

Spring Boot, Spring Data JPA & Lombok - Annotation Cheatsheet

Framework	Annotation	Description
Spring Boot	@Autowired	Auto-wires dependencies
Spring Boot	@Bean	Marks a method to return a Spring-managed bean
Spring Boot	@Component	Generic stereotype for a Spring bean
Spring Boot	@ComponentScan	Configures scanning for beans in packages
Spring Boot	@Configuration	Marks a class for bean definitions
Spring Boot	@Controller	MVC controller handling web requests
Spring Boot	@CrossOrigin	Enables CORS for REST APIs
Spring Boot	@EnableAutoConfiguration	Auto-configures beans based on classpath
Spring Boot	@EnableScheduling	Enables scheduling support
Spring Boot	@Lazy	Delays bean initialization until first use
Spring Boot	@Primary	Marks a bean as preferred when multiple candidates exist
Spring Boot	@Profile	Specifies environment-based beans (dev, prod)
Spring Boot	@PropertySource	Loads properties file into Spring Environment
Spring Boot	@Qualifier	Selects bean when multiple of same type exist
Spring Boot	@Repository	Specializes a class as a DAO component
Spring Boot	@RequestMapping	Maps HTTP requests to handler methods
Spring Boot	@RestController	Combines @Controller + @ResponseBody
Spring Boot	@Scope	Defines bean scope (singleton, prototype)
Spring Boot	@Service	Service layer annotation
Spring Boot	@SpringBootApplication	Combines @Configuration, @EnableAutoConfiguration, @ComponentScan
Spring Data JPA	@Entity	Marks a class as a JPA entity
Spring Data JPA	@Table	Customizes the table name for an entity
Spring Data JPA	@Id	Primary key field
Spring Data JPA	@GeneratedValue	Auto-generates PK values
Spring Data JPA	@Column	Customizes column mapping
Spring Data JPA	@Transient	Ignores a field in DB mapping
Spring Data JPA	@Enumerated	Maps enums to DB (ORDINAL / STRING)
Spring Data JPA	@Lob	Large object (CLOB / BLOB)
Spring Data JPA	@ManyToOne	Many-to-one relationship
Spring Data JPA	@OneToMany	One-to-many relationship
Spring Data JPA	@OneToOne	One-to-one relationship
Spring Data JPA	@ManyToMany	Many-to-many relationship
Spring Data JPA	@JoinColumn	Specifies FK column

Framework	Annotation	Description
Spring Data JPA	@JoinTable	Defines join table for many-to-many
Spring Data JPA	@Embedded	Embeds another entity in a table
Spring Data JPA	@Embeddable	Makes a class embeddable in another
Spring Data JPA	@MappedSuperclass	Superclass with mapping info but not an entity
Spring Data JPA	@Version	Enables optimistic locking
Spring Data JPA	@Query	Defines custom JPQL/SQL queries
Spring Data JPA	@Modifying	Marks a query as update/delete
Spring Data JPA	@Transactional	Wraps DB ops in a transaction
Spring Data JPA	@EnableJpaRepositories	Enables JPA repository scanning
Lombok	@AllArgsConstructor	Creates constructor with all fields
Lombok	@NoArgsConstructor	Creates no-arg constructor
Lombok	@RequiredArgsConstructor	Creates constructor for final/@NonNull fields
Lombok	@Builder	Enables Builder pattern
Lombok	@Data	Shortcut for getters, setters, toString, equals & hashCode
Lombok	@Getter	Generates getter methods
Lombok	@Setter	Generates setter methods
Lombok	@EqualsAndHashCode	Generates equals() & hashCode()
Lombok	@ToString	Generates toString()
Lombok	@Value	Immutable class (all fields final, getters only)
Lombok	@Slf4j	Logger support
Lombok	@Log, @Log4j2, @CommonsLog	Different logger backends
Lombok	@SneakyThrows	Skip checked exception handling
Lombok	@Cleanup	Auto-closes resources
Lombok	@NonNull	Null-checks method params
Lombok	@With	Creates new object with one modified field (immutables)
Lombok	@Singular	For builder collection fields
Lombok	@SuperBuilder	Builder for inheritance
Lombok	@FieldDefaults	Control field visibility (private final default)
Lombok	@Accessors	Chainable/setter style customization