**Core Java Questions:**

1. How-does-concurrent hash map-work-internally [<https://stackoverflow.com/questions/11793067/how-does-concurrenthashmap-work-internally>]
2. Can you implement own LRU Cache [<http://www.geeksforgeeks.org/lru-cache-implementation/>]
3. Latch & Cyclic barrier [<https://stackoverflow.com/questions/4168772/java-concurrency-countdown-latch-vs-cyclic-barrier>]
4. Blocking queue
5. What-are-fail-safe-fail-fast-iterators-in-java [<https://stackoverflow.com/questions/17377407/what-are-fail-safe-fail-fast-iterators-in-java>]
6. What-is-the-actual-use-of-interface-in-java [<https://stackoverflow.com/questions/4436087/what-is-the-actual-use-of-interface-in-java>]
7. How does hash map work internally
8. when-to-use-volatile-and-synchronized [<https://stackoverflow.com/questions/9851133/when-to-use-volatile-and-synchronized>]
9. What Are OOPs Concepts in Java ?
10. wait (), notify ()  and notifyAll ()
11. why-use-a-reentrantlock-if-one-can-use-synchronizedthis [<https://stackoverflow.com/questions/11821801/why-use-a-reentrantlock-if-one-can-use-synchronizedthis>]
12. <https://stackoverflow.com/questions/20212440/immutable-objects-and-hashmap-keys>

**SQL**

1. SELF JOIN/ INNER JOIN
2. <https://docs.oracle.com/cd/B28359_01/server.111/b32024/partition.htm>
3. SQL Query

**Design Pattern**

1. Template Design Pattern
2. Factory Design Pattern
3. Difference between factory and abstract factory design pattern
4. Singleton design pattern
5. Fly weight design pattern
6. Observer design pattern

**Design Principle**

1. <http://javarevisited.blogspot.in/2012/03/10-object-oriented-design-principles.html#axzz4r7b13uz7>
2. DRY
3. SOLID

**Data Structure** :

1. converting-roman-numerals-decimal-lying-1-3999
2. <http://www.geeksforgeeks.org/find-elements-present-first-array-not-second/>
3. Convert String to Integer [Write logic]

**Spring** :

1. Why-Use-the-Spring-Framework-.id-130098.html
2. Dependency injection/IOC
3. A large file you have to find more frequent used word.
4. Self join in table [emp.salary>manager.salary]
5. Immutable collection
6. other java question

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

JAVA:

\* Collections: linked list, array list

\* What is a HashMap

\* Internal working of a hashMap

\* What is concurrent hashMap and internal working

\* Design patterns

\* How to define a singleton class

\* String, stringbuilder and stringBuffer - differences; memory allocation?

\* Normal forms? normalization- 1f, 2f, 3f

\* Types of joins

\* practical scenario with DB designing for banking

\* problem solving - find a quadruple subset of an array such that sum of 1st 2 nos. is equal to the sum of other 2

1. Given RAM of 8 GB, need to sort the file which is huge volume(100 GB)
2. Each thread need to print the array incrementally and sequentially
3. Write a code to detect and resolve loop in a linked list

            (solution: use Floyd’s Cycle detection algorithm or Hashing or Mark visited nodes)

      4. How oracle index works internally and advantages

      5. Spring bean life cycle

      6.Advantages of spring

      7.Modules of Spring

       8.what is Spring boot and advantages

Please find below the list of question asked:

1. Difference between Hash Map and Concurrent Hash Map? In detail (Explain rehashing)
2. Design a linked list and find the nth last node.
3. Detect if linked list has loop or not?
4. Discussion on different Data Structures (Specifically on Array List and Linked List (Space and time complexity and Thread Safe)
5. Design a list of your choice (Thread safe, No performance issues)
6. Immutability in java w.r.t to Reference data members
7. Design a system which displays  latest 100 stock price of N companies (Single Thread and Multiple Thread)
8. Few Questions on Multi Threading, Deadlock, Concurrency, Spring.
9. Questions on B-Tree
10. Design all database tables for 1NF, 2NF, 3NF and write SQL queries
11. These are the set of questions discussed with me.
12. Round 1:
13. Difference between different data structures (Linked List, Array, Stack, Queue)
14. Discussion on how to design your own Map.
15. How to design your own blocking queue.
16. Basic question on Collection and Threading.
17. Java 8 features and explain few of them.
18. Linux related commands and how to you use them
19. How to debug a tomcat server on a Linux machine
20. Round 2:
21. Java : Collection related scenario and use case of few collections.
22. JUnit
23. Java 8 features
24. Hibernate : I said I am not aware
25. REST : Basic commands and how will you design your application
26. Spring : How to configure your source code and test code separately for a production env
27. Basic concepts of Spring
28. Database : A query which uses HAVING clause.
29. Linux : Different networking commands and main focus on regular expressions
30. How will you design same linux command in java

First round  
1 design payment system for a ecommerce ..payment method cc DC or netbanking and diff banks.  
  
  
2 a query to find all the employees of dept   
3 do it in hibernate.  
4 three is stream of no running find middle element.

Second round.  
  
Exception  
Threading very general.  
Diff of map  
Java 8 features that also just straight no cross question

1) Design parking lot, handling concurrency when multiple cars enter/exit at same time

2) Atomic variables, executor, volatile, cyclicbarrier-usage/internal implementation

3) Spring @Configuration, @Component

4) ConcurrentHashMap - Java 8 internal implementation

5) Previous project - design and improvement

6) Singleton best practices, factory, abstract factory

7) Spring - DI, beanfactory

8) 2 arraylists - How will you come up with a unique merged arraylist.

9) Lambda expressions

Question List:

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

1) Diff between Comparable and Comparator , With Employee Class example , followed by some scenario on that.

2) Collection hierarchy, real time used collection ?

3) which collection you use for fastest lookup in list of objects? and why ?   // HashMap

4) Access modifiers in java? Indirect question using car and key scenario.

5) Exception in java?  Exception hierarchy?

6) How you create your own exception in java? then use in car and key scenario.

7) what is synchronization in java? Race Condition?

8) diff Sleep() & Wait()? again  use case with car and key scenario.

9) Stress Management kind of scenario question?

 - How i can finish given task in with in time? if my tech lead is not there?

- Prioritized your work on what bases ?

- new technologies, how can you make task work with it, if you don't know ? your approach for it?

10) my last challenging work and how i handled it?

11) Angular 2 related some discussion => part of new technologies i show my interest then he asked about it.

Questions :

· Internal working of TreeSet

· Comparable , Comparator for custom objects

· compareTo method implementationof String class

· Finding second highest repeated/duplicated no in an array

· difference between HashMap, ConcurrentHashMap.

· How CHM works internally

· Immutable class

· Singletone class implementation, how multiple objects gets created in multithreading environment for some of Singletone implementation

What to do, If I want to use any custom object as a key in Map.

2. what if,only equals method is overridden.

3. what if,only hashcode method is overridden.

4. can you write the code for overriding the equals and hashcode.

5. How hashMap works.

6. How ConcurrentHashMap works.

7. Difference between hashMap and ConcurrentHashMap.

8. Tell some of usecases when you will use which collection class like (list,set,map).

9. Write the code for singalton class.

10. What is the use of ReentrantLock.

11. Advantage of ReentrantLock over synchronized block and method.

12. what is readwrite lock.

13. why to use Executor framework.

14. How to create threadPool and what is its advantage.

15. difference between submit and execute method.

16.Design for Cache and Parking lot in java.

17. Association between Employee and Address table in database (writing the code for one to one and one to many mapping).

18. NIO package advantage over IO package.

Java:

1. Internal working of ConcurrentHashMap
2. write your own immutable class (Most accurate Answer: perform deep-copy while object initialization and in getter methods return clones for the objects of dependent classes those are not immutable)
3. find nth last element in a custom single linked-list while iterating only once
4. write your own method that will take a map and return a synchronized map
5. alternative approach of synchronized block/method
6. what if we only override hashcode and NOT Equals method
7. internal working of hashmap
8. can you start a thread twice
9. AtomicInteger Implementations
10. implementation of CopyOnWriteArrayList
11. Re-entrant locks
12. implement multithreading to read multiple files from a location and process it
13. difference between singleton implementation in Spring and usual singleton pattern implementation
14. annotation based spring implementation
15. different ways to initialize spring beans
16. invoking a method while a bean is initialized
17. Object oriented design principles   
      
    More stress on Java collections and multithreading

SQL:

1. find the customer with multiple accounts
2. what is a materialized view(view vs materialized view)
3. remove duplicate entries is a table
4. given a scenario of banking, design  your database entities
5. Round 1:
6. Difference between different data structures (Linked List, Array, Stack, Queue)
7. Discussion on how to design your own Map.
8. How to design your own blocking queue.
9. Basic question on Collection and Threading.
10. Java 8 features and explain few of them.
11. Linux related commands and how to you use them
12. How to debug a tomcat server on a Linux machine
13. Round 2:
14. Java : Collection related scenario and use case of few collections.
15. JUnit
16. Java 8 features
17. Hibernate : I said I am not aware
18. REST : Basic commands and how will you design your application
19. Spring : How to configure your source code and test code separately for a production env
20. Basic concepts of Spring
21. Database : A query which uses HAVING clause.
22. Linux : Different networking commands and main focus on regular expressions
23. How will you design same linux command in java

**Morgan Stanley Interview questions**

**Round 1:**

1. Movie ticket booking system.
2. HashMap based scenario questions
3. Reversing a linked list
4. Inned Join Queries

**Round 2:**

Spring Application flow

Spring injection related questions

Scenarios for spring scopes

Bean life cycle

Sorting Scenario where data is 10 GB and RAM is 1 GB only

Max\_Heap and Min\_Heap scenarios

Online file uploading and downloading application designing and data structure

**Round 3:**

1. Parking lot designing
2. Collections Scenario based questions
3. String sorting and programing
4. Memory Management and Garbage collection
5. MongoDb Queries

**Round 4:**

1. Banking system designing using blocking queues and multithreading
2. SQL queries for banking system
3. Scenarios based on using collections
4. Concurrent collections and Various iterators used
5. Spring boot application flow
6. Webservices related questions

**Round 5:**

1. Applications on which I have worked.
2. OOPS principles
3. Logistics transport system designing
4. Singleton implementations
5. Immutable class implementation
6. Garbage collection, memory model, how objects move from young gen to old gen in memory
7. Implementation of collision algorithm for HashMap
8. How to catch exception in main which is thrown in child-thread
9. References (Soft and weak)
10. What is abstraction in java? realtime example.
11. Use of serialization
12. Problems:
13. Finding the middle node of a LinkedList
14. Finding if two LinkedList are merged or not
15. Finding all non repeating character in a string (may ask to write JUnit)
16. Database design for Facebook
17. Given a hospital scenario, what all REST services you expose
18. REST service for specific requirement like, list all patients data under one doctor/ dept
19. Query optimization

Rahul Baid (Java)

**Core Java Questions:**

1. How-does-concurrent hash map-work-internally [<https://stackoverflow.com/questions/11793067/how-does-concurrenthashmap-work-internally>]
2. Can you implement own LRU Cache [<http://www.geeksforgeeks.org/lru-cache-implementation/>]
3. Latch & Cyclic barrier [<https://stackoverflow.com/questions/4168772/java-concurrency-countdown-latch-vs-cyclic-barrier>]
4. Blocking queue
5. What-are-fail-safe-fail-fast-iterators-in-java [<https://stackoverflow.com/questions/17377407/what-are-fail-safe-fail-fast-iterators-in-java>]
6. What-is-the-actual-use-of-interface-in-java [<https://stackoverflow.com/questions/4436087/what-is-the-actual-use-of-interface-in-java>]
7. How does hash map work internally
8. when-to-use-volatile-and-synchronized [<https://stackoverflow.com/questions/9851133/when-to-use-volatile-and-synchronized>]
9. What Are OOPs Concepts in Java ?
10. wait (), notify ()  and notifyAll ()
11. why-use-a-reentrantlock-if-one-can-use-synchronizedthis [<https://stackoverflow.com/questions/11821801/why-use-a-reentrantlock-if-one-can-use-synchronizedthis>]
12. <https://stackoverflow.com/questions/20212440/immutable-objects-and-hashmap-keys>

**SQL**

1. SELF JOIN/ INNER JOIN
2. <https://docs.oracle.com/cd/B28359_01/server.111/b32024/partition.htm>
3. SQL Query

**Design Pattern**

1. Template Design Pattern
2. Factory Design Pattern
3. Difference between factory and abstract factory design pattern
4. Singleton design pattern
5. Fly weight design pattern
6. Observer design pattern

**Design Principle**

1. <http://javarevisited.blogspot.in/2012/03/10-object-oriented-design-principles.html#axzz4r7b13uz7>
2. DRY
3. SOLID

**Data Structure** :

1. converting-roman-numerals-decimal-lying-1-3999
2. <http://www.geeksforgeeks.org/find-elements-present-first-array-not-second/>
3. Convert String to Integer [Write logic]

**Spring** :

1. Why-Use-the-Spring-Framework-.id-130098.html
2. Dependency injection/IOC

The following are the interview questions by ( Hiring Manager ) Rajeev Sharma

1st round:

1. Explain concurrent constructs like count-down latch. it is scenario based question like given multiple threads and few threads to be executed after certain (say, producer) threads are executed.

2. Explain how concurrent hash map works, internally? How to control how many threads can perform concurrent reads/writes and how to configures it (ConcurrentHashMap constructor)

3. New features introduced in Java 8? Explain any main 3 with examples

4. Design Distributed cache? scenario based question. How communication happens between nodes when there are updates.

5. Do you have any questions?

2nd round:

1. Job responsibilities

2. Spring MVC context listener interface and its uses?

3. Few basic questions on spring, they were quite straight forward,

4. Do you have any questions? They explained the project details.

L1-Praveen Tiwari(F2F)

1)What are the collections used in last project.Then explain each scenarios why and when these collections are used.

2)Internal implementation of HashMap.How to synchronized the HashMap.Concurrent Hashmap implementation.

3)What happen if we do not override hashcode while key as an object in HashMap.

4)Singleton Class and immutable class(detail implementation).

5)List contains 100 names.Using six thread print the names(Names should not be print twice).

6)How JVM works.How garbage collector works.What is the concept behind using young gen and old gen.

7)How Spring beans works.

8)How URL authentication works.

9)Design BookMyShow.

L2-James(Telephonic)

1)Comple design of last project.

2)How Akka is different from Multithreading.Akka advantages.

3)How database behaves in multi threading  environment.

4)Acid Properties.