



Spring MVC Annot(dev.elma)

Annotation	Description
<code>@Controller</code>	Marks a class as a controller that handles HTTP requests.
<code>@RequestMapping</code>	Maps a URL pattern to a method or class.
<code>@PathVariable</code>	Binds a method parameter to a path variable in the request URL.
<code>@RequestParam</code>	Binds a method parameter to a query parameter in the request URL.
<code>@ModelAttribute</code>	Binds a method parameter to a model attribute.
<code>@RequestBody</code>	Binds the request body to a method parameter.
<code>@ResponseBody</code>	Indicates that the return value of a method should be serialized and sent as the response body.
<code>@ResponseStatus</code>	Specifies the HTTP response status code for a method.
<code>@ExceptionHandler</code>	Defines a method to handle a specific type of exception thrown by the controller.
<code>@SessionAttributes</code>	Specifies which model attributes should be stored in the HTTP session.
<code>@InitBinder</code>	Defines a method to initialize the <code>WebDataBinder</code> used for data binding in the controller.
<code>@RequestMapping(method)</code>	Maps a URL pattern to a method with a specific HTTP method (GET, POST, PUT, DELETE, etc.).
<code>@GetMapping</code>	Shortcut for <code>@RequestMapping(method = RequestMethod.GET)</code> .
<code>@PostMapping</code>	Shortcut for <code>@RequestMapping(method = RequestMethod.POST)</code> .
<code>@PutMapping</code>	Shortcut for <code>@RequestMapping(method = RequestMethod.PUT)</code> .
<code>@DeleteMapping</code>	Shortcut for <code>@RequestMapping(method = RequestMethod.DELETE)</code> .



Spring Security Annot (dev.elma)

Annotation	Description
<code>@EnableWebSecurity</code>	Enables Spring Security for a web application.
<code>@Configuration</code>	Indicates that a class is a configuration class that provides Spring bean definitions.
<code>@Secured</code>	Restricts access to a method to users who have been granted a specific set of authorities.
<code>@PreAuthorize</code>	Restricts access to a method based on a SpEL expression that evaluates to <code>true</code> .
<code>@PostAuthorize</code>	Restricts access to a method based on a SpEL expression that evaluates to <code>true</code> after the method has been called.
<code>@RolesAllowed</code>	Restricts access to a method to users who have been granted a specific set of roles.
<code>@EnableGlobalMethodSecurity</code>	Enables global method security for a Spring Security-enabled application.
<code>@AuthenticationPrincipal</code>	Provides access to the currently authenticated principal (user) in a controller method.
<code>@Order</code>	Specifies the order in which security filters should be applied to HTTP requests.



RestController(dev.elma)

Annotation	Description
<code>@RestController</code>	Marks a class as a RESTful web service controller.
<code>@GetMapping</code>	Maps HTTP GET requests to a specific method in the controller.
<code>@PostMapping</code>	Maps HTTP POST requests to a specific method in the controller.
<code>@PutMapping</code>	Maps HTTP PUT requests to a specific method in the controller.
<code>@DeleteMapping</code>	Maps HTTP DELETE requests to a specific method in the controller.
<code>@PathVariable</code>	Binds a method parameter to a path variable in the request URL.
<code>@RequestParam</code>	Binds a method parameter to a query parameter in the request URL.
<code>@RequestBody</code>	Binds the request body to a method parameter in the controller.
<code>@ResponseStatus</code>	Specifies the HTTP response status code for a method in the controller.
<code>@ExceptionHandler</code>	Defines a method to handle a specific type of exception thrown by the controller.



JPA Annotations (dev.elma)

Annotation	Purpose	Parameters
@Entity	Marks a class as a JPA entity	<code>name</code> , <code>catalog</code> , <code>schema</code> , <code>readOnly</code> , <code>dynamicInsert</code> , <code>dynamicUpdate</code>
@Id	Marks the primary key field of an entity	<code>name</code> , <code>columnDefinition</code> , <code>allocationSize</code>
@GeneratedValue	Specifies how the primary key value should be generated	<code>strategy</code> , <code>generator</code>
@Table	Specifies the name of the database table that the entity is mapped to	<code>name</code> , <code>catalog</code> , <code>schema</code> , <code>uniqueConstraints</code>
@Column	Specifies the mapping of a field to a database column	<code>name</code> , <code>nullable</code> , <code>unique</code> , <code>length</code> , <code>precision</code> , <code>scale</code> , <code>columnDefinition</code>
@OneToMany	Defines a one-to-many relationship between two entities	<code>targetEntity</code> , <code>cascade</code> , <code>fetch</code> , <code>mappedBy</code>
@ManyToOne	Defines a many-to-one relationship between two entities	<code>targetEntity</code> , <code>cascade</code> , <code>fetch</code> , <code>optional</code> , <code>mappedBy</code> , <code>joinColumns</code>
@ManyToMany	Defines a many-to-many relationship between two entities	<code>targetEntity</code> , <code>cascade</code> , <code>fetch</code> , <code>mappedBy</code> , <code>joinTable</code> , <code>joinColumns</code> , <code>inverseJoinColumns</code>
@JoinTable	Specifies the join table for a many-to-many relationship	<code>name</code> , <code>catalog</code> , <code>schema</code> , <code>joinColumns</code> , <code>inverseJoinColumns</code> , <code>uniqueConstraints</code>
@JoinColumn	Specifies the join column for a many-to-one or one-to-one relationship	<code>name</code> , <code>referencedColumnName</code> , <code>nullable</code> , <code>unique</code> , <code>insertable</code> , <code>updatable</code> , <code>columnDefinition</code> , <code>table</code>