

Programación

UD14 – Acceso a Bases de Datos Relacionales

Práctica 0

Ejercicios de Serialización y Deserialización en JSON y XML

1. Serializar y deserializar un objeto Persona en JSON y XML

- Crea una clase Persona con atributos nombre, edad y correo.
- Usa Jackson para convertir el objeto a JSON y guardarlo en un archivo.
- Usa JAXB para convertirlo a XML y guardarlo.
- Luego, deserializa los archivos y verifica que los datos sean correctos.

2. Serializar y deserializar una lista de productos en JSON y XML

- Define una clase Producto con atributos id, nombre y precio.
- Crea una lista de objetos Producto y conviértela a JSON y XML.
- Guarda los archivos y recupéralos deserializando su contenido.

3. Configuración de aplicación en JSON y XML

- Define una clase Configuracion con atributos como tema, idioma y volumen.
- Guarda su estado en un archivo JSON (config.json) y otro XML (config.xml).
- Al iniciar la aplicación, carga la configuración deserializándola desde los archivos.

4. Serialización con cifrado en JSON y XML

- Crea un objeto Usuario con id, nombre, y contraseña.
- Antes de guardar los datos, encripta la contraseña usando Base64 o un algoritmo de hash.
- Serializa el objeto a JSON y XML.
- Al deserializar, descifra la contraseña y verifica su integridad.

5. Serialización con Herencia en JSON y XML

- Define una clase Empleado con atributos id, nombre y salario.
- Luego, crea una subclase Gerente que añade bono.
- Serializa un objeto Gerente a JSON y XML.
- Deserializa los archivos y verifica que los datos de la subclase sean correctos.

Ejercicios sobre Interfaces de Formularios en Java (Swing o JavaFX)**1. Crear una ventana simple en Java Swing**

- Usa JFrame para crear una ventana.
- Agrega un JLabel con el texto "¡Hola, mundo!".

2. Crear un formulario de inicio de sesión

- Diseña una interfaz con JTextField (usuario) y JPasswordField (contraseña).
- Agrega un botón "Iniciar sesión".
- Muestra un mensaje si el usuario y la contraseña son correctos (ejemplo: "admin" / "1234").

3. Agregar eventos a botones en Swing

- Crea un formulario con un campo de texto y un botón.
- Cuando el usuario escriba su nombre y presione el botón, muestra "Hola, <nombre>!" en una etiqueta.

4. Crear un formulario de registro y guardar datos en MySQL

- Diseña un formulario con JTextField (nombre, correo) y un botón "Registrar".
- Al presionar el botón, guarda los datos en la base de datos MySQL.

5. Crear una tabla en Swing para mostrar datos de MySQL

- Usa JTable para mostrar una lista de usuarios obtenida de MySQL.
- Agrega un botón para actualizar la tabla con nuevos datos.

Ejercicios sobre Bases de Datos MySQL en Java (JDBC)

Para estos ejercicios, usaremos JDBC para conectar con MySQL.

1. Conectar Java con MySQL

- Configura un proyecto en Java y agrega el conector JDBC (mysql-connector-java).
- Escribe un programa que establezca una conexión con una base de datos MySQL y muestre un mensaje de éxito o error.

2. Insertar datos en una tabla MySQL

- Crea una tabla llamada usuarios con las columnas id, nombre y email.
- Usa un PreparedStatement para insertar registros desde Java.
- Asegúrate de manejar excepciones correctamente.

3. Consultar datos de la base de datos

- Recupera y muestra en consola los datos de la tabla usuarios usando ResultSet.
- Implementa una consulta que filtre usuarios por nombre.

4. Actualizar y eliminar datos en MySQL desde Java

- Crea un método que permita actualizar el correo de un usuario por su ID.
- Crea otro método que elimine un usuario de la base de datos por su ID.

5. Crear una interfaz con Java y MySQL

- Diseña una aplicación de consola que permita:
- Crear un nuevo usuario.
- Leer todos los usuarios.
- Actualizar datos de un usuario.
- Eliminar un usuario.
- Utiliza Scanner para ingresar datos desde el teclado.