These are commonly asked in technical interviews

**Basic Java Questions**

1. What is Java, and why is it platform-independent, main features?

2. What is the diff b/w JDK, JRE, and JVM?

3. What is the diff b/w `public`, `private`, `protected`, `default` access modifiers?

4. What is the significance of the `main()` method in Java?

5. What is the diff b/w `==` and `.equals()` in Java, what are primitive DT’s in Java?

6. What is autoboxing and unboxing in Java?

7. What is the purpose of the `final` keyword in Java?

8. What is a constructor, and how is it different from a method?

**Object-Oriented Programming (OOP) Questions**

9. What are the four pillars of OOP in Java?

10. What is the diff b/w inheritance and polymorphism, overloading and overriding?

11. Can you override a static method in Java? Why or why not?

12. What is an abstract class, and how is it different from an interface?

13. What is the `super` keyword used for in Java?

14. What is encapsulation, and how is it achieved in Java?

15. What is the diff b/w `this` and `super` keywords?

16. Can a constructor be private in Java? If yes, why?

17. What is the purpose of the `instanceof` operator?

**Collections and Data Structures**

18. What is the Java Collections Framework?

19. What is the diff b/w `ArrayList` and `LinkedList`?

20. How does a `HashMap` work internally in Java?

21. What is the diff b/w `HashSet` and `TreeSet`?

22. What is the diff b/w `List`, `Set`, and `Map`?

23. How do you make a collection thread-safe?

24. What is the `Comparable` vs. `Comparator` interface?

25. What is the significance of the `equals()` and `hashCode()` methods in collections? 26. What is an Iterator, and how is it used?

27. What is the diff b/w `fail-fast` and `fail-safe` iterators?

**Exception Handling**

28. What is an exception in Java?

29. What is the diff b/w `checked` and `unchecked` exceptions?

30. What is the purpose of the `try`, `catch`, and `finally` blocks?

31. What is the diff b/w `throw` and `throws` in Java?

32. Can you catch multiple exceptions in a single `catch` block? How?

**Multithreading**

33. What is multithreading in Java, and how is it achieved?

34. What is the diff b/w a `Thread` class and a `Runnable` interface?

35. What is the purpose of the `synchronized` keyword?

36. What is a deadlock, and how can it be avoided?

37. What is the diff b/w `wait()` and `sleep()` methods?

**Advanced Java**

38. What is the diff b/w `String`, `StringBuilder`, and `StringBuffer`?

39. What is garbage collection in Java, and how does it work?

40. What is the purpose of the `volatile` keyword?

41. What are Java annotations, and how are they used?

42. What is the diff b/w `Serializable` and `Externalizable` interfaces?

**Java 8 and Beyond**

43. What are the new features introduced in Java 8?

44. What is a Lambda Expression, and how is it used?

45. What is the Stream API, and how does it differ from collections?

46. What is the purpose of the `Optional` class in Java?

47. What is the diff b/w `default` and `static` methods in interfaces?