Phase 3

9 classes

Day 2 : June 5 2024

IOC and DI using annotation

Create maven project as DIUsingAnnotation

@Component : This annotation we can use on pojo or java bean class.

This annotation class level.

@Autowired : this annotation we need to use on complex or user defined reference level.

By default @Component annotation is not enable

To enable this annotation we can use

1. Xml file

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

Whenever we load xml file using spring classes.

Container check all those classes with @Component annotation and it will do DI for those classes.

1. Configuration class with few more annotation.

@Value annotation : this annotation we can use on primitive property level to set default value.

@Configuration : this annotation is replacement of xml file configuration.

@ComponentScan : this annotation is equal to <context:component-scan base-package=*"com"*></context:component-scan>in xml file.

Data Source : Provide data base connection in proper way with singleton concept. Before spring framework to achieve data source we were depending upon application server and using jndi look we were searching database connection.

But using spring with jdbc concept we can achieve data source features very easily.

We are creating spring with maven project to improve model layer

Ie JavaBean class, service class, dao class and resource class.

Core depending

Context depending to achieve di using annotation

Spring jdbc which provide data source features

Mysql connector

Spring jdbc module

Provide pre defined class ie DriverManagerDataSource which contains lot of property.

Ie driverName,url,password, username min which help to provide database connection in single with high secure manner.

Create table product(pid int primary key,pname varchar(10), price float);

@scope : this annotation we can use on class level to make singleton or prototype

@Repository : this annotation we can use on class level and that class contains database logic.

@Service : this annotation we can use on class level and this class contains business logic.