**Spring – Stereotype Annotations**

@Component is a class-level annotation. It is used to denote a class as a Component. We can use @Component across the application to mark the beans as Spring’s managed components. A component is responsible for some operations

Spring Framework provides us with some special annotations. These annotations are used to create Spring beans automatically in the application context. @Component annotation is the main Stereotype Annotation. There are some Stereotype meta-annotations which is derived from @Component those are

@Service

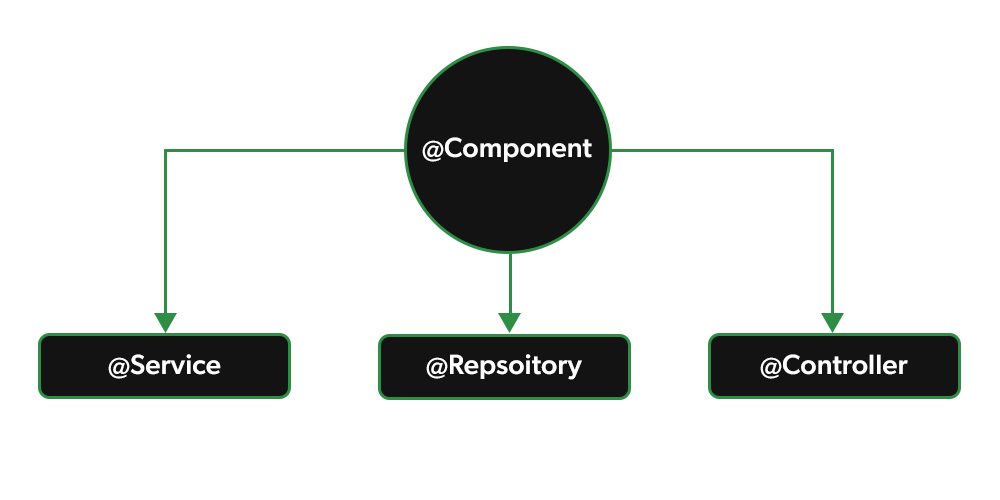
@Repository

@Controller

1: @Service: We specify a class with @Service to indicate that they’re holding the business logic. Besides being used in the service layer, there isn’t any other special use for this annotation. The utility classes can be marked as Service classes.

2: @Repository: We specify a class with @Repository to indicate that they’re dealing with CRUD operations, usually, it’s used with DAO (Data Access Object) or Repository implementations that deal with database tables.

3: @Controller: We specify a class with @Controller to indicate that they’re front controllers and responsible to handle user requests and return the appropriate response. It is mostly used with REST Web Services.



Bean Scope

The Spring Framework supports the following five scopes, three of which are available only if you use a web-aware ApplicationContext.

|  |  |
| --- | --- |
| **Sr.No.** | **Scope & Description** |
| 1 | **singleton**  This scopes the bean definition to a single instance per Spring IoC container (default). |
| 2 | **prototype**  This scopes a single bean definition to have any number of object instances. |
| 3 | **request**  This scopes a bean definition to an HTTP request. Only valid in the context of a web-aware Spring ApplicationContext. |
| 4 | **session**  This scopes a bean definition to an HTTP session. Only valid in the context of a web-aware Spring ApplicationContext. |
| 5 | **global-session**  This scopes a bean definition to a global HTTP session. Only valid in the context of a web-aware Spring ApplicationContext. |

If you are declaring bean inside xml file then we will have to declare scope there in bean declaration but if you are using stereotype annotation then we will have to use

@Scope annotation

<bean class=”” name=”” scope=”prototype”>

@Component

@Scope(“prototype”)

Class abc{

}

Spring

1. DI 🡪 SI, CI
2. How specify Bean in XML
3. ApplicationContext
4. Stereotype Annotation
5. **XML base configuration** in which we have specified beans in XML file only
6. **Mixed configuration** in which we didn’t specify beans in XML but we specified the packages in which our Bean classes are present and after that using stereotype Annotation like @Component, @Service and @Repository we specified beans.
7. **Java Based Configuration 🡪**

@Configuration

**@ComponentScan(base-packages=””)**

**Like xml you can specify bean with @Bean annotation**

**@ComponentScan :-** If I didn’t create a bean in bean.xml file, to treat the class as a component or bean where @componet annotation is specified.

Syntax:

<context:component-scan base-package = “com.demo3” ></ context:component-scan>

Note:- Give the package name component annotation is specified.