Spring Highlights

EJBs are heavyweight and tightly coupled and Springs are lightweight and loosely coupled.

Heavyweight uses OS libraries and lightweight doesn’t use.

Tightly coupled dependent on other layers.

Loosely coupled doesn’t dependent on other layers.

ORM tools: - JDBC, Hibernate, JPA, ibatis

Spring developed using Runtime Polymorphism and association (has-A relation).

IOC container reads XML documents and pass the reference to pojo.

**IOC containers: Core and J2EE Container.**

**Spring MVC Container: Web Container.**

Core Container = Bean Factory(Interface) 🡪 XMLBeanFactory

J2EE Container = ApplicationContext(Interface) 🡪 ConfigurableApplicationContext(Interface) 🡪 ClassPathXmlApplicationContext(Class)

MVC Conatiner = WebApplicationContext(Interface)🡪 WebApplicationContextUtil (Factory Class)

XMLBeanFactory Container creates instance of pojo in user request (Lazy container).

ApplicationContext Container creates instance of pojo on loading xml file in case of singleton scope (Eager container).

What containers do:-

1. Create instance of pojo classes.
2. Mange life cycle of pojo.
3. Do dependency injection in pojo.

In xml file configure dtd or xsd in main beans tag.

Scope of beans: - Singleton, Prototype

MVC Scope of beans: - Request, Session, Global Session (Context)

Spring can access private constructors.

Two types of Dependency Injection:- Constructor Parameters and Setter Method

Bean references.

Required annotation. To activate annotation use “RequiredAnnotationBeanPostProcessor” class as bean.

P- Namespace and C- Namespaces.

Auto wiring: Auto dependency injection. Can inject secondary types only.

Auto wiring using Annotations. Activate it using bean class.

Stereo Type

To create object by using auto scanning.

Use context-annotation to activate all annotations.

Inject static variables. “MethodInvokingFactoryBean”

How to create/get instance of java Singleton class, use factory-method tag in bean if static method.

If instance method use factory-bean tag also.

Create factory class by implementing factorybean interface.

Maintain pojo lifecycle 🡪 pojo implementing interfaces, configure in beans.xml, use annotations.

Classes to activate annotations.

DI by lookup method.3rd type.

What are lookup method? Interface methods, abstract method, and concreate method which we are overriding.

Method replacer.