CGI – L1: Java Full Stack Developer

1. **What is the Difference b/w HTML and HTML5?**
2. **What are the limitations of CSS?**
3. **What is Box model and what are the elements we need to add?**
4. **Print num of occurrences of each character in a given string.**
5. **What is faltMap in java 8?**
6. **features of Java8?**
7. **Print 2nd character of each string in a given list of strings.**
8. **Functional Interface**
9. **What is routing in angular?**
10. **How to create service in Angular?**
11. **How to create component in Angular?**
12. **Have u worked on Design Patterns**
13. **What is Singleton Design Pattern and any example?**
14. **What is Factory Design Pattern**
15. **Which package we will use to connect with APIs in angular?**
16. **If One MS fails how another MS will know, how u will handle this.**
17. **Communication b/w MS**
18. **Do u have hands on angular exp from scratch**
19. **What are the Junit annotations u have used?**
20. **Diff b/w Inject Mock vs Normal Mock**
21. **What is dependency injection and why we need to use?**
22. **Why Sping is famous rarther than other frameworks.**
23. **What is Criteria builder**
24. **What is API gateway**

Altimetrik- L1: Java Full Stack Developer

1. **How do you merge two unsorted arrays into single sorted array without duplicate using Streams?**
2. **"Given a String, find the first non-repeated character in it using Stream functions?**
3. **What is Service Discovery**
4. **What is Circuit Breaker**
5. **What u do if any MS fails?**
6. **Which are the SpringBoot annotations used in u r project**
7. **Difference b/w @Primary and @Qualifier**
8. **Write a query to fetch employee name who is earning more than his manager.**

Citius Tech -- L1: Java Full Stack Developer

1. **What is the o/p of below code and why?**

```
class A
{
void m1() throws ArrayIndexOutOfBoundsException
{
System.out.println("In m1 A");
}
}
class B extends A
{
void m1() throws IndexOutOfBoundsException
{
System.out.println("In m1 B");
}
}
public class Test {
public static void main(String[] args) {
A a=new B();
a.m1();
}
}
```

2. **What is the o/p of below code and why?**

```
class P
{
void m1()
{
System.out.println("In m1 P");
}
}
class C extends P
{
void m1()
{
System.out.println("In m1 C");
}
void c2()
{
System.out.println("In c2 C");
}
}
public class Test {
public static void main(String[] args) {
P p=new C();
p.c2();
}
}
```

3. **What is the o/p of below code and why?**

```
Class X {
void m1 (Object o) {
System.out.println("from Object ");
}

void m1 (String s) {
System.out.println("from String ");
}

class Y{
public static void main (String args[]) {

X x = new X();
x.m1(null);
}
}
```

4. What is the o/p of below code and why?

```
public class MainClass
{
   void method(int ... a)
   {
      System.out.println(1);
   }
   void method(int[] a)
   {
      System.out.println(2);
   }
}
```

5. Sort the below given code?

```
Set<Integer> set =  Collections.unmodifiableSet(new HashSet<Integer>
                     (Arrays.asList({ 51, 62, 72, 80, 11, 21, 3, 4, 33 })));
```

6. Print num of occurrences of each word in a given list of string.

```
List<String> list = Arrays.asList("abc", "abc", "frg", "lij");
```

7. How to write Junit test case for static methods?
8. What is composite key?
9. What is left join and right join?
10. What is cross join?
11. What is mean by stateless in MS.

Citius Tech – L2: Java Full Stack Developer

1. How the angular application works.
2. Print num of occurrences of each word in a given list of string without using map as o/p.

```
String str = "4,8,12,1,8,4,8,2";
```

3. Any AWS Exp?

Capgemini – L2: Java Developer

1. **Create Sample springboot main class**
2. **Create a Thread using Runnable**

Capgemini – L1: Java Developer:

1. **Two arrays given, need to merge, sort then get the middle value**
2. **Two anagram strings given, how u will check**
3. **Java 8 features**
4. **what is new implementation in java8 respective to Hashmap**
5. **what is Set in java.**
6. **What is TreeSet, how internally implemented.**
7. **Transient key word**
8. **What is Try with resources.**
9. **Comparable and Comparator**

Impetus – L1: Java Developer

1. **Types of ExecutorService in Java/J2EE**
2. **How to configure and setup multiple DBs (Insertion in one DB, updates in another DB)**
3. **What is POD.**
4. **Supplier**
5. **Java 8 Interface features**
6. **Functional Interface**
7. **List of functional interfaces before java 8, after java 8**
8. **Predicate**
9. **Consumer**
10. **Why Abstract class**
11. **Why Interface**
12. **Abstract vs Interface**
13. **When to use Interface, when to use Abstract.**
14. **Polymorphism**
15. **Method overloading**
16. **Method Overriding**
17. **Circuit breaker**
18. **Java Messaging Brokers**
19. **How M/S communicate each other**
20. **Why Feign Client**
21. **How to maintain good user exp using MS**
22. **How to connect with database**
23. **What is container and why we need that?**

Impetus – L2: Java Developer

1. **Microservice Architecture**
2. **How to communication happens b/w MS**

3. **If any MS fails how other MS will know**
4. **Sealed classes**
5. **Atomic variables**
6. **Servlet Filter**

Tavant Technologies – L1: Java Full Stack Developer

1. **Java 8 Features**
2. **Why Streams**
3. **What is Functional Interface and how its helps to write Lamda expressions**
4. **Use of Optional class**
5. **Method References**
6. **How MS communicate with each other**
7. **REST API implementation**
8. **Which Design pattern used in your project (MS).**
9. **What is HQL and Why?**
10. **What is mean by Collections.**
11. **How u will handle Exception handling in java.**
12. **OOPs concept.**
13. **Predicate and Supplier**
14. **What is mean by directives in angular.**
15. **What is mean by Components in angular.**
16. **What is data binding in angular.**
17. **What is String interpolation**
18. **What is mean by Template.**
19. **What is mean by Modules.**

EY – L1: Java Developer

1. **JDK, JRE and JVM**
2. **Write singleton class.**
3. **What are the memory areas in Java.**
4. **When we get Stack overflow error.**
5. **How to create Custom Checked Exception?**
6. **How to know whether the class is Checked exception class or unchecked?**
7. **4 pillars of OOPs and explain them?**
8. **OOPs scenario-based questions.**
9. **Diff b/w Throw and Throws.**
10. **Why Default method needed in Interfaces**
11. **SOLID principles**
12. **How can we achieve Open-Closed Principle**
13. **Print the values which starts with 1 in the given below list.**

    list (10,15,8,49,25,98,98,32,15)

14. **Print 3rd Highest salary from the table**
15. **Why Optional class**
16. **ArrayList vs LinkedList**
17. **ConcurrentModification exception and how to overcome that**
18. **HashMap Internal structure**

19. **What is the o/p below code?**

    ```
    HashMap<String, Integer> map = new HashMap<>();
    map.put(“a”, 10);
    map.put(“a”, 20);
    System.out.println(map);
    ```

20. **Equals() method**
21. **we have millions of data in DB which collection u will use to search/get the values**
22. **how to overcome ConcurrentModification without using concurrent hashmap and synchronized**
23. **How Threads communicate with eachother**
24. **ExecutorService framework**
25. **Race condition**
26. **@SpringBootApplication uses**
27. **Spring Bean methods**
28. **@Bean annotation**
29. **@Transactional annotation**
30. **@RestController vs @Controller**
31. **How to keep separate properties files for each env and how to read that**
32. **How MS communicate with each other**
33. **If MS fails to respond what u do in that case**
34. **Self Join Query**
35. **What is Hibernate**
36. **First level and 2nd level cache**

Capgemini – L1: Java Developer

1. **Java 8 Features**
2. **What is streams API**
3. **What are the intermediate operations in Stream**
4. **What are the terminal operations in Streams.**
5. **Functional Interface**
6. **What are the new Functional interfcaes introduced in Java8**
7. **About Supplier**
8. **About Consumer.**
9. **About Function**
10. **About Predicate**

CGI – L1: Java Developer

1. **Print the name of the employee whose salary is >20000 using streams api.**
2. **Print num of occurrences of each word in a given string.**

    String str = “big black bug bit a big black dog on his big black nose”

3. **What is HashSet in Java**
4. **@Configuration annotation**
5. **What is use of @Autowired annotation**

6. **@Primary vs @Qualifier**
7. **Spring Profiles**
8. **How MS communicate each other**
9. **POST vs PUT mapping in MS**

LTI Mindtree – L1: Java Developer

1. **Current Company and work**
2. **HashMap Internal structure**
3. **What is Method Overriding and Exceptions will handle**
4. **Can we Override static method?**
5. **Bean methods**
6. **Dependency Injection and types of Injections**
7. **Is the below code correct ?**

   ```
   @FunctionalInterface
   interface Test{
   void say(String msg);
   // Object class methods.
   int hashCode();
   }
   ```

8. **Print the emp name whose salary is >10000 in sorted order without any duplicates.**
9. **What is Circuit breaker.**
10. **Checked and Unchecked exceptions.**
11. **Query related to SQL sub query and Joins.**

Altimetrik – L2 Round – Java Full Stack Developer

1. **What is decorators in angular**
2. **What is mean by Lazy loading and how it works in angular**
3. **How MS communicate each other**
4. **What is template**
5. **What is component**
6. **What is mean by caching mechanism**
7. **What is Functional Interface**
8. **Why only one abstract method in Functional Interface**
9. **Why Default methods in Interfaces**
10. **SOLID principles**
11. **about @Transient annotation**
12. **How to do async operation in MS (@Async)**
13. **What is mean by entity class**
14. **What are the annotations used in JPA**
15. **How to connect with DB using JPA**
16. **How to handle token based authentication in angular**
17. **How you will get the data from API to angular**
18. **Observable and Promise**
19. **Observer vs Observable vs Observables**
20. **How to restrict user to access some pages/routes using Spring Security**
21. **What is mean by Index in SQL**
22. **What is Primary Key**

23. **Have u ever worked on Deployement**
24. **How to check code coverage and what is the min % u r maintaning in u r project**
25. **What is SpringBoot and why**

Altimetrik – EL Round – Java Full Stack Developer

1. **MicroService Architechture**
2. **Circuit Breaker**
3. **Messaging Qs**
4. **API Gateway and why**
5. **What u do if u r API gateway is down**
6. **Autoscaling**
7. **Diff b/w Immutable, Final**
8. **How you will handle Exceptions**
9. **Hashmap Internal Structure**
10. **What is Functional Interface and Why**
11. **Interceptors in angular and why**
12. **what is secured routing?**
13. **AuthGuard**

Cognizant – L1 Round : Java Developer

1. **Transient keyword**
2. **Synchronization**
3. **SingleThreadPoolExecutor**
4. **What is Microservice**
5. **What is SpringBoot**
6. **What is HashMap**
7. **How to ovirride HashCode**
8. **convert below string array to list**
   String[] str = { "hello", "world", "how", "are", "you", "I", "am", "fine" };
9. **Find min length of string in above array using java8**
10. **check "A11" is exist in this arrays or not using java8**
    String[] str1 = new String[] { "A01", "A02", "A01" };
    String[] str2 = new String[] { "A11", "A12", "A13" };
11. **What is Optional class and Why**
12. **What are the methods present in Optional class**
13. **what is Optional.of**
14. **what is Optional.ofNullable()**
15. **What is parallel streams**
16. **what is streams and why**
17. **What is @PathVariable**
18. **what is @RequestParam**
19. **what is mean by consumer and producer in MS**
20. **what is response entity in MS**
21. **what is Http codes and what are they**
22. **Intercepter**
23. **How can we connect multiple DBs in our project**
24. **what is mean by RequestMapping**

25. **How to handle Exceptions in MS**
26. **/employee/{id}?name=fname Explain this**
27. **When two controllers having same RequestMapping url then what will happen**
28. **Map<Integer, Sample> -> what it will print when u get the hasmap**
29. **how can u make any field to b mandatory in MS (required = true)**

CGI -- L1 Round - Java Full Stack Developer

1. **Self Intro**
2. **Abstract and Interfaces why we need those, which one to use when**
3. **HashMap and its internal instructure**
4. **ConcurrentHashMap why we need that**
5. **ConcurrentHashMap vs SynchronizedMap**
6. **How ConcurrentHashMap internally works**
7. **Fail-fast vs Fail-safe**
8. **What are the memories available in Java**
9. **Where String objects will store**
10. **What is the o/p of below code**

```
public class StringBasics {

public static void main(String[] args) {

String s1 = "hello";

String s2 = new String("hello");

String s3 = "hello";

if (s1 == s2) {

System.out.println("s1 and s2 equal");

} else {

System.out.println("s1 and s2 not equal"); --

}

if (s1 == s3) {

System.out.println("s1 and s3 equal");  --

} else {

System.out.println("s1 and s3 not equal");

}

}

}
```

11. **How many objects will b created by below code**

```
String s1 = new String("javaguides"); Heap

String s2 = "javaguides"; SCP
```

12. **Exception Hirarchy in Java**
13. **can we have try without catch**
14. **what is mean by try with resources**
15. **What is mean by sreams**
16. **What is Optional class and Why**
17. **find the num of occurnces of each letter in below String using java8**

    String str = "hyujikoyhfrtuj";

18. **Employee object -> sort the by employee name using Java8 and explain**
19. **Employee object -> group by gender using Java8 and explain**
20. **Spring vs SpringBoot**
21. **JDBC vs JPA**
22. **How to implement JPA in application**
23. **What is Microservice and its Architecture**
24. **What are the annotations u have used**
25. **@SpringBootApplication means**
26. **What is Spring Accuator and how to change the port num for this**
27. **What Http codes u have used**
28. **What are Http methods u have used**
29. **POST vs PUT vs Patch**
30. **Any Design pattern u worked on**
31. **Java8 features**
32. **Which annotation used to define primary key**
33. **Composite key annotation**
34. **How to connect with DB**
35. **What are services in angular**
36. **Decorators in angular**
37. **Pipes in angular, custom pipe**
38. **How to get response from backend to UI**
39. **Observables in angular**
40. **Routing in angular**
41. **package structure**
42. **How to deploy angular code**
43. **What is docker and dockerfile**
44. **What is ClassNotFoundException**
45. **Diff b/w ClassNotFoundException and NoClassDefException**

Virtusa – L1 : Java Full Stack Developer

1. **java8 features**
2. **What is functional interfaces**
3. **Is it mandatory to have @FunctionalInterface annotation**
4. **Diff b/w Interface and Class**
5. **why multiple inheritance is not possible in Java**
6. **What is polymorphism?**
7. **What is Encapsulation?**
8. **What is SCP?**
9. **String buffer vs String Builder**
10. **ArrayList vs Linked List**
11. **Hasmap vs Hashtable**

12. **Iterator vs List Iterator**
13. **What HashSet**
14. **What is Streams and why**
15. **Streams vs Collections**
16. **What is MicroService**
17. **PUT vs Patch**
18. **What is Generics**
19. **What is Wrapper class**
20. **HashMap internal structure**
    List = 10,15,8,49,25,98,32,10;
21. **Find the Even numbers from the list using java8**
22. **Find the Duplicate values from the given list using java8**
23. **find the count using java8**
24. **find the values which starts with "1" using java8**
25. **What is groupingBy and explain with an example**
26. **map vs filter in streams**
27. **What is predicate**
28. **What is Supplier**
29. **What is Consumer**
30. **what is Function**
31. **Checked vs un checked exceptions with example**
32. **Throw vs Throws**
33. **Controller vs RestController**
34. **Dependency Injection**
35. **What is pom.xml file**
36. **@Qualifier annotation**
37. **How to connect with Database from SpringBoot**
38. **@Repository, @Service, @Component**
39. **@Bean means when to use that**
40. **What is ORM tool**
41. **How many ORM tools u know**
42. **What is lazy loading**
43. **CRUDRepository vs JPARepository**
44. **What are the methods available in CRUDRepository**
45. **What are the methods available in JPARepository**
46. **How to fetch the data from DB**
47. **What is mean by lazy Loading**
48. **what is Transactional management**
49. **how many entity mappings are there and explain**
50. **Session factory**
51. **Write a native query to fetch EmpName, EmpDesignation using EmpID**
52. **Super Constructor**
53. **What is frontController**
54. **AssertEquals vs AssertNotEquals**
55. **How to create Hibernate Specific Immutable class**
56. **What is the use of @Query annotation**
57. **First level cache and 2nd level cahche**
58. **Write code login to read the Json/XML file**
59. **What is the entry point for Angular app**
60. **Can we have multiple @NgModule**
61. **Local Storage and Session Storage**

62. **What are components**
63. **What are Services in angular**
64. **Box model in CSS**
65. **Can we have same class for 2 diff divs**
66. **Singleton service**
67. **How Components communicate each other**
68. **Angular hooks**
69. **NgOnChnages ?**
70. **Constructor vs ngOnInit**
71. **Dependency Injection in angular**
72. **What is Pipe, give some examples**
73. **How to create custom pipe**
74. **Write piece of code for Custom pipe**
75. **What is mean by Incremental DOM**
76. **Hi String in JS**
77. **Regex Patterns in angular**
78. **How to give 10 spaces in paragraph**
79. **Selectors**
80. **Responsive Destination**
81. **Diff b/w padding and margin**

Wipro -- L1: Java FullStack Developer

1. **why Optional class in Java**
2. **What is Functional Interface in Java**
3. **Which all are Functional Interfaces available in Java**
4. **Predicate vs Consumer**
5. **Reverse a String str = "Java" without any predeifned methods**
6. **Print the distinct charactors in a given string**
7. **What is Serializantion and Why**
8. **What is the use of transient keyword**
9. **What is Atomic Integer in Java**
10. **Yield vs Join in multithreading**
11. **How to create Immutable class**
12. **how to add Date object in immutable class**
13. **Directives in angular**
14. **What is two way data binding**
15. **What is String Interpollation**
16. **What is Pipe**
17. **How to connect with backend API**
18. **Promise vs Observables**
19. **What is routing in angular**
20. **from which package router will be**
21. **from which package HttpClient will be**
22. **Template driven vs Reactive forms**
23. **How many annotations joined in @SpringBootAnnotation**
24. **How to add Eureka server**
25. **Circuit Breaker**
26. **API Gateway**
27. **Communication b/w MicroServices**
28. **How Kafka produces the topic**
29. **@RestController vs @Controller**

30. **What id Depedency Injection in SpringBoot**
31. **What is Autowiring**
32. **What is @Qualifier**
33. **what is @Primary**
34. **PUT vs PATCH mapping**
35. **Spring profiles**
36. **What is Saga Pattren**
37. **Truncate vs Delete**
38. **Left join vs Right Join**
39. **Why @ComponentScan**
40. **Bean Scope**
41. **What is application context**
42. **Synchronization in Java**
43. **How to create singleton class**
44. **How to break singleton pattren**
45. **How to fix the singleton pattren**
46. **What is SpringBoot and Why**
47. **SpringBoot acctuators**
48. **@Component vs @Bean**
49. **Checked and un-checked exceptions**
50. **How to create custom exceptions**
51. **Loggers in SpringBoot**
52. **How to achive SpringSecurity and explain**
53. **What is @Injectable**
54. **Concurrent hashmap**
55. **HashSet internal implementation**
56. **Monoletic vs MS**
57. **Use of DevTools in SpringBoot**

Boeing -- L1: Java FullStack developer

1. **Static class in Java**
2. **Dynamic Method dispacting**
3. **Write a pgm to acheive Runtime polymorphism**
4. **Write a pgm to check string2 is substring of string1 or not**
5. **Write a pgm to count the num of occ of each letter in a given string**
6. **What is deadlock and write a pgm for dead lock**
7. **SingleTon design pattren**
8. **What is Serialization and why**
9. **Why we need DI in springboot, what will happen without that**
10. **What is Spring IOC**
11. **Advantages of MS**
12. **Spring Security (Authorization and Authenticaction)**
13. **What is the use of Syncronized**
14. **Sort by value in a given map where String is key, Integer is value**
15. **File having 10L lines of text, read and store it in Hasmap**
16. **What is spring data jpa**
17. **Profiles in Springboot**
18. **Drawbacks by using MS**

FedEx -- Java FS

1. **Responsive design in angular**
2. **What is SPA**
3. **How to acheive secure routing**
4. **Promise vs Observables**
5. **What is the use of @ngOnInit, @ngOnChanges, @ngOnDestroy**
6. **Java 8 features u have used**
7. **What are Intermediate and Terminal operations in Streams**
8. **What is the use of Reduce**
9. **Flatmap vs map and example**
10. **What is the use of Optional class**
11. **Why default methods introduced in interfaces**
12. **Throw vs Throws**
13. **Why Throws required**
14. **HashMap vs Concurrent hasmap**
15. **Internal implementation of ConcurrentHashMap**
16. **Have u worked on any design pattrens**
17. **How to break Singleton**
18. **How to prevent Singleton breaking**
19. **Fault tolerance in MS**
20. **Constrains in mySQL**
21. **PUT vs PATCH**
22. **How to implement Global exceptions**
23. **How to trace the logs**
24. **What is mocking**
25. **What is TDD approch**
26. **What is mean by Immutable class**
27. **How to create custom immutable class**
28. **How can we break Immutable class**
29. **How can we fix once Immutable class is broken**

IBS Software -- L1 Java Full Stack Developer

1. **ReqParam vs Path variable**
2. **Global exception**
3. **How to create second level cache**
4. **Session vs Sessionfactory**
5. **File interfaces in hibernate**
6. **Criteria in Hibernate**
7. **HQL vs SQL**
8. **NgModule**
9. **NgGroup**
10. **Diff ways to submit a form**
11. **Validation in Backend**
12. **How to implement fiegn client**
13. **Group By**
14. **to create EmployeeReg form how many tables need to create**
15. **Agreegation**
16. **Delete vs Truncate**
17. **How to write delete query**
18. **Index in SQL**
19. **Java 8 features**

20. **Lamda expressions**
21. **Predicate**
22. **Map vs FlatMap**

String str = "JAVA";

23. **write a pgm to print non repeated values (O/P: J,V)**
24. **Spring MVC vs SpringBoot**
25. **@Qualifier**
26. **SpringBoot Acctuators**
27. **What are the spring boot starter dependencies**
28. **What is use of JPA**
29. **First level cache vs second level cache**
30. **Dialect in hibernate**
31. **How to create REST template**
32. **PUT vs POST**

EY -- L1 Java FS

1. **SOLID principles in JAVA**
2. **Design Pattrens**
3. **Write code to acheive SingleTon Pattren**
4. **What is Factory Pattren**
5. **What is Exception handling in Java and its hireracy**
6. **How to create Custom UnChecked Exception**
7. **What will happen if we extend a class with Exception**
8. **Try with resources in Java**
9. **What is Polymorphism in Java**
10. **Method Overloading vs Method Overriding**
11. **Write a pgm to find length of a String without using length, size**
12. **Find the num of occurances of each charactor in a string**
13. **You have 3 threads, once threads complete their execution we need to perform another operation, how will u write code for this**
14. **How amany ways to create a Threads**
15. **Thread vs Runnable**
16. **JavaScript vs TypeScript**
17. **Event Loop**
18. **Call backs**
19. **Write a query to find 5th Highest sal from Employee table**
20. **HashMap internal structure**
21. **HashCode, equals method contrast**
22. **What will happen if we wont implement hashcode method but implement equals method**
23. **Fail-fast vs Fail-safe**
24. **Iterator in Java**
25. **How to sort map using values in that**
26. **How internally lambda expressions work**
27. **What is Predicate**
28. **Memory areas of String**
29. **String Object creation Qns**
30. **SCP, Heap memory**
31. **Which are the annotations u have used in springboot**
32. **What is mean by @SpringBootApplication annotations**
33. **@Autowured vs @Bean**

34. @Component vs @Repository vs @Service
35. What is JPA
36. what is the use of having JPARepository
37. How to write native queries
38. drawbacks of native queries
39. JPQL vs Native query
40. How can we connect different kind of DBs in spingBoot app
41. Comparator vs Comparable
42. write pgm for compareTo, compare methods

EY: L2 - JAVA FS

1. **Hogging Threads in Java**
2. **.equals vs ==**
3. **What is mean by Pojo class in java**
4. **What is Observer pattern**
5. **What do u know about SAGA pattren**
6. **Why to useKafka**
7. **Producer and Consumer problem in Java**
8. **Write a SQL query to highest salary from employee from each dep**
9. **Write a query to fetch avg salary of an employee whose dep is HR**
10. **Rotate an array by right using case steps**
11. **HashMap Internal structure**
12. **Write a pgm to print length of each string in a list of strings**
13. **What is inexing in SQL**
14. **Joins in SQL**
15. **How many ways we can create strings**
16. **How the string value store in memory**

EY - L3(UI) Client round

1. **Observables in angular**
2. **sort the list using type script**
3. **How to write CSS to bring the content to the center of the page**
4. **How can we acheive live data to show each user who loggedin**
5. **How can we have a url to be readonly**
6. **Explain your role in angular side**
7. **How components communicate each other**
8. **How non-related components communicate each other**
9. **BehaviourSubject vs Subject**
10. **Have u worked on Ngrx**

HCL -- L1 Java Developer

1. **Java8 features**
2. **Streams vs Collections**
3. **Terminal and Intermediate operations in Streams with example**
4. **What is functional interface**
5. **Write custom functional interface**
6. **what is meta space and how it is diff from prem gen**
7. **What is SCP**
8. **How many memory allocations are there and why**
9. **Checked vs unChecked Exceptions with example**
10. **How to create custom Checked/unChecked exceptions**

11. Write a pgm to find 2nd highest num from the given array using Java streams
12. Write a pgm to find 2nd highest num from the given array without using Java Streams
int[] arr = {50, 40, 70, 10, 80, 90};
13. What is class loader
14. Types of class loader
15. Benefits of SpringBoot
16. How to create SpringBoot project
17. What is the use of @ComponentScan annotation
18. What is the use of Spring accuators
19. What is mean by stateless in Rest API
20. PUT vs PATCH
21. How MS communicate each other
22. What is mean by Async and sync communication
23. States of Object in hibernate
24. exaplain differnt mappings in hibernate
25. Which design pattren u worked
26. Exaplain about circuit breaker
27. Write a SingleTon class code
28. What is mean by production ready code
29. Any exp in React, Kafka

HCL: L2 - Java Developer

1. What is the use of AutoConfiguration in SpringBoot
2. Dependency Injection in SpringBoot
3. Drawbacks of field injection
4. When and why to use @SpringdataTest
5. Write code for Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid
6. Write a java pgm to get bookreleasedYear as a key and list of books as a value from the List of Books object

Genpact - L1 Java FS

1. Which Java8 fetaures u have used in u r project
2. Why default method introduced in Java8
3. What is functional inteface
4. What is Predicate and which method is present in that
5. What is consumer and which method is present in that
6. What is Supplier
7. What is streams and why
8. Why we need to use filter method
9. get the employee obj where emp sal is >50, <90 using streams
10. Which are the annotations u have used in u r project
11. What is @SpringBootAnnotation
12. @Component vs @Service vs @Repository
13. Why we need @Service, @Repository specifically
14. Can we have @Component, @Service at a same time
15. How to execute a method before calling PUT method

Mphasis : L1 Java Microservices

1. **Java8 features and explain them**
2. **How to avoid NullPointer Exception using Optional class**
3. **What is mean by Optional.isPresent()**
4. **Fail-Fast vs Fail-Safe and give examples**
5. **On which mechanism HashMap works**
6. **What is the new feature added as part of HashMap in Java8**
7. **How LinkedList becomes TreeMap in HashMap**
8. **SpringBoot accuators**
9. **SpringBoot startes**
10. **How many ways to create SpringBoot app**
11. **How to deploy SpringBoot app other than tomcat server**
12. **Which are the anotations u have used in u r project**
13. **WHat is the use of @SpringBootApplication**
14. **What is mean by @ComponentScan**
15. **How MS communicate each other**
16. **How Kafaka works**
17. **How to configure Two diff DBs in our application**
18. **MS architechture**

Mphasis: Client round - Java Microservices

1. **End to end project explanation**
2. **Java8 features**
3. **what is the use of optional class**
4. **What is functional interface**
5. **what is mean lamda expressions and why**
6. **What are streams**
7. **What new feature introduced in Java8 as part of Interfaces**
8. **Why default menthods introduced in Interfaces**
9. **What is collection**
10. **List vs Array List**
11. **HashMap vs TreeMap**
12. **HashMap internal implementation**
13. **HashSet implementation**
14. **What is singleton design pattren and why and when to use**
15. **What is Predicate**
16. **MicroService Architechture**
17. **Design pattrens in MS**
18. **Circuit Breaker Pattren**
19. **What is API gateway**
20. **How to write custom queries**
21. **Hibernate vs JPA**
22. **How to get API response in angular**
23. **What kind of work you used to do in UI side**
24. **What is your roles and responsibilities in current project**
25. **What is agile methodlogy**
26. **how to exclude tomcat server in springboot**

Circana : L1 - Java FSD

1. **Explain the Oops concepts**
2. **{51,34,70,4,26,55,4,19,55,23,17,70,42,57,51,70} find 2nd max from the given array using Java8**

3. String s="big black bug bit a big black dog on his big black nose"; find number of occurance of each word by using collections
4. Difference Between IdentityHashMap, WeakHashMap
5. HashSet implementation
6. hascode, equals contact
7. @RestController vs @Controller
8. explain Angular lifecycle hooks
9. constructor vs ngOnInit
10. How components communicate each other
11. How components communicate each other when they are different
12. Bean scope in spring
13. Types of bean scope
14. What are the annotations u have used in u r project
15. Use of @Qualifier annotation
16. Fail-fast vs Fail-safe
17. PathVariable vs @QueryParameter
18. Design Pattrens in java
19. How to valid the request body before processing it (using inerceptors)
20. How to implement lazy loading in angular app
21. Observables vs Promises
22. Which method will call in the below code

```
        class A {

        static int add(int a, int b)
        {
                return a + b;
        }
}

class B extnds A{

static int add(int a, int b)
        {
                return a + b;
        }
}
class C exetnds B{
        public static void main(String args[])
        {
                A b = new B();
                b.add(4, 6);
        }
}
```

23. Why to use static method
24. How to create Singleton class
25. How to make a Immutable class
26. What you to if you have collections of object in Immutable class
27. Java8 features
28. Interfaces vs Abstract class
29. What are the functional interfaces in java

30. How many ways we can create a form in angular

31. Once we click button popup should open and show the data which is passed from the main page

GainWell - L1 - Java FSD

1. **Which are the Java8 features u have worked**
2. **List<Integer> numbers = Arrays.asList(10, 20, 30,89,90,20,10); Sum the list using java8**
3. **Why SpringBoot used**
4. **Explain @SpringBootApplication**
5. **Post vs Put**
6. **How can you check the logs**
7. **Which security framework you have used**
8. **What is SpringBoot Actuators and why**
9. **How you have handled Exceptions in your current project**
10. **Which Design Pattern you have worked in your project**
11. **what is 405 error**
12. **What you do if any production issue comes**
13. **API gateway**
14. **How MSs will communicate each other**
15. **Angular hooks**
16. **Lazy loading in angular, how you will achieve**
17. **Data binding**
18. **Types of data binding and explain those**
19. **Observable vs Promise**
20. **How u will connect with backend API and fetch the data**
21. **What is Pipes in angular**
22. **How to create custom pipe in angular**
23. **Write a html code to convert IND currency to Dollar currency**
24. **How to pass data from one component to another**
25. **Margin vs padding in css**
26. **write a code to create a form with 2 fields, once you submit it the data should show**
27. **Sort the list of numbers using JS/TS**