# **spring常用注解的作用**

# ****一．注解:****

常用注解大致可分为两类，目的是修饰bean或注册bean。

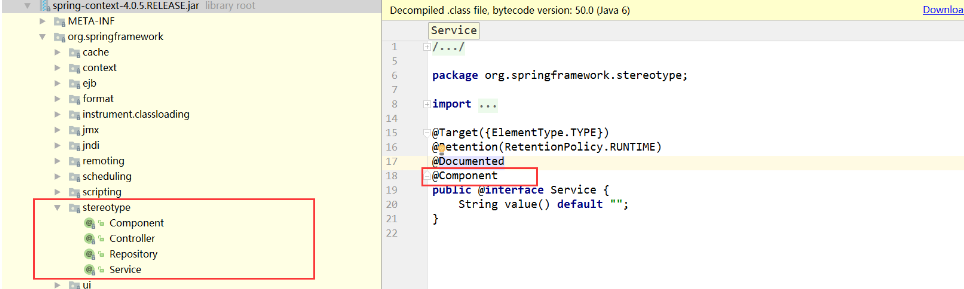
## ****1.@Bean注解****

@Bean 标识一个用于配置和初始化一个由SpringIoc容器管理的新对象的方法，类似于XML配置文件的<bean/>，一般与@Configration注解配合使用注册bean

****2.@Service注解****

如果一个类带了@Service注解，将自动注册到Spring容器，不需要再在applicationContext.xml文件定义bean了，类似作用的还包括@Component、@Repository、@Controller等

@Component ：标注一个普通的spring Bean类。 @Repository：标注一个DAO组件类。 @Service：标注一个业务逻辑组件类。 @Controller：标注一个控制器组件类。 这些都是注解在平时的开发过程中出镜率极高，@Component、@Repository、@Service、@Controller实质上属于同一类注解，用法相同，功能相同，区别在于标识组件的类型。@Component可以代替@Repository、@Service、@Controller，因为这三个注解是被@Component标注的。



## ****3.@Component注解****

## ****把普通pojo实例化到spring容器中，相当于配置文件中的  <bean id="" class=""/>泛指各种组件，就是说当我们的类不属于我们常见的各种归类时（非@Controller、非@Services等），我们就可以使用@Component来标注这个类。****

## ****4.@Repository注解****

用于标注数据访问组件，即DAO组件，再稍微大点的项目里，使用xml的bean定义来配置会大大增加代码体积且不易维护，所以引入了自动扫描的机制，它的作用和在[xml文件](https://www.baidu.com/s?wd=xml%E6%96%87%E4%BB%B6&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao)中使用bean节点配置组件时一样的。

## ****5.@Controller注解****

当组件属于交互用途的控制层时，则使用@Controller注解。被Controller标记的类就是一个控制器，这个类中的方法就是相应的动作。

说明：

1、被注解的java类均当做Bean实例，Bean实例的名称默认是Bean类的首字母小写，其他部分不变。@Service也可以自定义Bean名称，但是必须是唯一的！

2、尽量使用对应组件注解的类替换@Component注解，在spring未来的版本中，@Controller，@Service，@Repository会携带更多语义。并且便于开发和维护！

# **6.**@Configuration注解****

@Configuration用于定义配置类，可替换xml配置文件，被注解的类内部包含有一个或多个被@Bean注解的方法，这些方法将会被AnnotationConfigApplicationContext或AnnotationConfigWebApplicationContext类进行扫描，并用于构建bean定义，初始化Spring容器。

****需要注意的是：****

@Configuration不可以是final类型；

@Configuration不可以是匿名类；

嵌套的configuration必须是静态类。

Bean注解主要用于方法上，有点类似于工厂方法，当使用了@Bean注解，我们可以连续使用多种定义bean时用到的注解，譬如用@Qualifier注解定义工厂方法的名称，用@Scope注解定义该bean的作用域范围，譬如是singleton还是prototype等。

 说明：Spring 中新的 Java 配置支持的核心就是@Configuration 注解的类。这些类主要包括 @Bean 注解的方法来为 Spring 的 IoC 容器管理的对象定义实例，配置和初始化逻辑。使用@Configuration 来注解类表示类可以被 Spring 的 IoC 容器所使用，作为 bean 定义的资源。而使用@Component来注解类表示类，该类并不是归类于spring容器管理，而是类似于new bean()这样的一种方式。

## ****7.@Autowired 注解****

它可以对类成员变量、方法及构造函数进行标注，完成自动装配的工作。 通过 @Autowired的使用来消除 set ，get方法。

@Autowired注解可用于为类的属性、构造器、方法进行注值。默认情况下，其依赖的对象必须存在（bean可用）如果容器中包含多个同一类型的Bean，那么启动容器时会报找不到指定类型bean的异常，解决办法是结合@Qualifier注解进行限定，指定注入的bean名称

## ****8.@Resource注解****

@Resource和@Autowired注解都是用来实现依赖注入的。只是@Autowired按byType自动注入，而@Resource默认按 byName自动注入

****@Resource依赖注入时查找bean的规则****

1. ****既不指定name属性，也不指定type属性，则自动按byName方式进行查找。如果没有找到符合的bean，则回退为一个原始类型进行进行查找，如果找到就注入。****
2. ****只是指定了@Resource注解的name，则按name后的名字去bean元素里查找有与之相等的name属性的bean。****
3. ****只指定@Resource注解的type属性，则从上下文中找到类型匹配的唯一bean进行装配，找不到或者找到多个，都会抛出异常****
4. ****既指定了@Resource的name属性又指定了type，则从Spring上下文中找到唯一匹配的bean进行装配，找不到则抛出异常****

****除此之外还有很多注解****

# ****9.@Override注解****

标示当前的方法定义将覆盖超类中的方法他告诉我们同时也告诉编译器我们的这些方法肯定覆盖了类\*\*里面的方法，如果注释掉类\*\*里面的方法 那么就会报错，它需要全部覆盖某个接口的方法。

## ****10.@ComponentScan注解****

@ComponentScan告诉Spring 哪个packages 的用注解标识的类 会被spring自动扫描并且装入bean容器。

例如，如果你有个类用@Controller注解标识了，那么，如果不加上@ComponentScan，自动扫描该controller，那么该Controller就不会被spring扫描到，更不会装入spring容器中，因此你配置的这个Controller也没有意义。

## ****11.@PropertySouce注解****

通过@PropertySource注解将properties配置文件中的值存储到Spring的 Environment中，Environment接口提供方法去读取配置文件中的值，参数是properties文件中定义的key值。

## ****12.@ImportResource注解和@Import注解****

@Import注解是引入带有@Configuration的java类。

@ImportResource是引入spring配置文件.xml

## ****13.@RequestMapping****

是一个用来处理请求地址映射的注解，可用于类或方法上。用于类上，表示类中的所有响应请求的方法都是以该地址作为父路径。

## ****14.@responseBody注解****

@responseBody注解的作用是将controller的方法返回的对象通过适当的转换器转换为指定的格式之后，写入到response对象的body区，通常用来返回JSON数据或者是XML数据