



**SPRING**

**ANNOTATIONS**

THE ULTIMATE CHEATSHEET

Dependency Injection • MVC • Boot • Data • Cloud

Venkatesh Manchala

# Core Container & Beans

## Stereotype Annotations

|                    |  |
|--------------------|--|
| <b>@Component</b>  | Generic Spring-managed component.          |
| <b>@Service</b>    | Business logic layer.                      |
| <b>@Repository</b> | Data access (DAO) & exception translation. |
| <b>@Controller</b> | Web MVC controller.                        |

## Dependency Injection

|                   |  |
|-------------------|--|
| <b>@Autowired</b> | Injects dependencies (Constructor/Field/Setter). |
| <b>@Qualifier</b> | Selects a specific bean by name.                 |
| <b>@Primary</b>   | Default bean when multiple candidates exist.     |
| <b>@Value</b>     | Injects properties values ("\${key}").           |

## Configuration & Management

|                        |  |
|------------------------|--|
| <b>@Configuration</b>  | Source of bean definitions.            |
| <b>@Bean</b>           | Declares a method that returns a bean. |
| <b>@Scope</b>          | Defines scope (singleton, prototype).  |
| <b>@Lazy</b>           | Postpones initialization until access. |
| <b>@DependsOn</b>      | Enforces initialization order.         |
| <b>@PropertySource</b> | Loads external property files.         |
| <b>@Import</b>         | Imports other config classes.          |

# Spring Boot & MVC

## @SpringBootApplication

Combines @Configuration, @EnableAutoConfiguration, @ComponentScan.

## REST Request Handling

### @RestController

@Controller + @ResponseBody

### @RequestMapping

Maps HTTP requests to methods.

### @Get/Post/Put/DeleteMapping

Shortcut for specific HTTP methods.

### @PathVariable / @RequestParam

Extracts URI data / Query params.

**@RequestBody**      Deserializes JSON to Java Object.

## Global Exception Handling

### @ControllerAdvice

Global exception handling for multiple controllers.

**@ExceptionHandler**      Catches specific exceptions.

HTTP

# Data & Validation

## SQL / JPA (Relational)

|                       |                                 |
|-----------------------|---------------------------------|
| <b>@Entity</b>        | Declares a JPA entity.          |
| <b>@Table / @Id</b>   | Maps table name / Primary Key.  |
| <b>@Column</b>        | Maps a field to a DB column.    |
| <b>@OneToMany</b>     | Defines entity relationships.   |
| <b>@Transactional</b> | Manages transactions atomicity. |

## MongoDB

|           |                     |
|-----------|---------------------|
| @Document | Marks class as doc. |
| @Id       | Primary Key.        |
| @Field    | Key mapping.        |

## Cassandra

|               |                |
|---------------|----------------|
| @Table        | Defines table. |
| @PrimaryKey   | Key column.    |
| @PartitionKey | Partitioning.  |

## Validation (@Valid)

|                       |                           |
|-----------------------|---------------------------|
| <b>@NotBlank</b>      | String is not null/empty. |
| <b>@Size(min,max)</b> | Validates length.         |
| <b>@Email</b>         | Validates email format.   |
| <b>@Min / @Max</b>    | Numeric validation.       |

# Architecture & Security

## AOP (Aspect Oriented)

|                  |                                     |
|------------------|-------------------------------------|
| <b>@Aspect</b>   | Defines an aspect (logging, auth).  |
| <b>@Before</b>   | Runs before method execution.       |
| <b>@After</b>    | Runs after execution (finally).     |
| <b>@Around</b>   | Wraps execution (Pre & Post).       |
| <b>@Pointcut</b> | Defines reusable match expressions. |

## Spring Cloud

|                            |                                  |
|----------------------------|----------------------------------|
| <b>@EnableEurekaClient</b> | Registers with Service Registry. |
| <b>@FeignClient</b>        | Declarative REST client.         |
| <b>@CircuitBreaker</b>     | Fault tolerance logic.           |
| <b>@EnableConfigServer</b> | Centralized config server.       |

## Spring Security

|                                 |                            |
|---------------------------------|----------------------------|
| <b>@EnableWebSecurity</b>       | Enables security config.   |
| <b>@Secured / @PreAuthorize</b> | Role-based access control. |
| <b>@AuthenticationPrincipal</b> | Injects current user.      |

*Example: @PreAuthorize("hasRole('ADMIN')")*

# Testing & Async

## Testing Annotations

|                        |                                 |
|------------------------|---------------------------------|
| <b>@SpringBootTest</b> | Loads full application context. |
| <b>@WebMvcTest</b>     | Unit test for Controllers only. |
| <b>@MockBean</b>       | Mocks a dependency bean.        |
| <b>@DataJpaTest</b>    | Tests JPA repositories.         |
| <b>@Test</b>           | Marks method as a test case.    |

## Scheduling, Async & Caching

|                             |                            |
|-----------------------------|----------------------------|
| <b>@Async</b>               | Runs in a separate thread. |
| <b>@Scheduled(cron=...)</b> | Runs task on intervals.    |
| <b>@EnableCaching</b>       | Enables caching support.   |
| <b>@Cacheable("name")</b>   | Caches method result.      |

## Messaging (Kafka/WebSocket)

|                         |                          |
|-------------------------|--------------------------|
| <b>@KafkaListener</b>   | Listens to Kafka topics. |
| <b>@EnableWebSocket</b> | Enables WS support.      |
| <b>@MessageMapping</b>  | Maps WS messages.        |

