**Core Java**

1. What the difference between String, StringBuilder and StringBuffer
2. What is immutability. How to create immutable class.
3. Why string is immutable.
4. Internal implementation of HashMap
5. Explain SOLID principle.
6. What is Singleton class. How to create singleton class.
7. What are OOPS concepts.
8. How to handle exceptions in Java. How create customize exception class.
9. Explain Thread pool?
10. Difference of wait and sleep.
11. Explain Garbage Collector?
12. Explain JVM architecture.
13. Explain and create deadlock.
14. What is fail fast and fail safe.
15. What are class loader types.
16. Synchronization (Object level lock and class level lock)
17. Difference b/w ArrayList and Linked List
18. Difference b/w HashMap vs Hastable
19. If we want to store employee object as key what to do?
20. Given program need to check what’s the time complexity.
21. Difference b/w for loop and stream API, when to use and cross questions, variable assignment inside Stream API clarification,
22. Stream Filter based scenario questions.
23. Array List<String> arr = new ArrayList<>(), Why we aren't specifying type at right hand side.
24. How to find Max value in the array list? Discussion about implementation, time complexity
25. How to find most occurrence in the list.
26. int a; Integer b; System.out.println(b) Default values, will compile the program. What is Integer?
27. What kind of values will accept array list? what will happen if Array list specified with primitive type.
28. What is Restful service and clarification questions on RESP Api.
29. Difference b/w post and put and cross questions like when, why, Whynot
30. Rewrite a Java program that employs threads to print both odd and even numbers from 1 to 20.
31. Develop a method to determine the maximum and minimum values in an ArrayList.
32. Compare and contrast deep copy and shallow copy operations.
33. Explain the inner workings of the garbage collection process in Java.
34. Differentiate between ClassNotFoundException and NoClassDefFoundError in Java.
35. Determine when to use an abstract class or interface in your Java code.
36. Analyze the pros and cons of using loops versus the Stream API in Java.
37. Discover the most frequently occurring element in a list or collection.
38. Explore the uses and benefits of the ExecutorService in Java.
39. Distinguish between the purposes and applications of the "wait" and "sleep" methods in Java.
40. Examine the ConcurrentHashMap data structure in Java for concurrent data manipulation.
41. Define what a functional interface is and discuss its significance in Java.
42. Elaborate on the purpose and importance of the Optional class in Java, including when and why it should be used

**Sample Programs/Coding Questions**

1. Write program to find second max in given array (using loops) ?
2. Given array list find pairs of targeted sum
3. Reverse a string.
4. 3rd largest from an array
5. Code to print Fibonacci series for given number.
6. Remove duplicate from an array
7. Program to check loops in linked list
8. Occurrence of vowels in a string
9. Sort an array.
10. Find a loop in a linked list
11. reverse the words in each string. like,"This is SAP" -> "SAP is This" without using other loop and output variable.
12. Reverse the provided LinkedList like 1-2-3-4-null to 4-3-2-1-null
13. How to find most occurrence in the list?
14. Find second oldest employee from Employee table.
15. Find number of occurrences of each word in each paragraph.
16. Convert number to words. For Ex: input = 1234, output = one thousand two hundred thirty-four.

**Java 8+**

1. Uses of Streams
2. What are Java 8 features.
3. Difference b/w map and flatMap.

**Rest API**

1. Restful web services principles
2. Put and post difference.
3. Explain about statelessness.
4. Status codes : 201,204, 400, 401, 403
5. What is Restful service and clarification questions on RESP Api.
6. Application Gateway features and why we use it.

**Spring boot**

1. Advantages of Spring framework
2. Which one do you prefer? Spring or Spring Boot? Why?
3. Explain how Spring Boot handles dependencies (How Spring Boot works)
4. How do you handle Exceptions in your Spring Boot Application?
5. What are the annotations you've used in Spring Boot?

**Microservice**

1. What is monolithic and micro service Architecture
2. In Microservices how do we handle a breakdown of service in any of the 2 microservice Service1, Service2 ?
3. Microservice advantage and disadvantage
4. Microservice principles.

**Database**

1. What is a Left Outer Join?
2. Explain inner join
3. What's the difference between Relational and nosql

**Automation Testing(Only for QA roles):**

1. Explain POM
2. Scenario for Scroll and find element
3. Array and List to find 4th greatest numbers in given list and writing java code
4. Test Plan creation and Manual Execution
5. Difficulties and overcome with that solution in automation
6. Code to reverse a string and explanation of code
7. Inheritance questions
8. exception Handling questions(Try Except Finally)
9. Identifiers in selenium
10. Which is the most preferred Indentifier in selenium
11. difference between findElement and findElements
12. write expression for fetching first 10 row of table which has several columns using xpath
13. write query to join 2 tables
14. write query to fetch the data from table from 1st January 2023 to 31st March 2023
15. Different types of exception handling in selenium
16. what is bug and defect
17. what is test approach
18. what is test plan
19. Several questions from software testing

Note: They need Tester with Java core Expert and Junit and prepare more on Java basics then selenium and Katalon