



# Spring, Uso de XML

**CARRERA:** TECNOLOGÍA SUPERIOR EN DESARROLLO DE SOFTWARE

**ASIGNATURA:** Programación Distribuida

**ESTUDIANTE:** Victor Peralta

**DOCENTE:** Ing. Diego Maiquez

**PERÍODO ACADÉMICO:** SEMESTRE ABRIL – SEPTIEMBRE 2023

Portada

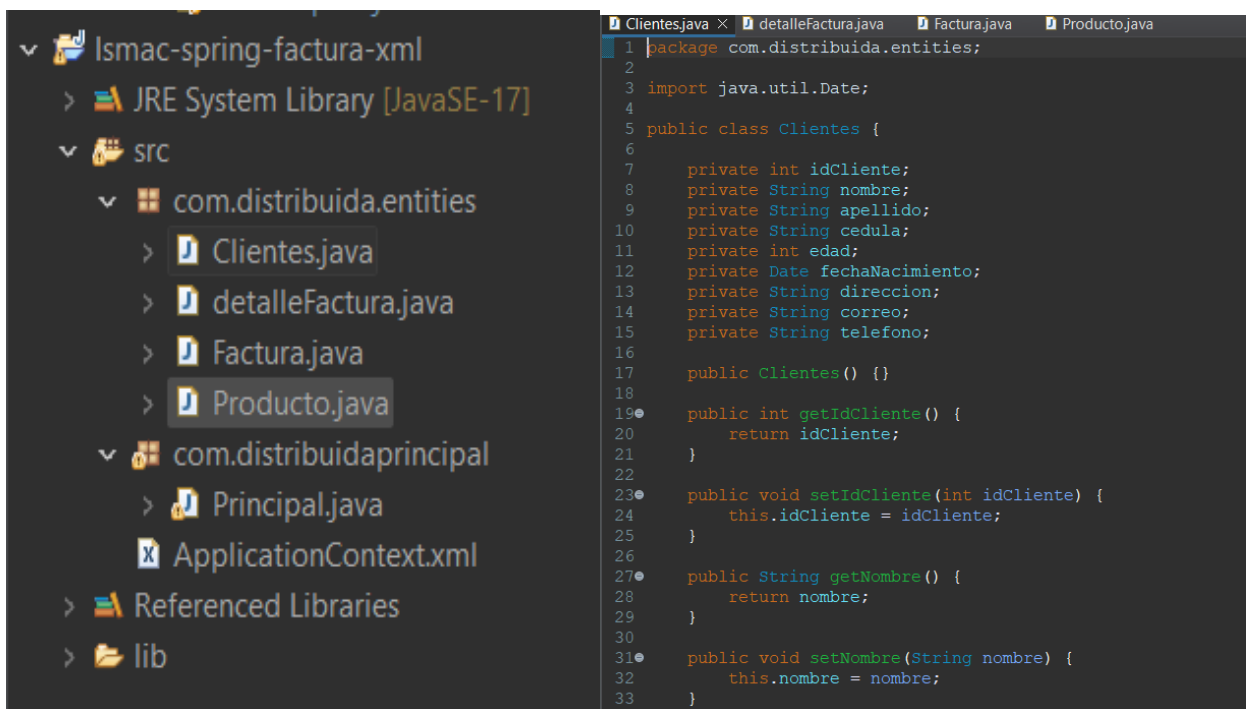
INFORME 3



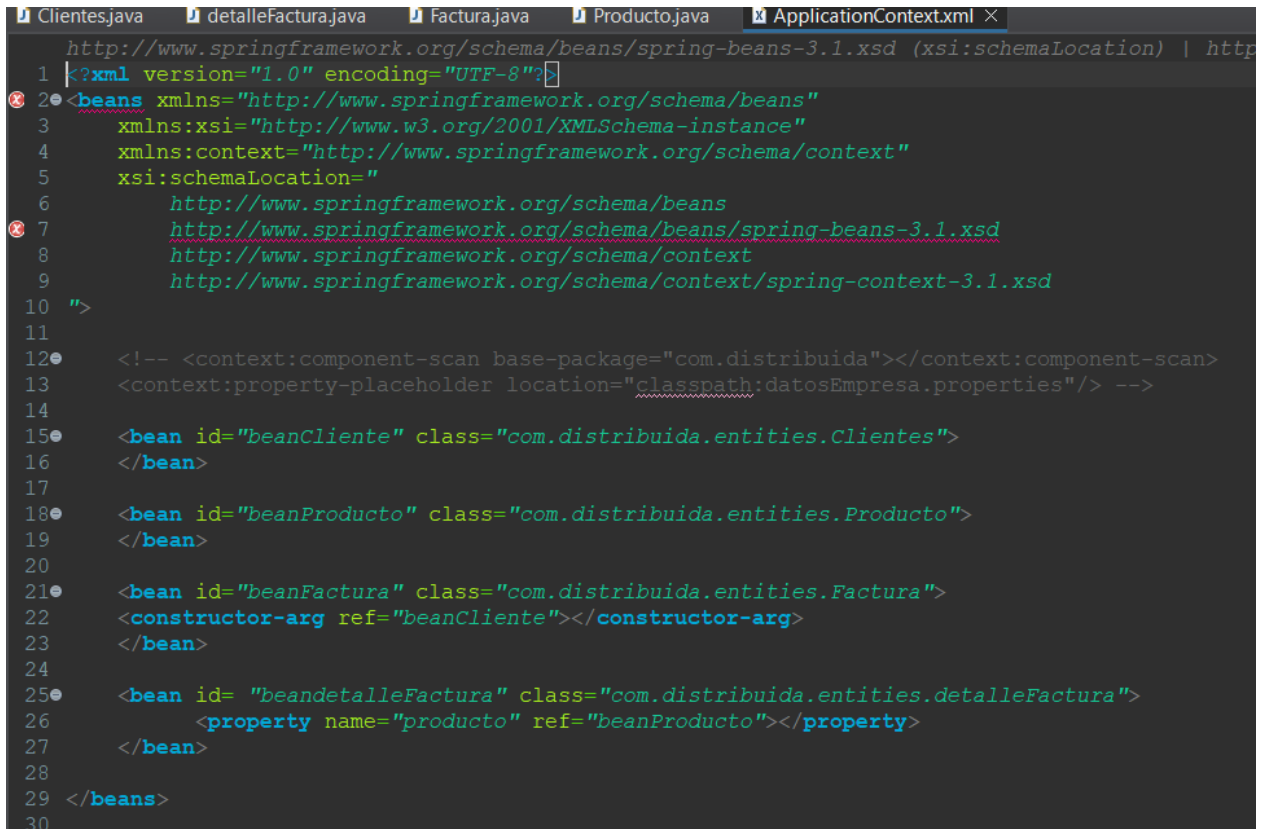
## Spring, Uso de XML

### Informe de la Practica

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.



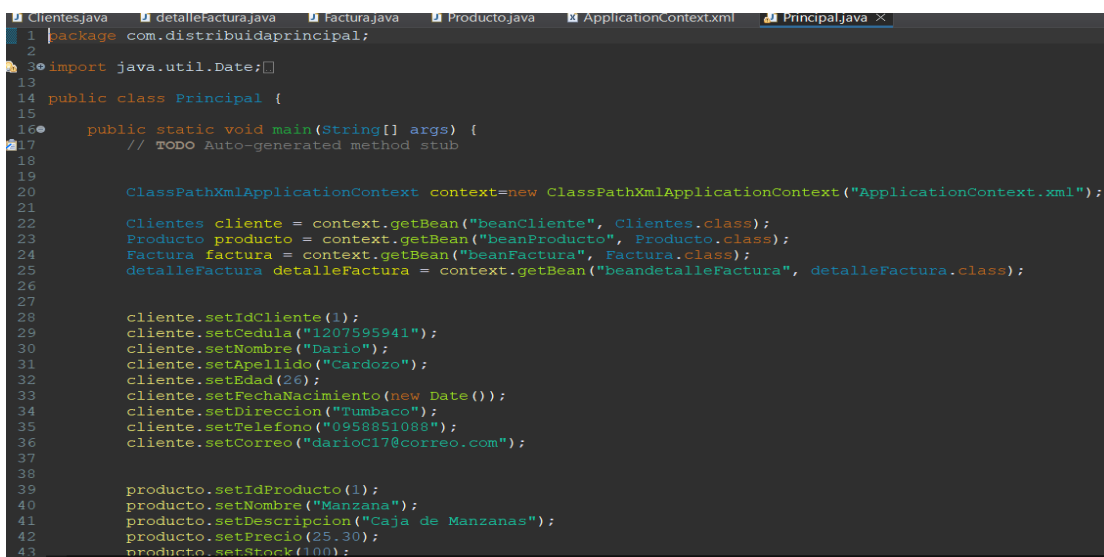
2. Deberá existir al menos un archivo de configuraciones [ApplicationContext.xml](#) con información de: metadatos (llamadas a librerías de spring), creación de beans e inyección de dependencias con las etiquetas *constructor-arg* o *property*



```

1  http://www.springframework.org/schema/beans/spring-beans-3.1.xsd (xsi:schemaLocation) | http
2  <?xml version="1.0" encoding="UTF-8"?>
3  <beans xmlns="http://www.springframework.org/schema/beans"
4      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5      xmlns:context="http://www.springframework.org/schema/context"
6      xsi:schemaLocation="
7          http://www.springframework.org/schema/beans
8          http://www.springframework.org/schema/beans/spring-beans-3.1.xsd
9          http://www.springframework.org/schema/context
10         http://www.springframework.org/schema/context/spring-context-3.1.xsd
11     ">
12     <!-- <context:component-scan base-package="com.distribuida"></context:component-scan>
13     <context:property-placeholder location="classpath:datosEmpresa.properties"/> -->
14
15     <bean id="beanCliente" class="com.distribuida.entities.Clientes">
16     </bean>
17
18     <bean id="beanProducto" class="com.distribuida.entities.Producto">
19     </bean>
20
21     <bean id="beanFactura" class="com.distribuida.entities.Factura">
22     <constructor-arg ref="beanCliente"></constructor-arg>
23     </bean>
24
25     <bean id="beandetalleFactura" class="com.distribuida.entities.detalleFactura">
26     <property name="producto" ref="beanProducto"></property>
27     </bean>
28
29 </beans>
30
  
```

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](#).



```

1  package com.distribuidaprincipal;
2
3  import java.util.Date;
4
5  public class Principal {
6
7      public static void main(String[] args) {
8          // TODO Auto-generated method stub
9
10         ClassPathXmlApplicationContext context=new ClassPathXmlApplicationContext("ApplicationContext.xml");
11
12         Clientes cliente = context.getBean("beanCliente", Clientes.class);
13         Producto producto = context.getBean("beanProducto", Producto.class);
14         Factura factura = context.getBean("beanFactura", Factura.class);
15         detalleFactura detalleFactura = context.getBean("beandetalleFactura", detalleFactura.class);
16
17         cliente.setIdCliente(1);
18         cliente.setCedula("1207595941");
19         cliente.setNombre("Dario");
20         cliente.setApellido("Cardozo");
21         cliente.setEdad(26);
22         cliente.setFechaNacimiento(new Date());
23         cliente.setDireccion("Tumbaco");
24         cliente.setTelefono("0958851088");
25         cliente.setCorreo("darioC17@correo.com");
26
27         producto.setIdProducto(1);
28         producto.setNombre("Manzana");
29         producto.setDescripcion("Caja de Manzanas");
30         producto.setPrecio(25.30);
31         producto.setStock(100);
32
33     }
34 }
  
```

```

1 Clientes.java 2 detalleFactura.java 3 Factura.java 4 Producto.java 5 Ap
32 cliente.setEdad(26);
33 cliente.setFechaNacimiento(new Date());
34 cliente.setDireccion("Tumbaco");
35 cliente.setTelefono("0958851088");
36 cliente.setCorreo("darioC17@correo.com");
37
38
39 producto.setIdProducto(1);
40 producto.setNombre("Manzana");
41 producto.setDescripcion("Caja de Manzanas");
42 producto.setPrecio(25.30);
43 producto.setStock(100);
44
45
46 factura.setNumFactura("FAC-0001");
47 factura.setFecha(new Date());
48 factura.setTotalNeto(56.32);
49 factura.setIva(0.12);
50 factura.setTotal(60.15);
51
52 detalleFactura.setIdDetalleFactura(2);
53 detalleFactura.setCantidad(8);
54 detalleFactura.setSubtotal(84.60);
55
56
57 System.out.println(factura.toString());
58 System.out.println(detalleFactura.toString());
59
60 context.close();
61
62 }
63
64 }

```

4. Instanciar al menos 1 objeto de cada clase asociado a las inyecciones necesarias.

```

cliente.setIdCliente(1);
cliente.setCedula("1207595941");
cliente.setNombre("Dario");
cliente.setApellido("Cardozo");
cliente.setEdad(26);
cliente.setFechaNacimiento(new Date());
cliente.setDireccion("Tumbaco");
cliente.setTelefono("0958851088");
cliente.setCorreo("darioC17@correo.com");

producto.setIdProducto(1);
producto.setNombre("Manzana");
producto.setDescripcion("Caja de Manzanas");
producto.setPrecio(25.30);
producto.setStock(100);

factura.setNumFactura("FAC-0001");
factura.setFecha(new Date());
factura.setTotalNeto(56.32);
factura.setIva(0.12);
factura.setTotal(60.15);

detalleFactura.setIdDetalleFactura(2);
detalleFactura.setCantidad(8);
detalleFactura.setSubtotal(84.60);

```

5. Imprimir resultados a través del método de sobrescritura toString();

```
Problems Servers Terminal Data Source Explorer Properties Console X
<terminated> Principal [Java Application] C:\Users\Lenovo.User\workspace\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.9.v20231028-0858\jre\bin\javaw.exe (10 dic. 2023 20:22:58 - 20:22:59) [pid: 14204]
Factura [idFactura=0, fecha=Sun Dec 10 20:22:59 ECT 2023, numFactura=FAC-0001, totalNeto=56.32, iva=0.12, total=60.15, cliente=Clientes [idCliente=1, n
detalleFactura [iddetalleFactura=2, cantidad=8, subtotal=84.6, factura=null, producto=Producto [idProducto=1, nombre=Manzana, descripcion=Caja de Manza
```