

Tema 12

Persistencia de Datos:

- Definición: La capacidad de los datos de sobrevivir más allá de la ejecución del programa que los creó.
- En POO: Capacidad de los objetos de existir más allá del ciclo de vida del programa.

Introducción a JDBC:

- JDBC (Java Database Connectivity) es el estándar de Java para la conectividad entre bases de datos y Java.
- JDBC API permite a las aplicaciones Java:
 - Conectarse a bases de datos.
 - Enviar sentencias SQL.
 - Procesar los resultados.

Componentes JDBC:

- **JDBC API:** Conjunto de interfaces y clases que los programadores usan para interactuar con las bases de datos.
- **JDBC Driver:** Software que permite la comunicación entre la API JDBC y una base de datos específica.

Drivers JDBC:

- Tipos de Drivers:
 - **Tipo 1 (JDBC-ODBC Bridge):** Usa ODBC. Desaconsejado.
 - **Tipo 2 (Native-API Driver):** Usa API nativa del DBMS.
 - **Tipo 3 (Network Protocol Driver):** Usa protocolo de red del DBMS.
 - **Tipo 4 (Thin Driver):** Driver puro Java. Recomendado.

Conexión a la Base de Datos:

- Proceso:
 1. Cargar el driver.
 2. Establecer la conexión con la base de datos.
- Clase DriverManager: Gestiona la carga de drivers y la creación de conexiones.
- Método getConnection(): Establece la conexión.

Interfaces Statement y PreparedStatement:

- Usadas para ejecutar sentencias SQL.
- Statement: Para sentencias SQL simples.
- PreparedStatement: Para sentencias SQL parametrizadas (más eficiente y seguro).
- Métodos para ejecutar: executeQuery() (para SELECT), executeUpdate() (para INSERT, UPDATE, DELETE).
- Método close(): Cierra la sentencia.

ResultSet:

- Resultado de una consulta SELECT.
- Permite acceder a las filas y columnas del resultado.
- Métodos:
 - next(): Moverse a la siguiente fila.
 - getXxxx(): Obtener el valor de una columna (donde Xxxx es el tipo de dato).

¿Cuál es la definición de "persistencia de datos" en el contexto de la programación?

- a) La velocidad de acceso a los datos.
- b) La capacidad de los datos de sobrevivir más allá de la ejecución del programa.
- c) La seguridad de los datos.
- d) La organización de los datos en estructuras complejas.

¿Qué significa la persistencia en la Programación Orientada a Objetos?

- a) La capacidad de los objetos de ser creados y destruidos rápidamente.
- b) La capacidad de los objetos de existir solo durante la ejecución del programa.
- c) La capacidad de los objetos de existir más allá del ciclo de vida del programa.
- d) La capacidad de los objetos de acceder a la memoria directamente.

¿Qué significa JDBC?

- a) Java Development Connection
- b) Java Database Connectivity
- c) Java Data Component
- d) Java Distributed Computing

¿Cuál es la función principal de JDBC?

- a) Definir la sintaxis del lenguaje Java.
- b) Proporcionar una API para la conectividad entre bases de datos y Java.
- c) Gestionar la memoria en aplicaciones Java.
- d) Crear interfaces gráficas de usuario.

¿Qué permite a las aplicaciones Java la API JDBC?

- a) Definir nuevas estructuras de datos.

- b) Conectarse a bases de datos, enviar sentencias SQL y procesar los resultados.
- c) Realizar cálculos matemáticos complejos.
- d) Controlar el acceso al sistema operativo.

¿Cuáles son los dos componentes principales de JDBC?

- a) Clases y objetos.
- b) Variables y constantes.
- c) JDBC API y JDBC Driver.
- d) Métodos y constructores.

¿Qué es un JDBC Driver?

- a) Una clase de Java para la gestión de errores.
- b) Una interfaz para la creación de interfaces gráficas.
- c) Software que permite la comunicación entre la API JDBC y una base de datos específica.
- d) Un tipo de base de datos.

¿Cuál de los siguientes NO es un tipo de Driver JDBC?

- a) Tipo 1 (JDBC-ODBC Bridge)
- b) Tipo 2 (Native-API Driver)
- c) Tipo 3 (Network Protocol Driver)
- d) Tipo 5 (SQL Driver)

¿Cuál es la principal desventaja de un Driver JDBC de Tipo 1 (JDBC-ODBC Bridge)?

- a) Es muy rápido.
- b) Es puro Java.
- c) Usa ODBC, lo que puede ser ineficiente.
- d) Es el más seguro.

¿Cuál es la principal ventaja de un Driver JDBC de Tipo 4 (Thin Driver)?

- a) Es el más rápido.
- b) Es nativo del sistema operativo.

c) Es puro Java, lo que lo hace multiplataforma.

d) Usa el protocolo de red del DBMS.

¿Cuál es el primer paso para conectarse a una base de datos usando JDBC?

a) Ejecutar la consulta SQL.

b) Cerrar la conexión.

c) Cargar el driver.

d) Procesar los resultados.

¿Qué clase de Java se utiliza para gestionar la carga de drivers y la creación de conexiones?

a) Connection

b) Statement

c) DriverManager

d) ResultSet

¿Qué método se utiliza para establecer una conexión con una base de datos en JDBC?

a) connect()

b) getConnection()

c) executeQuery()

d) executeUpdate()

¿Qué interfaces de JDBC se utilizan para ejecutar sentencias SQL?

a) Driver y Connection

b) Statement y PreparedStatement

c) ResultSet y DatabaseMetaData

d) SQLException y Warning

¿Cuál es la principal diferencia entre Statement y PreparedStatement?

a) Statement es más seguro.

b) PreparedStatement se usa para sentencias SQL simples.

c) PreparedStatement permite sentencias SQL parametrizadas, lo que es más eficiente y seguro.

d) No hay diferencia.

¿Qué método se utiliza para ejecutar una sentencia SQL de tipo SELECT?

a) executeUpdate()

b) executeQuery()

c) connect()

d) close()

¿Qué método se utiliza para ejecutar una sentencia SQL de tipo INSERT, UPDATE o DELETE?

a) executeQuery()

b) getConnection()

c) executeUpdate()

d) next()

¿Qué método se utiliza para cerrar una sentencia JDBC?

a) connect()

b) execute()

c) close()

d) next()

¿Qué representa un ResultSet en JDBC?

a) La conexión a la base de datos.

b) El resultado de ejecutar una consulta SELECT.

c) Una sentencia SQL.

d) Los metadatos de la base de datos.

¿Qué método se utiliza para moverse a la siguiente fila en un ResultSet?

a) get()

b) next()

c) read()

d) fetch()

¿Qué tipo de métodos se utilizan para acceder al valor de una columna en un ResultSet?

a) setXxxx()

b) getXxxx()

c) addXxxx()

d) deleteXxxx()

En el método getXxxx(), ¿qué representa "Xxxx"?

a) El nombre de la tabla.

b) El nombre de la columna.

c) El tipo de dato de la columna.

d) El número de la fila.

¿Cuál de las siguientes afirmaciones es CIERTA sobre PreparedStatement?

a) Es menos eficiente que Statement.

b) No se puede usar para sentencias parametrizadas.

c) Ayuda a prevenir inyecciones SQL.

d) No necesita ser compilado.

¿Por qué es importante cerrar las sentencias JDBC?

a) Para liberar recursos del sistema y de la base de datos.

b) Para abrir la conexión a la base de datos.

c) Para ejecutar la consulta SQL.

d) No es necesario cerrarlas.

¿Cuál de las siguientes opciones describe mejor la relación entre JDBC y una base de datos?

a) JDBC es un tipo de base de datos.

b) JDBC permite a Java comunicarse con diferentes bases de datos.

c) JDBC reemplaza a las bases de datos.

d) No hay relación entre JDBC y las bases de datos.