

Spring Annotation Cheat Sheet.

Core Spring Annotations

ANNOTATION	USE	DESCRIPTION
@Autowired	Constructor, Field, Method	Declares a constructor, field, setter method, or configuration method to be autowired by type. Items annotated with @Autowired do not have to be public.
@Configurable	Type	Used with to declare types whose properties should be injected, even if they are not instantiated by Spring. Typically used to inject the properties of domain objects.
@Order	Type, Method, Field	Defines ordering, as an alternative to implementing the org.springframework.core.Ordered interface.
@Qualifier	Field, Parameter, Type, Annotation Type	Guides autowiring to be performed by means other than by type.
@Required	Method (setters)	Specifies that a particular property must be injected or else the configuration will fail.
@Scope	Type	Specifies the scope of a bean, either singleton, prototype, request, session, or some custom scope.
@ComponentScan	Make Spring scan the package for the @Configuration classes.	
@Configuration	Mark a class as a source of bean definitions.	
@Bean	Indicates that a method produces a bean to be managed by the Spring container.	
@Lazy	Makes @Bean or @Component be initialized on demand rather than eagerly.	
@Value	Indicates a default value expression for the field or parameter, typically something like	

Stereotyping Annotations

ANNOTATION	USE	DESCRIPTION
@Component	Type	Generic stereotype annotation for any Spring-managed component.
@Controller	Type	Stereotypes a component as a Spring MVC controller.
@Repository	Type	Stereotypes a component as a repository. Also indicates that SQLExceptions thrown from the component's methods should be translated into Spring DataAccessExceptions.
@Service	Type	Stereotypes a component as a service.

Spring MVC Annotations

ANNOTATION	USE	DESCRIPTION
@Controller	Type	Stereotypes a component as a Spring MVC controller.
@InitBinder	Method	Annotates a method that customizes data binding.
@ModelAttribute	Parameter, Method	When applied to a method, used to preload the model with the value returned from the method. When applied to a parameter, binds a model attribute to the parameter. table
@RequestMapping	Method, Type	Maps a URL pattern and/or HTTP method to a method or controller type.
@RequestParam	Parameter	Binds a request parameter to a method parameter.
@SessionAttributes	Type	Specifies that a model attribute should be stored in the session.

Transaction Annotations

ANNOTATION	USE	DESCRIPTION
@Transactional	Method, Type	Declares transactional boundaries and rules on a bean and/or its methods.

Aspect Annotations

ANNOTATION	USE	DESCRIPTION
------------	-----	-------------

@Aspect	Type	Declares a class to be an aspect.
@After	Method	Declares a method to be called after a pointcut completes.
@AfterReturning	Method	Declares a method to be called after a pointcut returns successfully.
@AfterThrowing	Method	Declares a method to be called after a pointcut throws an exception.
@Around	Method	Declares a method that will wrap the pointcut.
@Before	Method	Declares a method to be called before proceeding to the pointcut.
@DeclareParents	Static Field	Declares that matching types should be given new parents, that is, it introduces new functionality into matching types.
@Pointcut	Method	Declares an empty method as a pointcut placeholder method.

JSR-250 Annotations

ANNOTATION	USE	DESCRIPTION
@PostConstruct	Method	Indicates a method to be invoked after a bean has been created and dependency injection is complete. Used to perform any initialization work necessary.
@PreDestroy	Method	Indicates a method to be invoked just before a bean is removed from the Spring context. Used to perform any cleanup work necessary.
@Resource	Method, Field	Indicates that a method or field should be injected with a named resource (by default, another bean).

Testing Annotations

ANNOTATION	USE	DESCRIPTION
@AfterTransaction	Method	Used to identify a method to be invoked after a transaction has completed.

@BeforeTransaction	Method	Used to identify a method to be invoked before a transaction starts.
@ContextConfiguration	Type	Configures a Spring application context for a test.
@DirtiesContext	Method	Indicates that a method dirties the Spring container and thus it must be rebuilt after the test completes.
@ExpectedException	Method	Indicates that the test method is expected to throw a specific exception. The test will fail if the exception is not thrown.
@IfProfileValue	Type, Method	Indicates that the test class or method is enabled for a specific profile configuration.
@NotTransactional	Method	Indicates that a test method must not execute in a transactional context.
@ProfileValueSourceConfiguration	Type	Identifies an implementation of a profile value source. The absence of this annotation will cause profile values to be loaded from system properties.
@Repeat	Method	Indicates that the test method must be repeated a specific number of times.
@Rollback	Method	Specifies whether or not the transaction for the annotated method should be rolled back or not.
@TestExecutionListeners	Type	Identifies zero or more test execution listeners for a test class.
@Timed	Method	Specifies a time limit for the test method. If the test does not complete

		before the time has expired, the test will fail.
@TransactionConfiguration	Type	Configures test classes for transactions, specifying the transaction manager and/or the default rollback rule for all test methods in a test class.

Spring Boot and Web annotations

ANNOTATION	DISCRIPTION
@SpringBootApplication	uses @Configuration, @EnableAutoConfiguration and @ComponentScan
@EnableAutoConfiguration	make Spring guess the configuration based on the classpath.
@ResponseBody	makes Spring bind method's return value to the web response body.
@RequestParam	bind HTTP parameters into method arguments.
@PathVariable	binds placeholder from the URI to the method parameter.