



Actividad 07/11/2024

Instituto Politécnico Nacional.
Escuela Superior de Cómputo.
Licenciatura en ciencia de datos.
Bases de Datos Avanzadas

Emiliano López Méndez.

Introduccion

En esta práctica de Bases de Datos Avanzadas, aplicamos conceptos esenciales de diseño y manipulación de bases de datos relacionales, enfocados en la creación de tablas, relaciones mediante claves foráneas y la inserción de datos de prueba. A través de este ejercicio, abordamos la importancia de estructurar adecuadamente los datos y optimizar la consulta de información. En particular, desarrollamos una vista que nos permite analizar las ventas por categoría de producto y mes, facilitando así la generación de informes clave para la toma de decisiones en contextos empresariales. Este ejercicio no solo refuerza las habilidades técnicas de SQL, sino también la comprensión de la utilidad de las bases de datos en el análisis de grandes volúmenes de datos.

Desarrollo de la Actividad:

Creación de Tablas: Inicialmente, creamos las tablas principales para almacenar datos sobre clientes, productos, tiempos y tiendas. Cada tabla incluye una clave primaria que asegura la unicidad de cada registro, fundamental para mantener la integridad de los datos. Esta fase permitió establecer la estructura básica de la base de datos y definir el esquema relacional.

Unión de Tablas mediante Llaves: Después de crear las tablas individuales, se establecieron las relaciones mediante claves foráneas, conectando las tablas auxiliares (dim_cliente, dim_producto, dim_tiempo, y dim_tienda) con la tabla principal hechos_ventas. Este paso es crucial para mantener la coherencia referencial en la base de datos y asegurar que solo se almacenen datos válidos y relacionados.

Inserciones en las Tablas: Se realizaron inserciones de datos en cada una de las tablas para simular un entorno de datos real. Estas inserciones fueron diseñadas para que los datos tuvieran coherencia y pudieran ser utilizados en consultas posteriores, incluyendo información relevante para cada cliente, producto, tiempo y tienda.

Creación de Vista para Análisis de Ventas: Finalmente, se creó una vista para obtener el total de ventas agrupado por categoría de producto y mes. Esta vista permite analizar fácilmente el rendimiento de cada categoría de productos a lo largo del tiempo, proporcionando una herramienta valiosa para la toma de decisiones en la gestión de ventas y la planificación estratégica.

Parte 1 Creación de tablas

En esta sección se crearon las tablas

```
create table dim_cliente(  
    id_cliente integer primary key,  
    nombre_cliente varchar(50),  
    edad integer,  
    genero char(1),  
    ciudad varchar(50)  
);  
  
create table dim_tiempo(  
    id_tiempo integer primary key,  
    anio integer,  
    mes integer,  
    dia integer,  
    trimestre integer  
);
```

```
create table dim_producto(  
    id_producto integer primary key,  
    nombre_producto varchar(50),  
    categoria varchar(50),  
    marca varchar(50),  
    precio_unitario decimal(10,2)  
);  
  
create table dim_tienda(  
    id_tienda integer primary key,  
    nombre_tienda varchar(50),  
    ciudad varchar(50),  
    estado varchar(50),  
    pais varchar(50)  
);
```

Parte 2 Unión de las tablas mediante llaves

En esta etapa, se realizó la creación y la unión de la tabla principal al igual que la unión de las llaves foráneas

```
33  
34 --Tabla principal de llaves  
35 create table hechos_ventas(  
36     id_venta integer primary key,  
37     id_tiempo integer,  
38     foreign key (id_tiempo) references dim_tiempo(id_tiempo),  
39     id_producto integer,  
40     foreign key (id_producto) references dim_producto(id_producto),  
41     id_cliente integer,  
42     foreign key (id_cliente) references dim_cliente(id_cliente),  
43     id_tienda integer,  
44     foreign key (id_tienda) references dim_tienda(id_tienda),  
45     cantidad integer,  
46     precio_total decimal(10,2)  
47 );  
48
```

Parte 3 Se hicieron las inserciones

En esta etapa se hicieron las inserciones para las tablas correspondientes, la igual que las inserciones tenga coherencia en si

```
--Inserts
INSERT INTO dim_cliente (id_cliente, nombre_cliente, edad, genero, ciudad) VALUES
(1, 'Juan Pérez', 35, 'M', 'Ciudad de México'),
(2, 'María Gómez', 28, 'F', 'Guadalajara'),
(3, 'Carlos López', 40, 'M', 'Monterrey'),
(4, 'Ana Sánchez', 25, 'F', 'Cancún'),
(5, 'Luis García', 33, 'M', 'Puebla'),
(6, 'Laura Martínez', 29, 'F', 'Tijuana'),
(7, 'Pedro Fernández', 38, 'M', 'León'),
(8, 'Sara Díaz', 45, 'F', 'Querétaro'),
(9, 'Ricardo Rodríguez', 32, 'M', 'San Luis Potosí'),
(10, 'Patricia Torres', 50, 'F', 'Mérida');
```

```
INSERT INTO dim_tiempo (id_tiempo, año, mes, día, trimestre) VALUES
(1, 2023, 1, 15, 1),
(2, 2023, 2, 20, 1),
(3, 2023, 3, 10, 1),
(4, 2023, 4, 5, 2),
(5, 2023, 5, 8, 2),
(6, 2023, 6, 25, 2),
(7, 2023, 7, 18, 3),
(8, 2023, 8, 30, 3),
(9, 2023, 9, 2, 3),
(10, 2023, 10, 13, 4);
```

```
INSERT INTO dim_producto (id_producto, nombre_producto, categoría, marca, precio_unitario) VALUES
(1, 'Camiseta Sport', 'Ropa', 'Nike', 250.00),
(2, 'Zapatos Running', 'Ropa', 'Adidas', 1200.00),
(3, 'Laptop Gaming', 'Tecnología', 'Dell', 15000.00),
(4, 'Smartphone', 'Electrónica', 'Samsung', 7000.00),
(5, 'Cámara Digital', 'Electrónica', 'Canon', 4000.00),
(6, 'Televisor 4K', 'Electrónica', 'Sony', 12000.00),
(7, 'Bicicleta MTB', 'Deportes', 'Trek', 8000.00),
(8, 'Smartwatch', 'Tecnología', 'Apple', 3500.00),
(9, 'Auriculares Bluetooth', 'Electrónica', 'Bose', 2500.00),
(10, 'Silla de Oficina', 'Muebles', 'Ikea', 1500.00);
```

```
INSERT INTO dim_tienda (id_tienda, nombre_tienda, ciudad, estado, pais) VALUES
(1, 'Tiendas Eléctrica', 'Ciudad de México', 'CDMX', 'México'),
(2, 'Deportes Outlet', 'Guadalajara', 'Jalisco', 'México'),
(3, 'Tecnología Express', 'Monterrey', 'Nuevo León', 'México'),
(4, 'Muebles Casa', 'Puebla', 'Puebla', 'México'),
(5, 'Moda Estilo', 'Tijuana', 'Baja California', 'México'),
(6, 'ElectroShop', 'León', 'Guanajuato', 'México'),
(7, 'Gadget World', 'Cancún', 'Quintana Roo', 'México'),
(8, 'Sportlife', 'San Luis Potosí', 'San Luis Potosí', 'México'),
(9, 'Tech Store', 'Querétaro', 'Querétaro', 'México'),
(10, 'Home Store', 'Mérida', 'Yucatán', 'México');
```

```
INSERT INTO hechos_ventas (id_venta, id_tiempo, id_producto, id_cliente, id_tienda, cantidad, precio_total) VALUES
(1, 1, 1, 1, 1, 2, 500.00),
(2, 2, 3, 2, 2, 1, 15000.00),
(3, 3, 4, 3, 3, 3, 21000.00),
(4, 4, 2, 4, 4, 1, 1200.00),
(5, 5, 6, 5, 5, 2, 24000.00),
(6, 6, 5, 6, 6, 1, 4000.00),
(7, 7, 7, 7, 7, 1, 8000.00),
(8, 8, 8, 8, 8, 1, 3500.00),
(9, 9, 9, 9, 9, 2, 5000.00),
(10, 10, 10, 10, 10, 1, 1500.00);
```

Parte 4 Vista para ver ventas por categoría de producto y mes

En esta parte del laboratorio creamos la vista para poder ver los productos vendidos por categoría, mes y cantidad vendida

```
--Vistas
create view ventas_categoria_producto_mes as
select
    dim_producto.categoria,
    dim_tiempo.mes,
    sum(hechos_ventas.precio_total) as total_ventas
from
    hechos_ventas
join dim_producto on hechos_ventas.id_producto = dim_producto.id_producto
join dim_tiempo on hechos_ventas.id_tiempo = dim_tiempo.id_tiempo
group by dim_producto.categoria, dim_tiempo.mes;

select *from dim_producto
select *from dim_tiempo
select *from ventas_categoria_producto_mes order by mes
```

Data Output			
<div><div><div><div></div></div><div><div></div></div><div><div></div></div><div><div></div></div><div><div></div></div><div><div></div></div><div><div></div></div><div><div></div></div><div><div></div></div></div></div>			
	categoría character varying (50)	mes integer	total_ventas numeric
1	Ropa	1	500.00
2	Tecnología	2	15000.00
3	Electrónica	3	21000.00
4	Ropa	4	1200.00
5	Electrónica	5	24000.00
6	Electrónica	6	4000.00
7	Deportes	7	8000.00
8	Tecnología	8	3500.00
9	Electrónica	9	5000.00
10	Muebles	10	1500.00

Conclusión

A través de esta práctica, logramos consolidar el conocimiento en diseño de bases de datos, aplicando conceptos de integridad referencial y normalización mediante el uso de claves primarias y foráneas. La creación de una vista para analizar las ventas por categoría y mes demostró cómo SQL puede ser utilizado para extraer información significativa de grandes volúmenes de datos de forma eficiente. Este ejercicio subraya la importancia de una buena estructura de base de datos, no solo para el almacenamiento de información, sino para facilitar el análisis y la generación de reportes. En conclusión, estas habilidades son esenciales para gestionar y aprovechar al máximo los datos en cualquier contexto organizacional.