**Spring原始注解主要替代<bean>的配置**



**！！！！！使用Spring注解开发时**

**要在xml中配置组件扫描——指定哪个包及其子包下的Bean需要进行扫描以便识别注解配置的类，字段和方法**

**组件扫描需要在xml配置文件中添加**



**1，@Component**



**该<bean>可被下图替代**



**如果一个类用@Component修饰，那么这个类默认为到 单例模式**

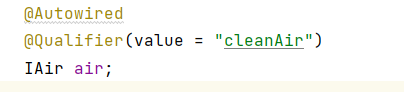
**Spring容器创建时，会扫描<contest:component-scan base-package>设置的包，**

**扫描到一个类含有@Component注解时，会读取@Component的value作为这个类的id，并且通过反射产生一个Bean对象，存储在Spring容器中**

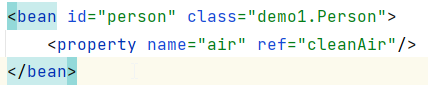
**所以@Component修饰的类，在Spring容器创建时就会创建一个Bean对象，存储在Spring容中，并用@Component的value作用该Bean对象的id**

**2，@Autowired和@Qualifier**

**①@Autowired和@Qualifier同时使用**



**在Spring容器中，查找id为cleanAir的Bean对象，注入给air，可替代下图**



**②@Autowired独自使用**



**在Spring容器中寻找类型匹配的Bean对象，注入给air**

**3，@Resource**

**相当于@Autowired+@Qualifier**

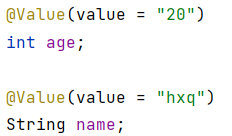


**4，@Controller，@Service，@Repository**

**三个标签跟@Component功能一样，更多的是起到标识作用**

**5，@Value**

**为普通数据类型注入值**



**通常配合配置文件使用**

**6，@PostConstruct**

**该注解修饰的方法，在Bean对象创建后执行用于初始化**

**相当于<bean>标签中的init-method**

**7，@PostDestory**

**该注解修饰的方法，在Bean对象销毁前执行**

**相当于<bean>标签中的destory-method**