

# **Proyecto Consultas SQL**

**Faber Geovanni Hernández Monguí**

**Laura Daniel Arteaga Marín**

## **INTRODUCCION**

Se cuenta con un sistema de base de datos transaccional diseñado para registrar y gestionar el ciclo completo de ventas. Este repositorio de datos se estructura en torno a un esquema relacional que captura la siguiente información clave por cada transacción:

- IDtransaccion: Identificador único de la transacción.
- NumeroPedido: Número de referencia del pedido.
- ItemLinea: Línea específica dentro de un pedido.
- FechaPedido: Fecha en la que se realizó el pedido.
- FechaEntrega: Fecha en la que fue entregado el producto.
- Cantidad: Número de unidades vendidas.
- IDcliente: Identificador único del cliente.
- IDtienda: Identificador único de la tienda.
- IDproducto: Identificador único del producto.

La base de datos contiene un volumen significativo de información, con un total de 27.594 registros históricos, los cuales representan la actividad comercial acumulada.

El objetivo principal de este análisis es explotar este activo de datos mediante la ejecución de un conjunto de consultas SQL estratégicamente diseñadas. El fin último es generar indicadores de rendimiento (KPIs) e información accionable que permita:

1. Optimizar la Gestión de Inventarios: Identificar productos con alta demanda y aquellos con bajo movimiento para ajustar los niveles de stock.
2. Mejorar el Rendimiento por Tienda: Analizar el desempeño de cada punto de venta en términos de volumen de ventas.
3. Evaluar el Desempeño de Productos: Determinar los productos más y menos rentables.

Este proceso de análisis de datos se posiciona como una herramienta fundamental para la toma de decisiones basada en evidencia, orientada a impulsar la rentabilidad y la competitividad del negocio.

## METODOLOGIA

Para garantizar la calidad e integridad de los datos en el proceso de análisis, se aplicó un proceso de normalización de la base de datos. Como resultado, se estructuró la información en cuatro tablas relacionales, definidas de la siguiente manera:

### *Tabla 1 Clientes*

Contiene la información demográfica y de ubicación de los clientes, con las siguientes variables: IDcliente, GeneroCliente, NombreCliente, CiudadCliente,

CodigoEstadoCliente, EstadoCliente, CodigoPostalCliente, PaisCliente, ContinenteCliente, FechaNacimientoCliente. Con llave primaria en la variable IDcliente.

### *Tabla 2 Producto*

Almacena los atributos descriptivos y de clasificación de los productos, con las siguientes variables: IDproducto, NombreProducto, MarcaProducto, ColorProducto, CostoProducto, PrecioProducto, IDSubcategoriaProducto, SubcategoriaProducto, IDCategoriaProducto, CategoriaProducto. Con llave primaria en la variable IDproducto+

### *Tabla 3 Tiendas*

Registra la información de localización y características de cada tienda, con las siguientes variables: IDtienda, PaisTienda, EstadoTienda, MetrosCuadradosTienda, FechaAperturaTienda. Con llave primaria en la variable IDtienda.

### *Tabla 4 Transacciones*

Consolida los datos transaccionales del negocio, con las siguientes variables: IDtransaccion, NumeroPedido, ItemLinea, FechaPedido, FechaEntrega, Cantidad, IDcliente, IDtienda, IDproducto. Con llave primaria en IDtransacciones y 3 llaves foráneas: IDtiendas, IDclientes y IDproducto.

Una vez definido el esquema normalizado, se procedió a implementar las tablas en MySQL Workbench, estableciendo explícitamente las restricciones de clave primaria y foránea. Esta estructura relacional garantiza la integridad y optimiza la ejecución de las consultas requeridas.

## CONSULTAS

A continuación, se desarrollarán las consultas para responder cada uno de los objetivos planteados:

### *Objetivo 1*

Identificar productos con alta demanda y aquellos con bajo movimiento

Código:

```
#Consulta 1
SELECT
    p.IDproducto,
    p.NombreProducto,
    SUM(t.Cantidad) as TotalUnidadesVendidas
FROM TRANSACCIONES t
JOIN PRODUCTO p ON t.IDproducto = p.IDproducto
GROUP BY p.IDproducto, p.NombreProducto
ORDER BY TotalUnidadesVendidas DESC
LIMIT 10;
```

Resultado:

	IDproducto	NombreProducto	TotalUnidadesVendidas
▶	419	Adventure Works Desktop PC1.80 ED180 Silver	42
	440	WWI Desktop PC1.60 E1600 Silver	39
	448	WWI Desktop PC1.80 E1801 Black	30
	1636	Contoso DVD 55DVD Storage Binder M56 Silver	27
	1743	MGS Combat Flight Simulator 3 E119	26
	1579	SV DVD Recorder L220 White	22
	458	WWI Desktop PC1.80 E1800 White	22
	1455	The Phone Company Pen Touch Screen Phones ...	22
	1660	Contoso DVD 15-Inch Player Portable L200 White	21
	1703	SV Hand Games for kids E30 Silver	20

Código:

```

SELECT
    p.IDproducto,
    p.NombreProducto,
    COALESCE(SUM(t.Cantidad), 0) as TotalUnidadesVendidas
FROM PRODUCTO p
LEFT JOIN TRANSACCIONES t ON p.IDproducto = t.IDproducto
GROUP BY p.IDproducto, p.NombreProducto
HAVING TotalUnidadesVendidas = 0 OR TotalUnidadesVendidas < 10
ORDER BY TotalUnidadesVendidas ASC
LIMIT 10

```

Resultado:

IDproducto	NombreProducto	TotalUnidadesVendidas
1	Contoso 512MB MP3 Player E51 Silver	0
2	Contoso 512MB MP3 Player E51 Blue	0
3	Contoso 1G MP3 Player E100 White	0
5	Contoso 2G MP3 Player E200 Red	0
6	Contoso 2G MP3 Player E200 Black	0
7	Contoso 2G MP3 Player E200 Blue	0
8	Contoso 4G MP3 Player E400 Silver	0
11	Contoso 4G MP3 Player E400 Orange	0
12	Contoso 4GB Flash MP3 Player E401 Blue	0
13	Contoso 4GB Flash MP3 Player E401 Black	0

## Objetivo 2

Analizar el desempeño de cada punto de venta en términos de volumen de ventas.

Consulta:

```

#consulta 2.1
SELECT
    tr.IDtienda,
    ti.PaisTienda,
    ti.EstadoTienda,
    SUM(tr.Cantidad * p.PrecioProducto) as IngresoTotal
FROM transacciones tr
JOIN tiendas ti ON tr.IDtienda = ti.IDtienda
JOIN producto p ON tr.IDproducto = p.IDproducto
GROUP BY tr.IDtienda, ti.PaisTienda, ti.EstadoTienda
ORDER BY IngresoTotal DESC;

```

Resultado:

	IDtienda	PaisTienda	EstadoTienda	IngresoTotal
▶	65	United States	West Virginia	108096
	47	United States	Hawaii	88264
	50	United States	Kansas	85222
	55	United States	Nevada	81870
	66	United States	Wyoming	77994
	54	United States	Nebraska	73480
	43	United States	Alaska	72582
	59	United States	Oregon	70300
	64	United States	Washington DC	67694
	48	United States	Idaho	65698
	61	United States	South Carolina	65211
	45	United States	Connecticut	62893
	57	United States	New Mexico	62207
	49	United States	Iowa	56726
	63	United States	Utah	56384
	53	United States	Montana	56382
	56	United States	New Hampshire	53475
	44	United States	Arkansas	53311
	62	United States	South Dakota	41058
	51	United States	Maine	39671

### Objetivo 3

Evaluar el Desempeño de Productos: Determinar los productos más y menos rentables.

Consulta:

---

```
#consulta 3.1
SELECT
    p.IDproducto,
    p.NombreProducto,
    p.PrecioProducto,
    p.CostoProducto,
    (p.PrecioProducto - p.CostoProducto) as MargenUnitario,
    SUM(t.Cantidad) as TotalUnidadesVendidas,
    (p.PrecioProducto - p.CostoProducto) * SUM(t.Cantidad) as MargenBeneficioTotal
FROM PRODUCTO p
JOIN TRANSACCIONES t ON p.IDproducto = t.IDproducto
GROUP BY p.IDproducto, p.NombreProducto, p.PrecioProducto, p.CostoProducto
ORDER BY MargenBeneficioTotal DESC
LIMIT 10;
```

## Resultado:

	IDproducto	NombreProducto	PrecioProducto	CostoProducto	MargenUnitario	TotalUnidadesVendidas	MargenBeneficioTotal
▶	1842	Litware Washer & Dryer 27In L420 Green	2652	878,66	1774	10	17740
	540	Proseware Projector 1080p DLP86 Black	2499	827,97	1672	10	16720
	1897	Fabrikam Refrigerator 24.7CuFt X9800 White	3199,99	1060,22	2139	7	14973
	1862	NT Washer & Dryer 27In L2700 Green	2652,9	878,96	1774	7	12418
	428	Adventure Works Desktop PC2.33 XD233 Brown	969	321,05	648	19	12312
	444	WWI Desktop PC2.33 X2330 Black	919	304,48	615	18	11070
	374	Adventure Works Laptop19W X1980 Silver	1299	430,38	869	11	9559
	2091	Contoso Water Heater 4.3GPM M1250 Blue	877,5	403,53	474	19	9006
	2085	Contoso Water Heater 7.2GPM X1800 White	1475	488,7	987	8	7896
	419	Adventure Works Desktop PC1.80 ED180 Silver	369	188,13	181	42	7602

## CONCLUSION

Después de revisar toda la información de ventas y productos, ya tenemos una idea más clara de cómo van las cosas. Lo más importante que hemos encontrado hasta ahora es:

Sabemos exactamente qué productos se venden rápido y cuáles se quedan meses en el almacén, así podemos comprar menos de lo que no se mueve y asegurarnos de no quedarnos sin lo que más pide la gente. También descubrimos que no todas las tiendas venden igual - algunas son mucho más eficientes que otras, así que podemos copiar lo que hacen las mejores tiendas para mejorar las que van más lentas. Y por último, ahora entendemos realmente cuáles productos nos dejan más ganancias y cuáles apenas generan utilidades, lo que nos ayuda a decidir mejor en qué enfocarnos y cómo poner precios. Ya sabemos dónde están los principales problemas y oportunidades para mejorar las ventas y ganar más dinero.

