

# Java SpringBoot Annotation

**Name: Syedanwar Hanzahusen Mashalkar**

## Core Annotations

### 1. @SpringBootApplication

- Combines @Configuration, @EnableAutoConfiguration, and @ComponentScan. Indicates a configuration class that triggers auto-configuration and component scanning.

## Configuration and Bean Annotations

### 2. @Configuration

- Marks a class as a source of bean definitions for the Spring IoC container.

### 3. @Bean

- Indicates that a method produces a bean to be managed by the Spring container.

## Dependency Injection Annotations

### 4. @Autowired

- Marks a constructor, field, setter method, or config method to be autowired by Spring's dependency injection facilities.

### 5. @Qualifier

- Specifies which bean should be injected when multiple beans of the same type are present.

## Component Scanning Annotations

### 6. @Component

- Indicates that a class is a "component" and a candidate for auto-detection when using annotation-based configuration and classpath scanning.

### 7. @Service

- A specialization of @Component, indicating a service layer class.

### 8. @Repository

- A specialization of @Component, indicating a data access object (DAO).

### 9. @Controller

- A specialization of @Component, used to define a web controller in Spring MVC.

### 10. @RestController

- Combines @Controller and @ResponseBody, indicating that data returned by methods will be written directly to the response body.

## Web Annotations

### 11. **@RequestMapping**

- Maps web requests to specific handler classes or handler methods.

### 12. **@GetMapping, @PostMapping, @PutMapping, @DeleteMapping, @PatchMapping**

- Specialized versions of `@RequestMapping` for HTTP GET, POST, PUT, DELETE, and PATCH requests, respectively.

## Additional Useful Annotations

### 13. **@Value**

- Injects values into fields from a properties file or environment variables.

### 14. **@PropertySource**

- Specifies the location of properties files.

### 15. **@EnableAutoConfiguration**

- Enables Spring Boot's auto-configuration mechanism.

These annotations are essential for configuring, managing, and simplifying the development of Spring Boot applications, providing a wide range of functionalities from dependency injection to web request handling.