



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION

PROGRAMACION ORIENTADA A OBJETOS

CONTROL DE LECTURA N° 2

TEMA: MODELO VISTA CONTROLADOR (MVC) Y PATRONES DE DISEÑO



REALIZADO POR:

Nayeli Scarleth Loachamin Tipan

DOCENTE:

Luis Enrique Jaramillo Montaña

Sangolquí – Ecuador

Modelo Vista Controlador (MVC) y Patrones de Diseño

Códigos

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS
Debug: AplicacionBiblioteca + - [ ] ... ^ X

PS C:\Users\essta\Documents\NetBeansProjects\Sistemadestiondebiblioteca> & 'D:\JDK\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:52202' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\essta\Documents\NetBeansProjects\Sistemadestiondebiblioteca\target\classes' 'com.mycompany.sistemadestiondebiblioteca.AplicacionBiblioteca'

Catálogo de Libros:
1 - Don Quijote de la Mancha por Miguel de Cervantes (Disponible)
2 - Cien años de soledad por Gabriel García Márquez (Disponible)
3 - El principito por Antoine de Saint-Exupéry (Disponible)

Mensaje: Libro agregado correctamente

Catálogo de Libros:
1 - Don Quijote de la Mancha por Miguel de Cervantes (Disponible)
2 - Cien años de soledad por Gabriel García Márquez (Disponible)
3 - El principito por Antoine de Saint-Exupéry (Disponible)
4 - 1984 por George Orwell (Disponible)

Mensaje: Libro prestado correctamente

Detalles del Libro:
ID: 2
Título: Cien años de soledad
Autor: Gabriel García Márquez
Estado: Prestado

Mensaje: El libro no está disponible para préstamo

Mensaje: Libro devuelto correctamente

Detalles del Libro:
ID: 2
Título: Cien años de soledad
Autor: Gabriel García Márquez
Estado: Disponible
PS C:\Users\essta\Documents\NetBeansProjects\Sistemadestiondebiblioteca>
```

Lín. 21, col. 21 Espacios: 4 UTF-8 CRLF {} Java Quokka Qodo Gen

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS
Debug: AplicacionTienda +v [icon] ... ^ x

Precio: $499.99
Stock: 20 unidades

Mensaje: Producto agregado al carrito

Mensaje: Producto agregado al carrito

Contenido del Carrito:
-----
Producto      Precio  Cantidad  Subtotal
-----
Laptop        $999.99   1        $999.99
Auriculares   $79.99    2        $159.98
-----
TOTAL:                $1159.97

Mensaje: Producto agregado al carrito

Contenido del Carrito:
-----
Producto      Precio  Cantidad  Subtotal
-----
Laptop        $999.99   1        $999.99
Auriculares   $79.99    2        $159.98
Mouse         $29.99    1        $29.99
-----
TOTAL:                $1189.96

Mensaje: Compra realizada con éxito

El carrito está vacío

Catálogo de Productos:
-----
ID  Nombre      Precio  Stock
-----
1   Laptop      $999.99   9
2   Smartphone $499.99  20
3   Auriculares $79.99   48
4   Teclado     $59.99   15
5   Mouse       $29.99   24
-----

PS C:\Users\essta\Documents\NetBeansProjects\TiendaOnline\productos>
```

Sistema Gestión de Productos

```
1 package com.mycompany.sisgestionproductos;
2
3
4
5
6
7 import java.util.ArrayList;
8 import java.util.List;
9
10 //Modelo
11 class Producto {
12     public int id;
13     public String nombre;
14     public double precio;
15     public int stock;
16
17     public Producto(int id, String nombre, double precio, int stock) {
18         this.id = id;
19         this.nombre = nombre;
20         this.precio = precio;
21         this.stock = stock;
22     }
23
24     // Getters y setters
25     public int getId() { return id; }
26     public void setId(int id) { this.id = id; }
27     public String getNombre() { return nombre; }
28     public void setNombre(String nombre) { this.nombre = nombre; }
29     public double getPrecio() { return precio; }
30     public void setPrecio(double precio) { this.precio = precio; }
31     public int getStock() { return stock; }
32     public void setStock(int stock) { this.stock = stock; }
33 }
34
35 // Vista
36 class ProductoVista {
37     public void mostrarProducto(Producto producto) {
```

```

48     System.out.println("\nDetalles del Producto:");
49     System.out.println("ID: " + producto.getId());
50     System.out.println("Nombre: " + producto.getNombre());
51     System.out.println("Precio: $" + producto.getPrecio());
52     System.out.println("Stock: " + producto.getStock());
53 }
54
55 public void mostrarListaProductos(List<Producto> productos) {
56     System.out.println("\nLista de Productos:");
57     for (Producto producto : productos) {
58         System.out.println("ID: " + producto.getId() +
59             ", Nombre: " + producto.getNombre() +
60             ", Precio: $" + producto.getPrecio() +
61             ", Stock: " + producto.getStock());
62     }
63 }
64
65 public void mostrarMensaje(String mensaje) {
66     System.out.println("\nMensaje: " + mensaje);
67 }
68
69 // Controlador
70 class ProductoControlador {
71     public List<Producto> modeloLista;
72     public ProductoVista vista;
73
74     public ProductoControlador(List<Producto> modelo, ProductoVista vista) {
75         this.modeloLista = modelo;
76         this.vista = vista;
77     }
78
79     public void agregarProducto(int id, String nombre, double precio, int stock) {
80         Producto producto = new Producto(id, nombre, precio, stock);
81         modeloLista.add(producto);
82     }

```

```

83     vista.mostrarMensaje("Producto agregado exitosamente");
84 }
85
86 public void actualizarProducto(int id, String nombre, double precio, int stock) {
87     for (Producto producto : modeloLista) {
88         if (producto.getId() == id) {
89             producto.setNombre(nombre);
90             producto.setPrecio(precio);
91             producto.setStock(stock);
92             vista.mostrarMensaje("Producto actualizado exitosamente");
93             return;
94         }
95     }
96     vista.mostrarMensaje("Producto no encontrado");
97 }
98
99 public void eliminarProducto(int id) {
100     modeloLista.removeIf(producto -> producto.getId() == id);
101     vista.mostrarMensaje("Producto eliminado exitosamente");
102 }
103
104 public void mostrarProducto(int id) {
105     for (Producto producto : modeloLista) {
106         if (producto.getId() == id) {
107             vista.mostrarProducto(producto);
108             return;
109         }
110     }
111     vista.mostrarMensaje("Producto no encontrado");
112 }
113
114 public void mostrarTodosLosProductos() {
115     vista.mostrarListaProductos(modeloLista);
116 }
117 }

```



```

27 // Controlador
28 class EstudianteControlador {
29     public Estudiante modelo;
30     public EstudianteVista vista;
31
32     public EstudianteControlador(Estudiante modelo, EstudianteVista vista) {
33         this.modelo = modelo;
34         this.vista = vista;
35     }
36
37     public void setEstudianteNombre(String nombre) {
38         modelo.setNombre(nombre);
39     }
40
41     public String getEstudianteNombre() {
42         return modelo.getNombre();
43     }
44
45     public void setEstudianteId(int id) {
46         modelo.setId(id);
47     }
48
49     public int getEstudianteId() {
50         return modelo.getId();
51     }
52
53     public void setEstudiantePromedio(double promedio) {
54         modelo.setPromedio(promedio);
55     }
56
57     public double getEstudiantePromedio() {
58         return modelo.getPromedio();
59     }
60 }

```

```

71 public void actualizarVista() {
72     vista.mostrarDetallesEstudiante(modelo.getNombre(),
73                                     modelo.getId(),
74                                     modelo.getPromedio());
75 }
76
77 public class DisGestionEstudiantes {
78
79     public static void main(String[] args) {
80         // Crear datos del estudiante de la base de datos
81         Estudiante modelo = new Estudiante("Juan Sanchez", 1, 9.5);
82
83         // Crear la vista para mostrar los datos del estudiante
84         EstudianteVista vista = new EstudianteVista();
85
86         // Crear el controlador
87         EstudianteControlador controlador = new EstudianteControlador(modelo, vista);
88
89         // Actualizar vista
90         controlador.actualizarVista();
91
92         // Actualizar datos del modelo a través del controlador
93         controlador.setEstudianteNombre("Juan Carlos Sanchez");
94         controlador.setEstudiantePromedio(9.8);
95
96         // Mostrar datos actualizados
97         controlador.actualizarVista();
98     }
99 }

```

```

Output - Run (Selected Project)
C:\Users\user>cd C:\Users\user\Documents\Proyectos\DisGestionEstudiantes\src\main\resources
Running the project...

----- Run (Selected Project) -----
Building DisGestionEstudiantes 1.0-SNAPSHOT
from pom.xml
[ 3m ]
----- Run (Selected Project) -----
C:\Users\user>cd C:\Users\user\Documents\Proyectos\DisGestionEstudiantes\src\main\resources
Building the project... all classes are up to date.
----- Run (Selected Project) -----
DisGestionEstudiantes
Nombre: Juan Sanchez
Id: 1
Promedio: 9.8
Estudiante:
Nombre: Juan Carlos Sanchez
Id: 1
Promedio: 9.8
BUILD SUCCESS
Total time: 0.004 s
Finished at: 2025-02-20T20:59:11-05:00

```

