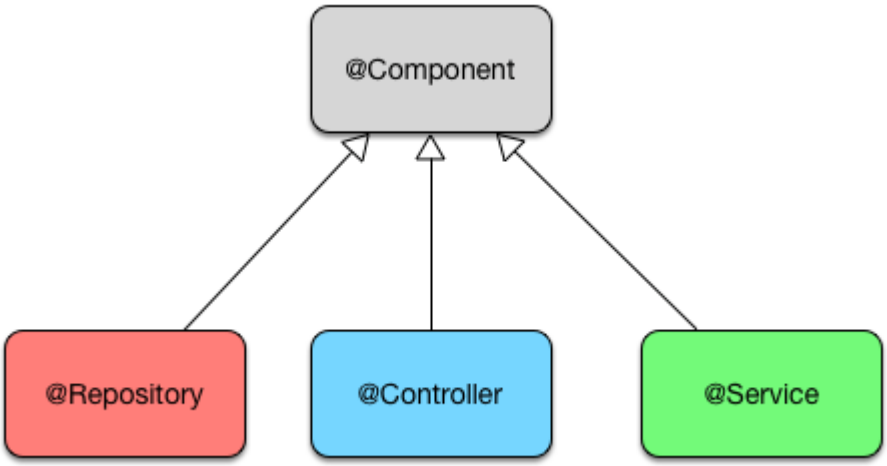
Spring Stereotypes ¿Como funcionan? . Spring define un conjunto de anotaciones core que categorizan cada uno de los componentes asociandoles una responsabilidad concreta. La anotación padre de todos es @Component



Spring Stereotypes

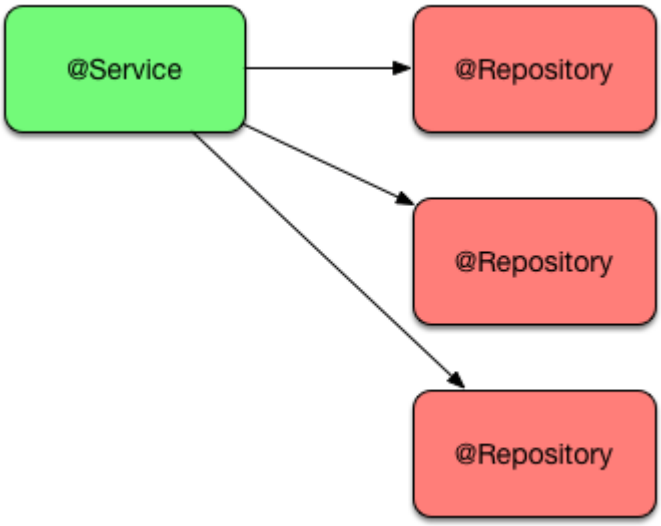
En estos momentos existen 4 Spring Stereotypes fundamentales :

@Component: Es el estereotipo general y permite anotar un bean para que Spring lo considere uno de sus objetos. Esto hará que Spring automáticamente lo registre y lo pueda inyectar como dependencia si es necesario. La clase se convierte en un componente de Spring Framework que soporta transaccionalidad etc. Esta anotación se usa mucho de apoyo cuando tenemos que configurar temas a detalle.

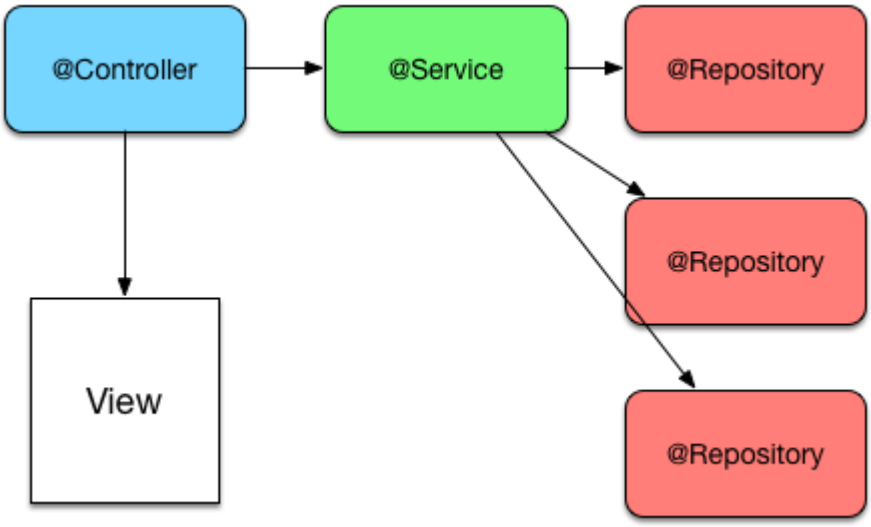
@Repository: Es el estereotipo que se encarga de dar de alta un bean para que implemente el patrón repositorio. Es el encargado de almacenar datos en una base de datos o repositorio de información que se necesite. Al marcar el bean con esta anotación Spring aporta servicios transversales como conversión de tipos de excepciones de JDBC a JPA a excepciones de datos manejadas por Spring.



@Service : Este estereotipo se encarga de gestionar las operaciones de negocio más importantes a nivel de la aplicación y aglutina llamadas a varios repositorios de forma simultánea. Su tarea fundamental es la de agregador implementa el patrón fachada y aglutina varios repositorios y llamadas a otros servicios.

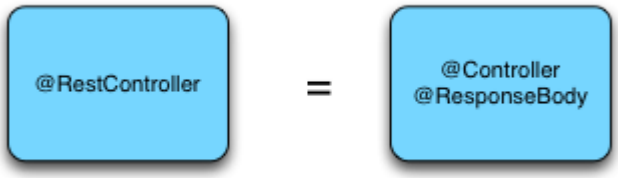


@Controller : El último de los estereotipos que es el que realiza las tareas de controlador y gestión de la comunicación entre el usuario y el aplicativo. Para ello se apoya habitualmente en algún motor de plantillas o librería de etiquetas que facilitan la creación de páginas.



@RestController

¿Existen más Spring Stereotypes dentro de Spring Framework? . Este es una pregunta muy habitual, la respuesta es No. Otras anotaciones que comunmente se ven como estereotipos no lo son . Un caso bastante clásico es el de @RestController que es una especialización de controller que contiene las anotaciones @Controller y @ResponseBody , pero realmente es una mera adaptación del clásico @Controller.



Los controladores de tipo RestController cada día se usan más en el universo de Spring Framework debido al mayor uso de las arquitecturas SPA en las aplicaciones clientes que usamos en el día a día y el uso de REST.

Otros artículos relacionados:

- Spring AOP y Aspectos

- Spring MVC
- Thymeleaf motor de plantillas