Final Big Homework

> *帮助梳理知识点和重点。对Tech Mock 和Client interview都很重要。*

Maven

1. What is the **lifecycle** of maven? could you tell me the details ?
2. what is the difference between **package** and **install** in maven lifecycle ?

Linux

1. Reading: https://www.yuque.com/fairy-era/yg511q/oeybmv

Java

Java Core & OOP

1. Write up Example code to demonstrate the three foundmental concepts of OOP. (reference Code Demo repo as example)
   a. Encapsulation;
   b. Polymorphism;
      i. Override, Overloading.
   c. Inheritance;
2. What is Java **garbage collection**?
3. What is **Runtime/unchecked exception**? what is Compile/Checked Exception?
   a. also reading： https://www.yuque.com/fairy-era/yg511q/gldkel#b861aad7
4. What is the difference between **throw** and **throws**?
5. Could you give me one example of NullPointerException?
6. how does Java hashmap internally work?
7. Collections：
   a. reading: https://www.yuque.com/fairy-era/yg511q/ksp07m#daf18702
      i. Not necessary to read how to implement the data structure
8. Reading: https://www.yuque.com/fairy-era/yg511q/qzv31t#347111bb
   a. 一些OA要用到IO

External Libs

1. What is Guava? (https://www.tutorialspoint.com/guava/index.htm). No need to learn it. it is good enough to hear about it and know its role.
2. List and expain some methods from Apache Commons/Collections
    a. https://www.baeldung.com/apache-commons-collection-utils)
    b. https://developer.aliyun.com/article/896751
3. How to convert json format string to java object? and how to convert java object to json string? write some demo codes.
    a. ObjectMapper from Jackson which is default in Spring framework.
    b. Gson grom google.

Java 8

1. List Several Java 8 new features and briefly explain them.
2. practice stream API at least **3** times (vendor 面试会给个情景让写stream，工作上也会大量用到。)
    a. https://blog.devgenius.io/15-practical-exercises-help-you-master-java-stream-api-3f9c86b1cf82

Multiple Threading

1. What is deadlock?
2. how to create a new thread(Please also consider Thread Pool case)?
3. Difference between Runnable and Callable？
4. what is the diff between t.start() and t.run()?
5. how do threads communicate with each other?
6. What is Atomic classes? when do we use it?
7. What is the cocurrent collections?
8. What is keyword synchronized, how do you understand it?

9. what kind of locks you know?
10. What is future and completableFuture?
11. leetcode for multiple threading (paste your solution code to here)
    - a. 1114- Print in Order
    - b. 1115-Print FooBar Alternately
    - c. 1116-Print Zero Even Odd

Database

1. what is the difference between RDBMS and NoSQL?
2. In which cases you choose Nosql or RDBMS?
3. Do you know Cassandra and MongoDB?
4. LEFT JOIN & RIGHT JOIN
5. SQL design: given two tables, product and order, how will you design?
6. How to improve the database performance?
7. LeetCode (paste your solution code to here)
    - a.

595. BIG COUNTRIES
1757. Recyclable and Low Fat Products
584. Find Customer Referee
183. Customers Who Never Order
175. Combine Two Tables
176. Second Highest Salary
177. Nth Highest Salary
181. Employees Earning More Than Their M
196. Delete Duplicate Emails
180. Consecutive Numbers
184. Department Highest Salary
596. Classes More Than 5 Students

Design Pattern

1. List several design pattern names. and write the code for Singleton and simple factory design pattern.

Rest

1. What are HTTP request methods?
2. What is the difference between Put and Post?
3. What are HTTP Status codes? and explain them.(200, 201, 302, 400, 404, 500)
4. Could you tell any endpoints you developed?
5. Design one set of APIs for managing the customers history orders

Spring

1. What is the IOC?
2. What is Dependency Injection?
3. Different ways of DI
4. Types of dependency injection
5. What is the Scope of a Bean?
6. could you give me the exmaple for singelton bean scope
7. What are the differences between @RequestParam and @PathVariable?
8. What is AOP? could you list any annotations and briefly explain it? could give me one example how do you use it?
9. What is spring batch?
10. What is cron/spring task?
11. how to monitor you application? (spring actuator)
12. Spring validation?
13. how to handler exception in spring?
14. What is Dispatcher Servelet/Front Controller? please describe the flow.
15. Difference between @Component and @Bean
16. Difference between @Component, @Service, @Repository and @RestController?
17. What is @ComponentScan?
18. Authentication vs. Authorization
19. What is JWT? (header.payload.signature) and what kind of tool/dependency in java can generate and parse JWT?
20. What will happen after click the URL?
21. how to call api in java code?
22. I want to get the **customer's** history orders between 01/02/2023 and 02/07/2023, and also want to get the suctomer's payments in this account. Please design the API url.

23. Develop the API to CRUD the customer's histroy orders.
    1. Need to write the full codes including Controller, Service & ServiceImpl, Repository, Entity
    2. You need to design the payloads(request, and response body).
    3. Use the correct status code for each API
    4. Use ResponseEntity
    5. Use Validation for request body
    6. Write unit tests.
    7. Exception handling.
    8. use log

Test

1. Functional test & Integrtion test & Unit test & Regression Test & Performance/Load test
2. What's the test framework in your team?
3. Talk about Mockito, PowerMockito and give an example
4. What is SonarQube?

Microservice

1. Microservice architecture in your recent project?
2. Microservice components
3. How to do server discovery?
4. What is API gateway and what we can do in API Gateway?
5. how to communicate each other between services?
6. How do you use RestTemplate? write the code example to call an API
7. How to scale up your microservice?
8. Kafka

a. What is broker?
b. are you a producer or consumer in your recent project?
c. What is topic? could you give me one example of the topics in your recent project?
d. What is partition?
e. What is group consumer?
f. What is offset?
g. Explain the concept of Leader and Follower in Kafka.
h. If consumer has a big lagging, how to solve it?
i. do you have any even name in your recent project?
j. Differentiate between Rabbitmq and Kafka.

Docker

Reference: https://www.interviewbit.com/docker-interview-questions/#docker-basic-questions

1. Can you tell something about docker container?
2. What are docker images?
3. What is a DockerFile?
4. What can you tell about Docker Compose?
5. What is the purpose of the volume parameter in a docker run command?
6. How to get logs of a container?
7. list all of the docker commands you know, and briefly explain it.

Kubernetes

1. What is Kubernetes?
2. What is pod, what is service, and what is deployment?
    - a. reading: https://www.educative.io/module/665EM3i03GVvJNjwJ/10370001/6231939383558144
        - i. only need to know theories.
    - b. reading: https://www.yuque.com/fairy-era/yg511q/eu30ue
3. Remember the below Kubectl commands
    - a. 查看所有命名空间：

      `kubectl get namespaces`
    - b. 查看所有Pod：

      `kubectl get pods`
    - c. 查看指定命名空间的所有Pod：

      `kubectl get pods -n <namespace>`
    - d. 查看所有运行中的Pod：

      `kubectl get pods --field-selector=status.phase=Running`
    - e. 查看指定Pod的详细信息：

      `kubectl describe pod <pod-name>`
    - f. 查看Pod的日志：

      `kubectl logs <pod-name>`
    - g. 进入一个正在运行的容器：

      `kubectl exec -it <pod-name> -- /bin/bash`
    - h. 删除一个Pod：

      `kubectl delete pod <pod-name>`
    - i. 扩展一个部署：

```
kubectl scale deployment <deployment-name> --replicas=<num-replicas>
```

a. 滚动更新一个部署：

```
kubectl rollout restart deployment <deployment-name>
```

Tools

1. What is splunk?
2. What is Jira? what is sprint? what is scrum?
3. how long is one Sprint? tell me your daily routine.
4. what is agile?
5. Can you talk about CI/CD?
6. How can you monitor your application? Like QPS, network, CPU usage, Memory Usage, APIs error rates (https://grafana.com/)
7. What kind of tools do you use to ensure your code quality? (Sonarqube)

Soft Skills (discuss it with team, no need to gather all members, at least 3 ppl)

1. write one set of soft skills you will use for interview. The goal is to have a good communication and make the interview be happly.
2. If you didn't get the questions,
3. If you misunderstand the question,
4. If you cannot catch up the hints from the interviewee,
5. If you totally don't know the answer for the question,
6. If you don't know the good answer for that question, however you have basic knowledge.
7. If you will know the answer when you have more info/details,
8. When doing the coding, if the interviewer like communication,
9. When doing the coding, if the interviewer doesn't like communication,

10. When doing the coding, if the interviewer communicate a lot then suddenly don't want talk since he is busy.
11. If the interviewer try to drive you do the system design or algorithm coding,
12. If the interviewer has diffrent solution with the solution you are writting.
13. How to make the communication be happy, and flow is smoothie.
14. How to make the interviewer like you.

Coding (discuss it with team, no need to gather all members, at least 3 ppl)

1. What is your steps to do the coding interview
2. how to do clarification?
3. how to design test cases for solution?
4. Coding style
5. write code on different platform
    a. google doc (notice coding style)
    b. online compiler (https://www.jdoodle.com/online-java-compiler/)
        i. 注意常用的导包
        ii. 注意怎么debug
        iii. 注意stream的包怎么导进去
    c. Local IDE
6. Practice the below coding in different platforms. in main method, design the **test cases** for your solution and make sure it can pass.
    a. 347. Top K Frequent Elements
    b. 206. Reverse Linked List
        i. you need to design the ListNode
    c. 3. Longest Substring Without Repeating
    d. 43. Multiply Strings

7. FYI
    a. If input array is **sorted**, then
        i. Binary Search
        ii. Two Pointers
    b. If asked for all **permutations/subset**s, then
        i. Backtracking
    c. If given a **tree**, then
        i. DFS
        ii. BFS
    d. If given a **linkedList**, then
        i. Two Pointers
    e. If **recursion is banned**, then
        i. Stack,
        ii. Queue
    f. If must solve **in-place,** then
        i. Swap corresponding values
        ii. store one or more different values in the same postion
    g. If asked for **maximum/minimum subarray/subset/options**, then
        i. DP
    h. If asked for **top/least K** items, then
        i. Heap - PriorityQueue + lambda
    i. If asked for **common strings** then
        i. Map
        ii. Trie
    j. Else
        i. Map/Set for O(1) time & O(n) space
        ii. Sort input for O(nlogn) time and O(1) space

OOD and SD

1. Parking lot
2. tiny url
3. shopping cart