

A partir del proyecto adjunto a este enunciado realizar los siguientes pasos

### Instrucciones para importar el proyecto en IntelliJ IDEA

1. **Descomprime el archivo ZIP** en una carpeta de tu ordenador.
2. Abre IntelliJ IDEA y selecciona **File > Open....**
3. Navega hasta la carpeta donde descomprimos el proyecto y selecciona la carpeta raíz del proyecto.
4. IntelliJ detectará automáticamente que es un proyecto Maven y mostrará una notificación para importar el proyecto. Haz clic en **Import Maven Projects**.
5. Si no aparece la notificación, haz clic derecho sobre el archivo **pom.xml** y selecciona **Add as Maven Project**.

### Cómo cargar las dependencias de Maven

- Al importar el proyecto, IntelliJ descargará automáticamente las dependencias definidas en el archivo **pom.xml**.
- **Si alguna dependencia no se descarga**, haz clic derecho sobre el proyecto y selecciona **Maven > Reload Project**.
- También puedes ir a la pestaña Maven (a la izquierda) y pulsar el icono de recarga.

### Estructura y explicación de los archivos y paquetes

- **src/main/java/com/gameshelf/model/Videojuego.java**: Clase que representa la entidad Videojuego con sus atributos y métodos.
- **src/main/java/com/gameshelf/dao/VideojuegoDAO.java**: Interfaz que define los métodos de acceso a datos para videojuegos.
- **src/main/java/com/gameshelf/dao/VideojuegoDAOImpl.java**: Aquí debes implementar la lógica de acceso a datos siguiendo la interfaz. Este es el **único archivo que debes modificar**, concretamente debes darle funcionamiento a los métodos marcados con un comentario que comienza por "TO-DO".
- **src/main/java/com/gameshelf/DatabaseFactory.java**: Clase de utilidad para obtener conexiones a la base de datos y realizar la inicialización.
- **src/test/java/com/gameshelf/dao/VideojuegoDAOTest.java**: Tests unitarios que comprueban el correcto funcionamiento de los métodos del DAO.
- **pom.xml**: Archivo de configuración de Maven donde se definen las dependencias y la configuración del proyecto.
- Sigue estos pasos para empezar a trabajar y consulta los tests para comprobar si tu implementación es correcta.

En la [página siguiente](#) se detallan más los requisitos que deben cumplir los métodos.

Implementa la clase VideojuegoDAOImpl que debe cumplir con la interfaz VideojuegoDAO proporcionada. Esta clase gestionará el acceso a la base de datos para la entidad Videojuego usando JDBC.

**Requisitos:**

- Implementa todos los métodos definidos en la interfaz VideojuegoDAO.
- Antes de insertar o actualizar un videojuego, valida que:
  - El título no es nulo ni vacío.
  - La plataforma no será nula ni vacía.
  - El precio sea mayor que 0.
  - El stock no es negativo.
  - Si alguna de estas **condiciones no se cumple, lanza** una excepción **IllegalArgumentException** con un **mensaje descriptivo**.
- Utiliza **PreparedStatement** para las operaciones de inserción, actualización y borrado.
- El método **precioMedioPorPlataforma**, debe devolver un mapa que contenga pares clave valor, dónde la clave es la plataforma y el valor es el precio medio de los videojuegos de dicha plataforma.
- El método **registrarVenta** debe disminuir el stock solo si hay unidades disponibles y registrar la venta en la tabla correspondiente. **Esto debe controlarse mediante una transacción.**
- El resto de métodos deben funcionar correctamente según lo especificado en la interfaz.
- Adjunta la implementación y asegúrate de que pasa los tests unitarios que validan tanto operaciones correctas como intentos de insertar o actualizar datos inválidos.

Una vez terminado y probado el programa para entregarlo debe comprimirse la carpeta del proyecto de IntelliJ a un archivo zip y darle el siguiente nombre:

**apellido\_nombre\_examenTema7.zip**

El archivo debe subirse al aula virtual en la entrega correspondiente antes de que termine la hora límite de entrega.