Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Curso: Sistemas Organizacionales y Gerenciales 2

Título: Práctica #1

Guatemala, Septiembre de 2025

Luis Godoy

200418365

```
import pandas as pd
import mysql.connector
from mysql.connector import Error
import seaborn as sns
import matplotlib.pyplot as plt
```

```
try:
    conn = mysql.connector.connect(
        host=host,
        user=user,
        password=password,
        database=database
    )

    query_clientes = "SELECT * FROM clientes;"
    query_productos = "SELECT * FROM productos;"
    query_metodos = "SELECT * FROM metodos_pago;"
    query_ordenes = "SELECT * FROM ordenes;"

    clientes = pd.read_sql(query_clientes, conn)
    productos = pd.read_sql(query_productos, conn)
    metodos_pago = pd.read_sql(query_metodos, conn)
    ordenes = pd.read_sql(query_ordenes, conn)

            Tablas cargadas correctamente")


                                e)

finally:
    if 'conn' in locals() and conn.is_connected():
        conn.close()
```

```
df = ordenes.merge(clientes, on="customer_id", how="left") \
            .merge(productos, on="product_name", how="left") \
            .merge(metodos_pago, on="payment_method_id", how="left")
```

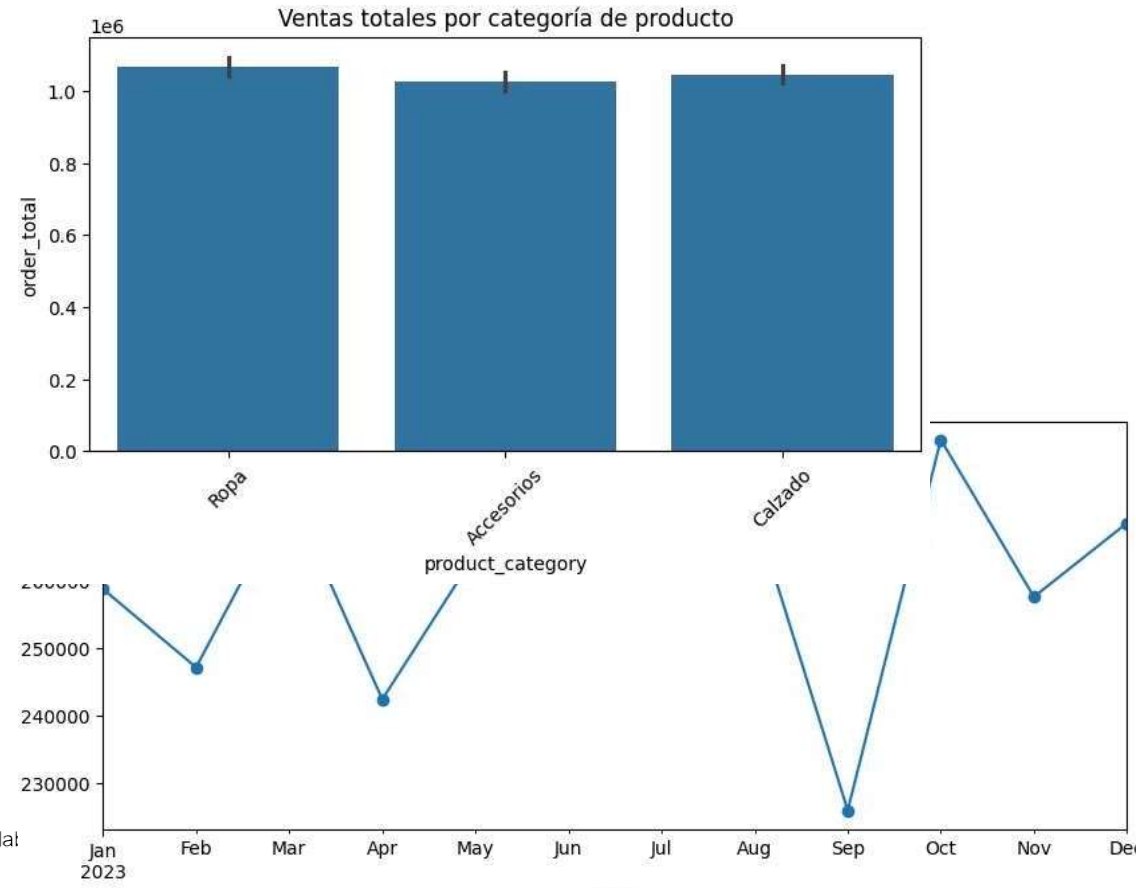
```
df["purchase_date"] = pd.to_datetime(df["purchase_date"], errors="coerce")

#media, mediana, moda
print("📊 Estadísticas Básicas:")
for col in ["customer_age", "product_price", "quantity", "order_total"]:
    media = df[col].mean()
    mediana = df[col].median()
    moda = df[col].mode()[0]
    print(f"{col}: Media={media:.2f}, Mediana={mediana:.2f}, Moda={moda}")
```

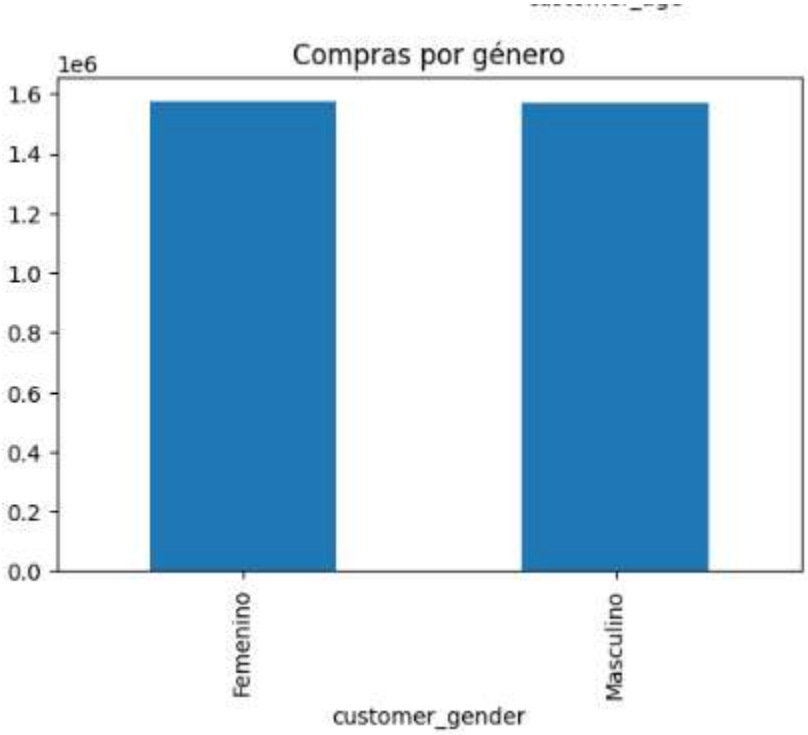
```
# Distribución de ventas por categoría de producto
plt.figure(figsize=(8,4))
sns.barplot(data=df, x="product_category", y="order_total", estimator=sum)
plt.title("Ventas totales por categoría de producto")
plt.xticks(rotation=45)
plt.show()
#Distribución de ventas por region
plt.figure(figsize=(8,4))
sns.barplot(data=df, x="shipping_region", y="order_total", estimator=sum)
plt.title("Ventas totales por región")
plt.xticks(rotation=45)
plt.show()
#Tendencia mensual de ventas
df.set_index("purchase_date", inplace=True)
ventas_mensuales = df.resample("M")["order_total"].sum()
ventas_mensuales.plot(kind="line", marker="o", figsize=(10,4), title="Tendencia mensual de ventas")
plt.show()
```

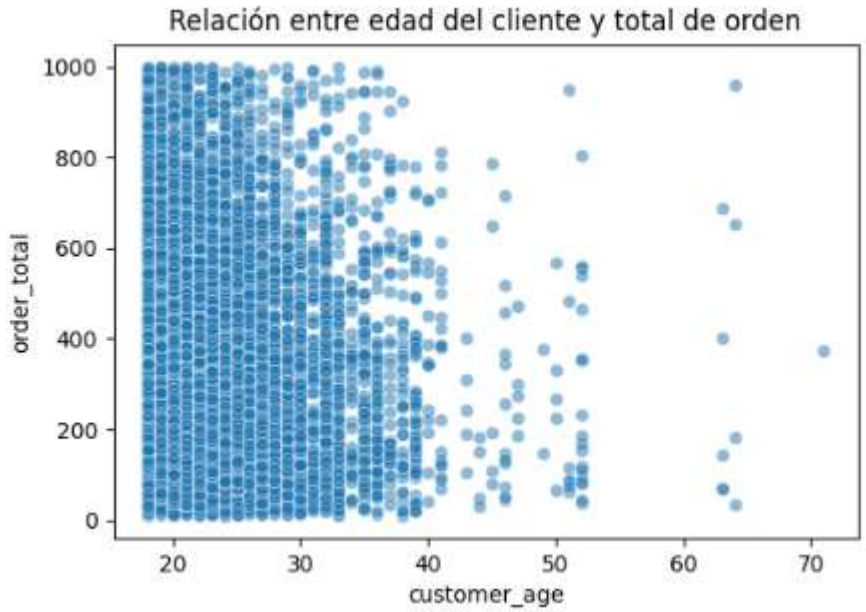
```
plt.show()

# Productos más vendidos
top_productos = df.groupby("product_name")["quantity"].sum().nlargest(10)
top_productos.plot(kind="bar", figsize=(10,4), title="Top 10 productos más vendidos")
plt.show()

# Segmentación por edad
plt.figure(figsize=(8,4))
sns.histplot(df["customer_age"], bins=15, kde=True)
plt.title("Distribución de clientes por edad")
plt.show()

# Compras por género
compras_genero = df.groupby("customer_gender")["order_total"].sum()
compras_genero.plot(kind="bar", figsize=(6,4), title="Compras por género")
plt.show()

# Correlación: edad vs total de orden
plt.figure(figsize=(6,4))
sns.scatterplot(data=df, x="customer_age", y="order_total", alpha=0.5)
plt.title("Relación entre edad del cliente y total de orden")
#
plt.show()
```

```
  7. CORRELACIONES
```

```
print("\n🔗 Correlación Edad vs Total Orden:", df["customer_age"].corr(df["order_total"]))

pivot = pd.pivot_table(df, values="order_total", index="product_category", columns="payment_method", aggfunc="sum", fill_value=0)
print("\n📊 Relación entre categoría de producto y método de pago:")
print(pivot)
```







a. ¿Cómo podrían los insights obtenidos ayudar a diferenciarse de la competencia?

Los insights permiten identificar:

Preferencias por categoría de producto: Al saber que ropa y accesorios tienen altas ventas, la empresa puede enfocar promociones, marketing personalizado o lanzamientos de productos en estas categorías.

Preferencias por región: Las regiones con mayores ventas (como Este y Norte) podrían recibir más inversión en logística, publicidad o eventos locales.

Métodos de pago más utilizados por categoría: Entender qué métodos de pago son más usados por tipo de producto permite ajustar pasarelas de pago, mejorar UX, o lanzar promociones específicas (por ejemplo: descuentos al pagar con PayPal para

calzado).

Esto ayuda a personalizar la experiencia del cliente, optimizar los recursos y posicionarse con estrategias más certeras que la competencia.

b. ¿Qué decisiones estratégicas podrían tomarse basándose en este análisis para aumentar las ventas y la satisfacción del cliente?

Optimización del inventario: Identificar los productos más vendidos (Sweater, Chaqueta, etc.) permite ajustar el inventario y evitar quiebres de stock.

Campañas segmentadas:

Por género (ya que ambos compran en cantidades similares).

Por edad (los clientes más jóvenes compran más; personalizar productos o campañas a este segmento).

Promociones mensuales: Dado que las ventas caen en abril y septiembre, se pueden lanzar promociones específicas en esos meses.

Diversificación de métodos de pago: Si ciertos métodos dominan por categoría, se pueden incluir nuevas opciones o hacer más visibles las existentes.

c. ¿Cómo podría este análisis de datos ayudar a la empresa a ahorrar costos o mejorar la eficiencia operativa?

Optimización logística: Focalizar entregas o refuerzos en las regiones con más ventas (Norte, Este).

Reducción de sobrestock: Evitar invertir en productos poco populares.

Ajuste de promociones y métodos de pago: No invertir en promociones ineficaces o métodos de pago poco usados.

Envíos personalizados: Aprovechando el conocimiento de regiones, adaptar métodos de entrega o alianzas con couriers.

d. ¿Qué datos adicionales recomendarían recopilar para obtener insights aún más valiosos en el futuro?

Feedback de clientes / NPS: Para conocer la percepción del servicio y producto.

Tiempo de entrega real: Para relacionarlo con la satisfacción del cliente o devoluciones.

Dispositivo o canal de compra: Saber si compran más desde móvil, web o app.

Historial de compras por cliente: Permitiría hacer retargeting, recomendaciones personalizadas y análisis de fidelidad.

División de tareas

El proyecto fue dividido en varias fases para una ejecución ordenada:

1. **Importación y limpieza de datos:** Conversión del archivo .csv en tablas SQL.
2. **Conexión a Google Cloud SQL desde Colab.**
3. **Análisis exploratorio de datos (EDA).**
4. **Visualización de datos y extracción de insights.**
5. **Documentación final.**

• Herramientas y tecnologías utilizadas

Herramienta	Uso principal
Google Colab	Entorno para ejecutar Python y conectar a la base de datos

Herramienta	Uso principal
Google Cloud SQL	Base de datos MySQL alojada en la nube
MySQL Workbench	Visualización y validación de la estructura de las tablas
Pandas, Matplotlib, Seaborn	Análisis de datos y visualizaciones gráficas
mysql.connector	Librería para conectar Python con MySQL desde Colab

¿Por qué estas herramientas?

- Colab ofrece un entorno accesible y potente para trabajar con análisis de datos.
- Google Cloud permite tener una base de datos disponible desde cualquier parte.
- Las librerías de visualización son altamente efectivas para análisis exploratorio.

• Plazos

Cada fase fue estimada de la siguiente manera:

- Conversión CSV → SQL: **1 día**
- Conexión a MySQL desde Colab: **1 día**
- Análisis EDA: **2 días**
- Visualizaciones + insights: **2 días**
- Documentación: **1 día**

B) Proceso de Análisis

• Limpieza y preparación de datos

1. **Carga del archivo** ventas_tienda_online.csv con pandas.
2. **Normalización de las columnas**: columnas como product_category, shipping_region y payment_method fueron homogeneizadas y transformadas en tablas independientes.
3. **Manejo de valores nulos**: en el método de pago se reemplazó NaN por "Desconocido".
4. **Generación de IDs** para cada categoría (product_id, customer_id, payment_method_id).

Se crearon y poblaron las siguientes tablas:

- clientes
- ordenes
- productos
- metodos_pago

• Decisiones tomadas en el análisis exploratorio

- Se usó la columna order_total para calcular **media, mediana y moda**.
- Se exploró la **distribución de ventas por categoría** y **región**.
- Se analizaron **tendencias mensuales** para detectar estacionalidad.
- Se segmentó a los clientes por edad y género.

- Se buscaron **correlaciones** entre edad y total de la orden, y entre categoría vs método de pago.

• Desafíos encontrados

Desafío	Solución
Timeout al conectar desde Colab a Google SQL	Se habilitó la IP pública de Colab en la configuración de redes
Error en formatos de valores numéricos (%d)	Se cambiaron a formatos flotantes con %.2f
Null en tablas de clientes	Se revisó y ajustó la lógica de carga y separación de IDs

---

C) Metodología de Visualización

• ¿Cómo se seleccionaron las visualizaciones?

Cada tipo de gráfico se eligió basado en la naturaleza del análisis:

Hallazgo	Visualización utilizada	Justificación
Ventas por categoría y región	Barras	Permiten comparar claramente múltiples grupos
Tendencia mensual	Línea	Ideal para ver comportamiento en el tiempo
Productos más vendidos	Barras (Top 10)	Para destacar claramente los productos líderes
Distribución de edad de clientes	Histograma + KDE	Muestra densidad de clientes por edad
Comparación de compras por género	Barras	Comparación directa entre dos grupos
Relación entre edad y total de orden	Diagrama de dispersión	Ver correlaciones o agrupaciones
Relación categoría vs método de pago	Tabla cruzada + resumen por promedio	Muestra diferencias sutiles entre preferencias de pago por categoría