Spring_fw_syllabus

Modules Of Spring Framework

Core Container

This container has the following four modules :

Spring Core: This module is the core of the Spring Framework. It provides an implementation for features like IoC (Inversion of Control) and Dependency Injection with a singleton design pattern.

Spring Bean: This module provides an implementation for the factory design pattern through BeanFactory.

Spring Context: This module is built on the solid base provided by the Core and the Beans modules and is a medium to access any object defined and configured.

Spring Expression Languages (SpEL): This module is an extension to expression language supported by Java server pages. It provides a powerful expression language for querying and manipulating an object graph, at runtime.

Spring Data Access/ Integration

It consists of the following five modules:

JDBC: This module provides JDBC abstraction layer which eliminates the need of repetitive and unnecessary exception handling overhead.

ORM: ORM stands for Object Relational Mapping. This module provides consistency/ portability to our code regardless of data access technologies based on object oriented mapping concept.

OXM: OXM stands for Object XML Mappers. It is used to convert the objects into XML format and vice versa. The Spring OXM provides an uniform API to access any of these OXM frameworks.

JMS: JMS stands for Java Messaging Service. This module contains features for producing and consuming messages among various clients.

Transaction: This module supports programmatic and declarative transaction management for classes that implement special interfaces and for all your POJOs. All the enterprise level transaction implementation concepts can be implemented in Spring by using this module.

Spring Web

Web layer includes the following modules:

Web: This module using servlet listeners and a web-oriented application context, provides basic web-oriented integration features like multi-part file upload functionality and the initialization of the IoC container.

Web-Servlet: This module contains Model-View-Controller (MVC) based implementation for web applications. It provides all other features of MVC, including UI tags and data validations.

Web-Socket: This module provides support for WebSocket based and two-way communication between the client and the server in web applications.

Web-Portlet: This module is also known as the Spring-MVC-Portlet module. It provides the support for Spring-based Portlets and mirrors the functionality of a Web-Servlet module.

Aspect-Oriented Programming (AOP)

AOP language is a powerful tool that allows developers to add enterprise functionality to the application such as transaction, security etc. It allows us to write less code and separate the code logic. AOP uses cross-cutting concerns.

Test

This module supports the testing of Spring components with JUnit or TestNG. It provides consistent loading of Spring ApplicationContexts and caching of those contexts. It also provides mock objects that we can use to test our code in isolation.