

Java Basics & Features

What is Java?

1. What is Java and what are its main features?
2. Explain "Write Once, Run Anywhere" principle in Java.
3. What are the advantages and disadvantages of Java?
4. How is Java platform-independent?
5. What is bytecode in Java?

Memory Management

6. Explain Java memory management and garbage collection.
7. What is the difference between stack and heap memory?
8. How does garbage collection work in Java?
9. What are memory leaks in Java and how to prevent them?
10. Explain different types of references in Java (Strong, Weak, Soft, Phantom).

Default Values & Variables

11. What are the default values assigned to variables in Java?
12. What happens if you don't initialize a local variable?
13. What are the different types of variables in Java?
14. Explain the difference between instance and static variables.
15. What is the scope of variables in Java?

Public Static Void Main

16. Explain each keyword in `public static void main(String[] args)`.
17. Why is the main method static in Java?
18. Can we have multiple main methods in a Java class?
19. Can we overload the main method?
20. What happens if we remove 'static' from the main method?

Packages & Access Modifiers

Packages

- 21. What are packages in Java and why are they used?
- 22. How many built-in packages does Java have?
- 23. What is the difference between import and static import?
- 24. Can a class belong to multiple packages?
- 25. What is the naming convention for packages?

Wrapper Classes

- 26. What are wrapper classes in Java?
- 27. What is autoboxing and unboxing?
- 28. When should you use wrapper classes vs primitives?
- 29. What is the difference between `==` and `.equals()` with wrapper classes?
- 30. Explain the Integer cache in Java.

Default Values & Variable Assignment

- 31. What are default values for different data types?
- 32. Can you change the default values of instance variables?
- 33. What is the difference between declaration and initialization?
- 34. Explain variable shadowing in Java.
- 35. What happens when you access an uninitialized variable?

System.out vs System.err

- 36. What is the difference between `System.out` and `System.err`?
- 37. Can you redirect `System.out` and `System.err`?
- 38. What is `System.in` used for?
- 39. Explain the `PrintStream` class.
- 40. How do you format output using `System.out.printf()`?

Classes & Objects

- 41. What is the difference between a class and an object?
- 42. How many objects can be created from a single class?
- 43. What is a constructor and its types?

44.Can a class exist without a constructor?

45.What is the difference between default and parameterized constructors?

Object-Oriented Programming

Explain OOPs Concepts

46.What are the four pillars of Object-Oriented Programming?

47.Explain encapsulation with an example.

48.What is inheritance and its types?

49.Explain polymorphism and its types.

50.What is abstraction and how is it achieved?

Abstract Class vs Interface

51.What is the difference between abstract class and interface?

52.When would you use an abstract class vs an interface?

53.Can an abstract class have constructors?

54.Can an interface have concrete methods? (Java 8+)

55.What are default and static methods in interfaces?

String & StringBuffer

56.What is the difference between String, StringBuffer, and StringBuilder?

57.Why are Strings immutable in Java?

58.What is the String pool?

59.When should you use StringBuilder over String concatenation?

60.Explain the intern() method in String class.

Collections Framework

Java Collections

61.What is the Collection Framework in Java?

62.What are the main interfaces in the Collection Framework?

63.Explain the hierarchy of Collection classes.

64.What is the difference between Collection and Collections?

65.When to use ArrayList vs LinkedList vs Vector?

List, Set, Map

- 66.What is the difference between List, Set, and Map?
- 67.Explain HashMap vs HashTable vs ConcurrentHashMap.
- 68.What is the difference between HashSet and TreeSet?
- 69.How does HashMap work internally?
- 70.What is the load factor in HashMap?

Coding & Pattern Problems

Basic Programming

- 71.Write a program to check if a number is prime.
- 72.Write a program to find factorial of a number.
- 73.Write a program to generate Fibonacci series.
- 74.Write a program to reverse a string.
- 75.Write a program to check if a string is palindrome.

Pattern Printing

- 76.Write a program to print a pyramid pattern.
- 77.Write a program to print diamond pattern.
- 78.Write a program to print number patterns.
- 79.Write a program to print Floyd's triangle.
- 80.Write a program to print Pascal's triangle.

Advanced Java Concepts

Exception Handling

- 81.What is exception handling in Java?
- 82.What is the difference between checked and unchecked exceptions?
- 83.Explain try-catch-finally blocks.
- 84.What is the difference between throw and throws?
- 85.Can you have multiple catch blocks for a single try?

Multithreading

- 86.What is multithreading in Java?

- 87.What are the ways to create threads in Java?
- 88.Explain synchronization in Java.
- 89.What is deadlock and how to prevent it?
- 90.What is the difference between wait() and sleep() methods?

Java 8 Features

- 91.What are lambda expressions?
- 92.Explain the Stream API in Java 8.
- 93.What is the Optional class?
- 94.What are method references?
- 95.What are functional interfaces?

Practical Scenario Questions

- 96.How would you design a simple library management system?
- 97.Explain how you would implement a parking lot system.
- 98.How would you handle a situation where multiple threads access shared data?
- 99.Design a simple calculator using OOP principles.
- 100.How would you optimize the performance of a Java application?

Bonus Questions for Depth

- 101.Explain JVM architecture.
- 102.What is reflection in Java?
- 103.What are annotations and how are they used?
- 104.Explain the Singleton design pattern.
- 105.What is dependency injection?

Core String Concepts

Basic Understanding

1. **What is String in Java?**
2. **Is String a primitive or derived type?**
3. **How do you create a String object in Java?**
4. **What is String pool in Java?**
5. **Explain String immutability in Java.**

6. What happens when you create a String using new keyword vs string literal?

String vs Other Classes

7. What is the difference between String, StringBuffer, and StringBuilder?
8. When should you use StringBuilder over String concatenation?
9. Difference between String in C and String in Java.
10. Why is char array preferred over String for storing passwords?

String Methods & Operations

Basic String Methods

11. How do you find the length of a String?
12. How do you check if a String is empty in Java?
13. How do you convert String to uppercase/lowercase?
14. How do you replace a character in a String?
15. How do you check if a String contains a substring?
16. What is the substring() method in Java?
17. Explain the charAt() method.
18. What is the indexOf() method used for?

String Comparison

19. What is the difference between == and equals() when comparing Strings?
20. How do you compare two Strings in Java?
21. What is the compareTo() method?
22. Explain the equalsIgnoreCase() method.

String Conversion & Manipulation

Type Conversion

23. How do you convert String to int in Java?
24. How do you convert int to String in Java?
25. How can a Java String be converted into a byte array?
26. How do you convert String to char array?
27. How do you convert char array to String?

Advanced Operations

- 28. What is `String.intern()` method?
- 29. Explain `String.format()` method.
- 30. What is `StringTokenizer` in Java?
- 31. What is `StringJoiner` in Java?

String Programming Questions

Basic Programming

- 32. Write a program to reverse a `String`.
- 33. Write a program to check if a `String` is palindrome.
- 34. Write a program to count vowels and consonants in a `String`.
- 35. Write a program to count occurrences of a character in `String`.
- 36. Write a program to remove spaces from a `String`.
- 37. Write a program to count words in a `String`.

Pattern & Logic Questions

- 38. Write a program to check if two `Strings` are anagrams.
- 39. Write a program to find first non-repeating character.
- 40. Write a program to compress a `String` (replace repeated characters with count).
- 41. Write a program to check if two `Strings` are rotations of each other.
- 42. Write a program to find all permutations of a `String`.
- 43. Write a program to remove duplicate characters from `String`.

Memory & Performance

Memory Management

- 44. How much memory does `String` occupy?
- 45. What is `String pool` and how does it save memory?
- 46. Explain `String interning` concept.
- 47. Why `String` is used as `HashMap` key?

Performance Questions

- 48. Which is faster - `StringBuffer` or `StringBuilder`?

- 49. What is the performance difference between String concatenation using + vs StringBuilder?
- 50. How does garbage collection work with Strings?

Advanced String Concepts

Thread Safety

- 51. Is String thread-safe?
- 52. Is StringBuffer thread-safe?
- 53. Is StringBuilder thread-safe?

Special Cases

- 54. Can you modify a String after creation?
- 55. What happens when you concatenate null with String?
- 56. How do you handle String encoding in Java?
- 57. What is the difference between String.valueOf() and toString()?

Common Coding Interview Questions

Easy Level

- 58. Find length of String without using length() method
- 59. Check if String contains only digits
- 60. Convert first character of each word to uppercase
- 61. Remove all white spaces from String
- 62. Find ASCII value of characters in String

Medium Level

- 63. Find longest palindromic substring
- 64. Count occurrences of each character in String
- 65. Find all substrings of a String
- 66. Check if String is a valid number
- 67. Implement String matching algorithm

Pattern-Based Questions (From Your Notes)

- 68. Print String in pyramid pattern

- 69. **Print String in reverse pyramid pattern**
- 70. **Print String in diamond pattern**
- 71. **Print String character frequency pattern**
- 72. **Print String in zigzag pattern**

Core Array Concepts

Basic Understanding

- 1. **What is an array in Java?**
- 2. **How do you declare an array in Java?**
- 3. **How do you initialize an array in Java?**
- 4. **What is the difference between array declaration and initialization?**
- 5. **Can you change the size of an array after creation?**
- 6. **What is the length property of an array?**

Array Properties

- 7. **What are the default values of array elements in Java?**
- 8. **Can an array contain elements of different data types?**
- 9. **What happens when you access an index outside array bounds?**
- 10. **What is ArrayIndexOutOfBoundsException?**
- 11. **Can you pass a negative number as array size?**
- 12. **What is ArrayStoreException in Java?**

Array Types & Advanced Concepts

Multidimensional Arrays

- 13. **What is a multidimensional array in Java?**
- 14. **What is a jagged array in Java?**
- 15. **How do you declare a 2D array?**
- 16. **What is the difference between regular and jagged arrays?**
- 17. **How do you initialize a 3D array?**

Anonymous Arrays

- 18. **What is an anonymous array in Java?**

19. How do you create an anonymous array?

20. When would you use anonymous arrays?

Array vs Other Data Structures

Array vs ArrayList

21. What is the difference between Array and ArrayList?

22. When should you use Array vs ArrayList?

23. Which is faster - Array or ArrayList?

24. Can ArrayList store primitive types directly?

Array vs Other Collections

25. What is the difference between Array and LinkedList?

26. What are the advantages of arrays over other data structures?

27. What are the drawbacks of arrays in Java?

28. What are the benefits of heap over sorted arrays?

Array Operations & Methods

Basic Operations

29. How do you find the length of an array?

30. How do you access elements in an array?

31. How do you sort an array in Java?

32. How do you copy an array in Java?

33. What is the Arrays.toString() method?

34. What is the Arrays.equals() method?

Search Operations

35. How do you search for an element in an array?

36. What is linear search in arrays?

37. What is binary search in arrays?

38. How do you check if an array contains a specific value?

Array Programming Questions

Basic Programming

- 39. Write a program to find the maximum element in an array.
- 40. Write a program to find the minimum element in an array.
- 41. Write a program to reverse an array.
- 42. Write a program to find the sum of array elements.
- 43. Write a program to find the average of array elements.
- 44. Write a program to count elements in an array.

Intermediate Programming

- 45. Write a program to find duplicate elements in an array.
- 46. Write a program to remove duplicate elements from an array.
- 47. Write a program to find the second largest element.
- 48. Write a program to rotate an array left/right.
- 49. Write a program to merge two arrays.
- 50. Write a program to find common elements between two arrays.

Pattern & Logic Questions

- 51. Write a program to find missing number in array.
- 52. Write a program to find the median of an array.
- 53. Write a program to check if array is sorted.
- 54. Write a program to find frequency of each element.
- 55. Write a program to separate even and odd numbers.
- 56. Write a program to find pairs with given sum.

Advanced Array Concepts

Memory & Performance

- 57. How much memory does an array occupy?
- 58. What is the time complexity for accessing array elements?
- 59. What is the time complexity for searching in arrays?
- 60. Can arrays be resized at runtime?
- 61. How does array memory allocation work?

Array Algorithms

- 62. How do you implement bubble sort on arrays?
- 63. How do you implement selection sort on arrays?
- 64. How do you implement insertion sort on arrays?
- 65. How do you implement quick sort on arrays?
- 66. How do you implement merge sort on arrays?

Common Coding Interview Questions

Easy Level

- 67. Find largest and smallest elements in array
- 68. Count positive and negative numbers in array
- 69. Check if array is palindrome
- 70. Move all zeros to end of array
- 71. Find intersection of two arrays
- 72. Convert array to ArrayList and vice versa

Medium Level

- 73. Find all subarrays with given sum
- 74. Find maximum subarray sum (Kadane's algorithm)
- 75. Implement array rotation by k positions
- 76. Find majority element in array
- 77. Rearrange array in alternating positive negative
- 78. Find triplets with zero sum

Hard Level

- 79. Find median of two sorted arrays
- 80. Implement next permutation of array
- 81. Find maximum product subarray
- 82. Trapping rain water problem
- 83. Find minimum in rotated sorted array
- 84. Search in rotated sorted array

Specialized Array Topics

String Arrays

- 85. How do you declare a String array?
- 86. How do you sort String arrays?
- 87. How do you search in String arrays?
- 88. How do you convert String array to List?

Object Arrays

- 89. How do you create arrays of custom objects?
- 90. How do you sort arrays of objects?
- 91. How do you compare arrays of objects?
- 92. What is shallow vs deep copy for object arrays?

Exception Handling with Arrays

- 93. What exceptions can occur with arrays?
- 94. How do you handle `ArrayIndexOutOfBoundsException`?
- 95. What is `NullPointerException` with arrays?
- 96. How do you validate array bounds?

Array Best Practices

- 97. When should you use arrays vs collections?
- 98. How do you optimize array performance?
- 99. What are common array programming mistakes?
- 100. How do you debug array-related issues?

Fundamental DSA Concepts

Basic Understanding

- 1. What are Data Structures?
- 2. Why do we need Data Structures?
- 3. What is an algorithm?
- 4. Explain time complexity and space complexity.
- 5. What is Big O notation?

6. What are linear and non-linear data structures?
7. What are the applications of data structures?

Time & Space Complexity

8. How do you calculate time complexity of an algorithm?
9. What is the difference between best, average, and worst case complexity?
10. Compare time complexities: $O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$
11. What is space complexity and how to optimize it?

Array-Based Questions

Basic Array Problems

12. Find the largest element in an array
13. Find the smallest element in an array
14. Reverse an array
15. Find sum of all elements in array
16. Find second largest element in array
17. Check if array is sorted
18. Remove duplicates from array

Intermediate Array Problems

19. Find missing number in array (1 to n)
20. Find duplicate elements in array
21. Rotate array by k positions
22. Merge two sorted arrays
23. Find pairs with given sum
24. Move all zeros to end
25. Find majority element

String-Based Questions

Basic String Problems

26. Reverse a string
27. Check if string is palindrome

- 28.Count vowels and consonants
- 29.Remove spaces from string
- 30.Check if two strings are anagrams
- 31.Find first non-repeating character
- 32.Count frequency of each character

Advanced String Problems

- 33.Find longest palindromic substring
- 34.Implement string matching algorithm
- 35.Check if strings are rotations of each other
- 36.Find all permutations of a string

Linked List Questions

Basic Linked List

- 37.What is a linked list and implement it
- 38.Difference between array and linked list
- 39.Insert node at beginning, middle, end
- 40.Delete node from linked list
- 41.Find length of linked list
- 42.Search element in linked list

Advanced Linked List

- 43.Reverse a linked list
- 44.Detect cycle in linked list
- 45.Find middle element of linked list
- 46.Merge two sorted linked lists
- 47.Remove duplicates from sorted linked list
- 48.Find intersection point of two linked lists

Stack and Queue Questions

Stack Operations

- 49.What is a stack and implement it

- 50. Check for balanced parentheses
- 51. Reverse a string using stack
- 52. Implement two stacks in one array
- 53. Next greater element using stack
- 54. Infix to postfix conversion

Queue Operations

- 55. What is a queue and implement it
- 56. Difference between stack and queue
- 57. Implement stack using queues
- 58. Implement queue using stacks
- 59. Circular queue implementation
- 60. Find first non-repeating character using queue

Tree-Based Questions

Binary Tree Basics

- 61. What is a binary tree?
- 62. Binary tree traversals (inorder, preorder, postorder)
- 63. Level order traversal
- 64. Find height/depth of binary tree
- 65. Count total nodes in binary tree
- 66. Find diameter of binary tree

Binary Search Tree

- 67. What is BST and its properties
- 68. Insert node in BST
- 69. Delete node from BST
- 70. Search element in BST
- 71. Find minimum and maximum in BST
- 72. Check if tree is valid BST
- 73. Find lowest common ancestor in BST

Sorting Algorithms

Basic Sorting

- 74. Implement bubble sort
- 75. Implement selection sort
- 76. Implement insertion sort
- 77. Compare time complexities of basic sorting algorithms

Advanced Sorting

- 78. Implement merge sort
- 79. Implement quick sort
- 80. Implement heap sort
- 81. When to use which sorting algorithm?

Searching Algorithms

Basic Searching

- 82. Linear search implementation
- 83. Binary search implementation
- 84. Binary search in rotated sorted array
- 85. Find first and last occurrence of element
- 86. Search in 2D matrix

Graph-Based Questions

Graph Basics

- 87. What is a graph? Types of graphs
- 88. Graph representation (adjacency list vs matrix)
- 89. Breadth-First Search (BFS)
- 90. Depth-First Search (DFS)
- 91. Detect cycle in graph
- 92. Find shortest path between nodes

Dynamic Programming Basics

DP Fundamentals

- 93.What is dynamic programming?
- 94.Fibonacci using DP
- 95.0/1 Knapsack problem
- 96.Longest common subsequence
- 97.What is memoization?

Hash-Based Questions

Hashing Concepts

- 98.What is hashing?
- 99.Handle collisions in hash tables
- 100.Find first repeating element using hashing
- 101.Count frequency using HashMap
- 102.Two sum problem using hashing

Advanced Problem-Solving

Pattern Recognition

- 103.Kadane's algorithm (maximum subarray sum)
- 104.Dutch flag problem (sort 0s, 1s, 2s)
- 105.Trapping rainwater problem
- 106.Sliding window technique
- 107.Two pointer technique

Recursion & Backtracking

- 108.Tower of Hanoi
- 109.Generate all subsets
- 110.N-Queens problem
- 111.Generate valid parentheses

MySQL Fundamentals

Basic Concepts

1. What is MySQL and how does it differ from other database management systems?
2. What are the advantages of using MySQL?
3. What is SQL (Structured Query Language)?
4. What is the difference between a database and a table in MySQL?
5. How do you create and delete a MySQL database?
6. What are the different data types available in MySQL?
7. Explain MySQL server and its components.

Data Types & Storage

8. What is the difference between CHAR and VARCHAR data types?
9. What are numeric data types in MySQL (INT, FLOAT, DECIMAL)?
10. What are date/time data types (DATE, TIME, DATETIME, TIMESTAMP)?
11. When should you use TEXT vs VARCHAR?
12. What is the AUTO_INCREMENT attribute and its purpose?

Basic SQL Operations

Database & Table Operations

13. How to create a table in MySQL? Write SQL statement for a "departments" table.
14. How do you alter a table structure in MySQL?
15. How do you drop a table in MySQL?
16. Write a SQL query to retrieve all records from a table named "employees".
17. How do you insert data into a MySQL table?
18. How do you update records in MySQL?
19. How do you delete records from MySQL?

Basic Queries

20. Write a SQL query to find the total number of employees.
21. Write a SQL query to select the top 5 highest-paid employees.
22. Write a SQL query to find all employees whose names start with 'A'.
23. How do you use the WHERE clause in MySQL?

- 24. What is the purpose of the GROUP BY clause?
- 25. Write a query using GROUP BY to summarize employee counts by department.

Keys & Relationships

Primary & Foreign Keys

- 26. What is a primary key and why is it used?
- 27. Explain the difference between a primary key and a foreign key.
- 28. What are foreign key constraints and how do they ensure data integrity?
- 29. Can a table have multiple primary keys?
- 30. What happens when you try to insert duplicate primary key values?

Database Relationships

- 31. What types of relationships are used in MySQL?
- 32. Explain one-to-one, one-to-many, and many-to-many relationships.
- 33. How do you implement relationships between tables?

Joins & Advanced Queries

Types of Joins

- 34. What are the different types of joins in MySQL?
- 35. Explain INNER JOIN with example.
- 36. Explain LEFT JOIN and when to use it.
- 37. What is the difference between RIGHT JOIN and LEFT JOIN?
- 38. What is CROSS JOIN and when is it used?
- 39. Write a SQL query to join "employees" and "departments" tables.

Set Operations

- 40. What is the difference between UNION and UNION ALL?
- 41. When should you use UNION vs UNION ALL?
- 42. How do you remove duplicates from query results?

Functions & Operations

Aggregate Functions

- 43. Write a SQL query to find the average salary of employees in each department.
- 44. What are COUNT, SUM, AVG, MIN, MAX functions?
- 45. How do you count the number of employees in each department?
- 46. What is the difference between COUNT(*) and COUNT(column_name)?

String Functions

- 47. What are common string functions in MySQL (CONCAT, SUBSTRING, LENGTH)?
- 48. How do you convert text to uppercase/lowercase in MySQL?
- 49. How do you trim spaces from strings?

Date Functions

- 50. Write a SQL query to retrieve employees hired in the last 30 days.
- 51. What are common date functions (NOW(), DATE(), YEAR(), MONTH())?
- 52. How do you format dates in MySQL?

Advanced Concepts

Views & Stored Procedures

- 53. What is a view in MySQL and how do you create one?
- 54. What are the advantages of using views?
- 55. What is a stored procedure?
- 56. Write a simple stored procedure that returns employee details by ID.
- 57. What is the difference between views and stored procedures?

Triggers

- 58. What are triggers in MySQL?
- 59. How many types of triggers are possible in MySQL?
- 60. Write a trigger that updates a "last_updated" timestamp.
- 61. When should you use triggers?

Indexing

- 62. What are indexes in MySQL and how do they improve performance?
- 63. What are the different types of indexes?
- 64. When should you create an index?
- 65. What are the disadvantages of using too many indexes?

Storage Engines

Engine Comparison

- 66. What are storage engines in MySQL?
- 67. What is the difference between MyISAM and InnoDB?
- 68. Which storage engine supports transactions?
- 69. When should you use InnoDB vs MyISAM?
- 70. What are transaction storage engines?

Transactions & ACID Properties

Transaction Concepts

- 71. What are transactions in MySQL?
- 72. Explain ACID properties in databases.
- 73. What are the transaction isolation levels?
- 74. How does MySQL handle concurrency?
- 75. What is Multi-Version Concurrency Control (MVCC)?

Performance & Optimization

Query Optimization

- 76. How do you inspect query execution plan in MySQL?
- 77. What is the EXPLAIN command used for?
- 78. How do you optimize slow queries?
- 79. What are temporary tables and how are they used?
- 80. What factors affect MySQL performance?

Database Design

- 81. What is normalization and its importance?
- 82. What are the normal forms (1NF, 2NF, 3NF)?
- 83. What are best practices for schema design?
- 84. When should you denormalize a database?
- 85. How do you handle schema migrations in production?

Backup & Security

Data Protection

- 86. How do you backup a MySQL database?
- 87. What is the difference between logical and physical backup?
- 88. How do you restore a MySQL database?
- 89. What are MySQL user privileges?
- 90. How do you secure a MySQL installation?

Advanced Topics

Scaling & Architecture

- 91. What is scaling in MySQL?
- 92. What is sharding in databases?
- 93. Explain MySQL's logical architecture.
- 94. What is the difference between OLTP and OLAP?
- 95. How do you handle high-traffic databases?

Coding Questions

Query Writing Practice

- 96. Write a query to find the second highest salary without using LIMIT.
- 97. Write a query to find employees with duplicate names.
- 98. Write a query to calculate running totals.
- 99. Write a query to find employees earning more than their department average.
- 100. Write a query to find the nth highest salary.

Spring Boot Fundamentals

Basic Understanding

1. What is Spring Boot and what are its benefits?
2. What makes Spring Boot superior to traditional Spring Framework?
3. What are the advantages of using Spring Boot?
4. How does Spring Boot simplify Java application development?
5. What are the key features of Spring Boot?
6. When should you use Spring Boot?
7. What companies use Spring Boot in production?

Core Components

8. What are the four key components of Spring Boot?
9. What is Spring Boot Auto-Configuration?
10. What is the purpose of `@EnableAutoConfiguration` annotation?
11. What is Spring Boot CLI?
12. What are Spring Boot Starter POMs?
13. What is Spring Boot Actuator?

Spring Boot Starters

Starter Dependencies

14. What is a Spring Boot Starter?
15. What are the most commonly used Spring Boot starters?
16. What is the difference between starter POMs and regular dependencies?
17. What is Spring Boot Starter Parent?
18. How do Spring Boot starters manage dependencies?
19. What is `spring-boot-starter-web`?
20. What is `spring-boot-starter-data-jpa`?

Annotations & Configuration

Key Annotations

21. What is `@SpringBootApplication` annotation?

- 22. What annotations are combined in `@SpringBootApplication`?
- 23. What is the difference between `@Component`, `@Service`, `@Repository`, and `@Controller`?
- 24. What is `@RestController` annotation?
- 25. What is `@RequestMapping` vs `@GetMapping`?
- 26. What is `@Value` annotation?
- 27. What is `@ConfigurationProperties` annotation?

Configuration Files

- 28. What is the purpose of `application.properties` file?
- 29. What is the difference between `application.properties` and `application.yml`?
- 30. How does Spring Boot handle externalized configuration?
- 31. What are Spring Boot Profiles?
- 32. How do you activate a specific profile?

REST API Development

Creating REST Services

- 33. How do you create a RESTful web service using Spring Boot?
- 34. What is the difference between `@Controller` and `@RestController`?
- 35. How do you handle HTTP methods (GET, POST, PUT, DELETE)?
- 36. What is `ResponseEntity` and when to use it?
- 37. How do you handle request parameters and path variables?
- 38. How do you validate request data in Spring Boot?

Exception Handling

- 39. How do you handle exceptions in Spring Boot?
- 40. What is `@ControllerAdvice`?
- 41. What is `@ExceptionHandler`?
- 42. How do you create custom error responses?
- 43. What is `@ResponseStatus` annotation?

Data Access & JPA

Spring Data JPA

- 44. What is Spring Data JPA?
- 45. How do you configure Spring Boot to use a database?
- 46. What is @Entity annotation?
- 47. What is @Repository annotation?
- 48. How do you define custom queries in Spring Data JPA?
- 49. What is the difference between JpaRepository and CrudRepository?
- 50. What is @Transactional annotation in Spring Boot?

Database Configuration

- 51. How do you configure multiple databases in Spring Boot?
- 52. What embedded databases does Spring Boot support?
- 53. How do you configure H2 database in Spring Boot?
- 54. How do you use Spring Boot with MySQL/PostgreSQL?

Testing

Testing Framework

- 55. How do you write unit tests in Spring Boot?
- 56. What is @SpringBootTest annotation?
- 57. What is @MockBean annotation?
- 58. What is @WebMvcTest?
- 59. What is @DataJpaTest?
- 60. How do you test REST controllers?
- 61. What is TestRestTemplate?

Security

Spring Security Integration

- 62. How do you secure a Spring Boot application?
- 63. What is Spring Boot Security Starter?
- 64. How do you implement authentication in Spring Boot?

- 65. How do you enable HTTPS in Spring Boot?
- 66. What is @PreAuthorize annotation?
- 67. How do you implement JWT authentication?

Embedded Servers

Server Configuration

- 68. What is Spring Boot's embedded server?
- 69. What embedded servers does Spring Boot support?
- 70. How do you change the default port in Spring Boot?
- 71. How do you configure Tomcat in Spring Boot?
- 72. Can you deploy Spring Boot as WAR file?
- 73. What are the advantages of embedded servers?

Monitoring & Management

Actuator Features

- 74. What is Spring Boot Actuator?
- 75. What endpoints does Actuator provide?
- 76. How do you implement health checks?
- 77. How do you monitor Spring Boot applications?
- 78. What is Micrometer in Spring Boot?
- 79. How do you secure Actuator endpoints?

Advanced Features

Caching & Async

- 80. How do you implement caching in Spring Boot?
- 81. What is @EnableCaching annotation?
- 82. What is @CacheEvict annotation?
- 83. What is @EnableAsync annotation?
- 84. What is @Scheduled annotation?
- 85. How do you implement asynchronous processing?

DevTools & Hot Reload

- 86. What is Spring Boot DevTools?
- 87. How does hot reloading work in Spring Boot?
- 88. What features does DevTools provide?
- 89. How do you disable DevTools in production?

Microservices with Spring Boot

Microservices Architecture

- 90. What Spring Boot features help develop microservices?
- 91. What is Spring Cloud and how does it relate to Spring Boot?
- 92. What is an API Gateway in microservices?
- 93. How do you implement service discovery?
- 94. What is Circuit Breaker pattern?
- 95. What is Spring Cloud Config?

Deployment & Production

Production Readiness

- 96. How do you deploy Spring Boot applications?
- 97. What is Spring Boot Maven/Gradle plugin?
- 98. How do you create Docker images for Spring Boot apps?
- 99. What are production-ready features in Spring Boot?
- 100. How do you optimize Spring Boot applications for production?