierName, t.name AS ToolName, bt.amount AS Quantity, bt.unitPrice AS UnitPrice, (bt.amount * bt.unitPrice) AS TotalPrice FROM buyTool bt JOIN purchase p ON bt.idPurchase = p.idPurchase JOIN supplier s ON p.idSupplier = s.idSupplier JOIN inventory i ON bt.idInventory = i.idInventory JOIN tool t ON i.idTool = t.idTool WHERE MONTH(p.date) = 1;

Reporte de inventario de la bodega de la calle 13 que incluya el nombre de la herramienta, cantidad y costo total (precio de compra * cantidad)

ALGEBRA RELACIONAL

$InventarioCalle13 \leftarrow \sigma(calle = "13")(BodegasInventarios)$
 $Resultado \leftarrow \pi(nombre_herramienta, cantidad, (precio_compra * cantidad) \rightarrow costo_total)(InventarioCalle13)$

SQL

SELECT t.name AS ToolName, i.amount AS Quantity, (i.amount * i.purchasePrice) AS TotalCost FROM inventory i JOIN warehouse w ON i.idWarehouse = w.idWarehouse JOIN tool t ON i.idTool = t.idTool WHERE w.street = 'Calle 13';

Reporte de compras de la herramienta “Tijera de jardinero” durante el año 2023 el nombre del proveedor, fecha de compra, cantidad, precio unitario y costo total

ALGEBRA RELACIONAL

$ComprasTijera2023 \leftarrow \sigma(nombre_herramienta = "Tijeradejardinero" \wedge año(fecha_compra) = 2023)(ComprasHerramientas)$
 $Resultado \leftarrow \pi(nombre_proveedor, fecha_compra, cantidad, precio_unitario, (cantidad * precio_unitario) \rightarrow costo_total)(ComprasTijera2023)$

SQL

SELECT s.name AS SupplierName, p.date AS PurchaseDate, bt.amount AS Quantity, bt.unitPrice AS

UnitPrice, (bt.amount * bt.unitPrice) AS TotalCost FROM buyTool bt JOIN purchase p ON bt.idPurchase = p.idPurchase JOIN supplier s ON p.idSupplier = s.idSupplier JOIN inventory i ON bt.idInventory = i.idInventory JOIN tool t ON i.idTool = t.idTool

Listado de responsables de las bodegas de la empresa con calle, número y teléfono

ALGEBRA RELACIONAL

$Resultado \leftarrow \pi(calle, numero, nombre_responsable, telefono)(Bodegas)$

SQL

SELECT w.manager AS Responsable, w.street AS Street, w.number AS Number, w.phone AS Phone FROM warehouse w;

Listado de contactos con los proveedores con nombre de contacto, nombre de proveedor, teléfono y correo electrónico

ALGEBRA RELACIONAL

$Resultado \leftarrow \pi(nombre_contacto, nombre_proveedor, telefono, correo_electronico)(Proveedores)$

SQL

SELECT s.contact AS ContactName, s.name AS SupplierName, s.phone AS Phone, s.email AS Email FROM supplier s;

Reporte de herramientas compradas cuyo precio unitario se menor o igual a \$250.00 que incluya nombre de la herramienta, fecha de compra y cantidad comprada en orden cronológico descendiente.

ALGEBRA RELACIONAL

$HerramientasBaratas \leftarrow \sigma(precio_unitario \leq 250)(ComprasHerramientas) Resultado \leftarrow \pi(nombre_herramienta, fecha)$

SQL

SELECT t.name AS ToolName, p.date AS PurchaseDate, bt.amount AS Quantity FROM buyTool bt JOIN purchase p ON bt.idPurchase = p.idPurchase JOIN inventory i ON bt.idInventory = i.idInventory JOIN tool t ON i.idTool = t.idTool WHERE bt.unitPrice <= 250 ORDER BY p.date DESC;

Reporte de herramientas en el inventario cuyo stock sea entre 5 y 20 piezas que incluya calle y número de la bodega, nombre de la herramienta, ubicación y cantidad en existencia

ALGEBRA RELACIONAL

$$\text{InventarioFiltrado} \leftarrow \sigma(\text{cantidad} \geq 5 \wedge \text{cantidad} \leq 20)(\text{InventariosBodegas})$$
$$\text{Resultado} \leftarrow \pi(\text{calle}, \text{numero}, \text{nombre_herramienta})(\text{InventarioFiltrado})$$

SQL

```
SELECT w.street AS WarehouseStreet, w.number AS WarehouseNumber, t.name AS ToolName, i.location AS Location, i.amount AS Quantity FROM inventory i JOIN warehouse w ON i.idWarehouse = w.idWarehouse JOIN tool t ON i.idTool = t.idTool WHERE i.amount BETWEEN 5 AND 20;
```

Reporte del stock de todas las bodegas que incluya calle, número, responsable, teléfono y total de herramientas almacenadas

ALGEBRA RELACIONAL

$$\text{Resultado} \leftarrow \pi(\text{calle}, \text{numero}, \text{nombre_responsable}, \text{telefono}, \text{SUM}(\text{cantidad}) \rightarrow \text{total_herramientas})(\text{BodegasInventarios})$$
$$\text{GROUP BY } \text{calle}, \text{numero}, \text{nombre_responsable}, \text{telefono}$$

SQL

```
SELECT w.street AS Street, w.number AS Number, w.manager AS Responsible, w.phone AS Phone, SUM(i.amount) AS TotalStock FROM inventory i JOIN warehouse w ON i.idWarehouse = w.idWarehouse GROUP BY w.idWarehouse;
```

Reporte de valor de inventario de todas las bodegas que incluya calle, estado y monto total de precio de venta de las herramientas almacenadas

ALGEBRA RELACIONAL

$$\text{Resultado} \leftarrow \pi(\text{calle}, \text{estado}, \text{SUM}(\text{precio_venta} * \text{cantidad}) \rightarrow \text{monto_total})(\text{BodegasInventariosHerramientas})$$
$$\text{GROUP BY } \text{calle}, \text{estado}$$

SQL

```
SELECT w.street AS Street, w.state AS State, SUM(i.storeSalePrice * i.amount) AS TotalInventoryValue FROM inventory i JOIN warehouse w ON i.idWarehouse = w.idWarehouse GROUP BY w.idWarehouse;
```

Conclusión

Las consultas a una base de datos desempeñan un papel fundamental en el funcionamiento eficiente de una distribuidora, ya que permiten gestionar y analizar información clave para la toma de decisiones. Mediante la implementación de consultas bien estructuradas, es posible obtener datos precisos sobre inventarios, ventas, proveedores y clientes, lo que facilita la optimización de recursos, el control de productos y la mejora del servicio al cliente.

Además, el uso de consultas parametrizadas y procedimientos almacenados contribuye a la seguridad y el rendimiento del sistema, reduciendo riesgos como inyección SQL y mejorando la rapidez en el procesamiento de datos. La capacidad de generar reportes en tiempo real sobre el desempeño de la distribuidora se traduce en ventajas competitivas al permitir adaptarse rápidamente a las necesidades del mercado y optimizar la cadena de suministro.

En conclusión, el manejo adecuado de las consultas a la base de datos es esencial para que una distribuidora funcione de manera organizada, eficiente y acorde con las demandas de un entorno empresarial dinámico.