

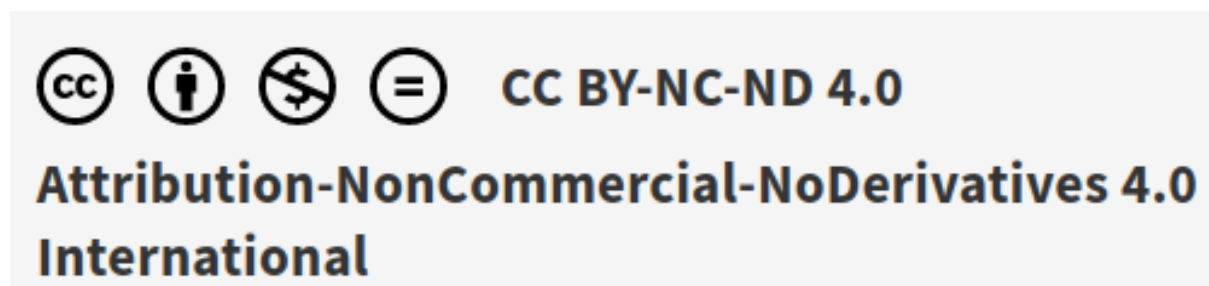
# Departamento de Informática

## Proyecto Excel2Database

Juan Gutierrez

Octubre 2025

Este documento se encuentra bajo una licencia Creative Commons de Atribución-CompartirIgual (CC BY-SA).



**Figura 1:** Atribución-CompartirIgual (CC BY-SA)

Esto significa que puedes:

- Compartir: copiar y redistribuir el material en cualquier medio o formato.
- Adaptar: remezclar, transformar y construir sobre el material para cualquier propósito, incluso comercialmente.

Bajo las siguientes condiciones:

- Atribución: debes dar crédito de manera adecuada, proporcionar un enlace a la licencia e indicar si se han realizado cambios. Puedes hacerlo de cualquier manera razonable, pero no de una manera que sugiera que el licenciante te respalda a ti o al uso que hagas del trabajo.
- Compartir igual: si remezclas, transformas o creas a partir del material, debes distribuir tus contribuciones bajo la misma licencia que el original.

Para más detalles, consulta la licencia completa.

---

## Índice

<b>RA2: “Desarrolla aplicaciones que gestionan información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión”*</b>	<b>3</b>
Proyecto: <i>Excel2Database</i> . . . . .	3
Descripción general . . . . .	3
Objetivos de aprendizaje . . . . .	3
Descripción funcional . . . . .	3
Requisitos técnicos . . . . .	4
Rúbrica de evaluación (RA2) . . . . .	5

## RA2: “Desarrolla aplicaciones que gestionan información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión”\*

### Proyecto: *Excel2Database*

#### Descripción general

El proyecto **Excel2Database** tiene como objetivo desarrollar una aplicación **Java (Maven)** capaz de **importar y exportar información** entre un archivo **Excel (.xlsx)** y una **base de datos MySQL**.

Este ejercicio permitirá al alumnado **consolidar los fundamentos del acceso a datos relacionales** mediante JDBC, el uso de ficheros, la manipulación de estructuras de datos y la configuración de proyectos Maven con dependencias externas.

#### Objetivos de aprendizaje

- Comprender el funcionamiento del **conector JDBC** y el proceso de conexión a MySQL.
- Aplicar el acceso a datos relacionales mediante **sentencias SQL** desde Java.
- Practicar la **lectura y escritura de ficheros Excel** mediante la librería **Apache POI**.
- Diseñar un programa estructurado y modular que cumpla un objetivo realista y completo.
- Configurar la aplicación a través de un fichero **.properties** externo.
- Integrar herramientas de gestión de dependencias (Maven) y documentación del código.

#### Descripción funcional

El programa debe ejecutarse desde línea de comandos con esta sintaxis:

```
1 excel2database -f fichero.xlsx -db nombreBD
```

#### Fases del proyecto:

##### 1. Lectura de configuración

- El programa cargará los parámetros de conexión desde un fichero **config.properties** :

```
1 host=localhost
2 port=3306
3 database=agenda
```

```
4 user=root
5 password=****
```

## 2. Análisis del libro Excel

- Cada **hoja** del libro representa una **tabla** de la base de datos.
- La **primera fila** contiene los nombres de los campos.
- La **segunda fila** define el **tipo de dato** (por ejemplo: `INT`, `VARCHAR(50)`, `DATE`).
- Las filas siguientes son los **registros**.

## 3. Creación de la base de datos y tablas

- Se generarán automáticamente las sentencias SQL `CREATE TABLE` según la estructura detectada.

## 4. Inserción de los datos

- Se leerán los registros del Excel e insertarán en la base de datos con sentencias `INSERT`.

## 5. Exportación inversa

- El programa permitirá la operación inversa: exportar una base de datos MySQL a un nuevo fichero Excel.

## Requisitos técnicos

- **Lenguaje:** Java 17 o superior
- **Gestor de dependencias:** Maven
- **Dependencias mínimas:**

```
1 <dependencies>
2   <dependency>
3     <groupId>org.apache.poi</groupId>
4     <artifactId>poi-ooxml</artifactId>
5     <version>5.2.5</version>
6   </dependency>
7   <dependency>
8     <groupId>mysql</groupId>
9     <artifactId>mysql-connector-j</artifactId>
10    <version>8.4.0</version>
11  </dependency>
12 </dependencies>
```

- El código debe incluir **tratamiento de excepciones**, **mensajes informativos** y **comentarios de documentación** (`/** ... */`).

**Rúbrica de evaluación (RA2)**

<b>Criterio de evaluación (según PD)</b>	<b>Indicadores observables</b>	<b>Excelente (9-10)</b>	<b>Notable (7-8)</b>	<b>Aprobado (5-6)</b>	<b>Insuficiente (&lt;5)</b>	<b>Peso</b>
<b>a)</b> Valora las ventajas e inconvenientes de los conectores.	Explica en memoria el uso del conector MySQL y las razones de su elección.	Justifica correctamente el uso de JDBC y las ventajas frente a otras alternativas.	Describe adecuadamente el uso de JDBC, con ejemplos.	Lo usa sin justificarlo.	No explica ni comprende su uso.	10%
<b>b)</b> Utiliza gestores embebidos o independientes.	Prueba la conexión con MySQL desde código y demuestra comprensión.	Configura correctamente la conexión y demuestra autonomía.	Configura con ayuda o correcciones.	Configura parcialmente.	No consigue conectar.	10%
<b>c)</b> Usa el conector idóneo en la aplicación.	Incluye correctamente el driver JDBC en <code>pom.xml</code> .	Correcta configuración y documentación.	Correcta pero sin documentar.	Usa dependencias redundantes o incompletas.	No funciona el conector.	10%
<b>d)</b> Establece la conexión.	Ejecuta correctamente <code>DriverManager.getConnection()</code> con parámetros del <code>.properties</code> .	Conexión estable y gestionada con cierre correcto ( <code>try-with-resources</code> ).	Conexión correcta, cierre manual.	Conexión funciona pero sin gestionar bien excepciones.	Error de conexión o parámetros.	5%
<b>e)</b> Define la estructura de la base de datos.	Crea las tablas desde el Excel analizado.	Genera correctamente el SQL dinámico <code>CREATE TABLE</code> .	Genera las tablas con pequeñas imprecisiones.	Las crea parcialmente.	No crea tablas válidas.	10%

<b>Criterio de evaluación (según PD)</b>	<b>Indicadores observables</b>	<b>Excelente (9-10)</b>	<b>Notable (7-8)</b>	<b>Aprobado (5-6)</b>	<b>Insuficiente (&lt;5)</b>	<b>Peso</b>
<b>f)</b> Desarrolla aplicaciones que modifican el contenido.	Inserta registros correctamente.	Inserta todos los datos sin error y controla duplicados.	Inserta correctamente con leves errores.	Inserta parcialmente.	No inserta datos.	15%
<b>g)</b> Define objetos para almacenar resultados.	Crea clases para representar tablas y campos.	Estructura clara con POJOs y uso de colecciones.	Clases correctas sin estructura óptima.	Estructura funcional mínima.	No modela los datos.	10%
<b>h)</b> Desarrolla aplicaciones que efectúan consultas.	Implementa lectura y exportación de datos a Excel.	Exporta correctamente con Apache POI.	Exporta con pequeños fallos.	Exporta parcialmente.	No exporta.	15%
<b>i)</b> Elimina objetos al finalizar su función.	Libera recursos (streams, conexiones).	Cierra todos los recursos correctamente.	Cierra parcialmente.	Cierre manual incompleto.	No libera recursos.	5%
<b>j)</b> Gestiona transacciones.	Aplica <code>commit</code> y <code>rollback</code> correctamente.	Controla transacciones correctamente con feedback al usuario.	Usa transacciones básicas.	Sin gestión explícita.	Genera inconsistencias.	10%

**Total:** 100%