Dear Candidate,

Below are few of the questions on JAVA.

4 principles of OOPS concept.

Inheritance.

Abstraction vs Encapsulation.

How to make a class Immutable. Then questions continue on why to make class final for Singleton.

How to make a class Singleton.

What are the different ways to break Singleton. (And : Multithreading , Cloning, Serialization, Reflection)

What is Cloning?

Deep cloning and shallow cloning? (with example)

What is serialization?

Abstract vs Interface.

For Collections

Internal working of HashMap, HashSet,

Which data structure would to prefer for Queue implementation?

Synchronized Collections.

Multi-threading

Simple multi-threading and executor framework.

Different implementations of Thread Pool.

How to stop a thread.

Wait vs Sleep.

Wait and notify method.

Why wait and notify methods are in Object class and not in Thread class.

Why wait and notify called from Synchronized block.

Spring Core.

Different type of dependency injection. Which is better in which scenario?

Different scope of beans.

Spring Singleton vs normal singleton.

Look-up methods in Spring.  
  
Rest Web services.

Which library did you use.

Different HTTP methods.

what all parameters decide which service methods will be called.

These are on OOPS, Problem solving, Logical, Execution Framework and Designing questions and Too many theory question.

Core Java

Oops concepts

Collections (Mainly Map)

Programs (Some coding questions)

Asp angular js(how does the framework work)

Union (what is row number,rank)

Indexes in SQL

Candidate 1

1) About immutable classes.

2) About Hashmaps.

3) Duplicate print from list of interger.

4) Concurrent Hashmap.

5) Inheritance and other oops concepts.

Interview discussion

Candidate 2   
1. Current project details, roles, responsibilities  
2. Have you work on rest?  
-- ans- just start working on it. Which package you have used for rest application .  
3. How you will design Make my trip ( user requested for flight details . make my trip will send request to no. Of service providers i.e indigo, spice jet, air India.  In response each service provider send response of flight details along with price.  
Make my trip will sort all response based on price in ascending order) how you will send request, how you will receive response object  from service provider ( exception scenario included ). Make my trip will start processing only after response from all service provider?  
Ans: request object will have following parameter i.e source place , destination,  journey start date, return date. Type of journey single/return, price range  
Response object will have flight details along with price, departure time , arriving time, duration.  
  
Implementation:  
Using executor threadpool request object will sent to service provider ( i.e. for indigo, spice jet, air India. 3 callable task which will work on request and send respone back in list as a flight details object ) list can contain multiple flight details object for requested date e.g. multiple flights running on same date for same service provider.  
each callable task will return future object. Make my trip will process on all future object to provide final output to user.  
Discussion on exception handling in above  
Discussion on if any service provider doesnt provide any response.  
4. Model -> service -> DAO-> third party application connect with database and return result or runtime exception if any. User will request user details based on user id?  
   Where you will handle exception scenario ?  
ans- service layer. Service layer will call to dao service if dao service return username which direct to user program. If no record found then dao will throw user not found exception . if any runtime exception will prapogated to caller method of service layer which will log exception details and send user friendly error to user program.  
  
5.have you work on spring, git?--no.   
Ans: i want to learn spring . work on svn for version controlling.  
6. Which software model do you work ?    
    Ans : Agile. discussion on current project developement  
7.What is coverage of testing? Have you work on junit?  
--no. discussion on current project testing .

Candidate : 1

1) Employee class has list of employees under that employee. Find the youngest employee in the organization. E.g. there is an employee object having information of manager and list of TL directly under him. Each TL object has developers under him. Find the youngest employee.

2) There is a car selling organization. That organization has many branches. Every day in the morning head office sends a list of registered car to all the branches. Those branches send back list of those car along with the best price in which they can sell. At the end of the day. Head office will decide to which branch that car will be given based on the best price. Write a code which will help head office to identify the branch who gave the best price for that car.

3) Sentence without spa e is inserted, output should be proper sentence. E.g. 'tomorrowisfriday' this will be input , output should be 'tomorrow is Friday'

Candidate : 2

1.      How is Abstraction achieved in Java?

2.      Discuss how do you decide when to use Interface or Abstract Class?

3.      Write the code for creating a thread-safe Singleton class.

a.      Now improve it to be much more performant.

4.      What does volatile do?

5.      ***Problem***:- Given an Employee {ID, Name, Salary} Class, how do you find the lowest ID employee ensuring uniqueness of ID.

a.      When you implement a TreeSet of Employee {ID, Name, Salary} Class, what modification will you do to Employee?

b.      Will you implement hashCode() method?

6.      ***Problem***:-

Given :

public Class Test{

                public int i;

                public int increment(){

                                return i++;

}

}

Questions:

a.      Is this code thread safe?

b.      How will you make it thread safe?

7.      ***Problem***:- Implement a recursive method ‘reverse’ that takes in a string and returns its reverse:

Public String reverse(String s);

8.      ***Problem***:- Implement the method sum() in Class A Given method in Class B.

public Class A {

                public int sum(List<B> list){

//returns sum of all int returned from call to

//method heavyProcessing() of each

//object of B in the list

}

}

Class B {

                public int heavyProcessing(){

//A very processor intensive method

}

}

Candidate : 3

Design a program that would compute a price of a machine that many have any no of parts. And the price of the parts are fetched from vendor system.

Focus was on the oops concept used, class design , fault tolerance and performance.

Rest he asked on singleton design pattern. Abstraction - interface and abstract classes.

Candidates : 4

* Executor framework + Threadpool working
* Rest API, when to use get , post
* How to return a list in JSON response
* Strategy and factory design pattern
* What to return in case of bad request for treat response
* Spring dependency injection

**Apart From That :**

Topic to prepare:

4 principles of OOPS concept.

Inheritance.

Abstraction vs Encapsulation.

How to make a class Immutable. Then questions continue on why to make class final for Singleton.

How to make a class Singleton.

What are the different ways to break Singleton. (And : Multithreading , Cloning, Serialization, Reflection)

What is Cloning?

Deep cloning and shallow cloning? (with example)

What is serialization?

Abstract vs Interface.

For Collections

Internal working of HashMap, HashSet,

Which data structure would to prefer for Queue implementation?

Synchronized Collections.

Multi-threading

Simple multi-threading and executor framework.

Different implementations of Thread Pool.

How to stop a thread.

Wait vs Sleep.

Wait and notify method.

Why wait and notify methods are in Object class and not in Thread class.

Why wait and notify called from Synchronized block.

Spring Core.

Different type of dependency injection. Which is better in which scenario?

Different scope of beans.

Spring Singleton vs normal singleton.

Look-up methods in Spring.  
  
Rest Web services.

Which library did you use.

Different HTTP methods.

what all parameters decide which service methods will be called.