

Taller 1: Modelado de Datos y SQL para la Gestión Industrial

Ingeniería Industrial – Sistemas de Información Gerencial

Febrero 2026

1. Contexto y Motivación

En la industria moderna, la eficiencia no solo se mide en la línea de producción, sino en la capacidad de gestionar la información que esta genera. Un ingeniero industrial que no comprende la estructura de sus datos está condenado a tomar decisiones basadas en intuiciones y no en evidencias.

Este taller busca que el estudiante aprenda a construir el “Plano de Información” de una planta (usando Modelado ER y UML) y a “interrogar” dicha información mediante SQL para obtener indicadores clave de rendimiento (KPIs).

2. Herramientas Requeridas

Para el desarrollo de este taller, utilizaremos herramientas de acceso libre que no requieren instalación:

- **Lucidchart:** Para el modelado gráfico. (<https://www.lucidchart.com>)
- **SQLite Online:** Motor de base de datos en el navegador. (<https://sqliteonline.com>)

3. Fase 1: El Plano de la Fábrica (Modelado ER y UML)

Antes de programar, debemos diseñar. En esta fase, los estudiantes actuarán como arquitectos de información de la empresa *Galletas Industriales S.A.*

3.1. Paso 1: Diagrama Entidad–Relación (ER)

El objetivo es evitar el caos de datos y asegurar la trazabilidad.

1. Ingrese a Lucidchart y cree un nuevo documento en blanco.
2. Active la biblioteca de figuras de Entidad–Relación.
3. Cree una entidad llamada `MATERIA_PRIMA`.
4. Defina como Clave Primaria (PK) el atributo `ID_Lote`.
5. Agregue los atributos: `Nombre`, `Peso_Kg`, `Fecha_Vencimiento`.
6. Cree otra entidad llamada `PROVEEDOR`.

7. Conecte ambas entidades con una relación llamada “Surte”.
8. Defina la cardinalidad: un proveedor (1) puede surtir muchos (N) lotes.

3.2. Paso 2: UML Industrial (Comportamiento)

Mientras el ER guarda datos, el UML diseña el comportamiento del sistema.

1. Active la biblioteca UML en Lucidchart.
2. Cree una clase llamada Maquina_Empaquetadora.
3. Defina los atributos:

- temperatura_actual
- velocidad_rpm
- estado_alerta

4. Defina los métodos:

- iniciar()
- detener_emergencia()
- calibrar_sensor()

4. Fase 2: Interrogando a la Planta (SQL Gerencial)

En esta sección utilizaremos SQL para transformar datos en decisiones.

4.1. Paso 3: Carga de la “Fábrica Invisible”

1. Ingrese a SQLite Online.
2. Ejecute el siguiente código:

```
CREATE TABLE Inventario_Bicis (  
  ID_Bici INT PRIMARY KEY,  
  Modelo TEXT,  
  Costo_Prod REAL  
);  
  
CREATE TABLE Ventas_Semana (  
  ID_Venta TEXT PRIMARY KEY,  
  ID_Bici INT,  
  Precio_Venta REAL,  
  Fecha DATE  
);  
  
INSERT INTO Inventario_Bicis VALUES  
(101, 'MTB-29', 300),  
(102, 'Ruta-Pro', 550),  
(103, 'Urbana', 150);  
  
INSERT INTO Ventas_Semana VALUES
```

```
( 'V-99' , 101 , 450 , '2026-02-20' ),
( 'V-100' , 102 , 800 , '2026-02-21' ),
( 'V-101' , 101 , 450 , '2026-02-22' );
```

4.2. Paso 4: Análisis de Rentabilidad (JOIN)

Realice una consulta que muestre:

- Modelo
- Precio de venta
- Costo de producción
- Ganancia neta (Precio_Venta - Costo_Prod)

Reflexione:

¿Qué significa si al realizar un LEFT JOIN aparece un valor NULL en ventas?

5. Instrucciones de Entrega

El informe debe entregarse en formato PDF por grupos de máximo tres personas antes del domingo 1 de marzo a las 23:59.

Debe incluir:

- Capturas del diagrama ER.
- Capturas del diagrama UML.
- Resultados de las consultas SQL.

6. Rúbrica de Evaluación

Criterio	Peso	Descripción
Modelado ER/UML	30 %	Correcta definición de entidades, claves y métodos.
Precisión SQL	30 %	Uso adecuado de JOIN y funciones de agregación.
Análisis Gerencial	30 %	Interpretación estratégica de resultados.
Calidad del Informe	10 %	Organización y claridad.

Guía Práctica: Gestión de Producción con SQL y SQLite

Ingeniería Industrial – Sistemas de Información Gerencial

Febrero 2026

1. Introducción

En la gestión moderna de producción, los datos deben ser íntegros y escalables. Esta guía muestra cómo pasar de un modelo relacional a una base de datos funcional para gestionar proveedores y materias primas.

2. Paso 1: Creación del Modelo (DDL)

Lo primero es definir nuestras entidades. La clave es la **Llave Foránea (FK)**, que conecta el insumo con su origen.

```
-- Tabla de proveedores (Maestra)
CREATE TABLE 'PROVEEDOR' (
  'ID_Proveedor' INT PRIMARY KEY,
  'Nombre' VARCHAR(100),
  'Telefono' VARCHAR(20),
  'Direccion' VARCHAR(150)
);

-- Tabla de materias primas (Operativa)
CREATE TABLE 'MATERIA_PRIMA' (
  'ID_Lote' INT PRIMARY KEY,
  'Nombre' VARCHAR(100),
  'Peso_Kg' DECIMAL(10,2),
  'Fecha_Vencimiento' DATE,
  'ID_Proveedor' INT,
  FOREIGN KEY ('ID_Proveedor') REFERENCES 'PROVEEDOR'('ID_Proveedor')
);
```

Listing 1: Creación de tablas con integridad referencial

3. Paso 2: Generación Masiva de Datos

Para simular un entorno real de producción, generamos 1,000 registros automáticamente usando recursividad.

```
INSERT INTO 'PROVEEDOR'
WITH RECURSIVE cnt(x) AS (
  SELECT 1 UNION ALL SELECT x + 1 FROM cnt WHERE x < 1000
)
SELECT x, 'Proveedor ' || x, '555-' || x, 'Sede Ind. ' || x FROM cnt;
```

Listing 2: Inserción masiva para pruebas de estrés

4. Paso 3: Análisis y Consultas (DML)

Una base de datos sirve para tomar decisiones. Aquí tres consultas vitales para un ingeniero de producción:

4.1. Trazabilidad Total (JOIN)

Permite saber qué proveedor entregó cada lote específico.

```
SELECT m.ID_Lote, m.Nombre, p.Nombre AS Proveedor
FROM MATERIA_PRIMA m
JOIN PROVEEDOR p ON m.ID_Proveedor = p.ID_Proveedor;
```

4.2. Control de Vencimientos

Identificar material que expira en los próximos 30 días para priorizar su uso (FEFO - First Expired, First Out).

```
SELECT Nombre, Fecha_Vencimiento
FROM MATERIA_PRIMA
WHERE Fecha_Vencimiento BETWEEN date('now') AND date('now', '+30 days');
```

5. Paso 4: Solución de Errores Comunes

Si recibes el error `UNIQUE constraint failed`, significa que intentas duplicar una Llave Primaria.

- **Solución:** Usa `INSERT OR REPLACE` si quieres actualizar el registro existente, o `DELETE FROM` tabla para limpiar antes de re-cargar.

6. Paso 5: Exportación e Integración

Para reportes gerenciales, exporta los resultados a **CSV** desde el motor SQL e impórtalos en **Power BI** o **Excel**. Esto permite crear Dashboards de inventario dinámicos.