SASVEN:

1. Prime numbers print.
2. Difference between @Transaction and @Transactional.
3. HashMap and Concurrent HashMap.
4. What is the difference between Array List and Vector?
5. Which Java Technologies Used?
6. Project Modules Explain.
7. Aggregation vs Composition.
8. Spring vs Spring Boot.
9. Conversion of List to Map and vice versa using stream Api
10. Spring Boot Security.

Landmark Group ( Watsonite )

1. List<> ----------🡪 It is ThreadSafe or not.
2. {0, 1, 1, 0, 2, 1, 0}



Sort it as



{0, 0, 0, 1, 1, 1, 2} 🡪 Without using Sorting.

1. Array = { 1, 2, 3, 4, 5, 6}

N = 2

Then first two elements go to last then Result array is:



{3, 4, 5, 6, 1, 2} 🡪 Time Complexity



🡪 Can we override Static Method.

🡪 Can we Override or overload the Main Method.

1. class A { class B extends A {



Method1(); m2;



} A a = new B();

a.m2(); // Here m2() is True or False.



}

1. @SpringBootApplication and @EnableAutoConfiguration
2. How we can class in Java as Singleton.
3. Ex: List<Integer> list = new ArrayList<>();

List.add(1), (2),(3) 🡪Concurrent Exception occurred.

**Dig Digital**

1. Array list = “Chris Hayes Thompson”;

Count Each elements in the list using Array.

1. Partial Views and Views in MVC

View: Basically contains a complete maseup which may contain master view( or Master Page).

Partial View: Is only apportion of page or small markup which don’t have Master page

It is User controlled and can be used at more than one place (i.e it is reusable)

1. CTE in MVC.

Common Table Expression 🡪 It is temporary named result Set that you can reference within CRUD Statement.

1. TempData in MVC

TempData is used to transfer the data from View to Controller and vice versa

Or from one controller to another Controller.

=====================================================================

**NeoSoft:**

1. StringPool & String Constant Pool.

Why String is immutable. Other immutable Objects?.(Wrapper classes).

1. Abstact Class and Interface.
2. Why interface and when abstract class already there.
3. How make loosely coupled and tightly coupled class in Core Java not in Spring.
4. Is-A and Has-A relationship.
5. Java 8 Features.
6. HasMap 🡪 Complete bucket concept.
7. Difference between JPA Repository and Crud Repository.
8. With different Return type 🡪 overload or override which is possible.
9. Modifiers all 🡪 can be override protected method.
10. JIRA(Practical Knowledge)
11. Project Module Explanation.
12. @Configuration class can be alternative of application.properties or yml
13. Difference between @Contoller and @RestControoler 🡪 Tell with Example where we can use that and which scenario.
14. Flow of class creation of Spring Boot like DB creation.

=====================================================================

**TechMahindra:**

1. Agile Methodology meeting and how to do meeting there.
2. Oracle DB 🡪Procedures, View, Join

Merge as one object of column.

EX: Column are id, name, rollno,. Merge all as one value.

1. Inner class, anonymous class

Write and show how to create and use of that.

1. Volatile Modifier
2. Stream Programme to sort the List based on age.
3. Custom property of application.properties.
4. HashMap internals.
5. Log4J and how Logger you see of your application.

==================================================================

1. Difference between sleep () and wait ().
2. Object class methods.
3. Conversion of GSON and JSON.
4. Collections: Which is fastest for storing and retrieving.
5. Multithread creation ways.
6. What is servlet.
7. Monolithic vs Microservices.
8. Spring Bean
9. Garbage Collector methods and how to call.
10. Which Java Garbage collector is called.
11. What are methods of Thread class.
12. Profiles and Actuators
13. HashMap internals.
14. Why collections give concurrent Modifications error.
15. Why String is immutable.
16. Spring MVC architecture
17. Spring Lifecycle, Scope, Custom Properties.
18. Thread methods
19. Functional Interface and Lambda expression
20. Bean scope -🡪 singleton(Default), prototype( Where to write)
21. Difference between Spring, Spring MVC, Spring Boot
22. Project module explanation which one you develop
23. Create MVC Controller Class.
24. String Constant pool and why immutable
25. . equals and == operator
26. Where you used the MVC in project
27. Custom application.properties file
28. Custom Exception handling.
29. array List and vector difference
30. main method can be private?
31. private method if override then what error it will give.
32. What is Singleton.
33. Factorial of a Number using recursion.
34. Singleton Program
35. String str = “Java Technologies”

Write a program to find the occurrence of above string

1. Create an ArrayList, add some names to that ArrayList. Print Arraylist.
2. Program for Executor service and where you implement in your project
3. Difference @Service and @Repository
4. Difference @Controller and @RestController
5. @Bean and @ComponentScan
6. What is Lambda
7. What are terminal Operations in Stream API
8. What are the Java 8 features
9. What is @FunctionalInterface
10. Write a stream API program
11. What is Dependency Injection
12. What is IOC. How its work
13. Types of Dependency Injection
14. Define Constructor injection.
15. How many ways a Thread can create?
16. Explain @Autowired annotation.
17. @Value and @Inject annotation
18. What is annotation.
19. What is servlet, Cofiguration file name used for servlet
20. Full form of CICD. What is CICD
21. What is PIPELINE in Jenkin. How it works.
22. If your application deploy in AWS, its working fine but next day its not working. How you resolve this problem.
23. What are the types of IOC container.
24. Two objects create in applications:

User u1 = new User();

U1.setEname

U1.setPhno

User u2 = new User();

U2.setEname

U2.setPhno

U1.equalsto(u2)

1. Complete flow of Application creation
2. User where uses the Microservice Architecture
3. Collection types and basic idea which are part of which
4. Different Bean creation annotation.
5. @Bean annotation and use ,can we use at class level
6. Benefits of microservices based architechture
7. Which independency injection is by default
8. Spring Aop, what is pointcut, jointpoint
9. Failsafe concept

**Soc-GEN:**

1. String pool, and internals
2. Different annotations, what if they are uses in different classes
3. Which letter is occurred maximum times in string given.

Ex: String sz = “aabbbacbbbb”;

Ans: maximum occurrence b=7 times

1. SpringBoot security(Each and every details) configuration etc.
2. How to send HTTP status-🡪 learn all
3. What are the Http methods
   1. Class A{ class B{

} }

What are the possibilities of objects creation and if method calls then which method is called.

===============================================================

**Happiest Mind :**

1. Marker interface--- Interface which does not have any methods or constants inside it. It is also called tagging interface. It provides the run-time information about the objects. Ex-🡪Serializable MI, Cloneable MI, Remote MI

Cloneable interface: if we try to clone an object that does not implement this interface, then JVM throws CloneNotSupportException. So it is indicator to the JVM that Object.clone() method is calling.

1. Functional Interface. Ex🡪 of functional interface
2. Data creation and join of data base table
3. Comparator/ comparable
4. String reverse program
5. Springboot security
6. Lambda Expression
7. While, do-while difference . how send 404 status code
8. Ex of Lambda interface : lang.Runnable()

util.comparator

io.FileFilter

util.concurrent.callable