

Ultimate Spring Boot Revision Question Bank (150+ Questions)

1. What is Spring Boot and why was it created?
2. How is Spring Boot different from the Spring Framework?
3. What are the major advantages of using Spring Boot?
4. How does Spring Boot simplify Spring application development?
5. What problems does Spring Boot solve in enterprise development?
6. What are the core features of Spring Boot?
7. In what real-world use cases should you choose Spring Boot?
8. Name major companies using Spring Boot in production and why.
9. What is Inversion of Control (IoC)?
10. What is Dependency Injection (DI)?
11. Difference between constructor injection, setter injection, and field injection.
12. Why is constructor injection recommended in Spring Boot?
13. What is a Spring Bean?
14. How does Spring manage, instantiate, and destroy beans?
15. What are the different bean scopes?
16. What is the difference between singleton vs prototype scope?
17. How does Spring Boot perform component scanning?
18. What is the role of @ComponentScan?
19. What is the Bean lifecycle in Spring Boot?
20. What are @PostConstruct and @PreDestroy?
21. What is BeanFactory vs ApplicationContext?
22. How does Spring resolve bean conflicts?
23. What is @Primary?
24. What is @Qualifier?
25. What are Lazy-loaded vs Eager-loaded beans?
26. What are the core components of Spring Boot?
27. What exactly is Spring Boot Auto-Configuration?
28. How does auto-configuration work internally?
29. What is @EnableAutoConfiguration?
30. What are conditional annotations?

31. What is Spring Boot CLI?
32. What are Spring Boot Starters?
33. Why are starter POMs better than manual dependency management?
34. What is Spring Boot Actuator?
35. What is a Spring Boot starter dependency?
36. List commonly used starters.
37. Difference between starter POM vs normal dependency.
38. What is spring-boot-starter-parent?
39. How do Spring Boot starters simplify dependency version conflicts?
40. What does spring-boot-starter-web include?
41. What does spring-boot-starter-data-jpa include?
42. What is @SpringBootApplication?
43. What three annotations are inside @SpringBootApplication?
44. Difference between @Component, @Service, @Repository, @Controller.
45. What is @RestController?
46. Difference between @RequestMapping and @GetMapping.
47. What is @Value?
48. What is @ConfigurationProperties?
49. Difference between @Bean and @Component.
50. What is application.properties used for?
51. Difference between properties and YAML?
52. How does Spring Boot load external configuration?
53. What is the order of property loading?
54. What are Spring Profiles?
55. How do you activate profiles?
56. How to create REST API in Spring Boot?
57. Difference between MVC and REST controllers.
58. How to handle HTTP methods?
59. What is ResponseEntity?
60. How to handle parameters?
61. How to validate input?
62. What is DTO pattern?

63. How to handle exceptions?
64. What is @ControllerAdvice?
65. What is @ExceptionHandler?
66. How to build custom error responses?
67. What is @ResponseStatus?
68. What is a Filter?
69. How to create custom filters?
70. What is HandlerInterceptor?
71. Differences: Filters vs Interceptors.
72. What is AOP?
73. What are join points and pointcuts?
74. What is @Aspect?
75. How does Spring AOP work?
76. What is Spring Data JPA?
77. How to configure database?
78. What is @Entity?
79. What is @Repository?
80. How to write custom queries?
81. CRUD vs JPA vs Paging repository?
82. What is @Transactional?
83. What is propagation & isolation level?
84. What is N+1 problem?
85. How does lazy loading work?
86. How to configure multiple databases?
87. Embedded DBs in Spring Boot.
88. Configure H2.
89. Connect to MySQL/PostgreSQL.
90. What is HikariCP?
91. How to write Spring Boot unit tests?
92. What is @SpringBootTest?
93. What is @MockBean?

94. What is @WebMvcTest?
95. What is @DataJpaTest?
96. How to test REST controllers?
97. What is TestRestTemplate?
98. How to use MockMVC?
99. How to secure Spring Boot?
100. What is Spring Security starter?
101. How to enable authentication?
102. How to enable HTTPS?
103. What is @PreAuthorize?
104. How to implement JWT?
105. What is embedded server?
106. Supported servers.
107. Change default port.
108. Configure Tomcat.
109. Deploy as WAR?
110. Advantages of embedded servers?
111. What is Actuator?
112. Actuator endpoints.
113. Implement health checks.
114. Monitor performance.
115. What is Micrometer?
116. Secure Actuator endpoints.
117. How to enable caching?
118. What is @EnableCaching?
119. What is @CacheEvict?
120. What is @EnableAsync?
121. Async execution.
122. What is @Scheduled?
123. What is DevTools?
124. How does auto-restart work?
125. DevTools features.

- 126. Disable DevTools for production.
- 127. Features for microservices.
- 128. What is Spring Cloud?
- 129. What is API Gateway?
- 130. Service discovery.
- 131. Circuit breaker.
- 132. Spring Cloud Config.
- 133. Load balancing.
- 134. Deployment methods.
- 135. Spring Boot Maven plugin.
- 136. Docker images.
- 137. Layered JARs.
- 138. Optimization for production.
- 139. Logging best practices.
- 140. Thread/memory tuning.
- 141. Externalizing secrets.
- 142. Spring Boot startup process.
- 143. ApplicationRunner vs CommandLineRunner.
- 144. How autoconfiguration is detected?
- 145. What is SpringFactoriesLoader?
- 146. Circular dependency resolution.
- 147. What is BeanPostProcessor?
- 148. What is BeanFactoryPostProcessor?
- 149. What is ApplicationContextInitializer?
- 150. How annotation scanning works internally?