

## Interviewed Question

### 1)daten Wissen (08/12/2024)

1. SQL find duplicate data in user table?
2. What type of dependency injection and why constructor injection are preferred
3. **Write program to reverse "Pankaj chaudhari" without using StringBuilder and StringBuffer**
4. Difference between controller and RestController
5. What is @PathVariable and @RequestParam
6. Can I get data without using @GetMapping? Can I use @PostMapping to get data?
7. In Docker, if I delete the image, then what happens? Will data be lost if I delete the volume?
8. In Swagger, does it happen at compile-time or runtime? If you run again, does it take previous or regenerate?
9. If there are 200+ APIs, does Swagger work? If yes, then how? If no, then what is the solution?
10. ArrayList in LinkedIn: If you want to search, then which data type is preferred? And for deletion and insertion, which data type is used and why?
11. What is a RESTful API?

### 2) ReBit (21/06/2024)

1. Difference between HashMap and Hashtable
2. Difference between == and equals method
3. Find missing number and find duplicate number
4. **How to add element in HashMap**
5. @Data annotation in Spring Boot
6. Define repository method to get data

### 3) WebOsmotic: (10/01/2025)

1. Java 8 feature
2. What is lambda function
3. What is stream API
4. What is a functional interface
5. What is abstraction how to achieve
6. What is encapsulation and how to achieve it
7. How to achieve polymorphism
8. What is the diff between string StringBuilder and StringBuffer
9. What is threadsafe and how you achieve threadsafe what are the ways
10. What is an equal method
11. if you want to compare employee object how you compare
12. What is comparable and comparator and its method
13. What are collections in Java
14. What are interfaces in the collection
15. Which is set and diff between list and set
16. Diff between array and list or ArrayList
17. What is hashmap
18. If you have to sort objects how you do
19. What is join and explain each and its different
20. If you have to optimized query how you do
21. What is indexing
22. What are aggregate function
23. What is spring ,spring boot diff
24. What is dependency injection di
25. What is DispatcherServlet
26. Which java version you used
27. **Program on longest uniq String aabaabbacd**
28. **Remove duplicate and print length**

#### **4) Taodigital(by ravi 11/1/25)**

how many ways to can print in console  
why arrays start with zero index in short  
what are the advantages and disadvantages of object cloning  
describe the java io and new io  
Http Status code and

**As an example, if the first string were ADOBECODEBANC and the second were ABC, the answer would be BANC.**

## 5) ncricle(16/01/25)

1. **What is the difference between abstraction and encapsulation?**
2. Why do we use encapsulation to hide data?
3. Why is multiple inheritance not supported in Java?
4. What is a composition in Java?
5. Write code using a `try-catch` block to catch a runtime exception.
6. What is the difference between `TreeMap` and `HashMap`?
7. When should you prefer `TreeMap` over `HashMap`?
8. Can we store `null` as a key in `HashMap`?
9. What are the types of collections in Java?
10. What are the benefits of inner classes?
11. How can we test a static method in JUnit? Provide a concise explanation.
12. **Program:**

**Given a string, write a function that returns the longest substring without repeating characters. Example: For the input string "abcabcbb", the output should be "abc", with a length of 3.**

## 6) L & T(1/16/25)

**What are the APIs that you work on?**

1. What is the latest version of Spring?
2. What are the features of Java 8?
3. What is the `static` keyword in Java?
4. What is a `static` block?
5. How to implement a custom exception?
6. When to use an exception and when to use a runtime exception?
7. What is the difference between checked and unchecked exceptions?
8. Is `ArrayIndexOutOfBoundsException` a runtime exception?
9. What is a lambda expression?
10. Implement a simple Java program using a lambda expression.

11. Why do we use the `this` keyword in a class?
12. Why do we use the `@FunctionalInterface` annotation?
13. What is AOP (Aspect-Oriented Programming), and where did you use it?
14. What is the difference between the 401 and 403 error status codes?
15. What is the difference between `Comparator` and `Comparable`?
16. What is `.orElse` in Java?

**find maximum occurring character in string**

**7)myNextHire(20/01/25)**

**Problem Statement:**

Given a string, find the first character that does not repeat. If all characters in the string repeat, return `-1`.

**Examples:**

1. Input:  
`"swiss"`  
Output:  
`'w'`
2. Input:  
`"aabbcc"`  
Output:  
`-1`
3. Input:  
`"leetcode"`  
Output:  
`'l'`

**Notes:**

- The string contains only lowercase English letters.
- The maximum length of the string is 100 characters.
- Solve the problem in one pass (i.e., traverse the string only once).

## 8) HCLTech(26/01/2025)

1. Tell me about yourself.
2. What is your project, and what technologies have you used in it?
3. Which version did you use in your project?
4. What is a functional interface?
5. What are the different types of functional interfaces?
6. In which scenario have you used the Consumer interface?
7. What is a HashMap?
8. What is the difference between HashMap and Hashtable?
9. Can I insert a null key in a HashMap?
10. If we pass a duplicate null key, what will happen?in hashmap
11. How does HashMap work?
12. What is the difference between LinkedList and List?
13. Which one is used the most?
14. Suppose we want to delete a value at a specific index in a LinkedList.
15. Why do we use functional interfaces instead of abstract methods?
16. Why do we use abstract methods in interfaces, and why not use abstract classes?
17. What is the difference between map and flatMap?
18. What is a parallel stream?
19. What is the difference between the filter and map methods?
20. What is a Spring Boot application?
21. What is the difference between a filter and an interceptor?
22. Suppose a client sends a request to your application; what will happen?
23. What is Hibernate and JPA repository?
24. Which one is better, Hibernate or JPA?
25. What is an auto-configuration condition?
26. What are @ResponseBody and @RequestBody annotations?
27. interceptor
- 28. I have a list of integers from 1 to 10. The numbers 5 and 6 are repeated. I want to know which numbers are duplicates. (Asked for Java code)**  
  
**Or find occurrence in list using stream**

## 9) TechBulls(29/01/2025)

1. Write code for a controller with a POST method.
2. What is a lambda expression? Implement `Runnable` using lambda.
3. What is a functional interface? Provide an example of a functional interface.
4. **Find the 2nd maximum number in an array** (If I have 1 lakh records, how can I find it in one shot?).
5. How do you inject dependencies in a controller?
6. What is an immutable class?
7. How is dependency injected in a controller?
8. List all methods of functional interfaces.
9. What is an immutable class? (*Repeated, you might want to remove one*)
10. What is a security filter?

#### 10) Jforce: 4/02/2025

1. Internals working of `Set` in Java.
2. Difference between `Runnable` and `Callable` interfaces.
3. What is `static` in Java? and where does it store
4. When we inject dependency using `@Autowired`, what happens in the background?
5. What is `ConcurrentHashMap` and how does it work?
6. Difference between `TreeMap` and `LinkedHashMap`.
7. **Write a program where:**
  - `EmpDTO` class contains `name` and `salary`.
  - Sort employees by name.
  - Find the third highest salary using Stream API.

#### 11) Human Cloud:5/02/2025

1. Can you introduce yourself?
2. Which features are used in Java?
3. Difference between `ArrayList` and `LinkedList`?
4. What is multithreading?
5. Suppose you are working on an application... (incomplete question)
6. I have an API for food orders; which HTTP method should I use?
7. You are using an external payment gateway, and there is a network issue. How do you handle it?
8. You need to design a Q&A system with multiple questions. How do you handle it?

9. Can you define monolithic architecture?
10. What is the difference between the `equals` method and `==` operator?
11. Can we write a private method inside the `main` method?
12. Can you define `StringBuffer` and `StringBuilder`?
13. What are wrapper classes?
14. Can we declare the same variables in the same class?
15. What is a functional interface?
16. Why do we use a functional interface?
17. What is `final`, `finally`, and `finalize`?
18. What are `volatile` and `synchronized` keywords?
19. Java 8 program to find duplicate elements in an array (output: 2 3 4 5 6).
20. What is the OPTIONS API?
21. How to create a POST and PUT API in Java (Spring Boot)?
22. Difference between `StringBuilder` and `StringBuffer`?
23. How do you design a Q&A system with multiple questions using Hibernate?
- 24. Write a program to remove duplicate number**
- 25.

### 2nd round:

1. Diff between post and put what will happen when post and put method call 2 time
2. What is lazy loading and what will happen with the example
3. String str="thetimeisthreeoclock" find 2nd max occurrence of character

### Find output: of program

```
class Box {  
  
    int value;  
  
    Box(int value) {  
  
        this.value = value;  
  
    }  
  
}  
  
public class Main {  
  
    public static void main(String[] args) {  
  
        Box box = new Box(10);
```

```

        System.out.println("Before method call: " + box.value);

        modifyBox(box)

        System.out.println("After method call: " + box.value);
    }

    public static void modifyBox(Box b) {

        b.value = 20;

        b = new Box(30);

    }
}

```

**what will be the output**

## 12) GlobalLogic:5/2/2024

1. What are the advantages of Stream API?
2. How to filter employees from different cities and find only those in Bangalore using Stream API?
3. What is the Optional class?
4. How to handle `Optional` in case no employee is found?
5. How to use `orElseThrow` with Optional?
6. Find the second-highest number from a list using Stream API.
7. Find the second-highest number without using sorting in Stream API.
8. What is the basic flow of a Spring Boot application?
9. What is dependency management in Spring Boot?
10. What are the commonly used annotations in Spring Boot?
11. How to integrate MySQL with a Spring Boot project?
12. How to implement caching in a project using Redis?
13. How to implement pagination in a project?
14. How to create a Spring Boot API to fetch employees from Bangalore using JPA?



15. SQL query to select employees from Bangalore (`city = 'Bangalore'`)

### GlobalLogic:2nd Round

1. Method Overloading and Method Overriding
2. Abstraction Feature in Java
3. Example of Abstraction in Java
4. Difference Between `finally`, `finalized`, and `final`
5. Difference Between Immutable Classes
6. What is a Functional Interface?
7. Difference Between String, StringBuffer, and StringBuilder
8. ArrayList and LinkedList - Difference
9. Difference Between Synchronized Method and Synchronized Block
10. Difference Between Runnable and Callable
11. Explain Dependency Injection
12. What is Spring Boot Starter?
13. How to Handle Exceptions in Spring Boot?
14. How to Implement Pagination in Spring Boot?
15. Hibernate and JPA - Difference
16. How Do You Execute an Update in a Repository (SQL)?
17. Design Patterns
18. Kafka in Microservices

### Infosys(8/2/2025)

1. what is diff between synchronized method and block
2. what is solid principle
3. diff between `@RequestParam` and `PathVariable`
4. what is spring security
5. features of spring and spring boot
6. what is multi threading
7. what is AOP
8. design pattern write singleton pattern code
9. internal working of hashmap

## **Interview question LinkedIn(java spring boot)**

1. How to use Lambda expressions and Stream API in Java?

Follow up:

Can you write a simple example where you filter a list of numbers and find the sum of even numbers using Java Streams?

2. What is the difference between synchronized and volatile in Java? Can you give an example of when each would be used?

3. Can you explain how Spring Boot's auto-configuration mechanism works? How does it help developers to quickly set up an application?

4. Can you explain the concept of Dependency Injection (DI) in Spring? How have you used DI in your Spring Boot applications? Can you give an example of constructor-based DI and setter-based DI?

5. How can you expose RESTful services in Spring Boot? Can you walk me through creating a simple REST API with a GET and POST method?

6. How to secure a Spring Boot application? Can you explain how you would use Spring Security for user authentication and authorization? Can you describe an example where you configured role-based access control?

7. How to deploy a Spring Boot application to an AWS EC2 instance?

8. Can you explain how AWS S3 works? How to use it in a Spring Boot application to store and retrieve files?

9. Can you explain how Maven manages the build lifecycle of a project? What are the main phases of the Maven lifecycle, and how have you used them in your projects?

10. How does Maven handle project dependencies? Can you explain how you manage version conflicts with Maven?

11. What are Maven profiles and plugins? How do you use them to customize builds for different environments or purposes?

12. How to write unit tests in Java using JUnit? Can you provide an example of a simple unit test for a Spring Boot service method?

13. Can you explain Test-Driven Development (TDD)? How have you applied TDD in your Spring Boot projects, and what challenges did you face while writing tests before the implementation?



## **AUTOMATION TESTING**

1. Can you explain a try-catch block with finally? What happens if an exception is thrown inside finally?

2. How does the String class differ from StringBuilder and StringBuffer?

3. Can we overload constructors? Provide an example.

4. Can we inherit constructors in Java? Why or why not?

5. Write a class with private variables and public getter/setter methods.
6. Can an abstract class implement an interface? Provide an example.
7. Write a program to find and count duplicate characters in a string.
8. How do you prevent memory leaks in Java applications?
9. Write a script to log in to a web application with a username and password, then verify the user has logged in successfully.
10. Write a script to select a value from a dropdown using the Select class.
11. What are implicit and explicit waits in Selenium? How do they differ?
13. Write a TestNG test suite to verify the login functionality of a website

### Interview ask programs:

1. Write program to reverse "Pankaj chaudhari" without using StringBuilder and StringBuffer
2. How to add element in HashMap
3. Program on longest unqi String aabaabbacd Remove duplicate and print length
4. As an example, if the first string were ADOBECODEBANC and the second were ABC, the answer would be BANC.
5. Given a string, write a function that returns the longest substring without repeating characters. Example: For the input string "abcabcbb", the output should be "abc", with a length of 3.
6. Problem Statement:  
Given a string, find the first character that does not repeat. If all characters in the string repeat, return -1.

### Examples:

4. Input:  
"swiss"  
Output:  
'w'
5. Input:  
"aabbcc"  
Output:  
-1
6. Input:  
"leetcode"  
Output:  
'l'

**Notes:**

- The string contains only lowercase English letters.
- The maximum length of the string is 100 characters.
- Solve the problem in one pass (i.e., traverse the string only once).