Inroduction To Spring

# Why Spring?

**Spring Introduction**

Spring framework is developed by Rod Johnson. Spring Framework is a java platform which provides a very rich architecture by which developer can build complex Java application with ease.

The reason behind the success of Spring framework is

1. It provides such infrastructure by which developer has to write less code, only can focus on business logic no other tasks like Logging, Transaction, Security etc.,

2. Spring does not reinvent the Wheel but it has provision to hook with any layer UI, business, DAO by use of popular technologies. So Spring is such a store where anything is available. It works on Spoke and Hub technology where Spring act as Hub and every popular technology like Hibernate, Struts, JSF, JMS, etc. act as Spokes.

Diagram: **Spoke and Hub**

**Why Spring?**

Spring has been developed to ease the Enterprise application development. To understand why we choose spring over normal java EE Stack let discuss about

Spring benefits and java EE pain points.

**Spring advantages**

1. Spring is very light weight so it can run on any server and size of spring also very small.

2. One of the pain points of java EE stack is when we use EJB it needs an Application server (Jboss,Weblogic etc.) which has EJB container to deploy artifacts ,but it is not run on Web server(like Tomcat) as lack of EJB container , But spring does not need any application server it can run on a webserver moreover we can run it standalone mode also.

3. Another pain point of java EE stack is developer has to implement many java stack specific interfaces or abstract class (like to create a servlet one has to extends HTTPServlet class), so application has strong coupling with Java EE so to remove Java EE from a project or introduce Java EE in a new Project is not an easy task to do lot of refactoring requires. Unlike Spring, to introduce spring no Spring-specific interface or abstract class requires may some annotations requires but believe me those are very few so we just work with POJOs (Plain Old Java Object) so remove or introduce spring in a project is very easy.

4. Spring container uses Dependency Injection (DI) so spring takes care of HAS-A or dependencies part. So Developer only bother about their business logic not the collaboration between Objects but in Java EE developer has to take care of collaboration.

5. Spring incorporate Separation of Concerns principle very well so the cross-cutting concerns like Logging, transaction, security which are not avoidable while coding but they are not core business logic, In Spring those are taken care of by AOP (Aspect Oriented Programming) and developer only implement business logic. In java EE we do not have such provision.

6. Spring uses Template pattern to suppress the boilerplate code and taking responsibility to call resource reclamation task (close costly resources like connection). In java EE developer has to write those code like loading database driver, getting a connection, closing connection etc.

7. As Spring provide templates for popular technologies, it is very easy to incorporate any popular technology in your project.

Due to above benefits, most of the projects default choice is Spring.