Lowes 1st round

Duration : 1hr

Format : 1 Algo question, Verbal questions

1. Write a program to provide integer pairs that add up to target value
   1. Int array = {1,2,10,4,5,6,3,7}
   2. Target = 11
2. Write a stream program where string input and hashmap output <character, count>?

string target t=2 a=1

1. Is java pass by value or pass by reference
2. What is a singleton design pattern? Why do you use singleton class when you can use Final and static classes?
3. Where to use comparable interface and comparator interface? How to sort a collection?
4. How does garbage collection works in Java
5. you have an exception in class which gets caught by catch block, in the finally block we have return statement. Will return statement work?
6. print all sets of numbers from a given array whose sum totals 5
7. Define an ArrayList with the employee objects having name, salary, department
8. Theory:  
   Abstraction & Interface  
   Annotations in Springboot - use and meanings  
   hashmap and synchronous hashmap
9. OOPS Concept
10. Create a user database in Java using Encapsulation
11. Create CRUD using Rest
12. What are the URL links of every operation in CRUD?
13. Basic Annotations in Spring Boot
14. Problem solving using Collections
15. SQL quries involving joins

1. Write a program to rotate a singly linked list ‘k’ times -> Answered

2. Point out all the errors in the following program: -> Answered

final abstract class SuperClass{

public SuperClass(){

}

public void method1() throws IOException{

System.out.println("SuperClass");

}

public abstract method2();

}

public class ChildClass extends SuperClass {

public void method1() throws Exception {

System.out.println("ChildClass");

}

public static void main(String[] args) throws Exception {

SuperClass childClass = new ChildClass();

childClass.method1();

}

}

3. What is the output of the following code snippet: -> Answered

public class Car{

public void break(){

sop "Vehicle stops"

}

public void accelerate(){

sop "Vehicle accelerates"

}

}

public class MercedesBenz extends Car{

public void break(){

sop "Vehicle stops"

}

public void accelerate(){

sop "Vehicle accelerates"

}

public void opensunroof(){

sop "sunroof opened"

}

}

public class Test{

psv main(string[] args){

Car car = new MercedesBenz();

car.opensunroof();

}

}

4. Given a list of employees, return a list containing age wise average salary of the employees using java8 features: -> Partially answered

List<Emp> employees = Arrays.asList(new Emp("A", 25, 50000),

new Emp("B", 25, 35000),

new Emp("C", 30, 56000),

new Emp("D", 25, 60000));

5. Write a REST API (Controller, Service and Repository class) to return a list of edam ids which have missing CDN info: -> Answered

Sample edam object: {

"edamId": "EDAM\_00307331",

"cdn": "https://ppgoogle.com/test/blueimages/36bdee6a-e628-48b8-b131-420a7bbbe771/00307331.jpeg",

"type": "Image",

"active": true,

"metaTags": [

"TAG\_image"

]}

6. Life cycle of a bean in Spring. -> Answered

Qn. Optimize the solution.

N

Array A = [4,7,9]

2 Operations

Input

Operation type = 1 or 2

1 = Add an element X in this array

1 11

[4,7,9,11]

2-> Find the max element

11

1 11

[4,7,9,11,11]

2

9

ans = 11

Map --4 -1, 7-.1,9->1,11->1

TreeSet 4 7 9 11

11

// ifvalue corresposndingkey ==1...find largestnkey

// 4 7 9 11 11

solve(List<Integer>list){

TreeSet<Integer> set = new TreeSet<>((x1,x2)-> x1> x2?-1:x1==x2?0:1);

set.last

int q =2;

wille(q!=0){

int q1 = 5;// System input

list.add(5);

..adding 5

n=5

//find max

int f=mp.getOrDefault(n,0);

if(f===1){

set.remove(n);

}

map.put(n,f+1);

if(f+1==1)

{

set.add(n);

}

}

}

Input

3

4 7 9

4

1 11

2

1 11

2

Output

11

9

has context menu

Implement TreeSet in decreasing order.

Ans: TreeSet<Integer> set = new TreeSet<>((x1,x2)-> x1> x2?-1:x1==x2?0:1);

Java 17(As I have used java17) vs Java 8

What tool u have used for Deployement?

Java 8 basic qns.(Functional Interface, Lambda Exp).

Lowes 2nd round

Duration: 1hr

Format : LLD System Design, API Design, Verbal Questions

1. Provide the LLD of a chess game, define how you will create chess pieces and their moves?
2. Provide the class structure for the design provided
3. For the above design provide the API design for it, define the endpoints, GET POST requests, their parameters, request body, Response types.
4. How will you provide the parameter, path params, query params, request body
5. Spring version, Spring boot versions
6. Annotations used in controllers, @Component, @Controller, @RestController, @ResponseBody.
7. Can you autowire a controller class?
8. JPA and hibernate how do you configure it in your application and configure a database.
9. How to create a new table using hibernate

React

1. What is the difference between Angular and React which is more consuming and why?
2. Difference between Class Based component and functional component
3. Tell all the hooks which are using in react and why we use useEffect what is the purpose to use?
4. How to build the react application?
5. How to optimize the react build?
6. What is micro front-end?
7. How to build the micro front-end code?
8. Props and state management
9. How to pass data in react in two ways?
10. Client-side rendering and server-side rendering
11. How to optimize micro frontend build?

**LOWES REACT QS:**

1. What are possible ways to create objects in javascript?
2. Difference between VAR, LET and CONST?
3. How many type of scopes are there From where you can declare variables ?
4. What is the use of map in React JS and how many parameter does map take?
5. How to pass data from child component to Parent component, can data flow from child to parent component?
6. Which library you are using for API calls or how you are calling API?
7. Which Hook you will use for API call or fetching data?
8. Apart of react any other framework you are using?
9. Is there any way to change props in react?
10. How to handle Asyncronous operators in javascript?

Programming 1:

const persons = [{name: 'personX', list: [1,2,3,4,5,6,7,8,9]}, {name: 'personY', list: [9,10,11,13,15]}]

output: {personX: [2,3,5,7], personY: [11,13]}

Programming 2 :

A diagram of a layout

Description automatically generated

By using HTML, CSS need to display above Image.

1. Explain the sequence of Redux state management.

2. Differences among the map (), filter (), and reduce. Can map () iterate over an array having conditions in it?

3. Explain how Inheritance works in JavaScript?

4. Brief about Event-looping system in JavaScript. Where does function, Promise, set Timeout sit in event loop?

5. what are closures? and write an executable sample for closure.

6.Purpose of strict mode in JavaScript and its advantages.

7.Can we write a read-only property in JavaScript and how do we achieve it?

8.Arrow functions and its advantages.

9. What is webpack? Have you worked on any Framework?.

10.How do we achieve Deep copy?

11. How do we achieve responsiveness in web app development.

12. Difference between Flex and Grid?

13. Difference between Grid and AG-Grid?

14. How do we achieve generalization of styling in CSS and how do we achieve it?

15. What are the font units for sizing?

16. rem vs em? Which one did you used the most.

17. Most frequently used react hooks and when do we used them?

18.What code have you written for unmounting using useEffect in your project?

19. How do we achieve performance optimization in react.

20. Write a reusable-code for the following.

//input

const persons = {personX: [1,2,3,4,5,6], personY: [7,8,9]}

//output

[{name: 'personX', list: [2,3,5]}, {name: 'personY', list: [7]}]

21. What is the result of the following?

const person = {name: 'personxX'}

const sampleFunc = (x) => this.x

sampleFunc(5)