

Instituto Superior Tecnológico ISMAC

PRÁCTICA 2 – Spring, Uso de XML

Asignatura: Programación Distribuida

Nombre: Darwin Loor

Fecha: 25/11/2023

1. Clases Entidades

Las clases entidad representan los elementos clave del sistema: Cliente, Producto, FacturaDetalle y Factura. Estas clases contienen atributos y métodos para modelar la información y funcionalidades necesarias para el sistema.

Cliente.java: Representa la información del cliente, con atributos como id y nombre.

Producto.java: Modela la información del producto, con atributos como id, nombre y precio.

FacturaDetalle.java: Contiene los detalles de una factura, incluyendo el producto, la cantidad y el subtotal.

Factura.java: Representa una factura que tiene un cliente asociado y una lista de detalles, calculando el total.

2. Archivo de Configuraciones ApplicationContext.xml

Este archivo XML configura el contexto de Spring, definiendo los beans que representan las instancias de las clases anteriores y sus relaciones.

<bean>: Cada elemento bean define la configuración de un objeto en el contexto de Spring. Se definen beans para Cliente, Producto, FacturaDetalle y Factura.

<property>: Se utiliza para inyectar valores a través de los métodos set de las clases.

<list>: Permite definir listas, en este caso, para los detalles de la factura.

3. Clase Principal de Control (IoC)

La clase MainApp.java actúa como la clase principal que controla la ejecución del programa.

ApplicationContext: Representa el contexto de Spring, que gestiona los beans.

ClassPathXmlApplicationContext: Implementación de ApplicationContext que carga la configuración desde un archivo XML en el classpath.

context.getBean("nombreDelBean"): Recupera una instancia de un bean del contexto de Spring.

4. Instanciar Objetos y 5. Imprimir Resultados

Estas secciones explican cómo se pueden instanciar objetos utilizando Spring y cómo imprimir los resultados.

Se utilizan las etiquetas de configuración definidas en el archivo ApplicationContext.xml para crear instancias de Cliente, Producto y Factura.

Luego, se imprime la representación de cada objeto utilizando el método toString().

