Objetivo General:

Implementar una solución de procesamiento y transformación de datos para una plataforma de e-commerce utilizando la arquitectura Medallion en MySQL(o una base de datos similar). El proyecto se centrará en garantizar la calidad de los datos en cada capa, utilizando dbt para la transformación, macros para la reutilización de lógica y pruebas automatizadas para la validación.

Fuentes de Datos (Simuladas):

Para este proyecto, los estudiantes trabajarán con conjuntos de datos simulados que representan las diferentes fuentes de información de un e-commerce. Estos conjuntos de datos se proporcionarán en formato CSV o Parquet para facilitar la ingesta.

1. **Datos de Clientes:**
    a. customers.csv: Contiene información sobre los clientes (ID del cliente, nombre, email, dirección, fecha de registro, segmento, etc.).
    b. **Posibles problemas de calidad:** Nombres inconsistentes, emails inválidos, direcciones incompletas, duplicados.
2. **Datos de Productos:**
    a. products.csv: Contiene detalles de los productos (ID del producto, nombre, descripción, categoría, precio, SKU, etc.).
    b. **Posibles problemas de calidad:** Descripciones inconsistentes, precios negativos o cero, categorías mal clasificadas.
3. **Datos de Órdenes:**
    a. orders.csv: Contiene información sobre las órdenes realizadas (ID de la orden, ID del cliente, fecha de la orden, estado de la orden, método de pago, etc.).
    b. **Posibles problemas de calidad:** Estados de orden inconsistentes, fechas de orden futuras o pasadas extremas, IDs de cliente o producto inexistentes.
4. **Datos de Detalles de Órdenes:**
    a. order_items.csv: Contiene los detalles de cada ítem dentro de una orden (ID de la orden, ID del producto, cantidad, precio unitario, descuento, etc.).
    b. **Posibles problemas de calidad:** Cantidades negativas o cero, precios unitarios inconsistentes con la tabla de productos, descuentos inválidos.
5. **Datos de Eventos de Navegación:**
    a. web_events.csv: Registros de la actividad de los usuarios en el sitio web (ID de sesión, ID del cliente, timestamp, tipo de evento, URL, etc.).
    b. **Posibles problemas de calidad:** Timestamps inconsistentes, URLs mal formadas, tipos de eventos desconocidos.

dbt (Data Build Tool):

Los estudiantes utilizarán dbt para construir las transformaciones de datos en las capas Silver y Gold. Esto incluirá:

- **Modelos (SQL):** Creación de archivos .sql dentro de la estructura de dbt para definir las transformaciones necesarias en cada capa.
    - **Bronze:** Cargar datos desde los csv como stg_customers, stg_customers, stg_products, stg_orders, stg_order_items, stg_wev_events.
    - **Silver:** Modelos aplicando limpieza básica (removiendo records inválidos, o aplicando un valor por defecto) y modelos combinados como dim_customers, dim_products, fct_orders.
    - **Gold:** Modelos para product_sales_by_category, daily_order_summary, etc.
- **Macros (Jinja):** Desarrollo de macros reutilizables para tareas comunes de calidad de datos y transformación. Ejemplos:
    - Una macro para verificar la validez de un email.
    - Una macro para estandarizar formatos de fecha.
    - Una macro para aplicar lógica condicional en la transformación (ej. asignar un segmento de cliente basado en el gasto).
- **Tests (YAML):** Implementación de pruebas automatizadas de calidad de datos utilizando el framework de testing de dbt. Esto incluirá:
    - **Tests genéricos:** not_null, unique, accepted_values, relationships (para asegurar la integridad referencial entre las tablas).
    - **Tests personalizados (SQL):** Creación de consultas SQL personalizadas para validar reglas de negocio específicas y condiciones de calidad de los datos. Ejemplos:
        - Verificar que los precios de los productos sean siempre positivos.
        - Asegurar que el estado de una orden pertenezca a un conjunto de valores válidos.
        - Validar que la fecha de la orden no sea futura.
        - Comprobar que la cantidad de ítems en una orden sea mayor que cero.

Tareas Específicas:

1. **Configuración del Entorno:**
    a. Configurar una base de datos local (MySQL o PostgreSQL)
    b. Instalar y configurar dbt en su entorno local.
    c. Inicializar un proyecto dbt.
2. **Ingesta de Datos (Bronze Layer):**
    a. Crear tablas externas o gestionadas en la capa Bronze y cargar los datos desde los archivos CSV simulados.

3. **Modelado y Transformación (Silver Layer):**
    a. Crear modelos dbt en la capa Silver para limpiar, estandarizar y conformar los datos de las diferentes fuentes en entidades de negocio coherentes.
    b. Utilizar macros para aplicar lógica de limpieza y estandarización de manera eficiente.
4. **Modelado y Agregación (Gold Layer):**
    a. Crear modelos dbt en la capa Gold para generar agregaciones y métricas útiles para el análisis del e-commerce.
5. **Implementación de Pruebas de Calidad de Datos:**
    a. Definir y escribir pruebas genéricas y personalizadas en dbt para validar la calidad de los datos en las capas Silver y Gold.
    b. Asegurarse de que las pruebas cubran los posibles problemas de calidad identificados en las fuentes de datos.
6. **Uso de Macros:**
    a. Desarrollar y utilizar al menos dos macros personalizadas que se apliquen en las transformaciones o pruebas.