**Yagnashri (Girish & Prateek Agarwal)**

1. What is a functional interface and what is a marker interface?

2. What is serialization and how to avoid serializing of a particular member of a class?

3. Can transient be applied on primitive datatypes or only on a java object?

4. What is a singleton design pattern and how do we control the creation of objects only five times?

5. In a multithreaded environment, which of the collection/Datastructure works on faster retrieval of data?

6. How to control creating pool of threads for doing a task in multithreading?

7. What is @QueryParam and @PathParam?

8. What are the necessities that are required to query the database to quickly retrieve a data?

9. What are views in SQL and how to create it?

10.How to do the connection pooling to improve the performance of the application?

**Shruti Tiwari (Pankaj Attree)**

1. What is POST n GET http methods.

2. Which http methods replaces the resource in server?

3. What is hamcrest maven dependency used for?

4. What is JSON schema validator used for?

5. What are the steps you will keep in mind while automating new pr?

6. How-to-send-JMeter-performance-testing-result-to-developer?

7. Difference between Path Parameter and Query parameter?

8. How to apply logical AND n OR operator in REST API?

9. What are the steps you will perform to create a automation project?

10. How will you let know ur developer about performance test results?

**Prachi Rajput (Gaurav Tak)**

1. Your roles and responsibilities in the previous project

2. OOPS Design patterns, was asked about observer design pattern in detail

3. Design pattern used in Kafka

4. What do you mean by asynchronous communication

5. Difference between asynchronous and multithreaded communication

6. Explain SOLID principles

7. Explain Microservice design patterns

8. Disadvantages of microservice architecture

9. Explain circuit breaker design pattern

10. Explain working of API gateway

11. What’s proxy and reverse proxy

12. Coding questions:

i) https://leetcode.com/problems/rotate-array/

ii) Sort a hashmap according to values

iii) Implement @GetMapping for rest api

13. What’s @Autowired annotation

14. Why were streams introduced, what advantage do they give

15. Difference between comparable and comparator

16. What is race condition, how do you handle it

17. What’s 2 phase commit

18. If multiple pods are trying to update the database at the same time, how would you resolve this issue.

19. Internal working of hashset

20. What are spring profiles, how do you stop a certain bean from being initialized in a particular profile

21. What is Blocking Queue and when do we prefer it over Queue data structure?

**Kamatchi (Sandeep Sridharamurthy)**

1. Rest controller write sample code for Get, POST, delete
2. what is idempotent in rest why it called as idempotent
3. {10, 23, 65, 20, 30, 3}
4. return Sum of even numbers using Streams
5. diff between couchbase & sql
6. OAUTH explanation
7. junit & mockito
8. design pattern in java
9. singleton design pattern is thread safe or not
10. how to handle exceptions
11. how to create custom exceptions

**Sumathi (Mahendran & Sudhamai)**

1.Tell about previous projects and technologies.

2.Do u have an experience in production support

3.In your work experience,do u face any critical issues and how to resolve that

4.How to increase performance in mysql

5.How to fix the production issue/defects

6.In production support's, what are the critical issues u faced and how to resolve that

7.Why we go for an hibernate

8.Diff between jdbc and hibernate

9.How to optimize the link(URL)

10.What are all the ways to increase the DB performance

**Kranthi Kumar Dheemu (Vineet Vijayan – MF talent)**

1. Section in COBOL program (give a brief explanation on each)
2. next sentence and continue in COBOL
3. scope terminator in COBOL and how is it used
4. sco4 and soc2
5. jes2 and jes3
6. IEBGENER
7. concurrency in DB2
8. left outer join and join what is the main difference between them.
9. what version controls have you used.
10. what's the use of commit and why is it used

**Shrishti Seth (Simerenjaut Kaur)**

1. About Project and roles and responsibilities.

2. Interfaces and difference between interface and abstract class. Which is better to use and why?

3. Marker Interface and purpose of using it?

4. Internal working of Hashmap and what is hash collision?

5. Polymorphism run time and compile time.

6. What is Agile Methodology?

7. if class A implements 2 interfaces B and C and both has test methods then which one is called and why?

8. some scenario based question...especially what would be the output of the following code type.

9. how to create custom marker interface.

10. Exception handling and finally block.

11. flow of springboot application

12. Rest APIs and microservice basics

13. difference between arraylist and linkedlist and why linkedlist is preferred over arraylist for data manipulation.

14. Singleton design pattern and how to transform a non-singleton class to a singleton class.

15. converting streams to collection with example.

16. Synchronization and deadlock?

17. how to remove deadlock?

18. string builder and string buffer and what do you mean by mutable and immutable

19.what do you understand by hashcode. also about hashcode and equals contract.

20. About multithreading and how to create thread and which is better way to create thread and what is daemon thread

21. Scenario based streams coding question.

**Thejaswini (Mahendran)**

introduce your self

Difference between Abstraction and interface

In functional interface what is difference b/w abstract and interface which is is used in better way

How hashmap internally maps

1)

class A{m1();}

class B extends A{m2();}

A a=new A();

a.m1(); // output

a.m2();

B b=new B();

b.m1();

b.m2();

A ab=new B();

ab.m1();

ab.m2();

2)

Map<Student,Integer> map=new Hashmap<student,Integer>;

Student s=new Student(1,"mahe");

Student s=new Student(1,"mahe");

map.put(s,1);

map.put(s,2);

how may objects can be created?

Map<Integer,Student> map=new Hashmap<Integer,student>;

map.put(1,s);

map.set(1,s);

map.get(1).getID()==?

**Aakriti (Mahendran)**

1. Tell about your tech stack, technologies you have worked on and about role in previous project.
2. What is Encapsulation
3. What is polymorphism and its example
4. What is an interface
5. What are abstract classes and where do you need to use an abstract class and an interface
6. Difference between list and set and coding questions based on that
7. Map interface and coding question based on that
8. Threads in java
9. Coding questions related to Inheritance
10. What is functional interface.

**Soundarya (Prateek & Girish)**

1. Print Duplicate values in array
2. sum(1)(2)(3)=8 3)Basic javascript questions var,let,const

**Saryu Patel**

1. What is JDK, JVM & JRE?
2. What is class Loader?
3. what is constructor? can objects be created without constructor?
4. What is marker interface?
5. what is abstract class?
6. what is interface and what is different bw interface and abstract class?
7. What is singleton class?
8. Difference bw reference object of Map and HashMap. which is better?
9. internal working of HashMap. Any change in newer version of Java?
10. Difference bw HashMap and HashSet.
11. Program to iterate a hashMap.
12. Insertion of duplicate Key or duplicate value in HashMap. what will be
13. the behavior of each case?
14. Difference bw ArrayList and LinkedList and when to use them?
15. Different ways to create a thread. which is better and why?
16. what is Daemon Thread?
17. what is deadlock and how can we prevent it?
18. what is wait(), notify() and notifyAll()?
19. Write a SQL query for inner join.
20. difference between Delete, truncate and Drop.
21. What is normalization and its type.
22. how to get unique values in SQL.

**Darshpreet Kaur**

1. what is stream api
2. what is functional interface and diff b/w dynamic vs static interface
3. terminal operations in stream
4. return type of .collect()
5. what is threadpool and how to create
6. what is Executor framework
7. what is dead lock and how to prevent.
8. what are constructors
9. other way to instantiate obj in java
10. what java 8 features you used and related questions from that
11. linkedlistVs arraylist
12. internal working of hashmap
13. Using agile or waterfall in your proj and agile related questions
14. scenario based questions and how to implement it -logic only EX: given-4,5,3,2,1,0,6,8,9/output-5,3,1,9,4,2,0,6,8(all odd number print first after that even number prints)
15. @controller vs @restcontroller

**Darshana Chopade**

1. Tell me about yourself.

2. what is a marker interface? How it works?

3. What is constructor? What is the purpose of using a constructor?

4. What is HashMap?

5. Explain about try catch block. How many types of Exception? Explain Checked and Unchecked exception with example.

6. What will be the output?

   try{

   SOP(A);

   throws IO Exception

   }

   catch{

   SOP(B);

   }

   finally{

   SOP(C);

   }

   SOP(D);

7. What is finally?

8. What is Static binding and Dynamic binding?

9. What is Overloading and Overriding?

10. Declare a hashmap. Add some values to it. Iterate.

11. Write logic to get employee names whose name that with A with help of java8 feature.

12. What is JDK,JRE and JVM? Can we execute a code with JRE only?

13. How user will consume Rest api? Explain on high level. Explain layers of spring application.

14. How to Persist Data?

15. What is default server in spring boot? Can we change the default server? How?

16. What is Singleton? Write a code.

17. How to create Thread? Which method is preferred to create thread and why?

18. What is Deadlock? What is Livelock?

19. What is entryset?

20.Is String Immutable? Why? How to mutate string? What is difference between String Buffer and String Builder?

21.Explain internal working of HashMap? What is HashMap collision? How we can avoid it?

22. What is Anonymous Class?

23. Difference between wait() and notify()

**Sarthak Mohapatra**

1. Tell about the last project on which you worked.
2. What is REST?
3. Is there any other specification apart from REST, which uses HTTP protocol?
4. What is the difference between POST method and PUT method?
5. Which version of Java have you used?
6. What are the features available in Java 1.8?
7. What is the difference between functional interface and lambda function?
8. How to implement security in a Spring Boot application?
9. How to implement security in a Java application?
10. Employee (name, id, salary)

id eg: dev001, devb002, m002

Id begins with dev

salary more than 10000

sort on basis of last 3 characters in id (i.e 001 in case of dev001)

1. Write a function which takes an array of n integers where array[i] is in the range of [1, n], returns an array of all the integers in the range [1, n] that do not appear in the array.

Example:

----------------

1) getDisappearedNumber([4,3,2,7,8,2,3,1]); // [5,6]

2) getDisappearedNumber([1,1]); // [2]

1. Which databases did you work on?
2. What is the difference between SQL database and MongoDB database?
3. Is there any difference between MySQL database, Oracle database, PostgreSQL database?
4. What is the disadvantage of using joins in SQL?
5. What is indexing in SQL?
6. What are the advantages and disadvantages of using microservices architecture?
7. Explain how you will shorten the incoming URLs in a microservice based application.
8. How will you implement the logic to shorten the URLs?
9. Which databases will you use for storing the URL details and why?
10. How will you optimize the microservices application for shortening the URLs to handle large number of requests to improve the performance of the application?

**Yagnashri(Simerenjaut Kaur,Yogendra Yadav)**

1. About roles and responsibilities

2. What is Deadlock and how to avoid deadlock?

3. How to achieve threadsafety?

4. Internal working of hashmap and hashSet.

5. Explain OUTER JOIN.

7. What is try with resources and which interface should the resources implement to autoclose? What happens if the resources does not implement Autocloseable interface?

8. How to expose Post api->explain the end to end working on how to persist a requestbody into the Database?

9. What is the purpose of Dialect mentioned in the prop file in a spring boot application?

10. What is Fail fast and fail safe iterator?

11. How to avoid concurrent modification exception?

12. If a thread1 is iterating on a collection and another thread2 tries to modify the collection, how does the thread1 knows that the collection has been modified?

13. How does the treeset stores the element in a sorted way?

14. What are the various ways in which u can print even numbers first and odd numbers post even numbers in a list? Explain one or two logic.

15. Concurrent HashMap working.

**Harsh Mohan Agarwal**

Past project overview and your role in it.

What is normalisation and why is it required.

Scenario based question on normalisation and it's impact on the overall performance.

Impact of JOINS on db performance

What are REST apis and difference b/w GET & POST request.

Java 8 features that you've worked on.

Updates in Time api in java8 vs java7

What is regex, real time application.

What are lambda expressions and why are they used.

How to check a given string is substring of another string using Regex.

Space and time complexity discussion of scenario based questions

Code the following problem

Write a function which takes an array of n integers where array[i] is in the range of [1, n],returns an array of all the integers in the range [1, n] that do not appear in the array.

Example:

----------------

1) getDisappearedNumber([4,3,2,7,8,2,3,1]); // [5,6]

2) getDisappearedNumber([1,1]); // [2

**Heeresh (Anshul, Prateek)**

1] Introduction yourself

2] Difference between monlithic and microservices Architecture

3] Drawback of microservices

4] Write a function which takes an array of n integers where array[i] is in the range of [1, n], returns an array of all the integers in the range [1, n] that do not appear in the array.

Example:

----------------

i) getDisappearedNumber([4,3,2,7,8,2,3,1]); // [5,6]

ii) getDisappearedNumber([1,1]); // [2]

5] For above program complexity space and time

6] Why String is immutable

7] what Database you used

8] Difference between SQL and NOSQL DB

9] explain Java 8 features

10] what is the difference between functional interface and lambda expression

**Viharika Parmar (QE profile)**

1. How to use and/or in API url   
2. 3 Libraries used in scripting   
3. Query/Path params   
4. Framework questions - why not used Karate instead of developing new framework similar to Karate   
5. Performance concepts

**Poornima (QE profile)**

1. Introduce about yourself and the projects you worked on.

2. Explain about rules and responsibilities in the projects

3. Explain about last project

4. Explain about the framework used for Automation

5. Explain about the feasibility analysis made while choosing the framework.

Ans: Whenever a framework is chosen, key areas we look into are Reporting of results, how best it is suited for the applications, How we pass the data, Whether it is suited for cross browser testing, for large data handling, has POM design pattern, added advantage is BDD.

6. How BDD Tests are writeen, are they written from non-engineers?

Ans: Depends on project. Automation tester will only write BDD, steps and Actions. In some projects BDD tests are written by non-engineers and logics/answers are written by automation tester.

7. If there is payment API which is unstable and you are not able to test the complete flow then what will be your approach to give project roll off considering the timelines.

Ans: Virtualisation/Mocking. If some integration is not working as expected due to stability issues, then we can mock the response and see if the test results are as expected.

8. What selectors are used in your automation project? Is the xpath/id faster?

Ans: IDs are faster in locating the elements. But not every project will have IDs, then XPaths are better.

9. If there is any breakage because of the xpaths? What is your approach?

Ans: Go for xpath which are static. Write Xpaths using text, contains, xpath by attribute thatwon't change/break in between the execution.

10. Difference b/w POST and PUT Methods?

Ans: Post is used for creation and PUT for update, POST is not idempotent and PUT is idempotent.

11. How do you write and/or operator?

Ans: cars?colour=blue&model=audi&inclusive=true

         Cars?colour=blur or red or yellow

12. What is the difference b/w Query parameter and Path parameters?

Ans: Path parameters are used to identify the resource and they are part of the URI which we pass in the request.

Query parameters are not compulsory parts of the URI. They are used for filtering and pagination?

13. Which is faster, API Automation or UI automation?

Ans: Considering the time lines, API automation is faster as it yields early results and no need to identify web elements and write logics to perform actions on these web elements.

**Anand Kumar Sharma**

1. What is interface ?

2. Working of hashmap.

3. Write a code of singleton class

4. How to protect singleton with multiple threads ?

5. How to restrict singleton to clone?

6. Diff between @restcontriller and @controller .

7. Write a method signature to use for @pathvarivale

8. Write a code to calculate letter aaabbccccddd ——> a3b2c3d3

9. How to call rest api in your java application and write the code ?

**Paritosh Gautam (Kapil & Gaurav Tak)**

1. About previous project- roles and responsibilities.
2. Streams
3. Spring Annotations.
4. Difference between Rest Controller and Controller.
5. Encapsulation
6. Has a relationship
7. Hash Set and Hashmap differences
8. Tree Map and its working
9. String and String buffer differences
10. Immutable and Mutable meaning
11. How Passwords are stored
12. What is REST
13. GET POST PUT
14. Lambda Functions
15. Terminal operations and Intermediary operations
16. Coding question related to inheritance, write a stream, one ques to write a lambda function
17. Spring Profiles
18. Serialization

**Chittirajulu Ramu (Simer and Yogi)**

1. what is deadlock and how to avoid deadlock
2. explain serialization and deserialization
3. what is transient variable
4. how to make a class as singleton
5. multithreading concepts
6. explain wait() and  notify() methods
7. difference between ArrayList and LinkedList
8. difference between hashmap and concurrent hashmap
9. explain fail fast and fail safe iterators
10. if class A implements 2 interfaces B and C and both has test methods then which one is called and why?
11. life cycle of spring boot application
12. difference between equals() and ==
13. difference between @RestController and @Controller
14. explain post and put mapping
15. write the declaration of hashmap and put the key,value pairs into hashmap and print those values.

**Heeresh (Mahendran & Sudhamai)**

1] Tell me about your previous experience

2] Difference between SOAP and REST in depth details

3]How to decide which service we need to use SOAP or REST in detail

4] Given an integer array t with a size of n which contains only numbers between 1..99. Print numbers that appear more than once.

Example

For t=[3,6,8,3,7,7,7,9,1,3] the expected output is 3,7

For t=[3,6,8,5,7,7,7,9,1,3,5] the expected output is 7,3,5

5]use case if you get URL from client and need to convert that into tiny URL what is your approach to design that product

**Ajit Sundaresan**

1)Interface and Abstract Class

2)How to map a class with database table by Spring JPA

3)Working of HashMap

4)ConcurrentHashMap

5)Spring Profile

6)Marker Interface

7)Cloneable Interface

8) @pathVariable and @Requestparam

9)Singleton class

10)difference between linkedlist and arraylist

11)try catch final

12)JDBC vs JPA(Hibernate)

13)if class A implements 2 interfaces B and C and both has test methods then which one is called and why?

14)ways to create thread and efficient one

15)what is wait(), notify() and notifyAll()?

16) How to run a specific query instead of method naming in Sping JPA?

**Bhavana (Anshul Gairola)**

1. Introduce yourself
2. What is singleton class, write a program for it
3. Difference between post and put,
4. Microservices disadvantages
5. Final keyword in detail
6. How do you select unique values from the table
7. Difference between mysql and postgres
8. How joins affect the query performance
9. What is normalization, how does it affect the query performance
10. Program to find missing numbers in an array.

**SivaParvathi**

1.Tell about your tech stack, technologies you have worked on and about role in previous project.

2. What is POST n GET http methods

3.What is JSON schema validator used for?

4.How to apply logical AND n OR operator in REST API?

5.Difference between Path Parameter and Query parameter?

6.Put and post also Identified?

7.What is oracle db what you done?

8.Table of stunednt student id and marks we need to find out marks of second higest?

9.Explain working of API gateway?

10.Implement @GetMapping for rest api?

11.Diff between jdbc and hibernate?

12.Insted of using inner query any operation we use?

13.What you approch production issues?

14.Have you involved production issues ,how you resolve that how to indentified?

15.what is the issuse in your application what are steps needs?

16.What are the colletions have you used?

17.when do we use Arrylist critiria ?

18.How do we make searching faster in Arrylist?

19.Difference between Array and ArrayList?

20.What is invortion control?

21.Differnce betwen dependency injection and invortion control?

22.How you implement loging funcutonality?

23.Why we are using @before and @after what we are using this?

24.what is differenc between @component and @ service?

25.@Service part of component?

26.Differnc between where and having clause?

27.Design pattrens have to you used?what is singlton pattren?

28.why do you went for singleton ?

29.What are the steps you will perform to create a automation project?

**Vijayakumar Tadavarthy**

1. Introduce yourself with experience  
2. Which version of Java you worked with  
3. Java HashMap internal implementation  
4. what is immutable in java  
5. what is the output of obj1 after second step;

 String obj1 = "vijay"  
String obj1 = obj1+"tadavarthy";  
6. how to access obj1 with the value "vijay" in the above code snippet.  
7. Explain stream operations.  
8. Explain Java stream functionality  
9. When stream operations are chained how to make sure the output from previous step is compatible.  
10. Singleton pattern functionality  
11. what tools you use for CI/CD implementation  
12. on XLR how deployment done  
13. explain java 8 new features used by you  
14. what is functional interface  
14. When java class extend functional interface and obstract call with same method name.  
15. How you resolve access issue for interface method and Obstract class method.

**Kalyani Gade**

1.SOLID principles

2.difference between comparator and comprable

3.Questions on sets and lists

4.write code on @postmapping

5.your roles and responsibilities in project

6.What is joins,procedure,where,groupby

7.method overloading and method overriding

8.scenario of project you are working on?

9.java8 features

10.default and static methods?

11.questuons on stream?

**Krishna Murari Sharma**

1.Introduce yourself?

2.What is additional benefits we get by Using @controller instead of @component?

3.what if we use @controller instead of @repository is DAO layer?

4.controller vs restController?

5.what happens if duplicate key is inserted into hashmap and how to avoid collision?

6.internal datastructure of hashmap and hashset?

7.solid principles with examples?

8.diff between where , groupby,having clause in sql?

9.Joins in SQL?

10.ACID properties?

11.what if two bank employees are trying to manipulate the data of the same person simultaneously?

12.comparator vs comparable?

13.java8 features?

14.default and static method in interface?

15.can we get data if we use @postmapping instead of @getMapping?

16.post vs put mapping?

17.Write a programme to rotate the String in most efficient manner?

18.Sort the Map using Stream API and explain the working of code ?

19.kafka artitechture?

20.how the data will save in hashmap if hash collision occur in 3 entries?

**Vibhor**

1.Introduce yourself

2.difference between encapsulation and abstraction

3.Questions on sets and Maps

4.API code for CRUD operations

5.your roles and responsibilities in project

6.What is having clause in sql

7.polymorphism difference between run-time and compile-time polymorphism

8.scenario based coding ques - sort employees with names and age using streams (Employee is a class with id, name and age as properties)

9.java8 features

10.java streams

11.@Autowired annotation, constructor and setter injection

12.ques on joins in sql and a simple query

13.difference between @RestController and @Controller annotation

**Anitha Angelin Nallamuthu**

* + - 1. Role in current project
      2. Day today activity in current project
      3. technical challenges faced in last 3 months
      4. Difference between Array and list
      5. Oops concepts
      6. When do we go for interfaces over abstract classes some real time examples
      7. public class C1 {

int x = 50;

}

public class C2 extends C1 {

int x = 40;

}

public class Check {

public static void main(String[] args)

{

C2 c2 = new C2();

System.out.println(c2.x); //40

C1 c1 = new C1();

System.out.println(c1.x);//50

C1 c3 = new C2();

System.out.println(c3.x);//50

}

}

Question:

----------------

8.Write a function which takes an array of n integers where array[i] is in the range of [1, n], returns an array of all the integers in the range [1, n] that do not appear in the array.

Example:

----------------

1) getDisappearedNumber([4,3,2,7,8,2,3,1]); // [5,6]

2) getDisappearedNumber([1,1]); // [2]

**Mohammed Asif**

1. Tell me about yourself.
2. What technology stack have you used in your recent project.
3. Which version of Java have you used in your previous project.
4. Scenario:

Interface A{

Default Void print() {

System.out.println(“Interface A”);

}

}

Interface B{

default Void print() {

System.out.println(“Interface B”);

}

}

Class Impl implements A,B {

Void print(){

System.out.println(“Impl class”);

}

}

Is there any problem/ambuity in the code, how to solve it. (Shared the screen and write the implementation).

1. Have you used stream api?
2. What does filter() method do?
3. Scenario:

Class Student{

Private int id;

Private String name;

//getters and setters

}

Class StreamExample {

Public static void main(String[] a){

List<Student> studentList = new ArrayList<>(); //consider list contains 10 items of Student class.

//write logic

}

}

Filter the student record whose id is less than 10 and print all the records. How do you do it. What is predicate().

1. Write a program to find disappeared numbers/missing numbers from a sorted array, in pair

Example:

1) array: [0, 1, 3, 50, 75], lower: 0, upper: 99

ranges: ['2', '4->49', '51->74', '76->99']

2) array: [30, 50, 70], lower: 25, upper: 75

ranges: ['25->29', '31->49', '51->69', '71->75']

void disappearedNumber(int[] originalArray, int lower, int upper) {

// write logic to print the missing numbers.

}

Explanation:

1) array: [0, 1, 3, 50, 75], lower: 0, upper: 99

Lower is specified from where the number starts in an array, and upper means where the array ends.

Output : ['2', '4->49', '51->74', '76->99']

So, the first item is 2 which comes between the input array 0,1 (items, not position).

Next, 4->49 as we have iterate the array till 1, then from 3 to 50 in an array, print the missing number as 4->49, because 4-49 is between 3->50.

1. Have you solved any complex issue in your recent projects? How did you solved those issues.
2. Which build tool and Repository tool you have used in your project.

**Ankur Singh**

1.Introduce yourself

2.your roles and responsibilities in project

3.difference between encapsulation and abstraction

4.polymorphism difference between run-time and compile-time polymorphism

5. Difference between array vs arraylist vs linked list.

6. Internal implementation of add method and contains method of arraylist, hashmap internal implementation.

7. JWT vs oauth

8. scenario based coding ques - sort employees with names and age using streams (Employee is a class with id, name and age as properties)

9.  java8 features  
10. java streams

11. Complex sql query related to join

12. Spring and maven related questions

13. How to protect password in network layer

**Shivam Gupta**

1. Introduction and project experience.
2. Questions from projects like What type of APIs you build?   What is the architecture used?   What technology used?
3. Internal data structure for HashMap?
4. Difference between ArrayList and LinkedList?
5. Internal data structure of LinkedList?
6. Difference between HashSet and TreeSet?
7. What is Asynchronous Communication?
8. Difference between Monolithic applications and Microservices?
9. Disadvantages of microservices architecture?
10. How to implement security in microservices? How we can use firewall ?
11. Load Balancing in microservices? Types of algorithms in load balancing used between instances?
12. Scenario based What happen to request for load balancing if multiple instances is deployed in multiple regions?
13. How to handle error in microservices while communication?
14. Sql scenario, Table1 User (userId, name) and Table 2 Comments (commentId, userId, comment) Write a sql query to find the name of user with highest number of comments?
15. Singleton classes? Write code for implementation of singleton classes? What is factory design pattern?
16. What are the scope of beans?

**Jayashree Korada**

1. Tell about yourself. Roles n responsibilities in your previous project
2. Java 8 new features? Streams in details
3. Functional interface
4. Write a logic to display the duplicates in given array

a = [1,2,3,3,7,7,7,7,4,8,1]

output should be [1, 3,7]

HashSet<Integer> nums= new HashSet<Integer>();

HashSet<Integer> dups= new HashSet<Integer>();

for (int i : inputarray) {

int current = nums[i];

if (nums.contains(current)) {

dups.add(current);

} else {

nums.add(current);

}

}

return dups;

}

1. Why Hashset
2. How do you improve performance in SQL
3. Will the below code compile if yes what will be the value of x in each case ?

Class M1{

int x =50;

}

Class M2 extends M1{

Int x = 40;

}

Main method:

M1 m1 = new M1();

Sysout m1.x

M2 m2 = new M2();

Sysout m2.x

M1 m3 = new M2()

Sysout m3.x

Ans: It compiles. 50, 40, 50

**Vivek Kumar Ghalout**

1. Explain how JavaScript works?
2. How JavaScript handles asynchronous operations?
3. What is event loop?
4. what are pure functions?
5. Why it is important to keep reducer a pure function?
6. Concept of destructuring?
7. Which ES6 features you have used?
8. Difference between map and forEach?
9. Explain array.filter and array.reduce in higher order functions?
10. Some coding questions?

**Satish Kumar Ramamoorthy**

1. Tell about yourself
2. Disadvantage of Micro service
3. Node js how internal working
4. javascript in lifecycle
5. oneApp different version so far in Amex
6. difference between oneApp v5 and v6
7. Service side rendering
8. when run jsx file what files will generate inside build folder
9. is javascript multithreaded
10. difference Asyncronous vs Threading
11. For Book my show asked to design using Microservices

**Kriti Pandey ( Simer and Yogesh)**

* + - 1. Project and Technology used
      2. Difference between abstract and interface
      3. Features of Java 8
      4. Why do we use optional class in Java 8 if we can have null pointer check in java7
      5. What is the use of default and static methods in Java 8
      6. What is executer framework
      7. What are fail fast and fail safe iterator. Internal working of fail safe iterators.
      8. How do threads communicate in multithreaded env
      9. Difference between sleep and wait method
      10. Internal working of hashset
      11. What is deadlock and how it can be avoided
      12. Internal working of concurrent hashmap
      13. Aaabbbbcccdd-> 3a4b3c2d. Write a code to get output (count of character occurrence)
      14. Write a code to get the second largest integer in a arraylist
      15. Write a code to get key and value pairs of hashmap
      16. What do you mean by joins in thread.
      17. What is functional interface.
      18. What is try with resource. What will happen if we will pass parameter which doesn’t implements autoclosable interface.

**Saurav Keshri**

1. Describe about the roles and responsibility in last project?

2. Which versioning tool used?

3. Which IDE is used?

4. What is microservices and whats the benifit of this?

5. How to connect between two microservices?

6. "Given an integer array t with a size of n which contains only numbers between 1..99. Print numbers that appear more than once.

For example:

For t=[3,6,8,3,7,7,7,9,1,3] the expected output is 3,7

For t=[3,6,8,5,7,7,7,9,1,3,5] the expected output is 7,3,5

The computational complexity of the optimal solution is O(n) with n being the length of array 't'.

public static void printRepeatedNumbers(int[] t) {

//TODO

} "

7. What is the error in the below code snippet

method(){

FileInputStream fs;

try {

fs = new FileInputStream(path);

//

fs.close();

} catch (Exception e) {

e.printStackTrace();

}

catch (IOException e) {

e.printStackTrace();

} catch (FileNotFoundException e) {

e.printStackTrace();

}

}

1. How we can expand or shorten the URL after clicking ?

**Komal Rodge (Mahendran and Sudhamai)**

1. explain abt yourself, your previous projs?

2. diff types of polymorphism. example of method overloading.

3. class A {

static m1(){SOP("A")}

m2(){SOP("A")}

}

class B extends A {

statis m1(){SOP("B")}

m2(){SOP("B")}

}

A a =new B()

a.m1()

a.m2()

4.

Set s = new HashSet<>()

s.add(new Student(1,"Mahe"))

s.add(new Student(1,"Mahe"))

s.add(new Student(1,"Sudha"))

5. coding using streams - display list of names whose lenght is greather than 10.

6. Question:

----------------

Write a function which takes three arguments, an array of numbers, a lower limit and the upper limit. The array is sorted integers where the range of elements are in the inclusive range [lower, upper], you need to return its missing range.

Example:

----------------

1) array: [0, 1, 3, 50, 75], lower: 0, upper: 99

ranges: ['2', '4->49', '51->74', '76->99']

2) array: [30, 50, 70], lower: 25, upper: 75

ranges: ['25->29', '31->49', '51->69', '71->75']

**Sagar (Mahendran and Sudhamai)**

1. Roles and responsibilities and previous projs and tech skills.

2. Biggest challenge which you resolved in the last 3 years.

3. Rate your scale on java from 1 to 5.

4. Explain OOPS concepts.

5. Output of the following code:

class A {

static void m1(){System.out.println("A");}

void m2(){System.out.println("A");}

}

class B extends A {

static void m1(){System.out.println("B");}

void m2(){System.out.println("B");}

}

class HelloWorld {

public static void main(String[] args) {

System.out.println("Hello, World!");

A a =new B();

a.m1();

a.m2();

6. Code example for inheritance.

7. Code example of method overriding

8. Diff betweeen REST and SOAP service.

9. write code for reversing string without using inbuilt method.

10. Finding duplicate characters in a string.

**Anitha (Simer and Selva)**

1. pseudo code to consume q POST service - Client Code
2. How will you consume json object
3. Write POJO class for given json
4. Maven command to find transitive dependency also from IDE
5. How will you ensure the jar/EAR has the latest code
6. How to download jenkins ear/jar from linux
7. How to serach a id in logs file using linux commands
8. How will you locate all the pom files in linux
9. grep command in large number of files
10. How to use grep command in recursive manner
11. Group by sql query
12. alias name in sql query
13. sql query to find all rows from the given date
14. regex expression in java also in query
15. program to toggle given String 'ABC5656ghghT' with complexity o(1)
16. Hash Map internal working.
17. What is deadlock and how to avoid it
18. Difference between HashMap and HashSet
19. How to create thread. Which is best one
20. Difference between wait and sleep
21. Sigleton class
22. SpringbootApplication life cycle

**Susanta Kumar Datta**

**FIRST ROUND**

1. Brief introduction about yourself.  
1. Maximum subarray sum problem(Brute force to optimised approach and discussion on time complexity and space complexity).  
2. Array vs ArrayList vs LinkedList(deep discussion on this).  
3. If ArrayList is dynamic till what time will elements be inserted in it?  
4. Singleton class definition(asked to write the code for this).  
5. Discussion on each lines of Singleton class(for eg- use of static, etc.)  
6. Discussion on caching(because I had use it in my project).  
7. Deep discussion on caching vs database(why caching is better and how).  
8. SQL vs NoSQL(also in respect of some real life use-case scenarios which will be better)  
for eg:- For bank transactions which type of database is used.

**SECOND ROUND**

1. Brief introduction about yourself.  
2. Find a leader element in an array(Brute force to optimised approach and discussion on time complexity and space complexity).  
for eg:- [12,3,4,7,6] -> O/P -12,7,6  
3. Why string is immutable?  
4. String builder and String buffer.  
5. System design of URL Shortening(long discussion on this)  
for eg:- requirements to build this, which architecture to follow, which database is to be used, at end what methods are used for url shortening)  
6. Asked to explain my previous project.  
7. Microservice vs monolithic(also in respect of some real life use-case scenarios which will be better)  
8. Why the founder of java has kept String immutable as whole? Because in case of C language, String can be immutable or mutable.

**Sathish Kumar Ramamurthy (Simer and Yogi)**

1. Self Intro and Current Project details
2. Marker interface - real time implementation example
3. Load balancing Vs reverse proxy
4. Sleep Vs wait
5. Microservices, advantages and disadvantages
6. Real time implementation of microservice
7. Deadlock and how to avoid deadlock
8. code for Situation to get deadlock
9. Program : To Print odd numbers first and even numbers last and arrange in ascending & descending order same – java 8
10. Dialect in springboot
11. crossorigin in springboot
12. Can we exclude packages from component scan? if yes, how to do
13. Program : To Print second largest number in java 8
14. Hashmap and concurrent hashmap
15. Second thread need to start after first thread completed - how will you do
16. Lock interface in java
17. Views in SQL
18. Race condition in java
19. Normalization and types
20. Cluster index
21. Group by syntax in java
22. Program: aaabbbbcccdd-> 3a4b3c2d. Write a code to get output (count of character occurrence)

**Sasmita Sahoo (Mahendran and Sudhamai)**

1.Tell about your tech stack, technologies you have worked on and about role in previous project.

2.What is Encapsulation

3. Polymorphism run time and compile time.

4.Example of runtime polymorphism

5.How hashmap internally works.

6.

class A{m1();}

class B extends A{m2();}

A a=new A();

a.m1(); // output

a.m2();

B b=new B();

b.m1();

b.m2();

A ab=new B();

ab.m1();

ab.m2();

7.set coding questions

8.Coding questions related to Inheritance

10.What is functional interface

**Richa Sharma (Anshul and Nitish)**

1. Multiple questions with respect to the past project which I have worked on.
2. What is more efficient to use Kafka or REST api during communication within Microservices?
3. Disadvantages of Microservices?
4. How we maintain the microservices?
5. How redis works why it is faster than other noSQL DB?
6. Difference between NoSql and Sql db?
7. Does NoSql maintain Normalizations like sql? How is primary key and indexing in NoSql works?
8. Difference between POST and PUT
9. How Kafka works? do we require any authentications for kafka?
10. Q. Write a function which takes an array of n integers where array[i] is in the range of [1, n], returns an array of all the integers in the range [1, n] that do not appear in the array.
11. :where n is the length of the array.
12. Example:
    1. getDisappearedNumber([4,3,2,7,8,2,3,1]); // [5,6]
    2. getDisappearedNumber([1,1]); // [2]

**Richa Sharma (Shilpa BR)**

1. What stack you worked on previous projects?
2. What is difference between Microservices and RestFul webservices?
3. Rate yourself in Java.
4. What is difference between error and exception?
5. Define custom exception? how to create one?
6. Abstraction vs Inheritance?
7. Example of abstraction? in what ways abstractions are achieved in java?
8. abstract vs interface?
9. Q.Given an integer array t with a size of n which contains only numbers between 1..99. Print numbers that appear more than once.
10. For example:
11. For t=[3,6,8,3,7,7,7,9,1,3] the expected output is 3,7
12. For t=[3,6,8,5,7,7,7,9,1,3,5] the expected output is 7,3,5
13. The computational complexity of the optimal solution is O(n) with n being the length of array 't'.
14. public static void printRepeatedNumbers(int[] t) {
15. //TODO
16. }
17. Q.Design Question- The client wants an api where he gives parameter of what DB connection it wants, and we need to provide that connection to respective request of client. There are multiple DBs like Mongo, oracle, cassandra. How you will design it?