

CUESTIÓN 3. Java CRUD

▪ **Gestión de base de datos relacionales.** Utilizando MySQL crea una base de datos en donde dispongamos de una tabla que permita almacenar lo siguientes datos de productos (idProducto, nombre, fechaEnvasado, unidades, precio, disponible). Disponible será un valor booleano.

```
CREATE DATABASE IF NOT EXISTS hito1t3;
```

```
USE hito1t3;
```

```
CREATE TABLE IF NOT EXISTS productos (
```

```
    idProducto INT AUTO_INCREMENT PRIMARY KEY,
```

```
    nombre VARCHAR(255) NOT NULL,
```

```
    fechaEnvasado DATE NOT NULL,
```

```
    unidades INT NOT NULL,
```

```
    precio DECIMAL(10, 2) NOT NULL,
```

```
    disponible BOOLEAN NOT NULL
```

```
);
```

```
INSERT INTO productos (nombre, fechaEnvasado, unidades, precio, disponible)
```

```
VALUES
```

```
    ('Producto A', '2024-04-29', 100, 15.99, true),
```

```
    ('Producto B', '2024-04-30', 50, 10.50, true),
```

```
    ('Producto C', '2024-04-28', 200, 25.75, false),
```

```
    ('Producto D', '2024-04-27', 75, 8.25, true),
```

```
    ('Producto E', '2024-04-26', 120, 18.49, true),
```

```
    ('Producto F', '2024-04-25', 80, 12.75, false),
```

```
    ('Producto G', '2024-04-24', 150, 20.00, true),
```

```
    ('Producto H', '2024-04-23', 90, 14.99, true),
```

```
    ('Producto I', '2024-04-22', 110, 30.25, false),
```

```
    ('Producto J', '2024-04-21', 70, 9.99, true);
```

- Realiza una aplicación que nos permita añadir, eliminar, actualizar y mostrar los productos. Realizar esta tarea por consola, utiliza funciones.

Módulo Producto.java

```
package hito1_cuestion_3;

public class Producto {

    private int id;
    private String nombre;
    private String fechaEnvasado;
    private int unidades;
    private double precio;
    private boolean disponible;

    public Producto() {}

    public Producto(String nombre, String fechaEnvasado, int unidades, double
precio, boolean disponible) {
        this.nombre = nombre;
        this.fechaEnvasado = fechaEnvasado;
        this.unidades = unidades;
        this.precio = precio;
        this.disponible = disponible;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getFechaEnvasado() {
        return fechaEnvasado;
    }

    public void setFechaEnvasado(String fechaEnvasado) {
        this.fechaEnvasado = fechaEnvasado;
    }

    public int getUnidades() {
        return unidades;
    }

    public void setUnidades(int unidades) {
        this.unidades = unidades;
    }
}
```

```

    }

    public double getPrecio() {
        return precio;
    }

    public void setPrecio(double precio) {
        this.precio = precio;
    }

    public boolean isDisponible() {
        return disponible;
    }

    public void setDisponible(boolean disponible) {
        this.disponible = disponible;
    }
}

```

Módulo ProductoDAO.java

```

package hito1_cuestion_3;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class ProductoDAO {

    private Connection con;

    public ProductoDAO(Connection con) {
        this.con = con;
    }

    public boolean insertarProducto(Producto producto) {
        String query = "INSERT INTO productos (nombre, fechaEnvasado, unidades, precio, disponible) VALUES (?, ?, ?, ?, ?)";

        try (PreparedStatement stmt = con.prepareStatement(query)) {
            stmt.setString(1, producto.getNombre());
            stmt.setString(2, producto.getFechaEnvasado());
            stmt.setInt(3, producto.getUnidades());
            stmt.setDouble(4, producto.getPrecio());
            stmt.setBoolean(5, producto.isDisponible());
            stmt.executeUpdate();
            return true;
        } catch (SQLException e) {
            e.printStackTrace();
            return false;
        }
    }

    public void mostrarProductos() {
        String query = "SELECT * FROM productos";
    }
}

```

```

        try (Statement stmt = con.createStatement();
             ResultSet rs = stmt.executeQuery(query)) {
            while (rs.next()) {
                System.out.println("ID Producto: " + rs.getInt("idProducto")
+ ", Nombre: " + rs.getString("nombre") + ", Fecha de Envasado: " +
                rs.getString("fechaEnvasado") + ", Unidades: " +
                rs.getInt("unidades") + ", Precio: " + rs.getDouble("precio") + ",
                Disponible: " + rs.getBoolean("disponible"));
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    public boolean actualizarProducto(Producto producto) {
        String query = "UPDATE productos SET nombre = ?, fechaEnvasado = ?,
unidades = ?, precio = ?, disponible = ? WHERE idProducto = ?";

        try (PreparedStatement stmt = con.prepareStatement(query)) {
            stmt.setString(1, producto.getNombre());
            stmt.setString(2, producto.getFechaEnvasado());
            stmt.setInt(3, producto.getUnidades());
            stmt.setDouble(4, producto.getPrecio());
            stmt.setBoolean(5, producto.isDisponible());
            stmt.setInt(6, producto.getId());
            int rowsAffected = stmt.executeUpdate();
            return rowsAffected > 0;
        } catch (SQLException e) {
            e.printStackTrace();
            return false;
        }
    }

    public boolean eliminarProducto(int idProducto) {
        String query = "DELETE FROM productos WHERE idProducto = ?";

        try (PreparedStatement stmt = con.prepareStatement(query)) {
            stmt.setInt(1, idProducto);
            int rowsAffected = stmt.executeUpdate();
            return rowsAffected > 0;
        } catch (SQLException e) {
            e.printStackTrace();
            return false;
        }
    }
}

```

Módulo Main.java

```

package hito1_cuestion_3;

import java.sql.*;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {

```

```

        try (Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/hito1t3", "root",
"")) {
            ProductoDAO productoDAO = new ProductoDAO(con);
            Scanner scanner = new Scanner(System.in);

            int opcion;
            do {
                System.out.println("Menú:");
                System.out.println("1. Añadir producto");
                System.out.println("2. Mostrar productos");
                System.out.println("3. Actualizar producto");
                System.out.println("4. Eliminar producto");
                System.out.println("5. Salir");
                System.out.print("Ingrese su opción: ");
                opcion = scanner.nextInt();

                switch (opcion) {
                    case 1:
                        agregarProducto(productoDAO, scanner);
                        break;
                    case 2:
                        productoDAO.mostrarProductos();
                        break;
                    case 3:
                        actualizarProducto(productoDAO, scanner);
                        break;
                    case 4:
                        eliminarProducto(productoDAO, scanner);
                        break;
                    case 5:
                        System.out.println("Saliendo del programa...");
                        break;
                    default:
                        System.out.println("Opción inválida. Por favor,
ingrese una opción válida.");
                }
            } while (opcion != 5);

            } catch (SQLException e) {
                e.printStackTrace();
            }
        }

        private static void agregarProducto(ProductoDAO productoDAO, Scanner
scanner) {
            System.out.println("Ingrese los datos del nuevo producto:");
            System.out.print("Nombre: ");
            String nombre = scanner.next();
            System.out.print("Fecha de envasado (YYYY-MM-DD): ");
            String fechaEnvasado = scanner.next();
            System.out.print("Unidades: ");
            int unidades = scanner.nextInt();
            System.out.print("Precio: ");
            double precio = scanner.nextDouble();
            System.out.print("Disponible (true/false): ");
            boolean disponible = scanner.nextBoolean();

```

```

        Producto nuevoProducto = new Producto(nombre, fechaEnvasado,
unidades, precio, disponible);
        boolean exito = productoDAO.insertarProducto(nuevoProducto);
        if (exito) {
            System.out.println("Producto agregado correctamente.");
        } else {
            System.out.println("Error al agregar el producto.");
        }
    }

    private static void actualizarProducto(ProductoDAO productoDAO, Scanner
scanner) {
        System.out.println("Ingrese el ID del producto a actualizar:");
        int idProducto = scanner.nextInt();
        System.out.println("Ingrese los nuevos datos del producto:");
        System.out.print("Nombre: ");
        String nombre = scanner.next();
        System.out.print("Fecha de envasado (YYYY-MM-DD): ");
        String fechaEnvasado = scanner.next();
        System.out.print("Unidades: ");
        int unidades = scanner.nextInt();
        System.out.print("Precio: ");
        double precio = scanner.nextDouble();
        System.out.print("Disponible (true/false): ");
        boolean disponible = scanner.nextBoolean();

        Producto productoActualizado = new Producto(nombre, fechaEnvasado,
unidades, precio, disponible);
        productoActualizado.setId(idProducto);
        boolean exito = productoDAO.actualizarProducto(productoActualizado);
        if (exito) {
            System.out.println("Producto actualizado correctamente.");
        } else {
            System.out.println("Error al actualizar el producto.");
        }
    }

    private static void eliminarProducto(ProductoDAO productoDAO, Scanner
scanner) {
        System.out.println("Ingrese el ID del producto a eliminar:");
        int idProducto = scanner.nextInt();
        boolean exito = productoDAO.eliminarProducto(idProducto);
        if (exito) {
            System.out.println("Producto eliminado correctamente.");
        } else {
            System.out.println("Error al eliminar el producto.");
        }
    }
}

```

Resultado:

```
Menú:
1. Añadir producto
2. Mostrar productos
3. Actualizar producto
4. Eliminar producto
5. Salir
Ingrese su opción: 1
Ingrese los datos del nuevo producto:
Nombre: zapatillas
Fecha de envasado (YYYY-MM-DD): 2020-10-12
Unidades: 2
Precio: 20
Disponible (true/false): true
Producto agregado correctamente.
```

```
Menú:
1. Añadir producto
2. Mostrar productos
3. Actualizar producto
4. Eliminar producto
5. Salir
Ingrese su opción: 2
ID Producto: 2, Nombre: Producto A, Fecha de Envasado: 2024-04-29, Unidades: 100, Precio: 15.99, Disponible: true
ID Producto: 3, Nombre: Producto B, Fecha de Envasado: 2024-04-30, Unidades: 50, Precio: 10.5, Disponible: true
ID Producto: 4, Nombre: Producto C, Fecha de Envasado: 2024-04-28, Unidades: 200, Precio: 25.75, Disponible: false
ID Producto: 5, Nombre: Producto Actualizado, Fecha de Envasado: 2024-04-30, Unidades: 150, Precio: 15.75, Disponible: false
ID Producto: 6, Nombre: Producto E, Fecha de Envasado: 2024-04-26, Unidades: 120, Precio: 18.49, Disponible: true
ID Producto: 7, Nombre: Producto F, Fecha de Envasado: 2024-04-25, Unidades: 80, Precio: 12.75, Disponible: false
ID Producto: 8, Nombre: Producto G, Fecha de Envasado: 2024-04-24, Unidades: 150, Precio: 20.0, Disponible: true
ID Producto: 9, Nombre: Producto H, Fecha de Envasado: 2024-04-23, Unidades: 90, Precio: 14.99, Disponible: true
ID Producto: 10, Nombre: Producto I, Fecha de Envasado: 2024-04-22, Unidades: 110, Precio: 30.25, Disponible: false
ID Producto: 11, Nombre: Producto J, Fecha de Envasado: 2024-04-21, Unidades: 70, Precio: 9.99, Disponible: true
ID Producto: 12, Nombre: Producto11, Fecha de Envasado: 2024-04-29, Unidades: 100, Precio: 10.5, Disponible: true
ID Producto: 13, Nombre: zapatillas, Fecha de Envasado: 2020-10-12, Unidades: 2, Precio: 20.0, Disponible: true
```

```
Menú:
1. Añadir producto
2. Mostrar productos
3. Actualizar producto
4. Eliminar producto
5. Salir
Ingrese su opción: 3
Ingrese el ID del producto a actualizar:
13
Ingrese los nuevos datos del producto:
Nombre: zapatos
Fecha de envasado (YYYY-MM-DD): 2020-10-12
Unidades: 1
Precio: 20
Disponible (true/false): true
Producto actualizado correctamente.
```

```
Menú:
1. Añadir producto
2. Mostrar productos
3. Actualizar producto
4. Eliminar producto
5. Salir
Ingrese su opción: 4
Ingrese el ID del producto a eliminar:
12
Producto eliminado correctamente.
Menú:
1. Añadir producto
2. Mostrar productos
3. Actualizar producto
4. Eliminar producto
5. Salir
Ingrese su opción: 5
Saliendo del programa...
```