

Interview Question – Core and Advance Java

1. What is the difference between `.equals()` and `==`?
2. What is the difference between `StringBuilder` and `StringBuffer`?
3. What is OOPS in Java? Explain its principles.
4. What is the difference between method overloading and overriding?
5. Is it based on return type?
6. In which cases is return type considered?
7. How do we achieve encapsulation in Java?
8. Why do we use private and public with setters and getters?
9. What is a marker interface and how does it work?
10. What are exceptions in Java and what are their types?
11. What happens if a checked exception is not handled?
12. How do we handle runtime exceptions?
13. What is the finally block and when is it executed?
14. Is finally always executed?
15. What is the return value if finally or exception block has a return?
16. How do you create a custom exception?
17. Explain multithreading in Java.
18. What is the Executor framework?
19. What is the difference between `Executor` and `ExecutorService`?
20. How does `ExecutorService` work internally?
21. What is the `synchronized` keyword and how does it work?
22. How do you use `synchronized` in your code?
23. What are the advantages and disadvantages of `synchronized`?
24. Write a program using a `synchronized` method.
25. What is the `volatile` keyword in Java?
26. What is a fail-fast vs fail-safe iterator?
27. What is `Spliterator` in Java?
28. What are the main features of Java 8?
29. Explain all Java 8 features in detail.
30. How does the Stream API work internally?
31. What do terminal operations return in Stream API?
32. How do you find even numbers using Stream API?
33. What is the difference between `ArrayList` and `LinkedList`?
34. When should we use `ArrayList` vs `LinkedList`?
35. How does `HashSet` use the `equals()` method?
36. How do we use the `contains()` method in collections?
37. How does `HashMap` work internally?
38. What is a hash collision in a `HashMap`?
39. What is the difference between `HashMap` and `Hashtable`?
40. What are the key memory areas in JVM?
41. What is the difference between stack and heap?
42. What are Java Records?
43. What is a setter method in Java?

44. What is the difference between @Component, @Service, and @Repository?
45. Can we replace @Service and @Repository with @Component?
46. How do you create a global exception handler in Spring Boot?
47. How does Spring Security work in a real-time project?
48. What is Authentication and Authorization?
49. How do you implement Spring Security in a project?
50. Which annotations are used for authorization in Spring?
51. What is the purpose of @PreAuthorize?
52. What are UserDetails and UserDetailsService in Spring Security?
53. What is a REST API?
54. What is the difference between PUT and PATCH in REST?
55. What is versioning in REST APIs?
56. What is the difference between @RestController and @Controller?
57. What are the different types of joins in SQL?
58. How does LEFT JOIN work in SQL?
59. What is the difference between WHERE and HAVING clauses?
60. How do you find duplicate records in SQL? (Write the query)
61. What is indexing in SQL and how does it work?
62. What are SOLID principles in software design?
63. Explain each SOLID principle with examples.
64. What is service discovery in microservices?
65. How do you secure microservices?
66. What is Docker and what is its purpose?
67. How do you use Docker with Spring Boot?
68. How do you set up Docker for a Spring Boot application?