Top 5 Problem Solving Questions for Java Interviews

Problem solving questions are new way to test engineers on interviews.

JAVINPAUL AND SOMA

NOV 25, 2023 • PAID

Hello friends,

Happy Thanksgiving and to celebrate the occasion I am also giving away free coupons to my Udemy courses. If you need one just comment the name of the course in this article and I will send you the coupon code. I am also offering a 30% discount on the annual plans on black Friday and Cyber Monday, just click this link to **subscribe now for just $35 per year**. (Only valid till 2nd December)

And now let's start with today's topic, problem-solving questions on interviews. I have been in this industry for more than 20 years now and I have given and taken 100s of interviews. Over the years I have also seen how interviews have changed from the early 2000s to the 2010s and the last few years.

There was a time when you could crack a Java interview by just knowing a few questions like ArrayList vs Vector, the internal working of HashMap, and Iterator vs Enumeration but that time has gone, just knowing API and concepts is not enough, you also need to be good at coding.

By the way, if you are preparing for Java interviews you can also check my **Java + Spring Interview + SQL Bundle on Gumroad**, use code **friends20** to get a 20% discount also

Coding problems are now making the first few rounds to screen candidates who cannot code but there is another interesting change that is making interviews more challenging than before, the introduction of problem-solving questions.

Companies now want to see how you apply your knowledge and skills to solve a real-world problem that you normally face in your day-to-day job. To be honest this is much better than asking you to invert a binary tree as this not only shows your practical skill but also is something a candidate cannot fake, it's a great way to see the real experience.

Questions like **What does it mean to make REST APIs stateless?** are a great way to judge how much a candidate knows. If he can mention the benefit stateless API brings and how you need to pass everything on the request to process that request is a great way to judge his experience.

Similarly, what would you do if you got OutOfMemoryError in your application is a great way to judge how you handle such a pressure situation. You can see how the candidate replies like taking a dump for debugging or restarting the app to immediately work, how does he analyze the situation? Is the error due to a sudden surge in data or a gradual increase? or is it a memory leak?

In this article, I Am going to share 5 such questions to both practice and learn how to solve problem-solving questions for tech interviews.

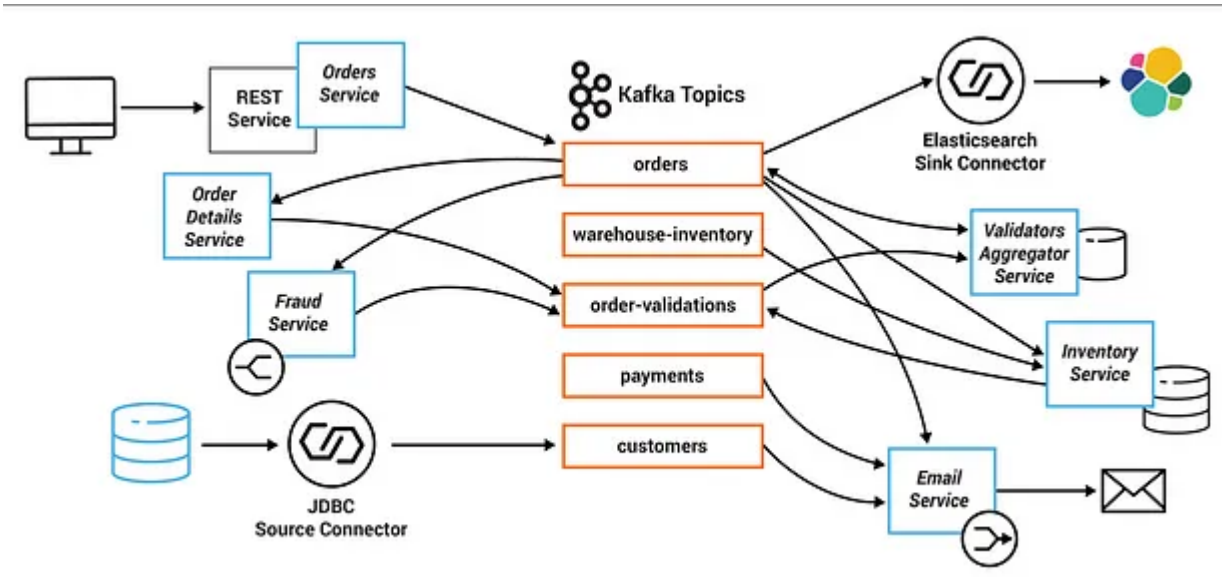
Here are some scenario-based microservices questions that are commonly asked to 5 to 10 years experienced developers:

1. Imagine you are working on a Microservice that's responsible for processing orders. However, due to some issues, it's currently down. What would you do to ensure that orders are not lost and can be processed once the service is back up?

Answer: This question is based on how you design systems so that they are not closely coupled. One possible solution would be to **implement a message queue** between the order creation service and the order processing service.

Orders could be stored in the queue until the processing service is back up, at which point they could be picked up and processed. For message queue, you can either use RabbitMQ or Apache Kafka, both allow you to create asynchronous Microservices that are not hard dependent on each other.

Here is what a **Microservice architecture with Apache Kafka** looks like:



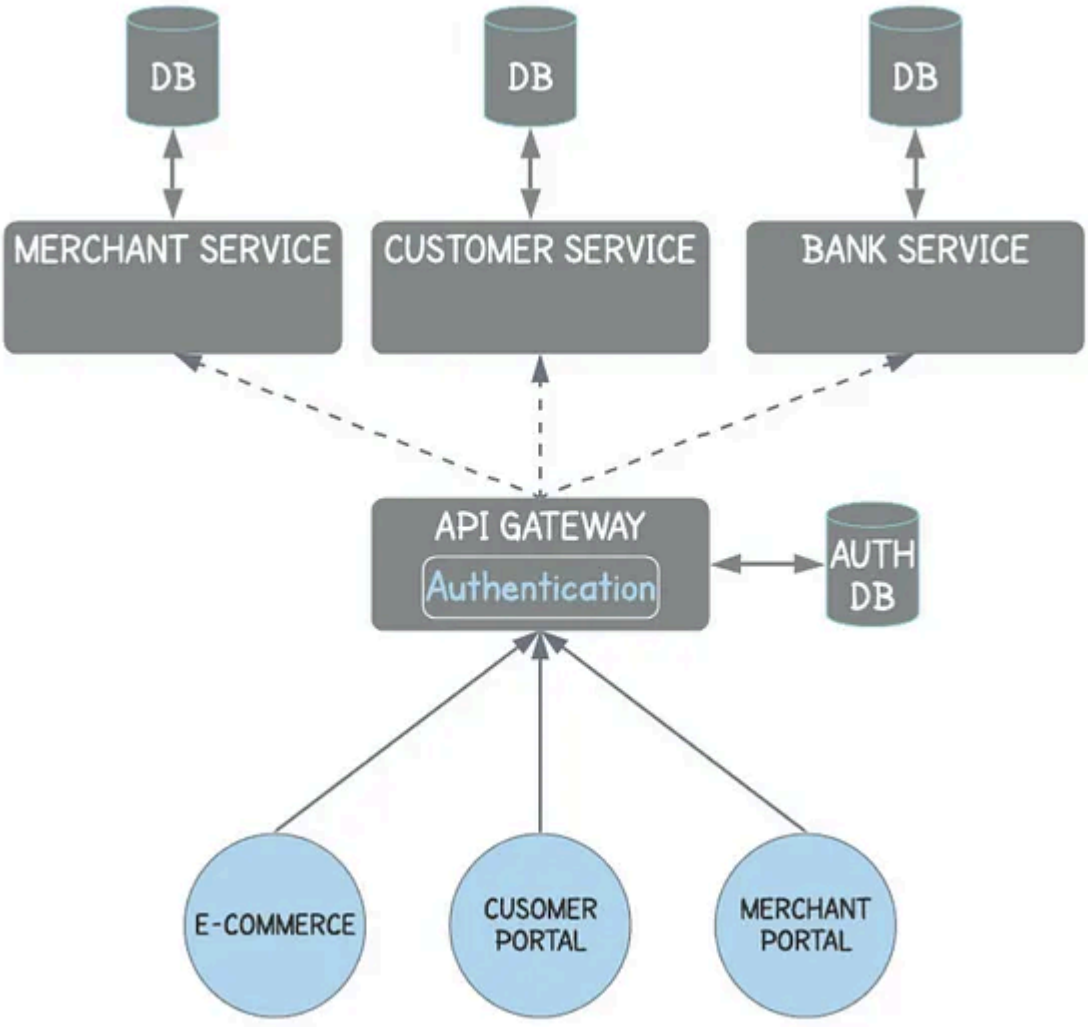
2. Suppose you have a microservice that's responsible for user authentication. How would you ensure that the service can handle a large number of requests and is highly available?

Answer: This question tests your design skill for designing a scalable and robust system that can handle millions of requests. One possible solution would be to **use load balancing and clustering.**

The service could be deployed on multiple servers, with a load balancer distributing incoming requests among them.

Additionally, the service could be **designed to be stateless**, meaning that each request can be handled independently without requiring access to a shared resource.

You can also use the **API Gateway design pattern** to implement user authentication in Microservices architecture.



3. Imagine you are working on a microservice that's responsible for generating reports. How would you ensure that the reports are generated correctly and efficiently while minimizing the impact on other services in the system?

Answer: This question is also similar to the previous question. One possible solution would be to **use caching and batching.** The service could cache previously generated reports and reuse them when possible, reducing the need to generate new reports from scratch.

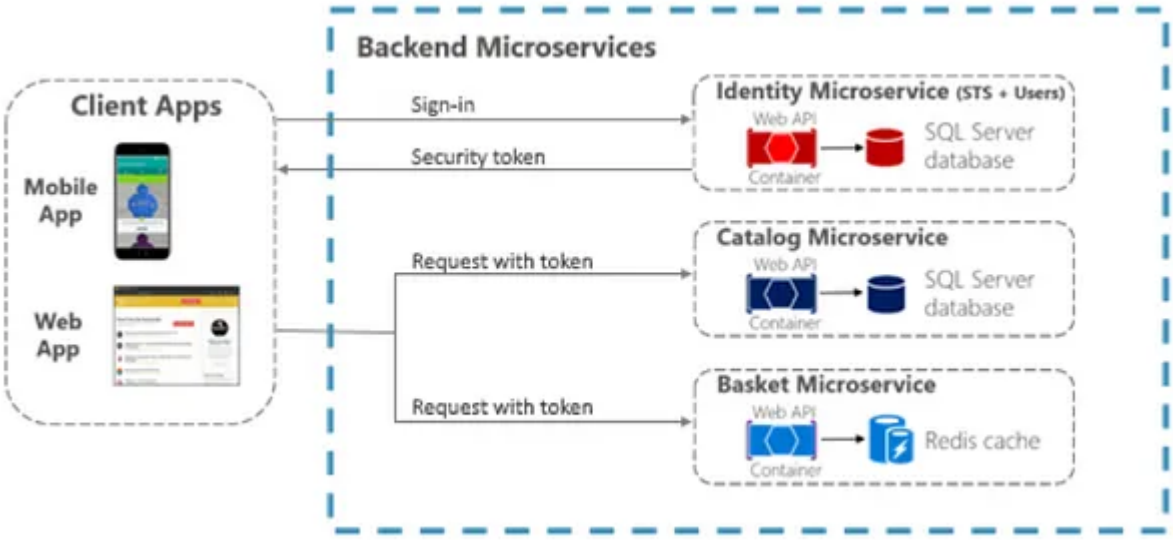
Additionally, the service could use batch processing to generate reports in batches, rather than on-demand, further reducing the load on the system.

4. Suppose you have a microservice that's responsible for handling payments. How would you ensure that the service is secure and that sensitive payment information is protected?

Answer: If you have done any microservice interviews then you may know that Payment processing is the interviewer's favorite topic as this requires transaction management and security. *You can't afford to lose data.*

One possible solution to this problem would be to **use encryption and tokenization**. Payment information could be **encrypted** before being transmitted to the service, ensuring that it's protected in transit.

Additionally, the service could use tokenization to store payment information in a secure manner, replacing sensitive information with non-sensitive tokens that can be safely stored and transmitted.



5. Imagine you have a Microservice that's responsible for handling user feedback. How would you ensure that the feedback is processed quickly and accurately, while also minimizing the risk of spam and abuse?

A: This question tests your skill in how you protect your system from abuse. One possible solution would be to use a combination of automated and manual moderation.

Automated moderation could be used to filter out obvious spam and abusive content, while more complex cases could be flagged for manual review.

Additionally, the service could implement **rate limiting** to prevent users from submitting large volumes of feedback in a short amount of time.

That's all in this **list of problem-solving questions for interviews**. If you like this post and want me to create more of these, please comment, your input is very valuable as I want to create content that is beneficial for you guys. You can also feel free to suggest any topic you would like me to cover.

And, since this is a Black Friday season, here are the best Black Friday and Cyber Monday discounts you can get for your learning and development. These are some amazing offers and you can save a lot of money now.

Here are the top 10 Black Friday deals for Software developers like us:

1. **85% OFF on Udemy Courses**
2. **50% OFF on Pluralsight Annual Subscription**
3. **50% OFF on Datacamp Annual Subscription**
4. **50% OFF on Codecademy Annual Plan**
5. **20% OFF on Educative Unlimited**
6. **27% OFF on Zero to Mastery Academy**
7. **ByteByteGo Black Friday 2023 Offer** (**30% OFF**)
8. **50% discount on 101 Blockchain Certifications**
9. **Udacity Buy one Gift one deal**
10. **45% OFF on Whizlabs Subscription and Courses**
11. **Baeldung Black Friday sale - Spring Course 30% OFF**
12. **Amazon Black Friday 2023 Deals** (**Bonus**)
13. **subscribe now for just $35 per year**

And, if you are preparing for Java interviews you can also check my **Java + Spring Interview + SQL Bundle on Gumroad**, use code **friends20** to get a 20% discount also

All the best for your learning and tech interviews guys, and happy Thanksgiving !!

4 Likes · 2 Restacks

← Previous

Next →

A guest post by

Soma
Java and React Developer

Subscribe to Soma

Comments

Write a comment...