



PRÁCTICA #2 - SPRING , USO DE XML

CARRERA: TECNOLOGÍA SUPERIOR EN DESARROLLO DE SOFTWARE

ASIGNATURA: Programación Distribuida

ESTUDIANTE: Dario Cardozo

DOCENTE: Ing. Diego Maiquez

PERÍODO ACADÉMICO: SEMESTRE ABRIL – SEPTIEMBRE 2023

PRACTICA 2



SPRING – USO DE XML

1. Usando el Framework Spring, desarrollar un programa “Factura” que contenga las clases entidades: “Cliente”, “Factura”, “FacturaDetalle” y “Producto” con sus correspondientes atributos, constructores, metodos getters and setters, etc.

```
eclipse-workspace - ismac-spring-factura-xml/src/com/distribuida/entidades/Clientes.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X
> ismac-factura [ismac-factura master]
> ismac-spring-factura-xml
> ismac-spring-factura-xml
> JRE System Library [JavaSE-17]
src
com.distribuida.entidades
  Clientes.java
  detalleFactura.java
  Factura.java
  Producto.java
com.distribuidapincipal
  Principal.java
  ApplicationContext.xml
Referenced Libraries
lib

1 package com.distribuida.entidades;
2
3 import java.util.Date;
4
5 public class Clientes {
6
7     private int idCliente;
8     private String nombre;
9     private String apellido;
10    private String cedula;
11    private int edad;
12    private Date fechaNacimiento;
13    private String direccion;
14    private String correo;
15    private String telefono;
16
17    public Clientes() {}
18
19    public int getIdCliente() {
20        return idCliente;
21    }
22
23    public void setIdCliente(int idCliente) {
24        this.idCliente = idCliente;
25    }
26
27    public String getNombre() {
28        return nombre;
29    }
30
31    public void setNombre(String nombre) {
32        this.nombre = nombre;
33    }
34
35    public String getApellido() {
36        return apellido;
37    }
38
39    public void setApellido(String apellido) {
```

```
eclipse-workspace - ismac-spring-factura-xml/src/com/distribuida/entidades/detalleFactura.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X
> ismac-factura [ismac-factura master]
> ismac-spring-factura-xml
> ismac-spring-factura-xml
> JRE System Library [JavaSE-17]
src
com.distribuida.entidades
  Clientes.java
  detalleFactura.java
  Factura.java
  Producto.java
com.distribuidapincipal
  Principal.java
  ApplicationContext.xml
Referenced Libraries
lib

1 package com.distribuida.entidades;
2
3 public class detalleFactura {
4
5     private int idDetalleFactura;
6     private int cantidad;
7     private Double subtotal;
8     private Factura factura; // private Factura fact
9     private Producto producto;
10
11    public detalleFactura() {}
12
13
14
15
16    public int getIdDetalleFactura() {
17        return idDetalleFactura;
18    }
19
20
21
22
23    public void setIdDetalleFactura(int idDetalleFactura) {
24        this.idDetalleFactura = idDetalleFactura;
25    }
26
27
28
29
30    public int getCantidad() {
31        return cantidad;
32    }
33
34    public void setCantidad(int cantidad) {
35        this.cantidad = cantidad;
36    }
37
38    public Double getSubtotal() {
39        return subtotal;
40    }
41
42    public void setSubtotal(Double subtotal) {
43        this.subtotal = subtotal;
44    }
45
46    public Factura getFactura() {
```

```
eclipse-workspace - ismac-spring-factura-xml/src/com/distribuida/entidades/Factura.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X
> ismac-factura [ismac-factura master]
> ismac-spring-factura-xml
> ismac-spring-factura-xml
> JRE System Library [JavaSE-17]
src
com.distribuida.entidades
  Clientes.java
  detalleFactura.java
  Factura.java
  Producto.java
com.distribuidapincipal
  Principal.java
  ApplicationContext.xml
Referenced Libraries
lib

1 package com.distribuida.entidades;
2
3 import java.util.Date;
4
5 public class Factura {
6
7     private int idFactura;
8     private Date fecha; //Date
9     private String numFactura;
10    private Double totalNeto;
11    private Double iva;
12    private Double total;
13    private Clientes cliente; //private int id_cliente;
14
15    public Factura () {}
16
17    public Factura (Clientes cliente) {
18        this.cliente=cliente;
19    }
20
21
22
23
24    public int getIdFactura() {
25        return idFactura;
26    }
27
28
29    public void setIdFactura(int idFactura) {
30        this.idFactura = idFactura;
31    }
32
33
34    public Date getFecha() {
35        return fecha;
36    }
37
38
39    public void setFecha(Date fecha) {
```

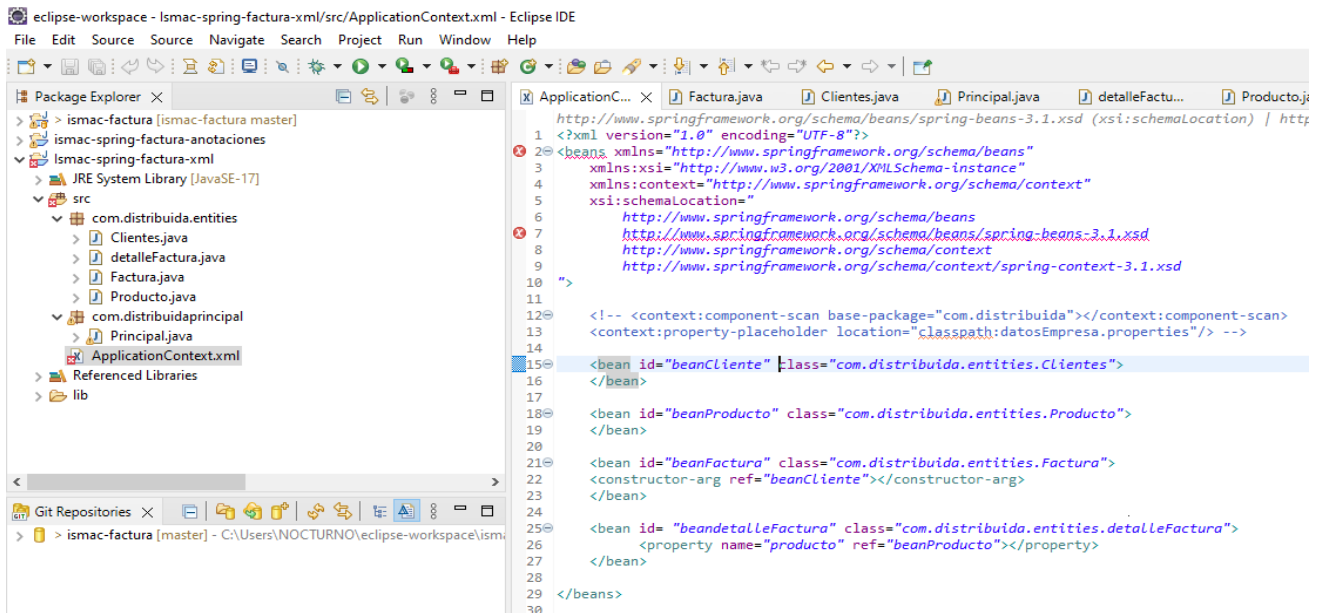
```
eclipse-workspace - ismac-spring-factura-xml/src/com/distribuida/entidades/Producto.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X
> ismac-factura [ismac-factura master]
> ismac-spring-factura-xml
> ismac-spring-factura-xml
> JRE System Library [JavaSE-17]
src
com.distribuida.entidades
  Clientes.java
  detalleFactura.java
  Factura.java
  Producto.java
com.distribuidapincipal
  Principal.java
  ApplicationContext.xml
Referenced Libraries
lib

1 package com.distribuida.entidades;
2
3 public class Producto {
4
5     private int idProducto;
6     private String nombre;
7     private String descripcion;
8     private Double precio;
9     private int stock;
10    public int getIdProducto() {
11        return idProducto;
12    }
13
14    public void setIdProducto(int idProducto) {
15        this.idProducto = idProducto;
16    }
17
18    public String getNombre() {
19        return nombre;
20    }
21
22    public void setNombre(String nombre) {
23        this.nombre = nombre;
24    }
25
26    public String getDescripcion() {
27        return descripcion;
28    }
29
30    public void setDescripcion(String descripcion) {
31        this.descripcion = descripcion;
32    }
33
34    public Double getPrecio() {
35        return precio;
36    }
37
38    public void setPrecio(Double precio) {
39        this.precio = precio;
40    }
41
42    public int getStock() {
43        return stock;
44    }
45
46    public void setStock(int stock) {
47        this.stock = stock;
48    }
49
50 }
```

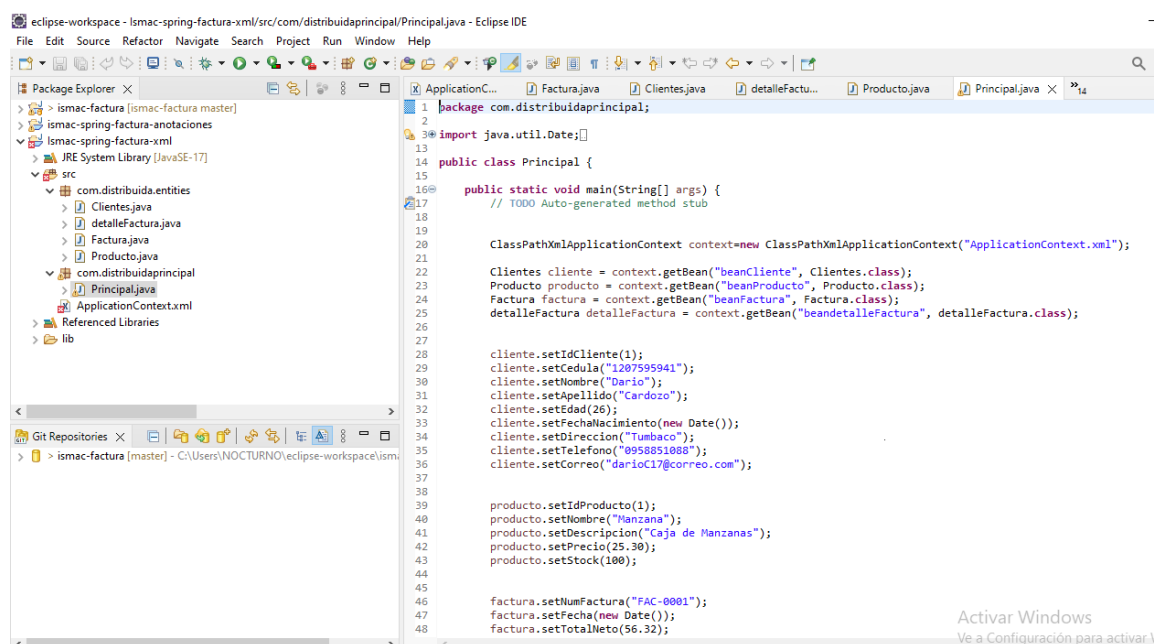
Deberá existir al menos un archivo de configuraciones [ApplicationContext.xml](#) con **REG-SA-0**

Página 3

2. información de: metadatos (llamadas a librerías de spring), creacion de beans e inyección de dependencias con las etiquetas *constructor-arg* o *property*



3. Realizar todas las inyecciones necesarias para gestionar la Factura usando los patrones IoC y DI, de tal forma que el código sea dinámico y desacoplado, estas instancias deberán estar centralizadas en otra clase principal de control (IoC) gestionada por la interfaz [ApplicationContext](#) y su implementación [ClassPathXmlApplicationContext](#).



```
27
28
29     cliente.setIdCliente(1);
30     cliente.setCedula("1207595941");
31     cliente.setNombre("Dario");
32     cliente.setApellido("Cardozo");
33     cliente.setEdad(26);
34     cliente.setFechaNacimiento(new Date());
35     cliente.setDireccion("Tumbaco");
36     cliente.setTelefono("0958851088");
37     cliente.setCorreo("darioC17@correo.com");
38
39
40     producto.setIdProducto(1);
41     producto.setNombre("Manzana");
42     producto.setDescripcion("Caja de Manzanas");
43     producto.setPrecio(25.30);
44     producto.setStock(100);
45
46
47     factura.setNumFactura("FAC-0001");
48     factura.setFecha(new Date());
49     factura.setTotalNeto(56.32);
50     factura.setIva(0.12);
51     factura.setTotal(60.15);
52
53     detalleFactura.setIdDetalleFactura(2);
54     detalleFactura.setCantidad(8);
55     detalleFactura.setSubtotal(84.60);
56
57     System.out.println(factura.toString());
58     System.out.println(detalleFactura.toString());
59
60     context.close();
61
62 }
63
64 }
65
```

4. Imprimir resultados a través del método de sobreescritura toString();

```
<terminated> Principal (1) [Java Application] C:\Users\NOCTURNO\ eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (6 dic. 2023 19:13:57 - 19:13:57) [pid: 2740]
.... Factura [idFactura=0, fecha=Wed Dec 06 19:13:57 ECT 2023, numFactura=FAC-0001, totalNeto=56.32, iva=0.12, total=60.15, cliente=Clientes [idCliente=1, nombre=Dario, apellido=Cardozo, ced
detalleFactura [idDetalleFactura=2, cantidad=8, subtotal=84.6, factura=null, producto=Producto [idProducto=1, nombre=Manzana, descripcion=Caja de Manzanas, precio=25.3, stock=100]]

/ src/com/distribuidaprincipal/Principal.java - Eclipse IDE
:h Project Run Window Help

C:\Users\NOCTURNO\ eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (6 dic. 2023 19:13:57 - 19:13:57) [pid: 2740]
Clientes [idCliente=1, nombre=Dario, apellido=Cardozo, cedula=1207595941, edad=26, direccion=Tumbaco, correo=darioC17@correo.com, telefono=0958851088]] ^
descripcion=Caja de Manzanas, precio=25.3, stock=100]]
```

CONCLUSIONES:

1. Utilizamos los métodos getter and setters para la creación de las entidades cliente, factura, detalle factura y producto con sus respectivos atributos y constructores lo que nos permite realizar el programa que nos ayuda arrojar datos a través de una clase principal donde colocamos los datos que se nos pide y poder ejecutar, imprimir el programa con un resultado exitoso.