Phase 3 : total 8 days

Implements framework the devops way

Day 1 : 25-07-2022

Spring Framework, Spring boot, Web Service and testing (junit)

Spring framework

According to phase 2

View -🡪 JSP or HTML

Controller 🡪 Servlet

Model -🡪 JavaBean / Entity, Service class, Dao class and resource class. Or EJB.

Container is a part of server responsible to create and maintain the life of object of servlet and jsp but not Model layer.

To improve the model layer we use EJB. Enterprise Java Bean.

To run the ejb program we require EJB container. EJB container is a part of application server.

EJB is very complex. Developing application using EJB is / was very complex.

Spring Framework came in picture.

Using spring framework we can improve model layer.

Difference Spring framework and EJB.

Framework : framework provide set of API which help to develop application very easily. Framework follow standard. The implementation of all design pattern is taken care by framework. If we develop any application using framework 70 to 80 percentage task is taken care by framework. Framework is just like a template or protocol but not a final product.

Struts : Struts is a open source web framework provided by apache company. It internally follow MVC architecture. It provided of classes to improve controller and model layer. It provided lot of tags to improve view layer. It internally follow front controller design pattern.

Struts is known as controller centric framework.

JSF : Java Server Faces : It is open source web framework provided by oracle company. It internally follow MVC architecture. It provided of classes to improve controller and model layer. It provided lot of tags to improve view layer. It internally follow front controller design pattern.

JSF is known as view centric framework.

Hibernate : Hibernate is ORM framework. It is replacement of JDBC. JDBC is use to improve view layer.

Spring framework : spring open source light weighted layer or onion architecture framework. Spring framework is replacement of EJB.

Spring framework provided lot of pre-defined modules

Like

1. Core module
2. Context module
3. Spring web
4. Spring MVC
5. Spring DAO
6. Spring MVC : Spring mvc internally follow MVC design pattern. And it provided lot of pre-defined classes and interfaces to improve all layer. Spring MVC is known as model centric framework.
7. Spring ORM
8. Spring AOP
9. Spring security
10. Spring cloud
11. Spring web service
12. Spring micro service
13. Spring data
14. Spring boot
15. Spring security
16. Spring WebFlux which allow use to use RxJS features

Etc

Spring core and context

Spring IOC : Inversion of control

Spring DI : Dependency Injection

IOC : Inversion of control : IOC is a concept. It is a design pattern or programming pattern. In place of creating or maintaining any resource or object explicitly. Allow to create and maintain by container. You have to pull it from a container whenever it required, use it and leave it. The life of resource maintain by container.

DI : DI is a implementation of IOC. Using DI container is push the dependencies for us base upon our requirement. As a developer we have to pull the any dependencies (resource or object) from a container whenever it required.

We can achieve DI lot of ways

1. Constructor base DI
2. Setter base DI

We can achieve using constructor base or setter base with

1. XML file configuration
2. Annotation base

Spring constructor and Setter base DI with XML configuration

Spring framework help us to create the object of POJO class (Plain Old Java Object). Spring container is a light weighted container part of jar file.

Web container or EJB container are heavy weighted container part of server. Those container will create the object of only those classes if class is type of servlet, or ejb or jsp file. For normal class those container doesn’t create the object.

To run the EJB Program we require application server like WebLogic or Jboss or WAS. Which contains EJB container .

By default spring container will create singleton object for every POJO class.

Singleton means only one memory created.

Day 2 : 26-07-2022

**Setter base DI**

**Constructor base DI must be achieve using fully dependencies. May be using empty constructor or parameterized. When we use the parameterized constructor order matter ie number of parameter as well as type of parameter.**

**Setter base DI we can achieve partially.**

Auto wired : autowiring a features provided by spring framework which enable dependency injection for complex object implicitly rather than explicitly using property or constructor ref.

If class contains primitive property container by default do the DI for primitive property.

But if class contains complex property we have to do di for that property using ref attribute explicitly.

Mainly we use two type of auto wired

1. byType : when we use byType spring container scan or search that type of bean declaration tag to inject the object in xml file. It must be contain only one bean tag declaration of that type.
2. byName : if two bean tag declaration of that type present that time we have to use byName. While using byName id name and reference name must be match.

DI concept using annotation base with partial xml file configuration

@Component annotation we have to write on POJO or Java bean class

@Autowired annotation on complex property.

By default @Component annotation is not enable we have to enable this annotation using xml file or java class with another annotation ie @Configuration annotation.

@Value : This annotation we have to use on primitive property to set the default value.

@Configuration // beans.xml file

@ComponentScan(basePackages = "com")

//<context:component-scan base-package="com"></context:component-scan>

27-07-2022

DataSource : Provide the source of the data in secure manner like database connectivity.

Old time we were configuring data source features in application server like web logic or Jboss and using JNDI look up we were getting the connection.

Using spring framework with help of spring jdbc dependencies we can get the Data source features.

With help of Spring framework we will get the benefits of service layer and dao layer using @Service annotation and @Repository.

If we want to get the Datasource reference we have to pass minimum 4 four information ie DriverName, url, username and password.

@component 🡪 we will write this annotation on POJO or Java Bean class

@Service -🡪 This annotation we will write on service layer class

@Repository 🡪 This annotation we will write on dao layer class.

Spring with JDBC ie DataSource to improve Model layer

Spring framework provided pre-defined API ie JdbcTemplate. JdbcTemplate wrap the core jdbc code

And provided set of method do improve DAO layer using JDBC.

Spring DAO (Data Access Object)

28-07-2022

Spring MVC : Spring Model View Controller. Spring MVC by default follow MVC architecture. It provided of lot of API to improve model layer(@Service, @Repository, @Component) and controller layer (@Controller as well as @RestController) and view can be HTML or JSP.

Spring MVC internally follow another one design pattern ie FrontController design pattern. Front controller is a type of design pattern flow of the application must be pass through that controller. Spring MVC provided pre-defined class ie DispatcherServlet . it is a type of Servlet controller provided spring mvc module.



We will create Simple Spring MVC Project.

Spring MVC means dynamic web project

29-07-2022

Spring MVC with ORM (Hibernate)

Spring MVC provide a features of DI for controller layer, java bean class, service class and dao class.

Hibernate provide features of ORM.

Spring MVC doesn’t provide any ORM features it provide dependencies which help to configure or integrate with any other ORM tool like Hibernate or Jpa or iBaties etc.

*org.springframework.orm.hibernate5.LocalSessionFactoryBean*

*this pre-defined class provided by spring MVC which help to configure spring mvc with orm tool.*

01-08-2022

Spring boot : Spring boot is another module of spring framework which help to bootstrap for the all spring modules.

Spring framework : developing any application using spring framework without boot were more complex in sense. Lot configuration like adding lot of jar files, configure the xml file, adding server etc.

Spring boot mainly bootstrap for spring application.

Spring boot = all spring modules – no xml file + few annotation + embedded web server ie tomcat.

Spring boot itself is a standalone project or core java project which help to develop any type of spring application using RAD (Rapid application development).

Spring boot component s

Spring boot starter : Spring boot provided lot of pre-defined starter which combine more than one jar file or set of jar files in one dependencies depending upon type of application we are developing

Like web starter, jdbc starter, jpa starter (spring boot doesn’t support hibernate it support jpa), testing starter, security starter, aop starter etc.

Spring boo auto configuration or configurator : before spring boot we were configure spring framework project using xml file. But in Spring boot xml file removed and we can use few annotation to do all types of configuration.

@SpringBootApplication = @Configuration + @ComponentScan + @AutoConfigurator

Pom.xml file is a part of maven tool not a part of spring framework

In spring boot if we want view as a JSP then we have to add one dependencies.