

INTRODUCCIÓN A LA CIENCIA DE LOS DATOS

**UNIVERSIDAD DEL VALLE
PROYECTO 1
HÉCTOR FABIO OCAMPO ARBELÁEZ**

INGENIERÍA EN SISTEMAS

**CARLOS ALBERTO CAMACHO CASTAÑO 202160331-3743
KEVIN ALEXANDER MARIN HENAO 202160364-3743
HARRISON INEEY VALENCIA OTERO 202159979-3743**

TULUÁ, VALLE DEL CAUCA

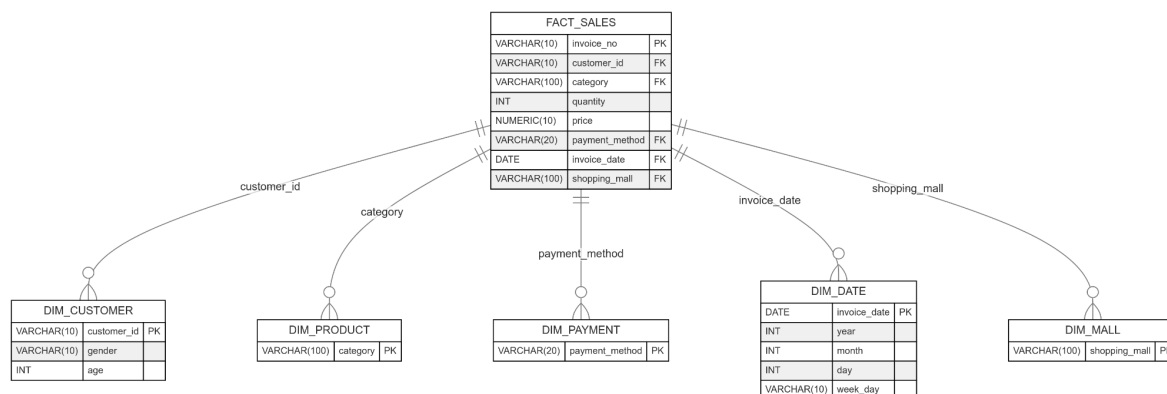
10/03/2025

1. Diseño del modelo de la bodega de datos

b. Dado que el dataset tiene una estructura de datos bastante simple y que está orientada a análisis de ventas, el modelo en estrella es la mejor opción, según nuestro criterio.

Ya que es más rápida a la hora de realizar consultas, porque requiere el uso de menos joins al mantener las dimensiones desnormalizadas. Las dimensiones cuentan con la información necesaria sin tener que dividirse en múltiples tablas.

c.



d.

```

CREATE TABLE dim_customer (
  customer_id VARCHAR(10) PRIMARY KEY,
  gender VARCHAR(10),
  age INT
);
  
```

```

CREATE TABLE dim_product (
  category VARCHAR(100) PRIMARY KEY
);
  
```

```

CREATE TABLE dim_payment (
  payment_method VARCHAR(20) PRIMARY KEY
);
  
```

```

CREATE TABLE dim_date (
  invoice_date DATE PRIMARY KEY,
  year INT,
  month INT,
  day INT,
  week_day VARCHAR(10)
);
  
```

```

CREATE TABLE dim_mall (
  shopping_mall VARCHAR(100) PRIMARY KEY
);
  
```

);

```
CREATE TABLE fact_sales (
    invoice_no VARCHAR(10) PRIMARY KEY,
    customer_id VARCHAR(10),
    category VARCHAR(100),
    quantity INT,
    price NUMERIC(10,2),
    payment_method VARCHAR(20),
    invoice_date DATE,
    shopping_mall VARCHAR(100),
    FOREIGN KEY (customer_id) REFERENCES dim_customer(customer_id),
    FOREIGN KEY (category) REFERENCES dim_product(category),
    FOREIGN KEY (payment_method) REFERENCES
dim_payment(payment_method),
    FOREIGN KEY (invoice_date) REFERENCES dim_date(invoice_date),
    FOREIGN KEY (shopping_mall) REFERENCES dim_mall(shopping_mall)
);
```

2. Explicación del proceso ETL

Extracción: Se usa Pandas para leer el dataset en formato CSV.

```
import pandas as pd

# Extracción: Cargar datos desde el archivo movies.csv
print("Extrayendo datos del CSV...")
csv_file = "customer_shopping_data.csv" # Ruta del archivo CSV
df = pd.read_csv(csv_file)
print("Datos extraídos correctamente.")
print('-----')

df.head() # Muestra las primeras filas para verificar
```

Extrayendo datos del CSV

Transformación: Se normalizan los datos, eliminan inconsistencias y estructuran según a el modelo en estrella.

```
# Transformación: Limpiar y modificar datos
print("Transformando datos...")

# Dimensiones del DataFrame
print("Dimensiones del DataFrame:")
df_customers = df[['customer_id', 'gender', 'age']].drop_duplicates()

df_products = df[['category']].drop_duplicates().rename(columns={'category': 'category'})

df_payment = df[['payment_method']].drop_duplicates()

df['invoice_date'] = pd.to_datetime(df['invoice_date'], dayfirst=True) # Convertir a formato de fecha con dayfirst=True

df_date = df[['invoice_date']].drop_duplicates().copy()
df_date['year'] = df_date['invoice_date'].dt.year
df_date['month'] = df_date['invoice_date'].dt.month
df_date['day'] = df_date['invoice_date'].dt.day
df_date['week_day'] = df_date['invoice_date'].dt.strftime('%A')

df_malls = df[['shopping_mall']].drop_duplicates()

df_fact_sales = df[['invoice_no', 'customer_id', 'category', 'quantity', 'price', 'payment_method', 'invoice_date', 'shopping_mall']]
```

Lo visualizamos de esta manera:

Transformando datos...

Dimensiones del DataFrame:

	customer_id	gender	age
0	C241288	Female	28
1	C111565	Male	21
2	C266599	Male	20
3	C988172	Female	66
4	C189076	Female	53

	category
0	Clothing
1	Shoes
4	Books
6	Cosmetics
10	Food & Beverage

	payment_method
0	Credit Card
1	Debit Card
2	Cash

	invoice_date	year	month	day	week_day
0	2022-08-05	2022	8	5	Friday
1	2021-12-12	2021	12	12	Sunday
2	2021-11-09	2021	11	9	Tuesday
...					
2	2021-11-09				Metrocity
3	2021-05-16				Metropol AVM
4	2021-10-24				Kanyon

Carga:

Se crea la conexión al servidor de PostgreSQL

```
print("Cargando datos en la base de datos PostgreSQL...")

# Configuración de conexión
DB_USER = "postgres"
DB_PASSWORD = "1234"
DB_HOST = "localhost"
DB_PORT = "5432"
DB_NAME = "proyecto-DS"

# Crear conexión a PostgreSQL
engine = create_engine(f'postgresql://{DB_USER}:{DB_PASSWORD}@{DB_HOST}:{DB_PORT}/{DB_NAME}')
print("Conexión exitosa")
```

Se cargan los datos a la BD en PostgreSQL

```
# Cargar clientes
df_customers.to_sql('dim_customer', con=engine, if_exists='append', index=False)

# Cargar productos
df_products.to_sql('dim_product', con=engine, if_exists='append', index=False)

# Cargar métodos de pago
df_payment.to_sql('dim_payment', con=engine, if_exists='append', index=False)

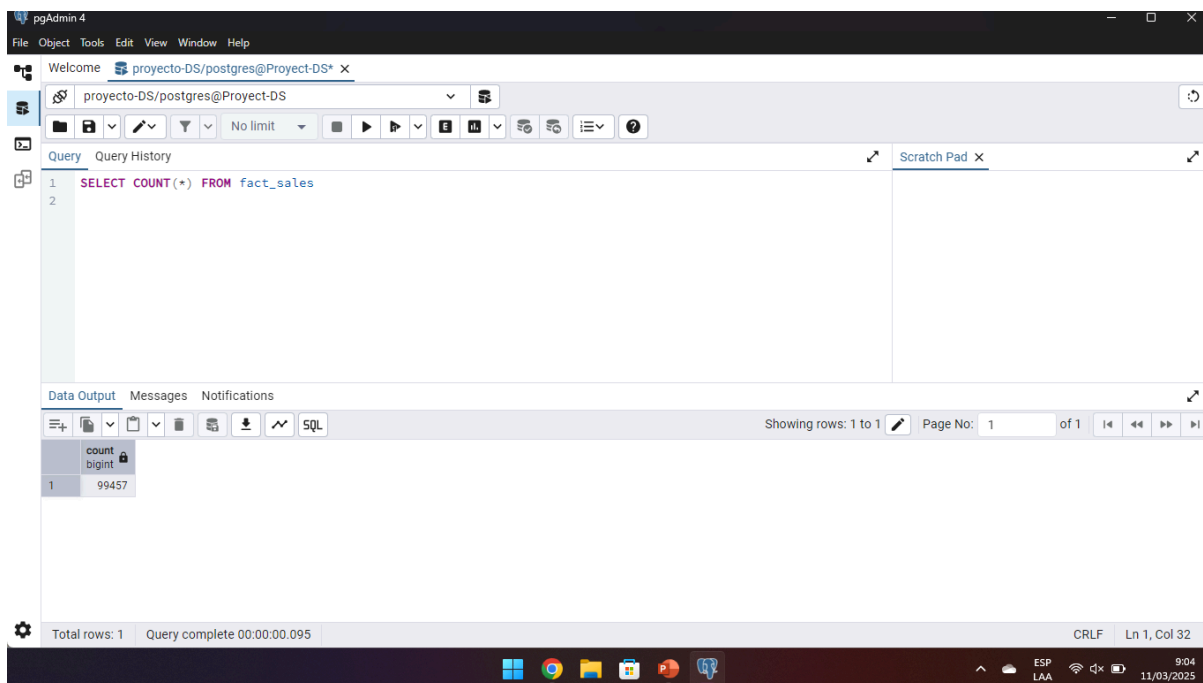
# Cargar fechas
df_date.to_sql('dim_date', con=engine, if_exists='append', index=False)

# Cargar centros comerciales
df_malls.to_sql('dim_mall', con=engine, if_exists='append', index=False)

print("Datos de dimensiones cargados correctamente")

df_fact_sales.to_sql('fact_sales', con=engine, if_exists='append', index=False)
print("Datos de fact_sales cargados correctamente")
```

Verificación: Se hace una consulta para verificar que los datos fueron cargados correctamente



The screenshot shows the pgAdmin 4 web interface. The top navigation bar includes 'File', 'Object', 'Tools', 'Edit', 'View', 'Window', and 'Help'. The main toolbar contains various icons for database management. The 'Query' tab is active, displaying the SQL query: `SELECT COUNT(*) FROM fact_sales`. The 'Data Output' tab shows the results of the query in a table with one row and one column.

count bigint
99457

The status bar at the bottom indicates 'Total rows: 1' and 'Query complete 00:00:00.095'. The system tray at the bottom right shows the date and time: '11/03/2025 9:04'.

3. Consultas analíticas en SQL

Resultados obtenidos de las consultas SQL

I. Total de ventas por categoría de producto

Data Output Messages Notifications		
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	category [PK] character varying (100)	total_ventas numeric
1	Clothing	113996791.04
2	Shoes	66553451.47
3	Technology	57862350.00
4	Cosmetics	6792862.90
5	Toys	3980426.24
6	Food & Beverage	849535.05
7	Books	834552.90
8	Souvenir	635824.65

II. Clientes con mayor volumen de compras

En este caso, son los 10 clientes con mayor volumen de compras

Data Output Messages Notifications				
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>				
	customer_id [PK] character varying (10)	gender character varying (10)	age integer	total_compras numeric
1	C101427	Male	32	26250.00
2	C101788	Female	66	26250.00
3	C101344	Female	31	26250.00
4	C100607	Female	54	26250.00
5	C101667	Female	23	26250.00
6	C100322	Female	24	26250.00
7	C100306	Female	35	26250.00
8	C101216	Male	68	26250.00
9	C100168	Male	48	26250.00
10	C104402	Female	69	26250.00

III. Métodos de pago más utilizados

Data Output Messages Notifications		
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗑️</div> <div>📄</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	payment_method [PK] character varying (20)	cantidad_transacciones bigint
1	Cash	44447
2	Credit Card	34931
3	Debit Card	20079

Aquí se cuenta cuántas veces ha sido usado cada método de pago

IV. Comparación de ventas por mes

Data Output Messages Notifications			
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗑️</div> <div>📄</div> <div>⬇️</div> <div>📈</div> </div>			
	year integer	month integer	total_ventas numeric
1	2021	1	9641614.62
2	2021	2	8772315.22
3	2021	3	9455359.38
4	2021	4	9389541.54
5	2021	5	9771756.97
6	2021	6	9286271.35
7	2021	7	10311119.68
8	2021	8	9630655.70
9	2021	9	9188165.62
10	2021	10	10263015.06
11	2021	11	9265555.29
12	2021	12	9585200.16
13	2022	1	9764311.14
14	2022	2	8344111.92
15	2022	3	9986685.16
16	2022	4	9326144.44
17	2022	5	9947574.13
18	2022	6	9647503.95
19	2022	7	10067602.95
20	2022	8	9651705.59
21	2022	9	9607629.29
22	2022	10	10282075.37
23	2022	11	8941584.66
24	2022	12	9869885.48
25	2023	1	9485599.83
26	2023	2	9508662.96
27	2023	3	2514146.79

4. Análisis Descriptivo y Visualización de Datos

Gráfico de barras: Total de ventas por categoría.

```
import matplotlib.pyplot as plt
import seaborn as sns
from sqlalchemy import create_engine

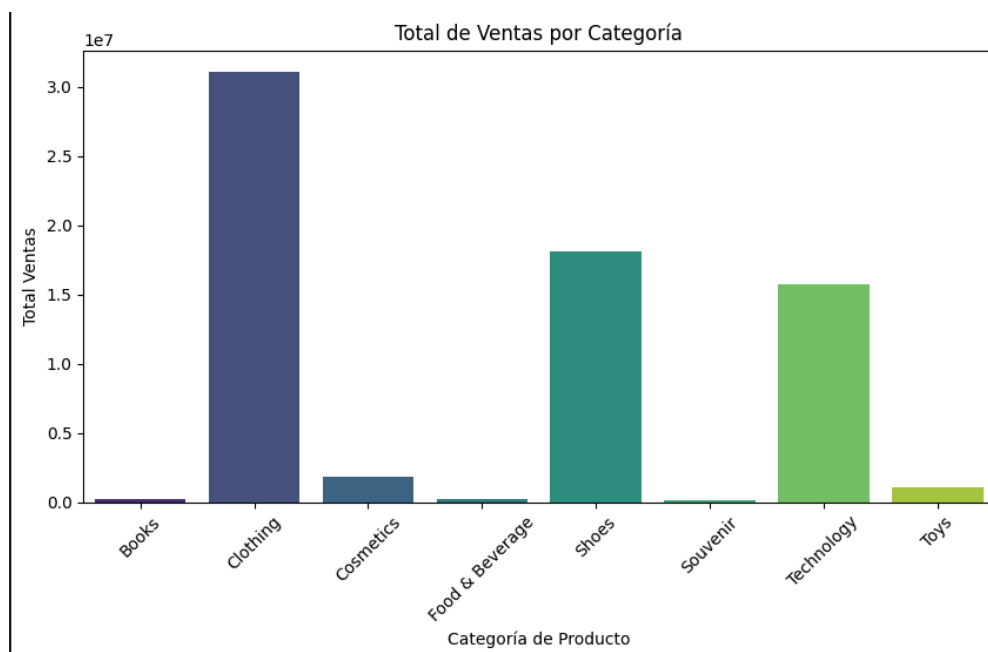
# Crear conexión a PostgreSQL
engine = create_engine(f'postgresql://{DB_USER}:{DB_PASSWORD}@{DB_HOST}:{DB_PORT}/{DB_NAME}')
print("Conexión exitosa")

# Carga del conjunto de datos de la tabla de hechos
df_fact_sales = pd.read_sql_table('fact_sales', engine)

ventas_categoria = df_fact_sales.groupby("category")["price"].sum().reset_index()

print("Ventas por categoría:")

plt.figure(figsize=(10, 5))
sns.barplot(x="category", y="price", data=ventas_categoria, palette="viridis")
plt.xticks(rotation=45)
plt.title("Total de Ventas por Categoría")
plt.xlabel("Categoría de Producto")
plt.ylabel("Total Ventas")
plt.show()
```



Analizando este gráfico podemos ver que la **Ropa y Zapatos**, son las categorías con mayores ventas, con ingresos superiores a los 31 millones y 18 millones.

Por otro lado vemos como los **Libros, Alimentos y Bebidas, y Recuerdos** tienen ingresos mucho menores, lo que sugiere que son productos menos vendidos o de menor precio unitario.

Los **juguetes** tienen un nivel intermedio de ingresos, lo que significa una demanda estable en comparación con las categorías de menor rendimiento.

Posibles mejoras del negocio:

- Podemos ampliar el inventario con los productos con mayores ventas, ofrecer promociones por volumen o implementar programas de fidelización para incentivar compras recurrentes.

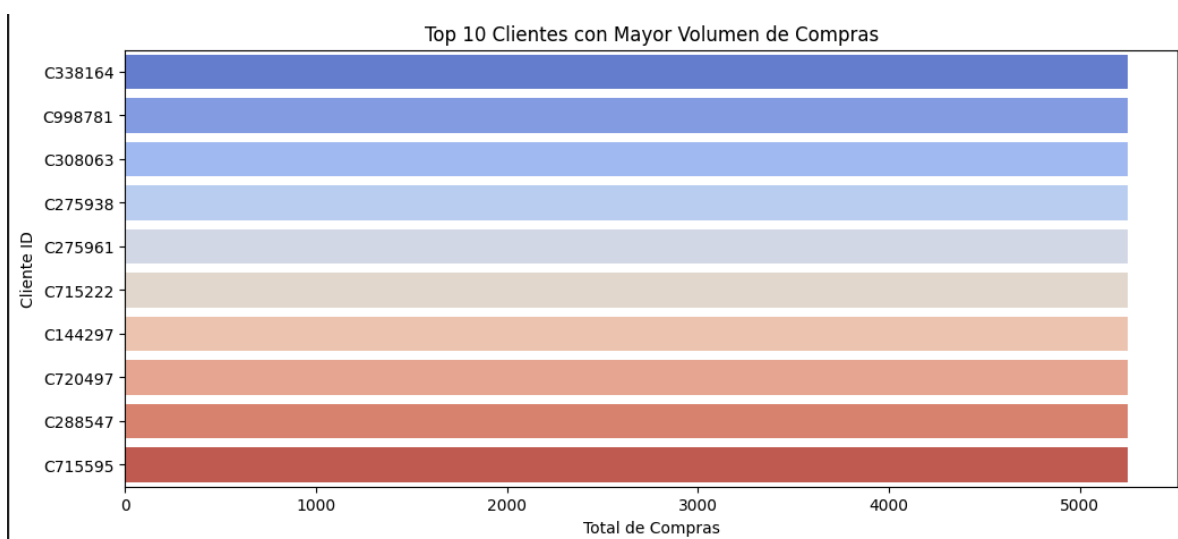
- Una menor demanda conlleva a mejorar su rendimiento, se pueden explorar estrategias como paquetes promocionales, campañas de marketing dirigidas a un público objetivo específicos.

Histograma: Top 10 clientes con mayor volumen de compras.

```
ventas_clientes = df_fact_sales.groupby("customer_id")["price"].sum().reset_index().sort_values(by="price", ascending=False).head(10)

print("Top 10 Clientes con Mayor Volumen de Compras:")

plt.figure(figsize=(12, 5))
sns.barplot(x="price", y="customer_id", data=ventas_clientes, palette="coolwarm")
plt.title("Top 10 Clientes con Mayor Volumen de Compras")
plt.xlabel("Total de Compras")
plt.ylabel("Cliente ID")
plt.show()
```



El **Cliente C715595** es el que más ha comprado, con un volumen de compras superior a **5000**, esto nos indica que es un cliente altamente activo e importante para el negocio.

Posibles mejoras del negocio:

- Analizando este gráfico podemos implementar nuevas estrategias, para incrementar las ventas con pequeños beneficios a los 10 clientes con mayor volumen de compras.

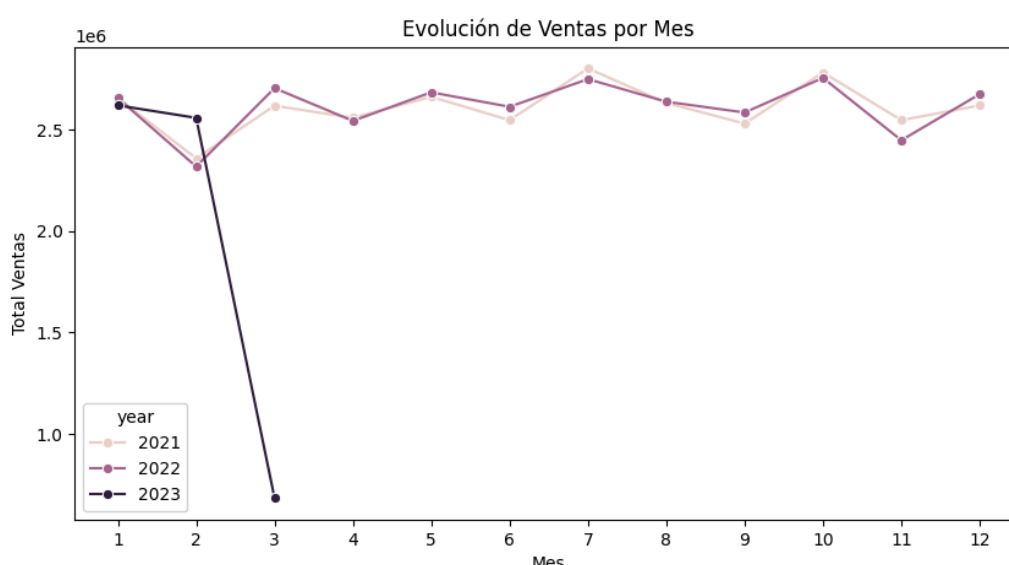
Gráfico de líneas: Ventas mensuales.

```
# Cargar la tabla dim_date desde la base de datos
dim_date = pd.read_sql_table('dim_date', engine)

fact_sales = df_fact_sales.merge(dim_date, on="invoice_date")
ventas_mes = fact_sales.groupby(["year", "month"])["price"].sum().reset_index()

print("Evolución de Ventas por Mes:")

plt.figure(figsize=(10, 5))
sns.lineplot(x=ventas_mes["month"], y=ventas_mes["price"], hue=ventas_mes["year"], marker="o")
plt.xticks(range(1, 13))
plt.title("Evolución de Ventas por Mes")
plt.xlabel("Mes")
plt.ylabel("Total Ventas")
plt.show()
```



Este gráfico de líneas muestra la evolución de las ventas por mes en los años 2021, 2022 y 2023.

Las ventas en 2021 y 2022 parecen seguir un patrón estable a lo largo del año, con fluctuaciones moderadas.

En 2023 vemos una caída brusca ya que solo hay datos hasta mediados del mes de marzo, pero vemos que en el mes febrero se tuvo unas ventas superiores a los años anteriores.

Posibles mejoras del negocio:

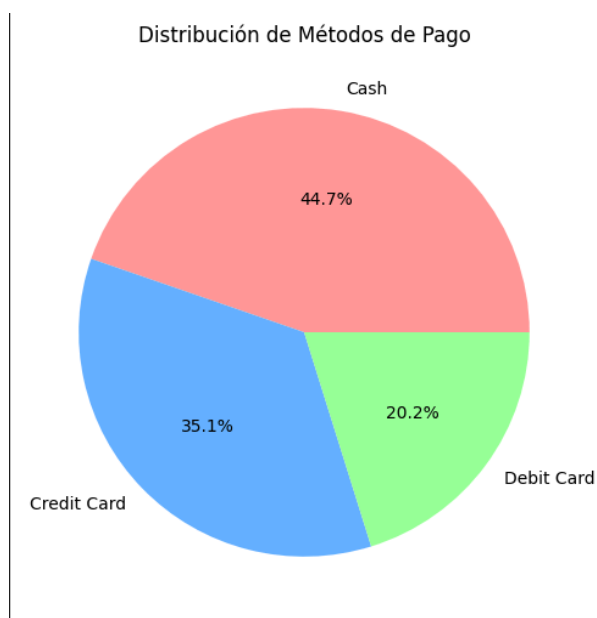
- Observamos un aumento en las ventas durante febrero de 2023. Para aprovechar esta tendencia, sería útil analizar qué productos tuvieron mayor demanda. Con esta información, podríamos ofrecer descuentos estratégicos en esos productos o en artículos complementarios, además de implementar campañas de marketing dirigidas para maximizar el impacto.
- Al analizar los meses con las ventas más bajas, podemos identificar las posibles causas de estos descensos y tomar medidas para evitar que se repitan en el futuro.

Gráfico circular: Uso de métodos de pago.

```
metodos_pago = df_fact_sales["payment_method"].value_counts()

print("Distribución de Métodos de Pago:")

plt.figure(figsize=(6, 6))
plt.pie(metodos_pago, labels=metodos_pago.index, autopct="%1.1f%%", colors=["#ff9999", "#66b3ff", "#99ff99"])
plt.title("Distribución de Métodos de Pago")
plt.show()
```



Este gráfico circular de métodos de pago nos permite analizar cómo prefieren pagar los clientes.

Las transacciones se realizan en **efectivo** El 44.7% , esto significa que muchos clientes prefieren este método.

Las **tarjetas de crédito** representan el 35.1%, lo que indica que quizás los clientes aprovechan beneficios como puntos o cashback.

Solo el 20.2% de las transacciones se hacen con **tarjeta de débito**, lo que significa que muchos clientes prefieren el crédito sobre el débito.

Posibles mejoras del negocio:

- Si el objetivo del negocio es reducir el uso de efectivo, se pueden ofrecer **descuentos o incentivos** para pagos con tarjeta.
- Si el objetivo es aumentar el uso de tarjetas de débito, se pueden **educar a los clientes** sobre sus beneficios o eliminar costos adicionales por su uso.

5. Diseño de la bodega de datos

- **Diseño eficiente:** Se optó por un modelo estrella para optimizar consultas.

- **ETL bien estructurado:** Se realizó la transformación y carga de datos de forma efectiva.
- **Consultas relevantes:** Se lograron responder preguntas clave del negocio.
- **Análisis visual útil:** Los gráficos permitieron identificar tendencias importantes.

CONCLUSIONES

Los gráficos generados proporcionaron información clave sobre tendencias y patrones de compra, facilitando la toma de decisiones estratégicas para mejorar la rentabilidad y eficiencia del negocio como.

Análisis de ventas y estrategias comerciales:

- Se identificó que las categorías de Ropa y Zapatos son las más rentables, lo que sugiere la importancia de ampliar el inventario y desarrollar estrategias de fidelización.
- Las categorías con menores ingresos, como Libros, Alimentos y Bebidas, y Recuerdos, requieren estrategias de promoción y marketing dirigidas para mejorar su rendimiento.
- El análisis de clientes reveló que algunos tienen un alto volumen de compras, lo que permite diseñar programas de beneficios exclusivos para incentivar la lealtad.

Identificación de patrones de ventas:

- Se observó un aumento significativo en las ventas en febrero de 2023, lo que sugiere una oportunidad para campañas de marketing específicas en este periodo.
- También se identificaron periodos de menor demanda, lo que permite desarrollar estrategias para mitigar caídas en las ventas y mejorar la estabilidad del negocio.