propuesta

Enfocamos el proyecto únicamente en:

- **Acceso masivo a una base de datos**
- **Proceso completo de data cleaning**
- **Documentación detallada de decisiones y prompts utilizados con IA**

💡 ETAPAS DEL TRABAJO

1. Selección y conexión a la base de datos

- Base sugerida: **Chinook** (simula una tienda de música, con clientes, facturas, artistas, etc.)
- Alternativa: algún dataset público en CSV que importamos a una base como PostgreSQL o SQLite.
- Documentar:
  - ¿Dónde está alojada? (local/servidor)
  - ¿Qué motor usamos? (PostgreSQL, SQLite, MySQL)
  - ¿Cómo nos conectamos? (`psycopg2`, `sqlite3`, etc.)

2. Extracción masiva de datos

- Accedemos desde Python con `pandas.read_sql` o similares.
- Justificamos si extraemos todo o solo un subconjunto.
- Guardamos logs o estadísticas del tamaño de los datos.

3. Detección de problemas

- Detección de:
  - Nulos
  - Tipos inconsistentes
  - Duplicados
  - Outliers (opcional)

- Se puede usar un script automatizado que nos diga:
  - % de valores nulos por columna
  - Tipos de datos detectados vs esperados

Prompt que puede ir en el informe:

> "¿Qué tipos de inconsistencias se pueden encontrar al analizar los datos extraídos de una base de datos relacional para limpieza?"

4. Decisiones de diseño

- ¿Se limpia directamente en la DB o fuera (en pandas)?
- ¿Qué hacer con cada tipo de error?
  - ¿Rellenamos nulos? ¿Eliminamos registros?
  - ¿Convertimos tipos?
  - ¿Agrupamos categorías?
- Cada decisión debe ser argumentada:
  - “Eliminamos filas con más del 50% de valores nulos porque...”
  - “Corregimos los tipos con...”

5. Aplicación de limpieza

- Implementación con `pandas`, `numpy` u otras herramientas.
- Aplicamos transformaciones.
- Documentamos los cambios.
- Opción: volver a cargar los datos limpios en una tabla nueva en la base.

6. Informe

- Estructura sugerida:
  1. Introducción
  2. Base de datos y conexión
  3. Extracción masiva

4. Problemas detectados
5. Decisiones de limpieza
6. Implementación del proceso
7. Resultados y verificación
8. Anexo: prompts de IA utilizados