Day 2 : 8 September 2024

Auto wired

Spring framework do the DI for primitive property implicitly. If any complex property like user defined object present in class. then we need to do explicitly for that property using property ref or constructor ref.

With help of auto wired features we can achieve DI for complex property implicitly rather than explicitly using ref attribute.

byType : container search that bean definition in xml file. If present automatically they do DI for that type.

in byType we need only one bean definition of that type.

byName : if more than one bean definition of that type present then we need to use byName. In byName reference name or object name and id name must be match.

Annotation base DI ie constructor base as well as setter base

@Component : This annotation we need use on POJO class or java bean class.

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

@Value : this annotation we need to use on primitive property to set initial value.

By default these annotation are not enable. To enable these annotation we can use

1. Xml file : xml file
2. Class with few more annotation : pure annotation

BeanFactory : core interface it provide only DI using xml base

ApplicationContext: specific interface which internally extends BeanFactory. It support annotation base di as well as provide some extra features.

AnnotationConfigApplicationContext : it support annotation base DI using configuration class.

If we want use purely annotation base then we need to create user defined class with @Configuration and @ComponentScan annotation