

### Larrie Shao - American Airline - 04/28/2025

1. Introduce yourself
2. coding:  
// Scenario:  
// You have a list of transactions from an e-commerce platform. Each transaction includes the following information:

```
// Transaction ID  
// User ID  
// Item ID  
// Quantity  
// Price per item
```

// Your task is to process this list of transactions to answer several business questions. You must use the Java Stream API to accomplish the following:

- // 1.Calculate the total revenue generated.
- // 2.Find the top 2 users who spent the most money.
- // 3.Identify the most popular item (the item with the highest quantity sold).
- 3. what is microservice
- 4. why need microservice
- 5. if we have monolithic application, has login logout account info features, how do you convert it to microservice
- 6. how do you perform security login in this flow
- 7. is jwt generated once or many times
- 8. vertical scaling vs horizontal
- 9. when to use rest api request between services
- 10. when to use message queue between services

### Colin Fan - BNY - 04/28/2025

1. Introduce yourself
2. what java 11 and java 17 or multithreading features you used in your project and how did you use it
3. what angular feature you have used
4. how do you send data between template and component
5. how do you pass data between components
6. how did you write multithreading code in your project (details)
7. what exception you need to handle when you write multithreading send request to diff endpoint
8. how Spring IOC work , all annotations and injection and bean types
9. why constructor injection
10. how do we do by name injections
11. by name vs by type
12. why constructor injection
13. have you used git, maven
14. what test frameworks you know
15. what exception do you need to handle when you write rest api which takes json as input content

### Larrie Shao - Capital One - 04/25/2025

1. Introduce yourself
2. what java version did you work on
3. new feature in java11
4. new feature in java 17
5. have you used

Pattern Matching for switch (preview), Sealed Classes, Text Blocks, Record classes

6. what is stream api
7. diff lambda expression and functional interface
8. what functional interface you have used before
9. diff functional interface and java interface
10. have you used parallel stream
11. have you used completable future
12. is java clone a shallow or deep clone
13. how to do deep clone
14. how to impl singleton in java
15. coding: input "DDDYYCC" => output "3D2Y2C"
16. what is SOLID principle
17. what is strategy design pattern
18. have you used spring aop , spring security and spring batch
19. have you used spring cloud
20. what is webflux
21. for rest api, do you impl code first or design schema first
22. what apigateway did you use in your project
23. what is circuit breaker
24. diff prototype bean scope and request bean scope
25. how do you handle global exception in spring boot
26. how do you ensure code quality
27. what's your ci/cd pipeline look like
28. nosql db exp
29. when to use mongodb and when to use cassandra
30. what aws service have you used before
31. how do you monitor aws service
32. frontend exp
33. which angular version did you use
34. how to send request from angular to backend
35. diff promise and observable
36. have you done frontend test / unit test
37. how to avoid sql injection
38. what is your biggest challenge
39. team size
40. daily work (agile scrum)
41. do you have kubernetes experience

Larrie Shao - Apple - 04/23/2025

Interview#1

1. Introduce yourself
2. Coding: return total stock profolio value given a csv file contains stock names and their quantities
3. Elasticsearch experience
4. How do you handle high concurrency
5. Have you used CompletableFuture
6. What thread pool have you used in your projects
7. Difference betw HashMap and ConcurrentHashMap
8. Difference betw ArrayList and LinkedList

Interview#2

9. What did your last project end
10. SDLC steps
11. How do you design RESTful APIs to get/create/update an object
12. Multithreading experience
13. When and why use multithreading
14. How do you handle race condition
15. Design the locking schema so that when a thread call method1(), it needs to until some other thread call method2()
16. Fail fast vs fail safe
17. Coding: Traverse binary tree in level order

18. Method override vs method overload
19. Sort a map by values
20. Explain how circuit breaker works

### Colin Fan - Swift - 04/22/2025

1. Intro
2. your project architecture
3. what's your project throughput / qps
4. where did you deploy your application , docker
5. what's the qps of your application
6. how did you fix production issues
7. what if you cannot find it from your log
8. how did you optimize the system performance(what issue, and how did you find it how did you fix it )
9. how did you optimize your db performance (story, business case)
10. any frontend exp
11. any python exp what python things you have down
12. any cloud exp, what cloud service you have done , or used
13. where did you use multithreading

### Sid - Walmart - 04/21/2025

1. Intro
2. coding: longest consecutive length of input unsorted array
3. sql coding:given order table and customer table, find largest price in 10 years and return price + customer name
4. how many ways to inject bean in spring and which one we use most
5. tell me new features of stream api
6. what are intermediate operations and what are terminal operation
7. what is functional interface
8. do we need @FunctionalInterface annotation?
9. how to prevent message failure in kafka
10. how to inject bean with same type
11. if we have microservices calling each other a -> b -> c , and if some of them are returning 500 / errors, what should we do

### Larrie Shao - Apple - 04/18/2025

1. self intro
2. project related questions
3. batch process experience
4. why batch process
5. elastic search experience, what problems, how to fix?
6. why latency using elastic search
7. Coding:

```

/*
/*
 *Find longest common prefix in an array of strings
 *["apple", "apples", "applepie"] -- apple
 *["airpod", "airtag"] -- air
 *["iPhone", "iPad", "innovation"] -- i
 */
*/
class Solution {
    public static void main(String[] args) {
    }
}

```

- a.
- b. write some unit tests

Pravesh - Disney - 04/09/2025

1. self intro
2. share screen and walk through the resume

Colin Fan - BNY - 04/07/2025

1. self intro
2. debug experience
3. ServiceNow experience
4. how to debug API error
5. SQL Query:

CH account: branch, currency,  
account\_no, entity,  
Lob: Lob\_name, entity, cost  
center

all accounts belongs to branch  
10 and lob\_name is asset  
servicing

6.

CH account: branch, account\_no,  
currency, balance

can you give me no of  
accounts and its total balances  
by currency

7.

8.

Aiden Xu - Walmart - 04/04/2025

1. Intro
2. What is java version you used

3. What are new features in java8
4. What is stream api
5. What is lambda
6. What is optional
7. How can you use optional
8. What is pattern matching
9. What is sealed class
10. What is thread
11. What is functional interface
12. What is executor
13. What is thread state
14. What is hibernate
15. What is jdbc
16. Hibernate vs jdbc
17. What is synchronized and how can you use it
18. What is binary tree

## Binary Tree Search

### Description

A binary search tree (BST) is a data structure where each node contains a value and has up to two child nodes. The left child node has a value less than its parent node, and the right child node has a value greater than or equal to its parent node. This structure allows efficient searching.

For each integer in a given list, determine if it exists in the BST. If it does, return 1; otherwise, return 0.

### Function Description

Complete the function *isPresent* in the editor with the following parameter(s):

*BSTreeNode root*: reference to the root node of a tree of integers

*int val[q]*: an array of integer items to search for

### Returns

*int[q]*: each element *i* denotes whether *val[i]* is found in the BST

19. Coding:

```
import java.io.*;
private static int isPresent(Node root, int val){
    /* For your reference
    class Node {
        Node left, right;
        int data;
        Node(int newData) {
            left = right = null;
            data = newData;
        }
    }
*/
}
```

20. What is your scrum and daily work

21. How do you production support

22. What are logger systems you used
23. What is hotfix
24. What is github
25. If you are writing code, how can you make your code better in anyway
26. How can you create concurrent environments in your java project
27. What is thread pool

**Pravesh - Disney (vendor) - 04/04/2025**

1. intro
2. previous work
3. Who you talk more
4. Full stack?
5. Resume question
6. What kind of team you work with
7. Disagreement with others
8. How to keep up new technology
9. Career goal in next 5 years
10. How to keep up with your team

**Bill Tu - Disney (vendor) - 04/04/2025**

1. self intro
2. project details
3. who is the end user and what features you provide
4. your responsibilities you role
5. why you match this position
6. 2nd project details
7. biggest challenge
8. what if you have conflict with other teammate
9. what do you see on criticize feedback
10. what do you see in 5 years
11. how do you keep yourself updated on new technologies
12. why you leave previous position
13. how do you handle fully remote job
14. team size
15. what if you cannot finish your task in time/ before deadline
16. why disney
17. this position is about full stack and implementing dashboard / ad team , do you have similar experience

**Larris Shao - Apple - 04/01/2025**

1. self intro
2. how to scale API
3. Coding: given an array and integer K, return num of the subarray whose sum is k
4. Coding:

**Problem: Word Ladder II**

Given two words, beginWord and endWord, and a dictionary wordList, find all the shortest transformation sequences from beginWord to endWord, such that:

only one letter can be changed at a time.

Each transformed word must exist in the word list.

Note:

Return an empty list if there is no such transformation sequence.

All words have the same length.

All words contain only lowercase alphabetic characters.

You may assume no duplicates in the word list.

You may assume beginWord and endWord are non-empty and are not the same.

/\*|

### Larris Shao - Apple - 03/31/2025

1. self intro
2. MongoDB experience
3. how do you do sharding
4. How does elastic search work?
5. REST API, how to fix slow response problem
6. how to read the data from a large file efficiently
7. SQL: join, group by and count

```

>  */
> input: Array of sorted integers
>
> output: Rearrange max/min in tandem/alternating
>
> Example:
>
> Input: [200, 312, 344, 422, 543, 678, 723]
> Output: {723, 200, 678, 312, 543, 344, 422}
>
> */

```

8. Coding:

### Manasa Annaparth - Walmart - 03/31/2025

1. self intro
2. details about your project
3. your responsibilities
4. your project pipeline
5. stream api coding: given list of student, find all unique department
6. stream api coding: given list of student, find all contacts(it's list structure), use flatmap

### Dave Du - Apple - 03/28/2025

1. self intro
2. kafka experience
3. bean scope
4. junit
5. java version
6. java 21 features
7. python experience

#### Dave Du - BNY - 03/28/2025

1. tell me about java memory management
2. how gc work
3. how memory heap work
4. where method stored
5. what parameters you know about gc , change gc parameters
6. what is thread local
7. what is wait and notify
8. can abstract class have no abstract method
9. then why we do this
10. what method can we extend from string
11. what is immutable class and how to make it immutable
12. diff synchronized and reentrantlock
13. what is fairlock
14. how hashmap work
15. can we have customized class as key
16. if we override hashCode not overriding equals
17. stream api vs collection for loop
18. how to use stream to filter people younger than 30
19. how to group people to key(age), value(list of people)
20. how to prevent memory leak
21. wait vs sleep
22. how to awake waiting thread
23. how to write spring boot to call from frontend to backend and save data to database
24. what is Executors library
25. what is blockingqueue where we use it
26. diff submit and execute
27. why api gateway
28. if you call 3rd party api and it fail what fall back mechanism
29. what library you use
30. how to secure db data
31. how to improve db performance
32. diff cluster and non-cluster index
33. what is sharding
34. what is kafka, topic partition, offset
35. what 's your cicd exp
36. what stages in cicd
37. how to encrypt ci/cd pipeline
38. how do you execute test in ci/cd
39. angular detection
40. angular service you know
41. how do you handle state manage in angular
42. git exp
43. diff git fetch and pull
44. git branch strategy
45. git stash
46. git rebase and git merge

#### Joe Xing - Apple - 03/27/2025

1. Coding: giving "aabbcccd" return "a2b2c2d1"
2. any corner cases you would consider ?
3. what if input is "a11" what's the output
4. how do we handle these cases

## Dave Du - Apple - 03/27/2025

1. self intro
2. how to manage mult-threading
3. searching experience

```
4  /*
5  Example 1:
6
7  Input: candidates = [2,3,6,7], target = 7
8  Output: [[2,2,3],[7]]
9  Explanation:
10 2 and 3 are candidates, and 2 + 2 + 3 = 7. Note that 2 can be used multiple times.
11 7 is a candidate, and 7 = 7.
12 These are the only two combinations.
13 Example 2:
14
15 Input: candidates = [2,3,5], target = 8
16 Output: [[2,2,2,2],[2,3,3],[3,5]]
17 Example 3:
18
19 Input: candidates = [2], target = 1
20 Output: []
21 */
22
```

## Lexie Kong - Sales Force - round 2 - 3/24/2025

1. Self and project intro
2. websocket vs http
3. css exp
4. how do you check and handle exception in react / frontend
5. how do you learn new technologies
6. react memo vs callback
7. react global state management
8. why redux with react, any benefits
9. what is useEffect
10. diff promise and async await
11. what is virtual dom
12. airflow? glue? etl exp
13. how do you do if you cannot finish job before deadline
14. do you prefer working independently or with team or mix
15. any exp with slack plugin
16. what is elastic search
17. why redis
18. diff vertical scaling and horizontal scaling
19. why we have horizontal scaling in Redis
20. what about session management
21. what is jwt and how jwt work
22. id token vs access token
23. how stack, jvm, gc work
24. final vs finally vs finalize
25. jvm tuning
26. concurrent hashmap vs hashmap vs treemap
27. oop 4 principles
28. tell me a excited story /challenge
29. how you analyze slow performance of the request
30. how you find the issue , which part is slow in microservice
31. if in one service how you find slow issue from which part of code / Thread
32. how do you analyze thread execution flow
33. Thread dump vs Heap dump
34. What do you analyze for GC when you use JProfiler? (pattern/pause)

35. how do you analyze sql query performance and how to tune it
36. SQL Coding: emp(dept\_id fk) table and dept table
  - a. dept\_id in emp table but not in dept table
  - b. count number of each one
  - c. remove count > 10

**Lexie Kong - Sales Force - round1 - 3/24/2025**

1. Self and project intro
2. Previous project, daily work
3. Tell about if you do optimization like pagination
4. How frontend and backend communicate
5. Experience work on front end
6. How microservices communicate
7. How service discovery work
8. If there external services want to visit you back, how you handle large volume request
9. what is Load balancer

**Lexie Kong - Sales Force - vendor - 3/24/2025**

1. Self and project intro
2. how to optimize
3. React memo
4. React callback
5. SQL Coding: get 3rd highest count of xx, if 3rd not present, get 2nd highest, or get top highest
6. Java Coding: remove duplicate adjacent characters
  - input: "axxxxa" => "aa" => ""
  - input "axxbxxa" => "aba"

**Anitha - Disney - 3/24/2025**

1. Self and project intro
2. What is stream api in java 8
3. Flat vs flatmap
4. Map vs filter
5. I have two classes, class A using primitive type to connect database, class B using wrapper type to connect database, what are difference between them
6. Comparable vs comparator
7. What is sington
8. monolithic vs microservices
9. What is cache memory. Pros and cons
10. I have table person, two columns, one is name another is age, I want to give me one records that tell me the name of the person and also the age of the oldest person
11. Local vs instance vs static variable
12. event driven vs request driven architecture
13. exceptions: checked, and unchecked
14. one class has catch Exception, another class catch sql exception. what are different between them. which one you like and why

**Dave Du - Walmart - 3/20/2025 -**

1. self intro
2. Education experience
3. Spring boot project implementation

## Springboot Java: COVID API

In this project, virtual data related to COVID-19 is provided for many countries. (Note that all the data is artificial.) Your task is to implement several REST endpoints to handle this data.

Here is an example of a COVID data JSON object:

```
{  
    "id":1,  
    "country":"MyCountry",  
    "continent": "antarctica",  
    "active":574,  
    "death":45,  
    "recovered": 7000  
}
```

The application should adhere to the following API format and response codes:

4.

`GET /covid/{id}:`

- Return the COVID entry with the given ID and status code 200.
- If the requested COVID entry does not exist, return status code 404.

`GET /covid/top5?by={by}:`

- Return the top 5 COVID entries sorted by the given field and status code 200.
  - Example: `/covid/top5?by=death` gives total deaths.
- If `by` is an invalid attribute, return status code 400.

`GET /covid/total?hv={hv}:`

```
GET /covid/total?by={by}:
```

- Return the sum of the given field and status code 200.
  - Example: /covid/total?by=active gives total active cases.
- If the *by* attribute is invalid, return status code 400.

```
GET /scan/report/scanDashboard:
```

- Generate a report like the JSON below. It needs to be grouped by *continent* and find the impact factor of each group.
- The status code should be 200, and the rounding precision is 3 decimal places.
- Exclude entries that have *continent* as null or empty.
- The *impactFactor* formula is as follows:  
$$\text{impactFactor(europe)} = \frac{\text{death}}{\text{sum(active+death+recovered)}}$$

- 5.
6. how to use @Transactional to manage the request
7. in one method, if there are db call, restapi call, service call, what will you consider?

### David Zhang - Swift- onsite- 3/19/2025

1. self-intro
2. Introduce the last project architecture(White Board)
3. Why use springboot?
4. Could AOP apply to the private method?
5. What's your responsibility in the microservices?
6. Any DevOps experience?
7. What AWS services did you work on?
8. What's AWS S3 used for?
9. Where is your frontend server deployed?
10. Is each service an independent spingboot?
11. Does each service have its own independent database?
12. What's the traditional solution of distributed transactions?
13. How do microservices communicate with each other?
14. Angular vs React, depends on what?
15. Any front-end test experience?
16. What did you build with Angular?
17. Rate your angular skill from 1 to 5
18. If the customer reports a product issue and there is no log for it, how are you going to solve it?
19. Coding: Print the first 5 Fibonacci number
20. Is there any other solution like recursion?
21. Time Complexity?
22. Do you have coding experience with AI models? Which model is the best for Java?

## Alan Mo - Digital Management - 03/19/2025

1. self intro
2. OOP
3. interface
4. class vs object
5. final variable vs method vs class
6. prevent sql injection
7. introduce what is Spring Framework
8. spring boot advantages
9. scale up vs scale out
10. SQL Query: Student table (firstname, lastname, major)
  - a. merge first name and last name column
  - b. how many students in each major

## Erkang xia - Finra - 03/04/2025

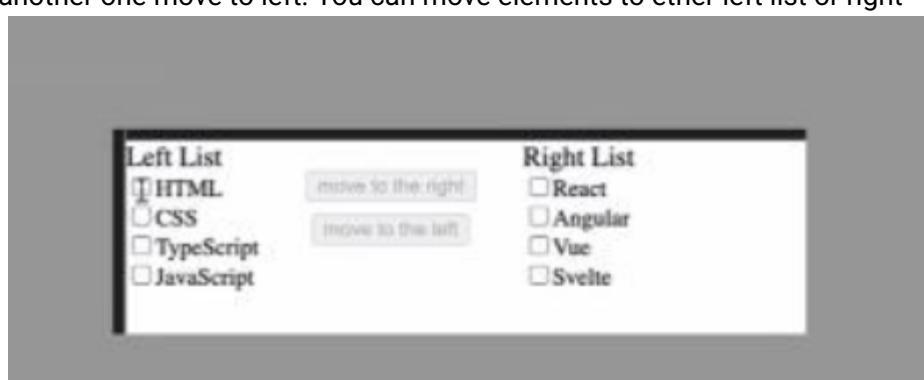
1. First round: coding:

```
1 // testing...
2 Id Name DeptId ManagerId Salary
3 1 Foo1 0 1000
4 2 Foo2 1 900
5 3 Foo3 1 850
6 4 Foo4 1 700
7 5 Foo5 1 720
8 6 Foo6 2 500
9 7 Foo7 2 400
10 8 Foo8 2 450
11 9 Foo9 2 425
12 10 Foo10 3 700
13
14 |
```

find top 3

earners for each department

2. Second round: using react to solve this question. You have two buttons, one move to left, another one move to right. You can move elements to either left list or right



- 3.
4. Coding: reverse a string without using string apis and loop
5. Write test functions to test all corner cases, tell me why

6. Use collection and reduce function to solve this question. reverse() is not allowed
7. How does reduce function works
8. What is output if you call Integer.Max\_value + 1 and why
9. What is generic type in java
10. What are design patterns you recently used
11. What is singleton
12. Write singleton programming
  - a. Eager one: when instance is loaded
13. What is the context in singleton
14. What is gc
15. Does does gc works
16. Write a programming that cause out of memory
  - a. Candidate answer: List<int> arraylist = new ArrayList<>();
    1. while(true){arraylist.add(1)}
  - b. What is size of int
  - c. How long this programming takes
  - d. If we change to string, does this take longer? And why
  - e. What is size of string
  - f. What is stringpool

#### Eric cao - Geode - 02/21/2025

1. Introduction about yourself
2. Primitive type vs wrapper type
3. Stack vs heap
4. What memory will give out of memory error
5. What is string pool
6. Class variable vs instance variable
7. What is an interface
8. Can you have multiple inheritance in java?
9. What is singleton design pattern
10. What is eager singleton
11. What is lazy singleton
12. What is ioc
13. What is DI
14. What is lambda function
15. What is stream api
16. What did you do with aws
17. What is restful api means
18. If we have one api and this api will call one method and we have two static variables in this method. What restful rule will break?
19. If we have sql table, student -> id and name. And designers forget to make primary type for id. How do you find duplicate id
20. How do you delete duplicate id and keep one
21. Do you have frontend experience?

#### Bowen Yang - Ebay - 02/19/2025

1. self intro
2. any challenge in your project
3. anything you don't want to do but have to do in project
4. coding: given orders
 

```
weight = 50
orders = (0, 30, 1), {orderid, weight, deadline}
(0, 20, 2),
(0, 30, 2),
(0, 40, 3),
(0, 20, 3)]
```

need to deliver the order before deadline, return minimum shipment we need to deliver all orders before deadline, if not possible, return -1

asked the interviewer: max size of orders? <= 10

asked the interviewer: can we ship many times per day? no, 1 shipment per day

asked the interviewer: what if there is no valid answer? return -1

5. followup: how to modify your code, if we want to track the shipping orders each day. Instead of printing the minimum shipping day, I want you to print day 1 : order xx , day 2 : order xxx
6. followup: how would you test your program, any other test cases you will use?

#### Emmanuel Sowunmi - Collebra (vendor tech) - 02/18/2025

1. self intro
2. java 17 new features
3. what is seal in java 17
4. how to keep thread safe
5. where did you use multithreading in project
6. what is concurrenthashmap
7. what is blocking queue
8. how to create thread
9. how many threads can read from concurrenthashmap at same time
10. how to analyze heap dump
11. how to optimize jvm
12. polymorphism
13. encapsulation
14. what is default method in interface
15. why we use it
16. final vs finally vs finalize
17. how to reuse database connection
18. comparator vs comparable
19. coding: use stack to impl queue

#### Joe Xing - State Street - 02/14/2025

1. self intro
2. your project architecture
3. what's your reposnsiblites
4. dive deep into project details (he mentioned recommendation system)
  - a. architecture and data flow of this feature
  - b. does position tag effect the result
  - c. how do you process pdf and word file efficiently
  - d. which part you worked on
5. how does spark cluster work
6. have you used hydoop
7. how does java handle multithreading
8. where did you use multirheading in your project
9. thread safe queue in java
10. diff fair lock and unfair lock
11. how do you design a fair lock if we don't have it in java
12. aws s3 multi region access point, if one region doesn't have copy yet after you uploading the file, how does s3 handle it
13. how does role work in aws
14. how kafka prevent data lost
15. what is producer ack
16. how do we prevent duplicate message
17. how do we handle duplicate message situation
18. coding: design LRU cache

#### Bowen Yang - Ebay - 02/04/2025

1. self intro
2. vanilla javascript coding: create a pop up window, and create a button. show the pop up window after clicking the button. display pop up window in the middle of the screen.

#### Bowen Yang - Ebay - 02/04/2025

1. self intro

2. what you have done in your project
3. talk about one service
4. what do you know about cache
5. coding: LRU
  - dry run your code
  - any other test cases / changes in your mind if we need to use this code in production
6. RDBMS vs NOSQL
7. if your application is slow what do you do
8. if you get requirements from PM what do you do, like different steps
9. what indexes do we have
10. give me example when do you use xx index

**Kevin Kong - Walmart - 02/10/2025 - Gary Rosales Zavala**

1. self intro
2. interesting part in your project

**Anitha - Global CDK (vendor tech round) - 02/07/2025**

1. self intro
2. why spring boot
3. do you have exp migrate from monolithic to microservice
4. what do you need to consider when you do it
5. spring boot version you used
6. new features from spring boot 3
7. what is webflux
8. have you used it in your project
9. what java version can we use with spring boot 3
10. what is discovery service , and why use it
11. what discovery service impl you used before
12. what annotations and configurations you did in eureka in spring boot
13. why AOP
14. where did you use AOP in application
15. what design patterns you used before
16. why singleton
17. where did you use singleton in your project(need example from project)
18. how to stop auto configuration in spring boot
19. what is CQRS pattern
20. have you used CQRS in your project
21. what is circuit breaker
22. how do you implement circuit breaker
23. if service A calling service B and no response from service B what will you do. Tell me all solutions to give user better experience
24. what monitors have you used
25. how do you increase microservice fault tolerance
26. what mechanism to maintain / keep fault tolerance(SLA, SLI,SLO ? )
27. for SLO what do you do to keep scalable microservice meet requirement
28. why s3 / why serverless
29. how do you keep class / method thread safe
30. how do you create immutable class
31. can you compile class with non-static main method
32. will you get exception and what exception name with non-static main method
33. what is deadlock
34. what is race condition
35. what is optimistic lock
36. have you experienced optimistic lock exception
37. what is pessimistic lock
38. where do you use pessimistic lock
39. what is spring boot actuator
40. what dependency name do you need when you use it
41. what annotations we use to configure customized actuator
42. how to secure sensitive data in application / database

43. how do we secure password
44. will you use char[] or string to keep password
45. string A = "123" how many objects created in heap and why
46. string A = new String("123") how many objects created in heap and why
47. how to handle exception
48. try catch vs exception handler which one you prefer and why
49. how many ways to load profile in spring boot (where and how)
50. what is class not found exception
51. how do we avoid this exception if we want to see this exception
52. what is out of memory, and how to prevent out of memory error
53. what is soft reference
54. where can we use soft reference in project
55. how to increase young generation size in heap
56. where do we configure it
57. how do we do it in spring boot and intellij (change young generation size)
58. what is blue green deployment
59. do we keep old servers
60. do we use it in stage env

#### Alan Mo - eBay - 02/06/2025

1. self intro
2. Angular Project:
  - a. create login page
3. Coding, use typescript implement  $\text{mul}(a)(b)(c) = a * b * c$

#### Bowen Yang - Ebay - 02/04/2025

1. self intro
2. what's your biggest challenge
3. why ebay
4. how request work, when you click url in browser
5. what is MVC
6. what is CORS
7. where can we set CORS (backend or frontend or both)
8. what is cookie
9. coding: write Angular component
  - a. send request to "<https://jsonplaceholder.typicode.com/users>"
  - b. get response, display username, name, company name in table (row + column)
  - c. change css, put 8px space between header, and add horizontal line separator between rows
10. coding: java validate parentheses

#### Sean Li - Walmart - 02/04/2025 - GRZ

1. self intro
2. project intro
3. frontend experience

#### Josiah Jiang - Walmart - 01/30/2025

1. self intro
2. Spring boot annotation
3. REST API
4. how to improve the system performance
5. http 1 vs 2
6. Kafka monitor tool
7. how to detect the lag of message in kafka
8. log vs metrics
9. how to query in mongoDB
10. Coding: search target on sorted matrix

**Charlotte Zhang - BNY - 01/28/2025**

1. self intro
2. your previous project
3. have you used message queue and where did you use it
4. kafka exp
5. where did you use kafka
6. how does kafka work
7. diff kafka and sqs
8. if we have too many messages in kafka queue what can we do

**Wincy Xu - Apple - 01/27/2025**

1. Bean scope
2. Profile
3. Hash table vs hash map
4. ArrayList vs linkedlist
5. Design pattern - Singleton
6. Nodal experience
7. Elastic Search: Must vs Should
8. Reflection
9. Kafka

**Jiahui Zou - BNY - 01/27/2025**

1. self intro
10. your project architecture
11. your project ci/cd pipeline
12. do you know how to configure ECS
13. what you include in your dockerfile
14. how did you develop frontend part of your project, do you get requirement from other people or who approve that feature
15. how did you handle fault tolerance
16. did you write any stored procedures
17. will you choose stored procedures or java hibernate logic
18. can you write hint in hibernate
19. what database did you use
20. do you see yourself as frontend or backend

**Alan Mo - Apple - 01/22/2025**

1. Coding: give a binary search tree and a number. Transverse the tree to find if there are any two nodes that sum up to the target.

```

1  /**
2  Given the root of a binary tree, return the maximum width of the given tree. The maximum
   width of a tree is the maximum width among all levels. Width of one level is the number of
   node between the left-most node and the right-most node, where the null nodes between the
   two are also counted towards the calculation.
3
4  For example
5
6  Tree-1:
7  ...|1
8  .../...\_
9  ...2...3
10
11 Max width = 2, which is the width at level 2
12
13 Tree-2:
14
15 ...|...|1
16 ...|.../...\_
17 ...|...2...3...
18 ...|...|\...|\
19 ...|...4...5
20
21 Max width = 3, which is the width at level 3 (including a null node between node 4 and 5)
22
23 */
24
25 /**
26 * Definition for a binary tree node.
27 * public class TreeNode {
28 *     int val;
29 *     TreeNode left;
30 *     TreeNode right;
31 *     TreeNode(int x) {val = x; }
32 * }
33 */
34

```

2.



Jiahui Zou - BNY - 01/22/2025

1. design dmv system

Max Ma - Swift - 01/22/2025

1. share screen, turn 360 to show around
2. singleton what is singleton. How to implement the singleton
3. restful api detail
4. is restfulapi stateful vs stateless
5. what is stateful what is stateless
6. cassandra how to implement it
7. how to do query in cassandra
8. What is cassandra? how you use cassandra, what is different between sql and cassandra
9. what is jpa stand for what jpa name
10. how you connect the database without jpa
11. how to connect the database in springboot
12. what is controller how you use controller how you implement controller
13. what is microservice
14. how you use microservice
15. how to connect microservice
16. Exception handle/ how to handle exception
17. Exception have two type what are they
18. what is runtime exception, what is compare time exception
19. how you implement exception in springboot
20. how you run java project in terminal
21. class not found
22. what is reflection how you use reflection

**Jeffery Chen - Swift - 01/22/2025**

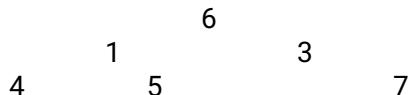
1. share screen, turn 360 to show around
2. self intro
3. rest api
4. stateful vs stateless
5. rest vs sock
6. microservice architecture
7. project details, responsibilities
8. stored procedure
9. Java lambda
10. Java Collection

**Alan Mo - Apple - 01/21/2025**

1. self intro
2. spring boot features
3. bean scope
4. static keyword
5. HashMap vs HashTable
6. StringBuffer vs StringBuilder
7. Redis Cache
8. Kafka Architecture
9. BFS
10. How to create Graph
11. no sql database

**Jiahui Zou - BNY – 01/17/2024**

1. introduction
2. coding: grandparent even number treenode current node val sum



result = 4 + 5 + 7 , because all of these nodes' grandparents are even

3. coding: jdbc connection talk to database and execute stored procedure
4. spring data jpa exp
5. native query exp
6. design patterns you have used
7. what database have you used
8. have you written any stored procedures before
9. angular exp
10. ngrx exp

**Jiahui Zou - T mobile(ENFEC) – 01/16/2024**

1. introduction
2. how did you use spring security
3. coding: create spring boot project and fetch url from 3rd party api
4. wait vs sleep
5. team size
6. ci/cd exp

**Calvin Tran - Capital One - 01/14/2024**

1. self intro
2. why switching job
3. how did you hear about capital one
4. JPA

5. AWS experience
6. ECR/ ECS
7. Microservice architecture in previous project
8. Most difficult challenge in your career
9. Angular version
10. CompletableFuture
11. java version
12. Maven
13. commonly used shell command
14. sql injection
15. github workflow
16. AWS Farget
17. AWS lambda code start
18. AWS SNS, SQS
19. experience in other coding language

Alex Shao - Apple - 01/13/2024

```

1  /**
2  Given the root of a binary tree, return the maximum width of the given tree. The maximum
   width of a tree is the maximum width among all levels. Width of one level is the number of
   node between the left-most node and the right-most node, where the null nodes between the
   two are also counted towards the calculation.
3
4  For example
5
6  Tree-1:
7  ...
8  ...
9  ...
10
11 Max width = 2, which is the width at level 2
12
13 Tree-2:
14
15 ...
16 ...
17 ...
18 ...
19 ...
20
21 Max width = 3, which is the width at level 3 (including a null node between node 4 and 5)
22
23 */
24
25 /**
26 * Definition for a binary tree node.
27 * public class TreeNode {
28 *     int val;
29 *     TreeNode left;
30 *     TreeNode right;
31 *     TreeNode(int x) {val = x;}
32 * }
33 */
34

```



Alex Shao - Apple - 01/10/2024

1. self intro
2. tell me your spring exp
3. why spring boot
4. what the java version you used
5. what spring version you used
6. what spring boot version you used
7. you said you used java 17 have you used mono / webflux

8. what is reactive programming
9. why message queue
10. advantage of rest api
11. disadvantage of rest api
12. how many regions do your application deploy to
13. how does data center do disaster recovery from one region to another region
14. why RDS in aws
15. how does RDS do disaster recovery from one region to another region
16. where does the RDS snapshot store
17. how do you use RDS snapshot to recover RDS db
18. how does s3 backup data
19. your cicd pipeline
20. what tools you used daily
21. how did you deploy to production

**Jichen Jiang - Apple - 01/09/2024**

1. self intro
2. project techstacks
3. your responsibility
4. messaging system advantage
5. java version used
6. springboot version
7. arraylist vs linked list

**Calvin Tran - BNY - 12/30/2024**

1. self intro
2. HTTP Trace, Options
3. K8s experience
4. microservice experience
5. Design Pattern
6. how to create annotation
7. in mongodb, how do you know what type attribute in a document
8. Coding: count the length of strings in a List
9. Angular experience
10. typescript vs javascript
11. version upgrades experience
12. Different types of Garbage collection
13. AWS experience
14. jdbc vs hibernate

**Hailey Jiang - Vanguard - 12/09/2024**

1. most recent project
2. what aws services have you used
3. what is 503 status code
4. what is your CI/CD pipeline
5. coding: find first occurrence index and last occurrence of input x in input array
6. where did you use lambda
7. if we send an event to lambda and lambda send event to message queue, how long will it take.
8. how you write restapi in spring boot
9. where did you use python script
10. why python over java on lambda
11. python tuple vs list
12. any other diff except immutable
13. python module vs package
14. agile scrum
15. have you used step function

16. questions related to his story, cache and TTL
17. how do you do fail tolerance over the region
18. where did you use multithreading
19. how did you deploy aws services, cloudformation
20. what was your deployment guideline, format, process to production
21. what version control did you use
22. how do you learn new things , technologies
23. tell me how aws glue work
24. tell me how did you use sagemaker
25. have you done production support
26. how did you read log, from cloudwatch or splunk
27. have you configured splunk alert or dashboard
28. what is event bridge
29. how do you keep your application secure
30. what did you use in your application to keep application secure

**Joe Xing - Plymouth Rock Assurance - 12/05/2024**

Round1

1. What is s3
2. What does ELB do
3. Do you know GC
4. have you used redis stream
5. what is application server
6. What does @Transactional do?
7. How did you process file upload in your service?
8. What is RDS?
9. What did you use DynamoDB for?
10. How did you use multithreading in your project?
11. What is Jenkins?

Round 2

12. Talk about your project
13. What application server have you used before?
14. What ORM have you used? Why ORM?
15. What does Index do in RDBMS?
16. What AWS service have you used.
17. What is the maximum upload size for S3
18. If I have 4 cores in my CPU, how to make sure you utilize all cores in your application
19. What's the difference between mvn build and mvn deploy?
20. If I have an older version of a library in maven, and I want to inject a new version of the library to the project. How can I make sure the project is using the new version
21. How do you test for private method using JUnit testing
22. Describe Spring MVC

**Eric Cao - Plymouth Rock Assurance - 12/02/2024**

1. most recent project
2. diff arraylist vs linkedlist
3. why stream api
4. why orm
5. linear search vs binary search
6. how does kafka work
7. how do you optimize kafka
8. how does message /request find the partition in kafka
9. your project data flow
10. where did you use multithreading in your project
11. why ASG in aws
12. how aws make service HA
13. what exception will you get if you use non-thread safe collection in thread env
14. Kubernetes component
15. how does spring mvc work
16. spring annotations

17. what is aop
18. what is advice
19. if we have 100 classes and there are many methods in those classes, how to track performance of each method.
20. AWS lambda function features / disadvantages
21. stream vs parallel stream
22. how to write smoking test
23. how to test private method
24. how to do unit test
25. how many objects can we store in S3
26. what's the max size of obj can we save in s3
27. biggest challenge
28. have you used load balancers
29. if there are 2 nodes behind the load balancer, one node fails during the transaction. What should we do and what will happen

**Joe Xing - Plymouth Rock Assurance - 11/27/2024**

1. most recent project
2. diff encapsulation and inheritance
3. diff checked exception and unchecked exception
4. diff error and checked exception
5. java 8 new features
6. diff stream and parallel stream
7. what is functional interface
8. what is functional programming
9. can we have static and default method in interface
10. why we put static, default method in interface
11. why spring
12. how many ways to inject bean
13. injection types
14. how did you monitor your application (jvm, log, api performance)
15. how did you deploy
16. if i send request to your endpoint, how does request go in your aws env
17. do you have ALB
18. how does api gateway connect with ALB
19. what is dispatcherservlet
20. diff rest controller and controller
21. any exp on kafka
22. what is sqs
23. is sqs pub sub
24. diff inner join and outer join
25. diff union and union all
26. what is docker
27. why do we need docker container
28. how did you test your endpoint

**Calvin Tran - BNY - 11/22/2024**

1. self intro
2. most recent project
3. Coding:

```

1 Given an integer array nums, find the
2 subarray
3 with the largest sum, and return its sum.
4
5
6
7 Example 1:
8
9 Input: nums = [-2,1,-3,4,-1,2,1,-5,4]
10 Output: 6
11 Explanation: The subarray [4,-1,2,1] has the largest sum 6.
12 Example 2:
13
14 Input: nums = [1]
15 Output: 1
16 Explanation: The subarray [1] has the largest sum 1.
17 Example 3:
18
19 Input: nums = [5,4,-1,7,8]
20 Output: 23
21 Explanation: The subarray [5,4,-1,7,8] has the largest sum 23.
22
23 class Solution {
24     public int maxSubArray(int[] nums) {
25
26     }
27 }
```

- a.
- b. Dry run test cases
- c. left join vs outer join
- d. What is SPA, Single Page Application?
- e. MVVC
- f. Spring AOP
- g. Authentication process
- h. functional interface
- i. java 21 new features

## Norfolk - eBay - 11/22/2024

1. self intro
2. Java Stream

```

1 You are given a list of Person objects, where the Person class is defined as follows:
2
3 java
4 Copy code
5 public class Person {
6     private String name;
7     private int age;
8     private String city;
9
10    // Constructors, Getters, Setters
11 }
12 Write a Java program using Streams to perform the following operations on a list of Person objects:
13
14 Filter out all persons whose age is less than 18.
15 Group the remaining persons by their city into a Map<String, List<Person>>.
16 Sort the persons in each city group by their age in descending order.
17 Print the results in the following format:
18
19 City: <city>
20 Name: <name>, Age: <age>
21 Name: <name>, Age: <age>
22 Example Input:
23 List<Person> persons = Arrays.asList(
24     new Person("Alice", 17, "New York"),
25     new Person("Bob", 20, "Los Angeles"),
26     new Person("Charlie", 22, "New York"),
27     new Person("David", 19, "Los Angeles"),
28     new Person("Eve", 18, "New York")
29 );
30 Expected Output:
31 City: Los Angeles
32 Name: Bob, Age: 20
33 Name: David, Age: 19
34 City: New York
35 Name: Charlie, Age: 22
36 Name: Eve, Age: 18

```

- 3.
4. Spring boot

```

1 //codeshare
2 You are tasked with creating a simple Spring Boot REST API that provides CRUD operations for a Book entity. The entity has the following fields:
3
4 id (long, primary key)
5 title (String)
6 author (String)
7 price (Double)
8 reviews (List<Review>)
9
10 Implement the following endpoints:
11 GET /books: Retrieve a list of all books.
12 GET /books/{id}: Retrieve a book by its ID.
13 POST /books: Create a new book.
14 PUT /books/{id}: Update an existing book.
15 DELETE /books/{id}: Delete a book by its ID.
16 Use an in-memory database (H2) for persistence.
17 Add basic input validation for the Book fields (e.g., title and author cannot be empty, price must be positive).
18
19
20
21 public class Book {
22
23     @Id
24     @GeneratedValue(strategy = GenerationType.IDENTITY)
25     private Long id;
26
27     @NotBlank(message = "Title cannot be empty")
28     private String title;
29
30     @NotBlank(message = "Author cannot be empty")
31     private String author;
32
33     @Positive(message = "Price must be positive")
34     private Double price;
35
36     // Constructors, Getters, and Setters
37
38 }
39
40 import org.springframework.data.jpa.repository.JpaRepository;
41
42 public interface BookRepository extends JpaRepository<Book, Long> {
43 }
```

5.

```

45
46
47 import jakarta.validation.Valid;
48 import org.springframework.beans.factory.annotation.Autowired;
49 import org.springframework.http.ResponseEntity;
50 import org.springframework.web.bind.annotation.*;
51
52 import java.util.List;
53
54
55 @RequestMapping("/books")
56 public class BookController {
57
58
59     private BookRepository bookRepository;
60
61     // GET /books: Retrieve all books
62     @GetMapping
63     public List<Book> getAllBooks() {
64         return
65     }
66
67     // GET /books/{id}: Retrieve a book by ID
68     @GetMapping("/{id}")
69     public ResponseEntity<Book> getBookById(Long id) {
70         return
71     }
72
73     // POST /books: Create a new book
74     public ResponseEntity<Book> createBook(@Valid @RequestBody Book book) {
75         Book savedBook = bookRepository.save(book);
76         return ResponseEntity.ok(savedBook);
77     }
78
79     // PUT /books/{id}: Update an existing book
80     @PutMapping("/{id}")
81     public ResponseEntity<Book> updateBook( Long id, @Valid @RequestBody Book bookDetails) {
82         return
83     }
84
85     // DELETE /books/{id}: Delete a book by ID
86     @DeleteMapping("/{id}")
87     public ResponseEntity<Void> deleteBook(@PathVariable Long id) {
88         return
89     }

```

6. Spring boot Dependency injection
7. Spring boot actuator
8. How to check system latency

### Ari Li - T mobile - 11/20/2024

1. most recent project
2. wait vs sleep
3. your responsibilities
4. project tech stacks
5. team size
6. how did you develop service
7. project architecture
8. ci/cd exp
9. why lambda expression
10. how did you use spring security
11. what annotations you used in spring
12. coding: given string, get max min freq characters

### Eric Cao - Plymouth Rock Assurance - 11/20/2024

1. most recent project
2. java 8 new features
3. diff stream and parallel stream
4. what is forkjoin pool
5. what are some functional interfaces
6. how many static methods can we create in functional interface
7. how many default methods can we create in functional interface
8. what is bifunction
9. diff runnable and callable
10. how many ways to create thread

11. diff future and completable future
12. why spring
13. how did you monitor your application (jvm, log, api performance)
14. how did you deploy
15. what gateway did you use
16. what ASG strategy did you use
17. how did you monitor log
18. how did your application send log to ELK
19. do you know actuator
20. have you used lambda service
21. where did you use it
22. diff sqs vs kafka
23. if sqs has message, and 2 consumers, will they get message at same time
24. how did your api gateway connect with application loadbalancer
25. diff join and inner join
26. diff union and union all
27. what isolation levels have you used
28. http method / operations you used before
29. diff put and post(idempotent)

#### TK Chu/ Calvin - Walmart - 11/18/2024

1. self intro
2. most recent project
3. front end experience
4. when you have conflicts with teammate
5. Team size
6. CICD experience
7. Coding:

```

File Edit Format View Help
Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid.
An input string is valid if:
Open brackets must be closed by the same type of brackets.
Open brackets must be closed in the correct order.
Every close bracket has a corresponding open bracket of the same type.

Example 1:
Input: s = "()"
Output: true
Example 2:
Input: s = "()[]{}"
Output: true
Example 3:
Input: s = "[]"
Output: false
Example 4:
Input: s = "([[]])"
Output: true

Constraints:
1 <= s.length <= 104
s consists of parentheses only '()[]{}'.

```

8. how to monitor and log
9. advantages of Spring boot Framework
10. how to improve kafka efficiency
11. in react, props vs states

#### TK Chu - BNY - 11/14/2024

1. most recent project
2. some questions related to his resume / story
3. coding: 3 sum, dry run your code
4. give me example where you use spring boot
5. how do you handle exception in spring boot
6. how do you validate input data in spring boot
7. tell me some design patterns you know

8. what is factory and why we use it
9. tell me different ways in java to load data from database
10. why orm

#### Ava Meng - Walmart - 11/13/2024

1. self intro
2. where did you use multithreading in your project
3. how does completable future work
4. completable future / multithreading in spring framework
5. what is synchronized
6. how thread pool work
7. diff hashmap and hashtable
8. when use hashmap
9. when use hashtable
10. what is race condition
11. give me an example of race condition
12. diff heap and stack
13. when will we get out of memory error
14. diff fail fast and fail safe
15. what exception does fail fast throw
16. what is stack overflow error
17. coding: move zero

#### Max Ma - American Airline - 11/05/2024

1. self intro
2. angular exp
3. microservice exp
4. interface vs extend class
5. bean scope
6. what is default method
7. why default method
8. what is SOLID principle
9. what is S
10. what is circuit breaker
11. what's your fallback mechanism
12. what is actuator
13. can we have multiple catch blocks
14. what's the output of following code:

```
try {  
    } catch (Exception ex) {  
    } catch (IOException r) {  
    }
```
15. coding: java 8 stream api to check palindrome

#### Bowen Yang - Vanguard (java + mobile position) - 10/30/2024

1. self intro
2. how did you design rest api
3. how did you design rest api contract
4. why kafka
5. why message queue
6. diff promise vs observable
7. how to store state in angular
8. have you used ng rx
9. promise vs observable
10. how did you document your api
11. swagger ? open api ?
12. are you using confluence without swagger and open api

13. mobile development exp
14. any ios exp
15. how did you deploy
16. google play store exp

#### Maxwell Sun - Vanguard (java + mobile position) - 10/30/2024

1. self intro
2. when did you graduate
3. how did you design rest api
4. how did you secure your rest api
5. how was your security flow
6. what was your security provider
7. how did you store jwt in angular
8. isn't local storage vulnerable
9. angular lifecycle
10. overriding vs overloading
11. abstract class vs interface
12. when abstract class
13. when interface
14. class extend from abstract class or normal class difference
15. AOP
16. IOC
17. Singleton
18. Angular lifecycle

#### Max Ma - Vanguard (java + mobile position) - 10/25/2024

1. self intro
2. are you still working with current project
3. do you know vanguard
4. did you need to go onsite / hybrid in your current position
5. tell me something about yourself
6. what have you done / responsibilities
7. anything you done with java ,web development
8. what have you done with mobile development
9. why this position ? why did you apply this position ?
10. are you android expertise
11. have you done IOS development
12. tell me a process / progress / project , and how did you impl it, and how did you evaluate it, and
13. how could you tell your process was success / how did you measure success
14. did you learn angular recently
15. choose some rest api you built, explain the flow / details including service name, layers, message queue, database.
16. your resume has oracle db, where did you use oracle db in your last project
17. tell me your ci/cd
18. where do you deploy your application (cloud, local)
19. where did your android app deployed to
20. why cache
21. when will you choose cache
22. how did you do the security part
23. tell me all security implementations to rest api
24. for jwt , do we need identity provider
25. how do you call different service, service to service
26. do you have any retry and circuit breaker
27. what is circuit breaker, and how does it work
28. diff synchronous request and asynchronous request
29. any angular exp
30. for mobile app, how much exp you have

31. tell me a project you started from scratch (service, design) -> how did you start, learn, design , document
32. biggest challenge, walk me your process, and the result
33. have you done production support
34. what tools in you used to track project
35. explain your agile scrum, jira process, how do you track story, progress

#### Norfolk Hu - Charter Communications - 10/25/2024

1. self intro
2. project/ data center
3. where do you deploy
4. Kubernetes
5. Java 11 vs Java 17
6. spring boot new features
7. what is AOP
8. where do you use AOP
9. linux experience
10. how to compare two files
11. how to change mode
12. security experience
13. kafka experience
14. splunk experience

#### Jeffery Chen - Freddie Mac - 10/18/2024

1. self intro
2. Angular version
3. Angular test framework
4. Selenium experience
5. Cucumber experience
6. Oracle database version
7. how to put an element in the middle by CSS
8. Kafka experience, how to handle consumer failure
9. jenkins pipeline

#### Calvin Tran - Freddie Mac - 10/18/2024

1. self intro
2. projects
3. multi-threading use case in project
4. how to handle exceptions
5. different types exception
6. volatile vs synchronized
7. jdbc vs hibernate
8. Angular version
9. how to put an element in the middle by CSS
10. @Profile
11. git merge strategy
12. @Component vs @Service

#### TK Chu - Freddie Mac - 10/17/2024

1. why you left previous project
2. were you hired by apex through out the year

3. self intro
4. who design system
5. who design flow
6. questions based on your resume (each line / each keyword)
7. how did you optimize performance
8. story about multithreading
9. CI/CD
10. development process
11. git strategy
12. why spring boot for microservice
13. why did you choose xx major
14. if you have conflict with teammate / leader / architect, what will you do
15. where did you use elastic cache
16. what does endpoint + request body, response body look like when we create book

### Sam Hsiao - Penske - 10/09/2024

1. self intro
2. when your service calls two endpoints and then send result to front end, how do you do
3. java 8 new features
4. functional interface
5. stream API
6. Coding: Given an array of string, use java stream to find the string with the longest length
7. Coding: find the longest palindromic substring in a given string
8. @Controller vs @RestController
9. how to tune database

### Cayden Liu - Penske - 10/08/2024

1. self intro
2. project
3. equals() vs ==
4. equals() and Hashcode
5. hash collision
6. final, finally, finalize
7. spring security
8. beanfactory vs factorybean
9. @Transactional
10. TDD
11. unit test vs integration test
12. doAnswer() in Mockito
13. inner join vs outer join
14. Promise vs Observable

article	author	price
0001	B	3.99
0002	A	10.99
0003	C	1.69
0004	B	19.95
0005	A	6.96

- 15.
16. find the article with the highest price for each author

### Hailey - Walmart - 10/04/2024

1. self intro
2. explain your project

3. biggest challenge
4. responsibilities
5. what if someone leave different idea in feedback when you were doing pull request code review
6. how do you solve bugs

#### Hailey - Walmart - 10/04/2024

1. self intro
2. react version
3. spring boot version
4. node js exp
5. suspense
6. error boundary
7. promise
8. callback function
9. CORS
10. virtualization in react
11. git fetch vs git pull
12. mockito exp
13. what is spy
14. what annotations you used in spring boot
15. explain dependency injection
16. circular injection
17. factory pattern
18. coding: input String s, List<Integer> roll.

example1

```
s = "abc", roll = [3, 1]
increase first 3 characters by 1 "abc" -> "bcd"
increase first 1 character by 1 "bcd" -> "ccd"
```

example2

```
s = "zzz", roll = [1, 2]
increase first 1 characters by 1 "zzz" -> "azz"
increase first 2 character by 1 "azz" -> "baz"
```

#### Cayden Liu - Walmart Final 30 mins - 10/02/2024

1. self intro
2. project intro
3. what is the project data flow
4. the difficulties you face
5. dive into project details

#### Cayden Liu - Walmart - 10/01/2024

1. self intro/ project
1. useEffect in react
2. virtualization in react
3. pagination in react
4. useParams in router
5. CORS issue
6. Spring boot, dependency injection
7. ways of dependency injection
8. abstract factory pattern
9. @PathVariable @RequestParam
10. JWT
11. git fetch vs git pull
12. unit test library
13. @Spy

## Split the Array

### Description

The array  $nums$  contains  $n$  non-negative integers. Determine the number of ways to split  $nums$  into three non-empty subarrays  $A_1$ ,  $A_2$ , and  $A_3$  with the sum of elements of the subarrays as  $S_1$ ,  $S_2$ , and  $S_3$  respectively, such that  $S_1$ ,  $S_2$ , and  $S_3$  satisfy  $S_2 \leq S_1 + S_3$ .  
Return the value modulo  $(10^9 + 7)$ .

### Note:

- A subarray is a contiguous part of an array. For example, the subarrays of  $[1, 2, 3]$  are  $[1]$ ,  $[2]$ ,  $[3]$ ,  $[1, 2]$ ,  $[2, 3]$  and  $[1, 2, 3]$ .
- All three subarrays should contain at least 1 element.
- Each element of the array must belong to one of the three subarrays  $A_1$ ,  $A_2$ , or  $A_3$ .
- The subarray  $A_1$  must be followed by subarray  $A_2$  which must then be followed by  $A_3$ . For example in the array  $[1, 4, 3]$  it is not possible to take  $A_1 = 1$ ,  $A_2 = 3$ , and  $A_3 = 4$ . We must take  $A_1 = 1$ ,  $A_2 = 4$  and  $A_3 = 3$  only.

### Example

$nums = [1, 2, 3, 4]$

14. Coding: There are 3 valid solutions:

```
Java 17 Autocomplete Loading... 
1 > import java.io.*;-
14 class Result {
15
16     /*
17      * Complete the 'countWays'
18      * function below.
19      *
20      * The function is expected
21      * to return an INTEGER.
22      *
23      * The function accepts
24      * INTEGER_ARRAY nums as
25      * parameter.
26
27     */
28
29
30
31 > public static int countWays
(List<Integer> nums) {
// Write your code here
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59 > public class Solution {
```

Hailey - Walmart - 09/30/2024

1. self intro
2. what you good at / your interest on
3. why microservice
4. why spring
5. have you used aop, where
6. why spring use IOC
7. what's your frontend skill set
8. how does kafka work(at least one)
9. coding: two String, s1, s2, s2 length = 3,  $1 \leq s2$  length  $\leq 10^5$ . count the number of times s1 appear as substring in s2
10. draw project architecture on whiteboard

Esther Gong - Macquarie Group - 09/30/2024 - UI Developer Position

1. self intro
2. ngrx experience
3. how to change the state in ngrx
4. react experience
5. different types of directive
6. pure pipe vs impure pipe
7. let vs var
8. when dealing with number in js, what should be careful
9. switchmap
10. takeUntil
11. async pipe
12. git/ jira
13. how to solve the git conflict

14. git rebase
15. how to test async code
16. CSS, SCSS
17. flexbox
18. **BEM** (Block, Element, Modifier)
19. SOLID principles
20. abstract vs encapsulation
21. design pattern
22. responsible UI experience

### Calvin Tran - Walmart - 09/27/0204

1. self intro
2. what is factory pattern
3. diff abstract factory and factory method
4. diff promise and callback function
5. what design patterns you used in react
6. what is error boundary
7. what is suspense
8. diff git pull and git fetch
9. diff path variable and request param
10. is request param mandatory
11. have you used mockito before
12. coding: minimum length of subarray with at least k unique elements
13. coding: group anagram, return size of the group

### Stella Huan - Finra - 09/27/2024

1. self intro

Here is a card game called "Deck Duel" where two types of professionals, a Doctor and an Engineer, draw cards from a deck to accrue points.

**Game Specification:**

**1. Players:**

- Doctor: Specializes in drawing hearts and diamonds.
- Engineer: Specializes in drawing clubs and spades.

**2. Cards:**

- The deck consists of standard playing cards (52 cards, divided into four suits: hearts, diamonds, clubs, and spades).

**3. Scoring Rules:**

- When a Doctor draws a heart with an even number, they score 10 points; if an Engineer draws the same, they score 5 points.
- When a Doctor draws a diamond with an odd number, they score 8 points; an Engineer scores 4 points.
- Conversely, when an Engineer draws a club with an even number, they score 10 points; a Doctor scores 5 points.
- When an Engineer draws a spade with an odd number, they score 8 points; a Doctor scores 4 points.

**4. Gameplay:**

- Which player gets to draw a card first is decided among the players before the game starts.
- Players take turns drawing from a shuffled deck until all cards are drawn.
- Calculate scores based on the card's suit and number in relation to the player's profession.
- Determine the winner based on the scores accrued.

**Objective:**

In this exercise, we would like you to prioritize designing a robust and comprehensive solution architecture for the given problem. Focus on defining the overall structure, including classes, interfaces, and method signatures. Aim to create a skeletal but complete design in Java code, rather than trying to implement every method in detail.

Your design should illustrate how the various components interact and how you plan to address the core requirements. The goal is to demonstrate your understanding of object-oriented principles and your ability to create an extendible and maintainable solution.

Remember, the clarity and completeness of your design in Java code are more important than fully implementing every detail.

2. Good luck!

### Sam Hsiao - Walmart - 09/27/0204 - manager 30 min round

1. self intro
2. previous exp
3. based on project - why use nosql, why use kafka
4. challenge in last project
5. how do you handle conflicts with team member

## Cayden Liu - Walmart - 09/27/2024

1. self intro
2. project
3. sql vs no-sql
4. what are microservices
5. kubernetes
6. use python to code

```
# Online Python – IDE, Editor, Compiler, Interpreter
Roman numerals are represented by seven different symbols: I, V, X, L, C, D and M.
Symbol      Value
I            1
V            5
X            10
L            50
C            100
D            500
M            1000
For example, 2 is written as II in Roman numeral, just two ones added together.
12 is written as XII, which is simply X + II.
The number 27 is written as XXVII, which is XX + V + II.
```

- 7.

## Jian ma - Goldsman - 09/26/2024

1. Self intro
2. What is injection
3. What is DI
4. What are ways to inject beans
5. Why do you use spring boot
6. What are features you used in spring boot
7. How to authenticate a user
8. How do you do security
9. What is jwt
10. How does jwt generate
11. How do you do testing
12. How do you test your apis
13. How can you pass value into your apis when you are using postman
14. How do you do validation
15. What database you used
16. What is cassandra
17. What is partition key
18. Who picks partition keys
19. What is acid
20. How does atomic, isolation, durable work in cassandra
21. If you have 10 m user records and you have 5 partition, how can you distribute loads into your database
22. What is @Async annotation in spring boot
23. How do you use kafka

## Jolene Pei - paypal - 09/25/2024

1. Self intro
2. what is @autowired
3. what is @Qualifier
4. what are http methods
5. diff post and put

6. when will you use patch
7. coding: design following rest api
  - a. list all books
  - b. get books by author
  - c. get books by category
  - d. create new book
  - e. get top 100 books
8. coding: 3 sum , return all unique combination
9. coding: merge interval
10. dry run your code

#### Sam Hsiao - Walmart - 09/24/2024

1. self intro
2. git pull vs git fetch
3. git process
4. git add vs git commit
5. git stash
6. spring boot, java version you worked on
7. annotations in spring boot
8. Circular dependency injection
9. dependency injection
10. CORS problem
11. js: Promise
12. Redux
13. Hook
14. sharding
15. what is the websocket
16. setTimeout(js)
17. what is npm
18. what is package.json used for?
19. pathvariable vs requestparam
20. design pattern: Factory
21. Coding: Group Anagram
22. react optimization
23. what are hooks in react and explain them
- 24.

#### Sophia Zhang - paypal - 09/23/2024

1. Self intro
  2. given react project, including few components, and a poll data structure
    - a. poll manager components contains <vote> and <result> child components
    - b. vote component will vote for this person
    - c. view result will display result on the page
- there are 3 buttons, 2 buttons to vote for different people, another button to see result  
you need to modify project and implementing following features
- a. click vote button will increment votes for the specific person
  - b. click view result button will display result for votes and should disable all buttons

Design a Poll Manager app that allows users to vote for one of two options and view the winner. The app should be built using TypeScript and React. Certain core React functionalities are already implemented.

The application has three components: *PollManager*, *Vote*, and *Results*.

The *Vote* component should

- display the text of each option.
- provide a "Vote" button for each option to cast a vote.
- disable the "Vote" button for each option when the winner is declared.

The *Results* component should

- display the current leader or tie status.
- display the result text based on the following conditions.
  - If no votes have been cast, the result text should be empty: "".
  - If the same number of votes have been cast for both options, the result text should be: "It's a tie".
  - If one option has more votes than the other, the result text should be: "{LEADER} is leading by {VOTE\_DIFFERENCE} vote(s)".
- The "View Winner" button:
  - If no votes have been cast, the button should be disabled.
  - Clicking the "View Winner" button should:
    - display the result text: "It's a tie" if the same number of votes have been cast for both options.
    - display the result text: "{LEADER} won by {VOTE\_DIFFERENCE} vote(s)" if one option has more votes than the other.
    - disable the "View Winner" button
    - disable the "Vote" button for each option

```
export const pollData: Poll = {
  question: 'Who is your favorite superhero?',
  options: [
    { id: 1, text: 'Superman', votes: 0 },
    { id: 2, text: 'Batman', votes: 0 },
  ],
};
```

2. What is the o/p of the following program.

```
public class PayPalTests {  
  
    public static void main(String[] args) {  
        System.out.println("a");  
        try  
        {  
            System.out.println("b");  
            throw new IllegalArgumentException();  
        }catch (IOException e)  
        {  
            System.out.println("c");  
        }  
        catch(Exception e)  
        {  
            System.out.println("c");  
        }  
        finally  
        {  
            System.out.println("d");  
        }  
        System.out.println("e");  
  
    }  
}
```

=====

what could be the o/p,  
public class PayPalThread {  
  
 public static int counter = 0;

```
    public void counter() {  
        for (int i = 0; i < 1000; i++) {  
            counter++;  
        }  
    }
```

```
    public static void main(String[] args) {  
        // i could recollect same syntax below, there could be little change as well  
        // basically when you start 2 threads , and call join method on each thread, in this  
        scenario,what happens to counter value.  
        var thread1 = new Thread(() -> {new PayPalThread().counter();});  
        var thread2 = new Thread(() -> {new PayPalThread().counter();});  
        thread1.join();  
        thread2.join();
```

```
        thread1.start();  
        thread2.start();  
        System.out.println("Counter after threads finish: " + counter);  
  
    }  
}
```

=====

what would be the o/p  
class T {  
 public static void main() {  
 print("begin")  
 T t = new T();

```

        t = null;
        System.gc();
        print("end")
    }
    public void finalize() {
        print("this is finalize")
    }
}
=====
what would be the o/p
class T {
    static T t;
    public static void main() {
        print("begin")
        t = new T();
        t = null;
        System.gc();

        t = null;
        System.gc();
        print("end")
    }
    public void finalize() {
        print("this is finalize");
        t = this;
    }
}
=====
```

what would be the o/p

```

class Test extends Thread {
    public void run() {
        print(..);
    }
    public static void main() {
        Test t = new Test();
        t.run();
    }
}
```

how many thread created in this program?  
how to fix it

=====

what would be the o/p

```

class Parent {
    public Parent() {
        System.out.println("Parent");
    }
}
class Child extends Parent {
    public Child() {
        System.out.println("Child");
    }
}

main() {
    Parent p = new Parent();
    Child c = new Child();
}
```

=====

what would be the o/p

```

class Parent {
```

```

public void print() {
    System.out.println("Parent");
}
}

class Child extends Parent {
    public void print() {
        System.out.println("Child");
    }
}

public class PaypalPoly {

    public static void main(String[] args) {
        Parent parent = new Parent();
        parent.print();

        Parent child1 = new Child();
        child1.print();

        Child child2 = new Child();
        child2.print();
    }
}
=====
```

Design API for banking system. only API design, no coding needed.

- 1) deposit
- 2) withdrawal
- 3) transfer

At the end few scenarios - how do you handle transactions when 2 DBs involved, like deposit occurs on one DB  
withdrawal occurs on another DB.

```
=====
Design API for students and courses
student
courses.
```

### Max Ma - Goldman Sachs - 09/19/2024 - round 1

1. Self intro
2. leetcode : 4. Median of Two Sorted Arrays
3. leetcode: 42. Trapping Rain Water

### Sophia Zhang - paypal - 09/17/2024

1. Self intro
2. == VS ===
3. why interface
4. generic in typescript
5. how does js asynchronous work
6. how does event loop work
7. why typescript
8. var vs let vs const
9. what is modules in js
10. client server mode
11. undefined vs null
12. what is elastic search
13. docker and kubernetes

#### 14. coding1:

```
1 // What will console.log print for the following code?
2 // What if the comparison is '===' instead
3 let s = "3"
4 let x = 3
5 ↘ if (s == x) {
6   console.log("dog");
7 ↗ } else {
8   console.log("cat");
9 }
0
1 // Are there any safety issues with the following code?
2 const name = user.personalInfo.name;
3
4 // Complete the following function to create a deep clone of a JSON object
5 function deepClone(obj) {
6
7 }
8
9 // Given a string, write a function counting occurrences of each character in the string
10 function countCharacterOccurrences(str) {
11
12 }
```

#### 15. coding2:

```
// This code should filter the array but has an error, find and fix it
const numbers: number[] = [1, 2, 3, 4, 5];
const evenNumbers = numbers.filter((num) => num % 2);
console.log(evenNumbers);

// What is optional chaining? How can we simplify line 13 using it?
const user = {
  personalInfo: {
    name: 'Bob'
  }
}
const username = user && user.personalInfo && user.personalInfo.name;

// The following code has an error on line 19. How could we fix this error using union types?
let stringOrNumber: string;
stringOrNumber = 'hello';
stringOrNumber = 3;

// Please complete the following code for the "getArea" function
interface Circle {
  kind: 'circle';
  radius: number;
}

interface Square {
  kind: 'square';
  sideLength: number;
}
```

```

// Please complete the following code for the "getArea" function
✓ interface Circle {
    kind: 'circle';
    radius: number;
}

✓ interface Square {
    kind: 'square';
    sideLength: number;
}

const myCircle: Circle = { kind: 'circle', radius: 5 };
const mySquare: Square = { kind: 'square', sideLength: 4 };

Interviewer 1
type Shape = Circle | Square;

function getArea(shape: Shape): number {
}

console.log(getArea(myCircle)); // 78.53981633974483
console.log(getArea(mySquare)); // 16

```

### 16. coding3:

```

var arr1 = [{id: 1, x: 1}, {id: 2, x: 9}];
var arr2 = [{id: 3, x: 5}];
var arr3 = [{id: 1, x: 2, y: 3}, {id: 2, x: 3, y: 6}];
var arr4 = [{id: 2, x: 10, y: 20}, {id: 3, x: 0, y: 0}];
var arr5 = [{id: 1, b: {b: 94}, v: [4, 3], y: 48}];
var arr6 = [{id: 1, b: {c: 84}, v: [1, 3]}];

✓ /**
 * @param {Array} arr1
 * @param {Array} arr2
 * @return {Array}
 */
var join = function(arr1, arr2) {

};

console.log("1: ", join(arr1, arr2)); // Output: [{id: 1, x: 1}, {id: 2, x: 9}, {id: 3, x: 5}]
console.log("2: ", join(arr3, arr4)); // Output: [{id: 1, x: 2, y: 3}, {id: 2, x: 10, y: 20}, {id: 3, x: 0, y: 0 }]
console.log("3: ", join(arr5, arr6)); // Output: [{id: 1, b: {c: 84}, v: [1, 3], y: 48}]
|

```

## 17. coding4:

A subsequence of a string is obtained by deleting zero or more characters from the string while maintaining order. Given a string, generate an array of all subsequences sorted alphabetically ascending, omitting the empty string.

### Example

$s = "xyz"$

Not including the empty string, the sorted subsequences are  $["x", "xy", "xyz", "xz", "y", "yz", "z"]$ .

### Function Description

Complete the function *buildSubsequences* in the editor below.

*buildSubsequences* has the following parameter(s):

*str s*: the string to process

### Returns

*str[]*: an array of strings comprising all subsequences of the given string sorted alphabetically, ascending.

### Constraints

- $1 < \text{length of } s < 16$
- $s$  is a string of distinct lowercase English alphabetic letters  $\text{ascii}[a-z]$ .

Sophia Zhang - paypal - 09/16/2024

1. Self intro
2. Project intro
3. What were your responsibilities
4. The biggest challenge
5. Follow up question on challenge: what is your data volume.
6. Do you know what id LRU
7. Coding: LRU cache

Maggie Yang - Finra - 09/10/2024

1. self intro
2. most recent project
3. where do you see yourself in 5 years later
4. unit test experience
5. data volume of your project

6. coding:

```
4 /* Name of the class has to be "Main" only if the class is public.
5 Consider a phone dial pad:
6
7 -----
8 |   | ABC | DEF |
9 | 1 | 2 | 3 |
10 -----
11 | GHI | JKL | MNO |
12 | 4 | 5 | 6 |
13 -----
14 | PQRS | TUV | WXYZ |
15 | 7 | 8 | 9 |
16 -----
17 |   |   |   |
18 | * | 0 | # |
19 -----
20 Given a list of words and a 7 digit integer, return the words that could be written by typing the integer on the dial pad.
21
22 For example, given the list of words:
23 "foo", "dog", "mom", "bar", "ninja", "potato", "cap", "cry"
24
25 And the number:
26 3662279
27
28 Return:
29 "foo", "bar", "cap", "cry"
30
31 This is because the words can be typed out using the following correlating numbers when using the dial pad:
32 [366, 227, 227, 279]
33 and all of these numerical phrases are subsets of the given phone number. |
```

7.

Frank Ge - Expedia - Aug/12/2024

1. self intro && previous project
2. team size
3. daily work
4. give me a production support example
5. your responsibilities in project and talk me about data flows
6. how did you monitor system
7. daily user of your application

Sarvan - Walmart - 09/06/2024

1. self intro
2. kafka, react experience
3. db experience
4. kafka optimization
5. kafka compression type
6. in React, what is the props and states
7. coding: Returns true if the first string is a subsequence of the second string. IC.
8. CICD pipeline experience
9. Code report - SonarQube

Austin Wang - BNY Mellon - 09/06/2024

1. Why choose us? What do you know about our project?
2. self intro and tech skills
3. most recent project
4. EC2 vs ECS
5. how to scale
6. microservice experience

Stella Huan - Walmart - 9/5/2024

1. self intro
2. project detail
3. code coverage. code review

4. microservice - circuit breaker
5. kafka experience
6. how kafka is used in the project
7. kafka optimization
8. coding: Returns true if the first string is a subsequence of the second string. IC.

### Maggie Yang - Finra - 09/05/2024

1. self intro/ project
2. sql query

<i>t_orders</i>	<i>t_customers</i>
id   bigint(20)	+---->  id   bigint(20)
order_id   bigint(20)	name   varchar(100)
customer_id   bigint(20)  -----+  contact_name   varchar(100)	
employee_id   bigint(20)	address   varchar(255)
order_date   varchar(255)	city   varchar(40)
shipper_id   bigint(20)  ---+  zip_code   varchar(20)	
+-----+-----+      country   varchar(40)	
	+-----+-----+
<i>t_shippers</i>	
+-----+-----+	
id   bigint(20)  <-----+	
name   varchar(255)	
phone   varchar(20)	
+-----+-----+	

- a. which country has the most orders
- b. create a new table with columns: **country**, **order\_vol** (representing the number of orders), and **vol\_category**, where the category is based on the following conditions:
  - Less than 5 orders → **low**
  - Between 5 and 15 orders → **medium**
  - More than 15 orders → **high**

For a given stock, we are given a list of stock prices that increment every minute. For example:

Prices(p): [4, 5, 9, 5, 3, 9, 2, 6]

Determine the biggest profit you can make through one buy and one sell  
 Solution: Buy at t=4, Sell at t=5, Profit = t(5) - t(4) = 9 - 3 = 6

3.

### Austin Wang - BNY - 08/30/2024

1. self intro/ project intro
2. interesting thing you have done
3. what do you learn from it
4. Coding: binary tree, common ancestors
5. spring AOP
6. what is ngrx

### TK Chu - Finra - Aug/27/2024

round1

1. self introduction
2. in stored procedure how to read table
3. after for loop cursor what will you do(close cursor)
4. any exp on trigger , stored procedure, function, package
5. give me a use case of trigger
6. how does trigger work
7. if you cannot write trigger, how do you impl same logic
8. how to design one to many and many to many tables

9. for emp table empid, empname, salary, ssn, what index will you apply
10. why hibernate
11. diff jpa and hibernate, can you use them together
12. have you used @query
13. if a query was about 5s to return data, then one day it needs 2min, how do you debug
14. what could be the reason and how do you solve it

**round2**

1. self introduction and project
2. what design patterns you used in your project
3. where did you use kafka and what mq can we replace kafka
4. what db used in your last project
5. have you changed isolation levels before
6. what is finally block
7. what is bigdecimal
8. and what's the limit of it
9. have you used @ParameterizedTest, and how to use it
10. how to test backend api
11. have you used angular before
12. have you done angular test before
13. what's your strong point
14. diff having and where
15. can you come up a question about having for junior developer

**round3**

1. self introduction
2. can subclass has larger access modifier scope or smaller and why
3. why stream api
4. what is lambda expression, what's the limit
5. why not using for loop
6. js vs java
7. what java package have you used before
8. what is IOC
9. why IOC
10. if we have a for loop and collection.get(idx), will you prefer get(idx) or iterator

**round4**

1. self introduction
2. project architecture
3. do you have cloudfront
4. team size
5. what was your responsibilities
6. daily work
7. how did you get ticket / story
8. any static code report
9. how to create clean code
10. if you are going to modify someone's code , and it doesn't have any test cases, what will you do
11. how to test private method
12. what is mock
13. what do you think about code coverage, how much of it do we need
14. how to create clean test code/ good test
15. what test did you do in your last project
16. how many ways to inject bean in spring
17. what is serializable in java
18. how did you secure your rest api (private services)
19. what is auto scaling
20. diff horizontal and vertical scaling
21. why spring / spring boot

2. Stringbuilder vs StringBuffer
3. Volittle
4. Spring security
5. Did You use json or xml?
6. Normalization in database

Jolene Pei - Walmart - Aug/20/2024

**round1**

1. self introduction
2. coding: two sum
3. coding: invert binary tree
4. coding: lc654
5. what is cdc
6. how does spring load bean(reflection flow)
7. what is SAGA pattern
8. any azure exp

**round2**

1. self intro
2. system design: design shopping cart
  - a. high availability
  - b. high performance
  - c. 200 millisecond response time
  - d. 1 million active user

Travis Wang - Walmart - Aug/15/2024

1. self introduction
2. describe a feature you done in your recent project
3. daily work
4. how to deploy
5. load of your project
6. post production
7. if user send similar query to your application, do you have anything to speed up the process
8. which cache strategy did you use
9. if 3rd party api not working, any mechanism to handle it
10. how does hystrix work
11. messaging queue experience

TK Chu - Finra - Aug/15/2024

1. self introduction
2. coding: longest substring without duplicate
3. dry run your code(like debug mode)
4. what is foreign key
5. can we delete primary key before removing foreign key data
6. ui exp
7. stored procedure experience
8. where did you use trigger in your previous project

John Doe - Apple - Aug/13/2024

1. self intro && previous project
2. What's your latest tech stack for?
3. What's your experience in AWS?
4. If we have an on-premise server, if the server is running slow, what could be the problem?
5. How do you solve it?
6. What kind of sql query will be slow queries?
7. How do you identify it?
8. Coding: given a string "red blue green yellow black", return a map of color - frequency of color
9. (follow up) Coding: sort the map based on value

John Doe - Apple - Aug/13/2024

10. self intro && previous project
11. What's your experience in sql database?
12. What's your experience in mongoDB?
13. What's your experience in kafka?
14. Coding: given String "[[1,2,3], [4,5,6]]", return a list of lists of Strings
15. Coding: in sql database, given a lecture table, student table and registration table, return a list of lectures with number of students registered. Please include those lectures in which no one registered.

#### **John Doe - Fannie Mae - Aug/12/2024**

1. self intro && previous project
2. Coding: giving a list of products with id, category and price, return a map where key is the category and value is number of different products. Do it in stream API.
3. (follow up)Coding: Filter out price more than 5 dollars
4. Coding: unit test with Mockito
5. What is SQL database
6. What's the difference between Left join and Left outer join
7. What noSQL database have you worked with?
8. What microservice architecture have you implemented?
9. What is strict mode in javascript?
10. What is lazy loading? How do you do it in Angular?
11. What's bootstrap in Angular main.ts?
12. What services have you worked with?
13. Given 5GB data in S3, how do you load it to a postgreSQL?
14. (follow up) Why lambda instead of an API for this data migration?

#### **Austin Wang - Expedia - Aug/12/2024**

1. self intro && previous project
2. How do you separate modules when you migrate from monolithic to microservice?
3. What kind of data migration have you done in your last project?
4. What's your experience in aws?
5. What's your experience in cloudformation?
6. What's your experience in automation testing?
7. what's your typical day look like?
8. How do you collaborate with your teammates?

#### **Frank Ge - Expedia - Aug/12/2024**

1. self intro && previous project
2. project details
3. explain one of your data pipeline you worked on
4. how to keep application operational stable / healthy
5. do you follow agile scrum
6. have you mentored other people before
7. what was your most proud thing

#### **Namratha Gonuguntla - 08/12/2024**

1. self intro
2. project 1
3. project 2
4. Coding:

```
//method called public void identify pairs : take 2 parametrts (list of integers and integer value)
//Sample input : numbers = 8, 2, 4, 6, 1, 4
//Sample input 2 = value= 8
// Identify all the pairs that add up to that value
// Just all the pairs in each line 6, 2, next line 4 ,4
```

### TK Chu - BNY - Aug/05/2024

1. self introduction
2. explain security flow
3. explain project architecture
4. who is the client
5. who use application
6. details about some data flow
7. coding: three sum
8. coding: swap 3 numbers without using temp value

### TK Chu - BNY - Aug/05/2024

1. coding1: change minimum number of character to make string with no same chars next to each other
2. coding2: send request to endpoint: [http://hackerrank/..](http://hackerrank/)
  - a. cannot use resttemplate , use urlconnection
  - b. can use gson to convert response json to object
  - c. need to get all pages result
  - d. filter result by some conditions
  - e. return sum of valid count

### Austin Wang / Frank Ge - Expedia - Aug/05/2024

1. self intro && previous project
2. Mongodb VS Dynamodb
3. how to do cache on mongodb
4. Mongodb vs SQL
5. what is kubernetes
6. what is container in kubernetes
7. what project(s) is based on Spring in your previous experience
8. (follow up) What's the responsibility for your promotion service?
9. What automation testing/suit have you built before?
10. How did you use kafka in your previous project?
11. how kafka partition strategy
12. what is zero copy in kafka
13. What is the difference between SQS and Kafka?
14. Coding: A unique character is one that appears only once in a string. Given a string consisting of lowercase English letters only, return the index of the first occurrence of a unique character in the string using 1-based indexing. If the string does not contain any unique character, return -1.
15. (more than dozens of follow up for your coding: time complexity, space complexity, corner case, for the unique character for only lower case, ensure to use a fixed array rather than a map. Then there is also follow up on how to use ascii code to validate the lower case only. then how about exception throwing, how will you handle any of those corner case via exception throwing)
16. Coding: convert coding to stream api
17. SOLID

### TK Chu - BNY - Aug/05/2024

1. self intro && previous project
2. education background(timeline from bachelor to master)
3. what is volatile
4. if we have number i = 0 , two threads, one change, one read, what value will the reader thread get.
5. if we put volatile , what will happen
6. if we put atomic integer, what will happen
7. how does atomic integer work
8. time complexity of hashmap
9. how does hashmap work
10. concurrenthashmap vs hashmap
11. how does concurrenthashmap work
12. what is Redblacktree

13. what is balanced tree
14. why balance tree / red black tree
15. if we have a ticket selling system, two user buy ticket from system, and it oversales the ticket, how do you modify it and change the database structure
16. how does optimistic lock work
17. how database prevent two users update same row
18. what locks does database have
19. what does exclusive lock do
20. if we click browser , <https://www.amazon.co.uk/> , what will happen
21. what does dns do
22. why we want 3rd party dns
23. how does ip packets transferred to uk (ip connection)
24. where is cable
25. is there any services before server (CDN/Security/Nginx)
26. what is CDN
27. which datacenter will use visit , UK or US CDN?
28. what is Nginx
29. what is https , the S
30. besides the encryption , any other benefits from HTTPS
31. what is certificate , why we need it

#### **Stella Huan - National Science Foundation - Aug/01/2024**

1. self intro && previous project
2. How to ensure high availability and failure tolerance in microservice architecture?
3. how to implement microservice architecture in spring cloud?
4. Other than EKL, have you used anything else in Spring Cloud for logging and monitoring?
5. How to ensure high availability in AWS?
6. how to restrict access for components in Angular ?
7. How to manage states in React?
8. How to improve the performance of the React loading if the loading is slow?

#### **John Doe - Apple - Aug/01/2024**

1. self intro && previous experience
2. How are you going to design an API for getting employees based on employee ID?
3. (follow up) Where are you going to start from for this design?
4. (follow up) How are you going to connect databases in your spring boot project?
5. (follow up) how are you going to connect to database with different data source, such as postgresSQL and MongoDB
6. (follow up) Say we have a couple of requests sending for the above business, one sent to MongoDB, one sent to PostgreSQL, another sent to another service, how are you going to do it?
7. (follow up) If you are going async, how are you going to do it in spring boot?
8. (follow up) if you are using CompletableFuture, how are you going to execute them?
9. (follow up) if you need to chain those CompletableFuture up, how are you going to do it?
10. Coding: given a list of string, i.e. "Max Tong", "John Doe", returns a map where the key is first name and value is last name.
11. Coding: given a BinarySearchTree root and a target value,, return a list of integers which contains all the values for you to go through from root to target.
12. Coding: Given a 2d string array, where ["Fruit", "Apple"], ["Fruit", "Banana"], ["Fruit", "Orange"], ["Car", "Ford"], ["Car", "BMW"], ["Car", "Mercede"], return a map where key is the category and value is list of all real items.

#### **Namratha Gonuguntla - 7/31/2024**

1. coding: fibonacci sequence
2. coding: char frequency in string

#### **Max Tong - SWIFT - 7/26/2024**

1. machine learning experience
2. AWS experience
3. Python experience
4. python framework in data analysis

5. cloudformation
6. open to relocate?

### Karina Shen - 7/26/2024 - Capital One

1. self intro
2. coding: Given a json string with multiple entities with each of them containing three fields: id, dependencies and waitPeriod. You need to find out the execution order for all those entities

### Max Tong - 7/25/2024 - SWIFT

1. self intro
2. Future vs CompletableFuture pros and cons
3. BiConsumer
4. static block vs init block
5. Coding: given an int array, find out all pairs in which sum is perfect square
6. Follow-up Coding: revise logic into stream API
7. When a task is close to deadline but cannot be accomplished, what are you going to do?
8. When there is a disagreement between team members, what are you going to do?
9. Why do you wanna have a change from current project

### Mike Shen - 7/19/2024 - Paypal

1. self introduction
2. if we have infinite list, how to find certain number from it
3. why spring boot
4. what is @springbootapplication
5. application properties vs yaml file
6. coding: LRU
7. coding: given list of arrival time and list of duration time. return maximum number of events we can host

### Max Tong - 7/19/2024 - Swift

1. self intro
2. Why are you looking for a new role?
3. Try with resource
4. Virtual thread
5. Ok to relocation
6. How do you learn new tech

### Mike Shen - 7/19/2024 - Paypal

1. self introduction
2. biggest challenge
3. how does hashmap work
4. if i have customized class as key , what to do with hashmap
5. loose coupling vs cohesion
6. if i want to have a cohesion implementation, how to do it
7. stream api
8. do you know reactive programming
9. why Spring
10. we have multithreading to use beans, will it be issues
11. what do you need to consider when you design Rest api
12. how to traverse a tree
13. scenario where you want to use pre-order, in order, post order
14. describe how will you create a dictionary in java
15. if we have exception in finally block, what will happen
16. if we want to sort list student, what should we do
17. tell me some design patterns you know
18. diff inheritance and composition
19. interface vs functional interface
20. what is redblacktree
21. sql vs nosql

22. diff post and patch
23. what is idempotent

### Austin - 07/17/2024 - Finra

#### 1st round

1. self intro
2. what working style do you prefer
3. interesting part of previous project
4. biggest challenge
5. coding question

Please write a method for me that takes an input string and returns an output that consists of characters that appear once and only once.  
EX: "ABCDEF" => "ABDF"

- 6.
7. Why stringbuilder?
8. Can you generate some test cases for above coding problem
9. what is dependency injection
10. downside of dependency injection

#### 2nd round

1. self intro
2. testing experience
3. TDD vs BDD
4. how do you choose the development framework
5. how do you analyze the risk in testing
6. how do you make sure your test cases covers all aspect
7. A service consumes B service's API, how to do end to end test
8. what is test pyramid strategy
9. what is your current role
10. How do you make sure the test case is reusable?
11. test suite takes more time running on jenkins, what strategy to use to reduce time
12. what is the most complicated sql query you wrote
13. sql vs no sql

### Austin - 07/09/2024 - Swift

1. self intro
2. Python experience
3. Machine learning Library
4. Machine Learning Model
5. Usage of machine learning in your project
6. most recent project
7. how many microservice are you responsible for
8. project microservice architecture
9. how do you rate your python skills out of 10

### Stella Huan - 06/26/2024 - finish line

#### 1st round

1. the case you think out of box to solve the problem
2. how do you learn new tech skills
3. your tasks are based on other teammates' tasks. But they can't finish on time, what do you do?
4. talk about a scenario you make decision between short term and long term goal
5. When you have conflicts with teammates in system architecture, you think you are right, he thinks he is right, what should you do?
6. agile experience
7. design experience

## 2nd round

1. self intro/ project
2. experience talk with diverse people, different level of communication
3. your role in the team
4. how do you deliver message that people doesn't want to hear, like can't finish task on time
5. design experience
6. team size
7. a time when you improve the performance of design
8. what is your strength
9. what is your weakness

### Karina Shen - 06/14/2024 - CVS

1. self intro/ project
2. Node js experience
3. spring bean scope
4. how to create a bean
5. how to define prototype
6. how to add log in all the incoming request
7. what is transient keyword in java
8. message queue experience
9. rabbitMQ vs kafka
10. System Design: User Management System

### namratha gonuguntla - 06/13/2024 - Walmart

1. self intro/ project
2. project follow up: rabbitMQ vs Kafka
3. challenge in moving rabbitMQ to kafka
4. java & spring boot experience
5. database used in your project
6. why index, how does it work
7. react experience

### Stella Huan - 06/11/2024 - Swift - onsite

1. draw your project microservice architecture on whiteboard
2. data flow between services, which database did you use?
3. how many Java lines did you write? How many classes have you created in recent projects?
4. how do these services communicate? Which ones use API calls? Which ones use other communication?
5. do you have multi-threading in your project? where do you use it?
6. how many threads did you use? What do these threads work for?
7. where to make sure thread-safe in your service, give more details, where do you add synchronization?
8. You talked about you used Kafka, how many partitions you have and how many brokers
9. what exception handler you created in your project?
10. how does one service know another service?
11. If you couldn't write data into database, what exception should be thrown?
12. After writing data into database, then database fails, we can restart it, what needs to do next to make everything back to normal?
13. how to use docker in microservice?
14. what do you do after you get a ticket?
15. final, finalize()
16. testing experience
17. Spring aop
18. spring security
19. a file has multiple fields, how to read a file and exclude one field in file using Java?
20. do you use linux command? give some examples and their command
21. in microservice architecture, how to figure out where the issue is?

22. what is your graduation year? When did you come to the US?
23. Why did you work when you were at school?
- 24.coding: first write a function to swap two int numbers. then write a main function to call this swap function; check if there is any issue, how to update codes if there's an issue

#### Eddy Han - 06/11/2024 - Walmart

1. Self intro
2. Project intro
3. Did you used java for your last project
4. What are rest api method you used
5. What are other methods you used
6. Put vs post
7. Is post idempotent
8. Is put idempotent
9. High level architecture of kafka
10. how does consumer get messages
11. how does consumer know what messages to get
12. how does kafka know message has been processed
13. Coding: leetcode 1268

#### Stella Huan - 06/04/2024 - Techwish

1. self intro
2. java experience
3. spring ioc
4. bean scope
5. annotation in spring
6. how to secure sensitive data
7. encryption
8. how to design the payment process
9. database experience, sql, nosql

#### Stella Huan - 06/04/2024 - Freddie Mac

10. self intro
11. coding: first Non Repeated Character

#### Ryan Wang - 05/24/2024 - eBay - 1st round

1. self intro/project
2. why change your job
3. Coding: group anagrams
4. Coding: Leetcode: meeting arrangement

#### Alex Wu - 05/17/2024 - final Ebay

1. self intro/ project
2. kafka exp
3. scala exp
4. spark exp
5. no-sql exp
6. ok to work offshore?

#### Jerry Jin - 05/17/2024 - final Ebay

1. self intro/ Project
2. kafka exp
3. scala exp
4. spark exp
5. what do you do if i give you a spark job, how do you start
6. what will you do if you get a tight deadline or changed deadline
7. can you work on some support , or on call

Alice Lin - 05/16/2024 - final Sam's Club

1. self intro/ project/ challenge

We need to design a data structure to represent a basic spreadsheet. This spreadsheet should store values in cells and allow us to manipulate them using two methods:  
get\_value(cell\_address): This method retrieves the value stored in a specific cell identified by its address (e.g., "A1").  
set\_value(cell\_address, value): This method sets the value of a cell at a given address. The value can be a number or text.  
Additionally, this spreadsheet should handle a simple formula (SUM). The formula can reference other cells within the spreadsheet using their addresses.

Alex Wu - 05/15/2024 - 1st round Ebay

1. self intro/ Project

```
// Given a integer dictionary A of unknown size,
// where the numbers in the dictionary are sorted in ascending order,
// determine if a given target integer T is in the dictionary.
// Return the index of T in A, return -1 if T is not in A.

// Assumptions
// dictionary A is not null
// dictionary.get(i) will return null(Java)/INT_MIN(C++)/None(Python) if index i is out

// Examples
// A = {1, 2, 5, 9, .....}, T = 5, return 2
// A = {1, 2, 5, 9, 12, .....}, T = 7, return -1

// Solution
/*
 * interface Dictionary {
 *     public Integer get(int index);
```

- 2.

Jerry Jin - 05/14/2024 - 1st round Ebay

1. self intro/ Project
2. coding: given a list interval, which has flight taking off time and landing time , give me a maximum number of flights in the sky at same time
3. coding: verify input is tree or not

Leo Liu - 05/06/2024 - Walmart

1. self intro && previous project
2. what is kafka
3. why we need to use kafka
4. What are the qps for your kafka MQ?

5. coding: leetcode 98

Harry Lu - 05/02/2024 - Vistrada LLC(vendor)

1. self intro/ project
2. any kafka exp
3. how kafka work
4. point to point vs pub sub mode of kafka
5. list vs set
6. difference between arraylist vs linkedlist
7. how does hashmap work
8. how does hashmap locate the slot of key
9. comparator vs comparable
10. how GC work
11. java 8 new features
12. java 8 interface new features
13. stringbuffer vs string
14. volatile
15. is volatile i++ thread safe
16. how do you design microservice
17. aws exp

Iran Niu - 05/01/2024 - 1st round Apple

1. self intro/ project
2. ElasticSearch experience
3. CICD experience
4. SDLC, what me the process from requirement collections to setting up a project
5. spring data jpa

```
class Solution {

    enum FoodType { FRUIT, VEGETABLE, MEAT, DAIRY }
    private static Map<String, FoodType> basket = Map.of(
        "strawberry", FoodType.FRUIT,
        "broccoli", FoodType.VEGETABLE,
        "banana", FoodType.FRUIT,
        "yogurt", FoodType.DAIRY,
        "kale", FoodType.VEGETABLE,
        "cheese", FoodType.DAIRY
    );

    public static void main(String[] args) {
        // 1. Get the number of type VEGETABLE in the basket.
        long count = 0; //?

        // 2. Get all FRUIT items' names in the basket.
        Set<String> fruitNames = Set.of(); //?

        // 3. Get all FRUIT items' names in the basket that have "berry" in the name.
        Set<String> berryNames = Set.of(); //?
    }
}
```

6. coding:

Jerry Jin - 04/18/2024 - 2nd round eBay

1. self intro/ Project
2. coding: design a class /data structure which takes user requests at any time. User will send "person id, travel start time, travel end time"  
x, 12, 2  
x, 2, 4  
and provide a function to query average travel time by using person id.
3. follow up: what if user travel info / avg time needs to be reset every day.

### Jerry Jin - 04/18/2024 - 1st round eBay

1. self intro/ Project
2. input int array and int k  
you can pick number from begin or from end, return maximum sum  
[1, 2, 1] k = 1 -> return 1  
[1, 2, 1] k = 2 -> return 2
3. write Junit test case for it
4. write get / post http method for it
5. write Junit test for controller
6. maven build your application

### Andy Wu - 04/17/2024 - 2nd round eBay

1. self intro/ Project

```
Given n unique positive integers, number k and target, find all possible k integers where their
sum is target
1 <= k <= n
target > 0

Examples:
Input: [1,2,3,4], k=2, target=5
Output: [[1,4],[2,3]]

Input: [4,3,1,6], k=3, target=8
Output: [[1,3,4]]

Input: [1,3,4,6], k=3, target=2
Output: []
```

- 2.

### Victor Wei / jerry jin/andy chen- 04/16/2024 - ebay

1. self intro/ project
2. questions based on project introduction
3. what was your contribution to the project
4. are you part of on-call team
5. sli vs slo about on call
6. sli vs slo vs sla
7. how did you monitor and how did you do alert
8. how did you know if performance of application is good
9. biggest challenge about your on call things
10. what kind of alert you have created about your on call team
11. you said, you have worked on cassandra upgrade , diff key value store and document store
12. what is clustered index
13. diff partition key vs clustered key
14. devops / api / unit test which one you like most
15. 5 things you need to do to create a good api
16. what do you know about rate limiter
17. why do we need rate limiter
18. do you know TLS
19. what's your biggest strength
20. give me a situation where you don't know this technology or area, and they want you to start it as soon as possible
21. give me a situation where you did something more valuable to the project than others.
22. which area you want to grow, or you are not strong enough
23. what have you done to improve your weaknesses you mentioned before
24. tell me about the most positive experience in project
25. tell me about the most negative experience in project
26. biggest challenge(technical challenge)

1. self intro/ project
  1. Implement a REST API which takes in 3 values fromCurrency, toCurrency and amount and returns the final amount after conversion.
  2. Call the foreign exchange rate API to get the current exchange rate in realtime from <https://www.frankfurter.app/docs/>
  3. Response schema

```
{  "fromCurrency": "USD",  "toCurrency": "AUD",  "amount": 100}
```
- 2.

### Andy - 04/15/2023 - CVS

1. self intro && previous project
2. what's your tech skill sets
3. What are the features you developed?
4. Which java version have you used?
5. java 8 features
6. design patterns you used before
7. How do you prioritize multiple urgent tasks?
8. Do you know TDD?
9. How do you do pair programming?
10. How do you solve conflicts during pair programming?
11. agile/scrum experience
12. thread pool experience
13. SQL vs NoSQL
14. What is the motivation for your team to migrate from sql to nosql?
15. How do you migrate data records from sql to nosql ?
16. What's the most complicated thing you have done in React?
17. What do you think is good in angularJS which you wish to have in React?
18. Angular vs AngularJS
19. React Redux
20. What's your motivation in your career path?
21. How do you upskill around?
22. What is the factory design pattern?
23. Where did you use the factory design pattern?

### Heng-Yu Wu(Alex) - 04/12/2024 - Ebay

1. self intro && previous project
2. Can you introduce yourself?
3. Can you describe a previous project you've worked on?
4. What specific features did you build in this project?
5. How many people were on your team?
6. What does a typical day look like for you? How much time do you spend in meetings versus working?
7. What did you like most about your team?
8. What did you like least about your team?
9. Why did you leave your last project?
10. In the software technical aspect, which part do you like the most (e.g., system design, REST APIs, etc.)?
11. Under your preferred technical area, which component do you think is the most important?
12. Can you describe a challenging situation, how you solved it, and what was the impact?
13. What makes you think your team is a good team, and which part do you value the most?

**Grace Lu - 04/08/2024 - Walmart(Infobahn)**

1. self intro && previous project
2. diff microservice and monolithic
3. diff arraylist and linkedlist
4. how to keep thread safe
5. write rest endpoint for CRUD

**Marshall Jiao - 03/28/2024 - FHL bank**

1. self intro && previous project
2. Are you more on spring boot or angular expertise?
3. What are different directives we have in angular?
4. How do you pass value in between components in Angular?
5. How does a http request digest in a spring boot project in different layers?
6. async vs sync call
7. Are you familiar with the software development life cycle?
8. What is @Transcational annotation for?
9. What to design a restful endpoint?
10. What is the cloud platform you are using?
11. Any containerization experience?
12. What is AWS Batch and Step Function?

**Marshall Jiao - 03/25/2024 - Apple**

1. self intro && previous project
2. how good are you in mongoDB
3. are you familiar with any annotation-driven development with MongoDB
4. Have you written mongoDB native aggregation query?
5. Coding:  
input: Array of sorted integers  
output: Rearrange max/min in tandem/alternating  
Example:  
Input: {200, 312, .344, 422, 543, 678, 723}  
Output: {723, .200, .678, .312, .543, .344, 422}
6. mongoDB design: given a 10GB file, each line is a json document, how to insert it to mongoDB?

**Steven Song - 03/14/2024 - SWIFT - onsite**

1. talk about what you learn in the past 5 years
2. how did you use gRPC in your project
3. In your project, you use both Kafka and gRPC, show me the process where you use Kafka, and where you use gRPC and why
4. your project is based on spring, what annotation you use for RESTful API, what annotation you use for gRPC API?
5. For the service with gRPC, how do you initialize gRPC?
6. Name a gRPC error.
7. What does spring do while initialization?
8. How to create a docker image for a project?
9. Concatenate two strings without using java string api?

**Norfolk Hu - 03/13/2024 - Apple - onsite**

1. self intro && previous project
2. How do you handle metrics in your project? What are those metrics for?
3. How do you monitor high latency for your rest api?
4. How do you handle high latency?
5. Why Cassandra?
6. what is partition key/clustering key for in Cassandra
7. Can the partition key in Cassandra consist of multiple columns? What's the purpose for that?
8. In DB, if we need a field for order id, what data type should we use?
9. What if we cannot use UUID for order id?
10. What if int is not big enough for a unique order id?
11. what if we use zookeeper to keep global unique order id, what is the pro and cons?
12. Coding: two sum

### Steven Song - 03/12/2024 - Apple

1. self intro && previous project
2. What have you done in the team? web app development or something else
3. any database development on model?
4. What database have you used?
5. what is the traffic volume is in your project
6. what is the delay for your API
7. what are the things cost that delay
8. have you met any situation with a slow API
9. How do you fix it?
10. coding:

```
import java.io.*;
import java.util.*;
/*
*. Exercise-1: Create List of persons, print the object by sorting on .
firstname
Exercise-2: Put persons list into personDept Map and do couple of
looks up by Person object
*/
/*
class Solution {
public static void main(String [] args) {
}
}
class Person{
private String name;
private int age;
private String state;
}
class Dept {
private String name;
private int id;
}
11. git cmd
12. how to check out via git cmd
13. maven lifecycle
14. how to skip test via maven cmd
15. difference between maven compile vs install
16. where do you store log
17. How do you query splunk logs?
```

### Victor Wei - 03/11/2024 - Medimpact

1. Self intro
2. What are the parts you worked on?
3. Why did you use the circuit breaker?
4. When the circuit breaker happens, what messages do you return to the user?
5. How to manage state globally in react
6. How to create a new component in angular
7. What is http interceptor? What can you use it for
8. What is @scope annotation?
9. What are bean scope you know?
10. What is prototype bean scope
11. What are java classes you often to use in kafka
12. How do you create a kafka topic?
13. Suppose you have student and class classes, what is relationship between them
14. In sql, how do you count how many students are in a class?
15. What is aggregation function you can use for this sql
16. Put vs patch
17. Tell me what are steps to solve production issues in your work?
18. How do you send and consume third party apis?
19. How do you create an api that involves a database?
20. What are annotations you used in controller class?

21. Suppose you have an api that takes 2 minutes to finish its business logic, the requirement requires api should return to user immediately whenever user call this api, how do you do and what message you should return to user firstly? (hint: thinking about multithreading)
22. What part of previous work you are proud
23. Given a code: how many ways to eliminate if statements
 

```
if(a == 1) then
if(b == 2) then
if(c == 3) then
if(d==4) then
if(e == 5) then
```

#### Norfolk Hu - 03/04/2024 - Apple

1. self intro && previous project
2. cache experience in apple
3. cassandra experience in walmart
4. what is partition key for in cassandra
5. what is clustering column in cassandra
6. given a cassandra table with primary key as last name, how to query based on first name
7. how many nodes do you have for your cassandra db
8. coding: given a string "a##b", each "#" consider a deletion of one character before #. so the output in this case will be "b".

#### Alan Yu - 03/01/2024 - Compass Group

1. all following questions are coding questions
2. how do you do security
3. where did you and how did you store password
4. how does JWT work
5. how did you impl spring security
6. why security context
7. if we have different authentication ways, how do you implement them
8. but if we use providers manager it will loop through them all the time, anyway to speed up process
9. how do you configure spring security
10. how to configure filter in spring security
11. how do you verify roles of different user
12. how to write rest api to insert user
13. how to configure controller or write controller
14. how to handle different date format issue between client and server
15. what if some fields are mandatory cannot miss
16. how to configure different patterns for some field
17. how to write code connect to database
18. transaction propagation levels
19. coding: stream api, get max salary employee
20. coding: stream api, group employee by name , return map
21. can we pass in counter into stream api
22. can we throw exception in stream api
23. how to throw checked exception in stream api
24. how do we do it if we cannot use int counter in stream api
25. do you know CORS
26. angular exp
27. angular lifecycle hook
28. how to communicate between services in microservice
29. how to configure resttemplate
30. limitation of lambda aws
31. for aws lambda, if we have file from s3 to rds, file size is 2gb, can we use lambda
32. CICD pipeline

#### Ben Xu - 03/01/2024 - Walmart

1. self intro && previous project
2. no-sql database experience

3. coding: leetcode 116

**Jeffery Li - 02/27/2024 - Goldman Sachs - 4th round**

1. all following questions are coding questions
2. write java class and design producer and consumer, producer can send msg, and consumer can consume message
3. what queue should we use
4. class ProducerConsumer {

```
        private final BlockingQueue<Integer> queue = new ArrayBlockingQueue<>();
```

}
5. create consumer class to consume msg
6. any exception should we throw in consumer class , how to write it
7. create producer class to send 5 data into it
8. create a new consumer class, i want to return a value
9. create a method, and executor service, handle this new class and get result
10. write spring controller for CRUD (get, put, post, delete), including URL design and status code
11. what does request param look like in url
12. diff 200 and 201
13. sql select all account
14. sql select account group by city

**Jeffery Li - 02/23/2024 - Goldman Sachs - 3rd round**

1. questions related to your resume
2. have you used cdk aws
3. tell me serverless services
4. any cloudformation experience
5. what is step function
6. lambda maximum java package size
7. what languages can we use in lambda
8. how to run java in lambda
9. lambda limitation
10. how to package your java program and upload to lambda
11. how about the dependencies in your java application
12. what is sqs
13. diff queue and broker
14. what is dead letter queue
15. how do you send notification to developer if some errors in queue
16. how do you use cloudwatch monitor sqs(metrics)
17. what is event bridger
18. diff load balancers in aws
19. what are 2 most frequently used method in s3
20. how did you use aws in the work
21. how to create thread
22. disadvantage of runnable
23. how to get result from thread
24. how to reuse thread
25. what types of threadpool in java
26. what is the interface name of thread pool
27. if we create child thread in main thread, how to wait child thread in main thread
28. volatile
29. how to write jdbc
30. rest api method
31. how to configure version of api
32. filter
33. interceptor
34. what is aop
35. joinpoint
36. point cut
37. spring batch experience
38. log level

Dylan Ge - 02/21/2024 - Walmart - 1st round

1. Self intro/ Project
2. Java stream
3. Functional interface
4. What is Dependency Injection
5. What is functional test
6. How to do functional test
7. Spring bean scope
8. @Autowired, find by type, or name
9. Cassandra experience
10. Spring Data
11. Why JPA

Alex Wu - 02/20/2024 - Walmart - 1st round

1. Self intro
2. Experience with Cassandra, Kafka?
3. What is partition
4. Reactive programming experience

Jeffery Li - 02/20/2024 - Goldman Sachs - 2nd round

1. questions related to your resume
2. why microservice
3. and how to create one service in microservice
4. how do you use spring boot
5. why spring boot
6. how do you secure your rest api
7. if you using 3rd party package or 3rd party api, how do you prevent vulnerability
8. how to ensure code quality
9. if we have producer a , and consumer b, they are connected with non-secured protocol. how to we transfer data
10. are you good at data structure
11. hashmap vs hashtable
12. can we put null into hashmap or hashtable
13. if we put(null, 10) into hashmap, then put(null, 100) what will happen

Jeffery Li - 02/16/2024 - Plymouth Rock Assurance - 1st round

1. questions related to your resume
2. are still in current project
3. what's your role / responsibilities in your latest project
4. how does hashmap work
5. tell me about java immutable collection
6. how to write ajax call
7. how to handle CORS from ajax call
8. jquery before, after, append difference
9. when did you graduate
10. how to handle exception in spring boot
11. any aws experience
12. how to delete duplicate rows in sql
13. how to test application without code
14. your code coverage
15. how to make list thread safe
16. any problems we will meet when use java collections
17. how do you debug ajax in browser

Victor Wei - 02/16/2024 - Walmart

1. self intro && previous project
2. what is spring dependency injection
3. what is Optional.ofNullable
4. how to configure spring beans , what annotations you have used
5. where did you deploy it

6. diff singleton and prototype
7. how to handle bean confliction

**Jeffery Li - 02/16/2024 - goldman Sachs 1st**

1. questions related to your resume
2. how does gc work
3. what can you do if you get out of memory error
4. diff abstract class and interface
5. how does hashmap work
6. diff equals and hashCode
7. what is hash collision
8. if we need to use customer class as key , what should we do
9. tell me some hashmap methods you know
10. what is immutable in java
11. how to create immutable class in java
12. why immutable
13. diff abstract class and interface
14. diff hashmap vs concurrenthashmap
15. how does concurrenthashmap work
16. what aws parts have you used in daily life

**Hunter Zheng - 02/15/2024 - Walmart**

1. self intro && previous project
2. most recent project
3. role and responsibilities
4. architecture of your project
5. tech stack of your project
6. leetcode: 332

**Ray Guo - 02/15/2024 - Walmart**

1. self intro && previous project
2. kafka experience
3. what is purpose for using kafka
4. how does kafka solve peak
5. how do you determine number of partition and topic
6. what if kafka is during runtime and meets a high peak?
7. What if the Kafka downstream consumer is down?
8. what is your project architecture before kafka
9. CompletableFuture with Async experience (problem 8 follow up)
10. given a restful endpoint with latency, how will you improve performance
11. What is microservice?
12. How do microservice modules communicate with each other in your project?
13. pros and cons for microservice
14. Tell me what are the ways to make your web app fail tolerance? (database, microservice, cloud)
15. circuit breaker -> how do you use hystrix?

**Derek Shen - 02/15/2024 - Walmart**

1. self intro && previous project
2. kafka experience
3. reactive kafka
4. what is functional interface
5. how to send multiple requests if we don't need to wait for response
6. coding: /\* Count the number of "xx" in the given string. We'll say that overlapping is \* allowed, so "xxx" contains 2 "xx". \*/
7. coding: /\* Given start and end numbers, return a new array containing the sequence of \* integers from start up to but not including end, so start=5 and end=10 \* yields {5, 6, 7, 8, 9}. The end number will be greater or equal to the \* start number. Note that a length-0 array is valid. \*/

### James Wang - 02/14/2024 - Plymouth Rock Assurance - 1st round

1. self intro && previous project
2. what's your location
3. are still in current project
4. what's your role / responsibilities in your latest project
5. vector vs arraylist
6. tell me about java immutable collection
7. what is immutable in java
8. coding: given this picture , write pseudo code

```
test.txt
3 will
1 F
6 test
2 testing
10 code
4 coding

1
2 3
4 5 6
7 8 9 10

I will test code --
```

9. document ready vs windows loading jquery
10. how to write ajax call
11. how to write restful api in spring boot (from scratch)
12. challenge problem in your most recent project , and how did you resolve it
13. how did you handle exception in spring boot
14. how did you handle database transaction in spring boot
15. how does java handle fail fast vs fail safe

### Matthew Chen - 02/13/2024 - Walmart

1. self intro && previous project
2. coding: given int array, sort it in java
3. coding: remove duplicates in the sorted array

### Shrek Wang - 02/13/2024 - Apple

1. introduce myself
2. exp from the previous project
3. explaining specific application components on my resume and how they work in the flow.
4. Spring exp
5. Spring DI - and how do we implement let's say a Car class with type of Tesla and Ford; methods of start and end
6. Spring AOP exp
7. Spring AOP advice, before, after, joinpoint
8. Java collections and when you use them
9. HashMap, hashtable and hashing
10. How to iterate a collection
11. Multithreading executable, how to optimize?
12. Exp on MongoDB, how to handle multiple updates
13. data size and how to optimize the read and write?
14. coding question: Input "aabccc" output "a2b1c3" => then "aabcccabc" to "a2b1c3a1b1c1"

### Noah Cao - 02/13/2024 - Kroger

1. Ask resume: spring boot 2.5 -> 2.7 update
2. Snyk problem solving experience
3. Github actions and k8s experience
4. Non tech challenge

5. Tech challenge
6. Api endpoints not used any more, what to do?
7. Parallel coding experience
8. Is there any performance problem you solved when an api happens
9. Kafka zookeeper
10. How to solve the disagreement in team
11. How to solve everyone is remote then no well communications
12. All dev no QAs how to have great work
13. Do you have database design experience?
14. Do you know cassandra?
15. Do you know CAP theorem?
16. In java where to use Singleton design pattern?
17. How do you choose yourself? Backend or Frontend?
18. Any questions for us?

**Matthew Chen - 02/12/2024 - Walmart**

4. self intro && previous project
5. coding: design thread pool in java
6. array vs arraylist
7. what is blocking queue

**Marve Mao - 02/06/2024 - Plymouth CommonWealth(on-site)**

With TL round:

1. Introduction, explain recent project.
2. What tech stack used?
3. Questions about AWS, experience in AWS and use cases.
4. Questions about backend development, what kind of Microservices?
5. Questions about how services communicate?
6. Give a form of data, asked about how to get data if received a request from front end?
7. Based on the above, how to filter the data and return a count?
8. How to change data under specific needs?
9. Like the above form, if doing front end, how we create the form?
10. We want to click a button and update the form if we make a modification, how in angular?
11. If we want to click a button, update the form and show something based on the response from backend, how in angular?
12. If we want only one click, front end would send 3 http requests: 2 post and 1 get, how can we do that sync and async in angular? Chaining update post request, AJAX, waiting for backend to response, if failed, branch, if success, chain to next.
13. How we create multi-threads in Java?
14. Completable future and future.
15. Experience in Kafka?
16. Experience in deploying? Have you used Kubernetes?
17. Experience in CICD?
18. Any questions?

With solution engineer round:

1. Intro, explain recent project.
2. Intro tech stack.
3. Good with Java 8? What is Default? What is functional interface? What is consumer in it? What is supplier?
4. How to create a singleton class? Under what situation we used singleton class.
5. Talk about design pattern? Any idea about factory? How it works? Show me pseudo code.
6. Lets talk about spring, u used springboot?
7. Tell me something about annotation.
8. @SpringBootApplication, is this annotation a must? Your understanding of it.
9. Focusing on other annotations, what r annotations we used on methods?(he specified that in a controller class).
10. What should be the return value of that method?(I asked about the annotation on the class).

11. What if it is a RestController? Return value? What if it is a Controller, return value? Why if Controller, we return String? Under what architecture we used this model-view? Why prefer RestController (difference from Controller)?
12. If a request arrived, show me the flow. Exception-handling.
13. What is BeanScope. What is prototype? What is default beanscope in springboot?
14. How we do multi-threading in java?
15. What about javascript? Async and sync? Promise and callback?
16. Tell me experience in kafka.
17. Tell me experience in Hibernate? How you do CRUD, SpringDataJPA.

### **Alan Yu - 02/05/2024 - Apple**

1. Introduce yourself
2. coding: merge intervals
3. why using List list = new ArrayList
4. what is method generic
5. resume questions
6. mongodb exp
7. why Mongolian
8. aws exp

### **Desmond wang - 01/30/2024 - Ebay**

1. Introduce yourself
2. Project intro and what did you do?
3. What is a scalable web application?
4. How to make an application scalable and what tech you will use? For example, make your application can hold 2000 users to few millions
5. Follow-up questions based on questions 4:
  - a. what is nosql and sql database
  - b. What is sharding and partition
  - c. At what time do you need auto scaling
6. What is your strength
7. What is your behavioral weakness and technical weakness
8. What is key value database and document database
9. Why are looking for new job
10. How do you prioritize your task, if you have two important tasks are coming
11. What is the most challenging part of last job
12. What is acid in database
13. What is cap
14. Is cassandra ca or ap?
15. Why did you end last project
16. What are you going to do to let last company keep you stay
17. Why do you want to join ebay

### **Desmond wang - 01/30/2024 - Ebay**

1. Self intro
2. Implement method in library, medium, allow Google search  
Create a library project using IDE(IntelliJ/Eclipse) with maven.  
groupId: com.ebay.interview  
artifactId: common  
version: 1.0.0  
Implement utility class StringUtils with static method shiftString(input, times), and return shifted string. E.g. input: "abcdefghijklmnopqrstuvwxyz", shift right 2 times, the result is: "yzabcdefghijklmnopqrstuuvwxyz"
3. Generate the library artifact.
4. Install it into local mvn
5. Create a github repo and upload it
6. Creating another micro service project with importing StringUtils that you created

### Marve Mao - 01/30/2024 - Plymouth Rock Assurance

1. self intro && previous project
2. any microservice experience
3. where did you deploy application
4. how to create spring boot application
5. diff spring boot and spring
6. how did you deploy spring boot application
7. how do you configure docker port / image path when you deploy docker
8. when did you graduate
9. previous project (all projects on resume)
10. what is lambda
11. why lambda
12. how did you write lambda

### Marve Mao - 01/26/2024 - BNY

1. self intro && previous project
2. trunk-based development
3. pros and cons for development methodology metnioed above
4. teamwork experience
5. how to ensure code quality
6. how to sync up with your team
7. cicd experience
8. agile experience

### Andy Mou - 01/04/2024 - Ebay

1. Intro yourself
2. what did you do in last project
3. how did you impl idempotent service
4. have you done sql tuning / service tuning
5. how did you make sure scalability of the application

### Marshall Jiao - 01/04/2024 - Ebay

1. self intro && previous project
2. what are different ways to create threads in java
3. what is a deadlock
4. given a slow rest API, how do you identify the problem
5. What are the benefits to use Spring framework
6. coding: write a thread safe singleton class named MyService
7. coding: longest common prefix
8. coding: palindrome number
9. coding: java stream API (stream creation, stream filtering, stream print out)

### Andy Mou - 01/03/2024 - Ebay

1. Intro yourself
2. QPS of your rest api
3. volatile
4. linkedlist, arraylist, vector, which one would you choose to remove / insert a lot of elements
5. what is thread local
6. diff fail safe and fail fast
7. how does fail fast work
8. kafka if we have 10 partitions ,how many consumers do we need
9. if one consumer shutdown what will happen
10. is finally block always executed, when do we use finally block
11. coding1: get longest palindrome substring from input string
12. coding2: get longest length of substring which only contains unique characters

### Andy Mou - 12/29/2023 - Ebay

1. Intro yourself
2. Write a restful api for user registration, service layer, repository layer, controller layer
3. Given a long url, this long url function will produce a short url. And short url will call a function to retrieve this long url, how do you store long and short urls, what database you will use.
4. you have two services db1 and db2. Db1 and db2 have the same input [data1,data2,data3,clickid]. Write a query to tell which data comes from which service
5. Write a query to compare the difference data between two db service
6. What is circuit breaker
7. What is multithreading, where do your project use
8. Coding: given an integer of array, remove all duplicates and not allow use collections

### Henry Jiang - 11/21/2023 - BNY

1. Self Intro.
2. who is using your project
3. why didn't your company buy a product from market
4. what did you do in that project
5. how did you design microservice
6. what's your frontend exp

### Frank Chang - 11/21/2023 - BNY

1. Self Intro.
2. frontend exp
3. any exp with copilot
4. daily work / agile scrum
5. team composition

### Javey Lu - 11/21/2023 - Walmart

1. Self Intro.
2. any multithreading experience
3. how to keep thread safe
4. give me some example multithreading methods you used
5. coding:

Given an array of integers temperatures represents the daily temperatures, return an array answer such that answer[i] is the number of days you have to wait after the  $i^{\text{th}}$  day to get a warmer temperature. If there is no future day for which this is possible, keep answer[i] == 0 instead.

#### **Example 1:**

**Input:** temperatures = [73,74,75,71,69,71,72,76,73]  
**Output:** [1,1,4,2,1,1,0,0]

#### **Example 2:**

**Input:** temperatures = [30,40,50,60]  
**Output:** [1,1,1,0]

#### **Example 3:**

**Input:** temperatures = [30,60,90]  
**Output:** [1,1,0]

6. tell me about your challenge in the recent project
7. any bugs you faced in project (real world example)
8. any bugs you fixed on java side (real world example)

### Flex Fu - 11/12/2023 - SWIFT

1. Self Intro.

2. Cassandra consistency,
3. consistency levels,
4. what is quorum,
5. what are the differences between QUORUM and LOCAL\_QUORUM in Cassandra,
6. What is the mathematical expression of the quorum replications?
7. Whiteboard coding: Given a string, consisting of characters and digits, return an integer value that only extracts the number out of this string.
8. Reverse String
9. Java: Which variables of the above are allocated on the stack, which are on the heap?
10. Java: Tell me all the methods of the Object class that you can think of.
11. What C++ project experiences do you have?
12. What are the difference between smart pointers and normal pointers
13. What are the differences between C++ string and Java string,
14. what is the last character of a C++ string, underlying type of a C++ string
15. Linux command: How would you find all the Java processes running on a Linux server
16. Linux command: Change all 'xyz' in the file to 'abc'.
17. Linux command: How to display the processes that are running on a specific port.
18. Difference between docker container and docker image.
19. Hands-on experience with docker.

#### **Henry Jiang - 11/10/2023 BNY**

1. Self intro and previous project
2. How to scale your frontend and backend in your previous project
3. Would you like to do more with frontend?
4. Design apis for book system.
5. What database you use for this system
6. How to handle exception
7. What are options you can return to caller if an exception happens
8. Are you comfortable with aws?
9. Are you comfortable with docker?
10. What is dockerfile

#### **Henry Jiang - 11/02/2023 - Health Care Service Corporation**

1. self intro && previous project
2. how much time you spend on frontend or backend
3. angular version
4. how to route in angular
5. how to share data between components
6. angular data binding
7. what components have you implemented in angular
8. how to let angular load faster
9. if you deployed an application into production, but you get 500 when you send request to server, how do you fix the issue.
10. what databases have you used
11. what orm have you used
12. any issues with nosql /or challenges when you use nosql
13. which java version have you used
14. any exp on java 8 stream api
15. code review:

```

    @Autowired
    RestTemplate rs;

    public String getBooks() {
        String url = "...";
        HttpHeaders ..;
        ResponseEntity<String> responseData =      rs.exchange(.., String.type);
        return responseData.body();
    }

```

#### **Frank Chang - 11/02/2023 - BNY**

1. Introduce yourself
2. Talk about your responsibilities
3. Do you have experience with frontend
4. Did you use typescript or javascript
5. Creating apis for book store, say you will implement CRUD. describe your backend structure
6. What http method to use for updating, adding books.
7. How do you handle any exceptions for these apis

#### Berry Wang - 10/26/2023 - T Mobile

1. Explain me your last project(with using the whiteboard)
2. How to ensure consistency in Kafka(my project used kafka a lot)
3. Coding problem: Valid Parentheses
4. How do you do exception handling
5. If our service having 500 requests coming from external, what will you do to handle it
6. What is CAP
7. What do you know about ThreadPool

#### Watson Peng - 11/01/2023 - SWIFT

1. self intro
2. Waterfall or Agile
3. Team size and composition
4. Scrum meeting schedule
5. white board coding: given a string, get all the characters from its even index, then build a new string of it.
6. white board coding: given an array of numbers, and an index array. We want to print the numbers in ascending order without making changes on the number array.
7. What is Cassandra consistency
8. What are the consistency levels and how do you configure it
9. What is the partition key of Cassandra
10. How do you fetch and manage data from Cassandra database
11. Linux command: list all process
12. Linux command: list out the 10 most recently modified files in the current directory
13. Linux command: display the process of the port that you are currently monitoring
14. Linux command: change all 'xyz' in the file to 'abc'
15. docker experience

#### Henry Jiang - 10/18/2023 - Hexaware Technologies Inc

16. self intro && previous project
17. daily work
18. agile experience
19. difference between sql and nosql
20. scenario to use nosql to process large volume data, why sql is not suitable.
21. how nosql to solve foreign key
22. scenario to design a microservice
23. how to deal with transaction through multi microservice. eg. one microservice is lost, how transaction will be.
24. question about AWS ECS + cloudwatch according to project.
25. SQL coding: find the second max value in table
26. angular question:
  - a. design a pipe to sort data
  - b. two way binding
  - c. ngModel
  - d. file transfer in angular application (not sure question it is)
  - e. dynamic binding
27. java question coding:two string swap without use third variable.
28. Linux command

#### Watson Peng / Jimmy Wang - 10/13/2023 - Morgan Stanley - Krishna Das

1. self intro && previous project
2. angular vs react
3. (talked about previous career experience) why you wanna join morgan stanley

4. what do you know about morgan stanley
5. tell me more about the project you have done with the financial background
6. your project qps/ tps
7. what clients are using project
8. (follow up) What have you built in that project?
9. how do you delete transactions
10. do you know elastic search
11. What is your motivation to switch into the financial industry?

Watson Peng / Jimmy Wang - 10/13/2023 - Morgan Stanley - Natalia Govor, Daniel Lee

12. self intro && previous project
13. biggest challenge in previous project
14. let, var, const
15. if your frontend project is going to be used on different devices, such as computer, pad, phone. How do you render the page to fit into different size of the screen?
16. why typescript
17. redux
18. What is the problem? (can use label = 'Counter: \${}')

```
import { useState, useEffect } from "react";

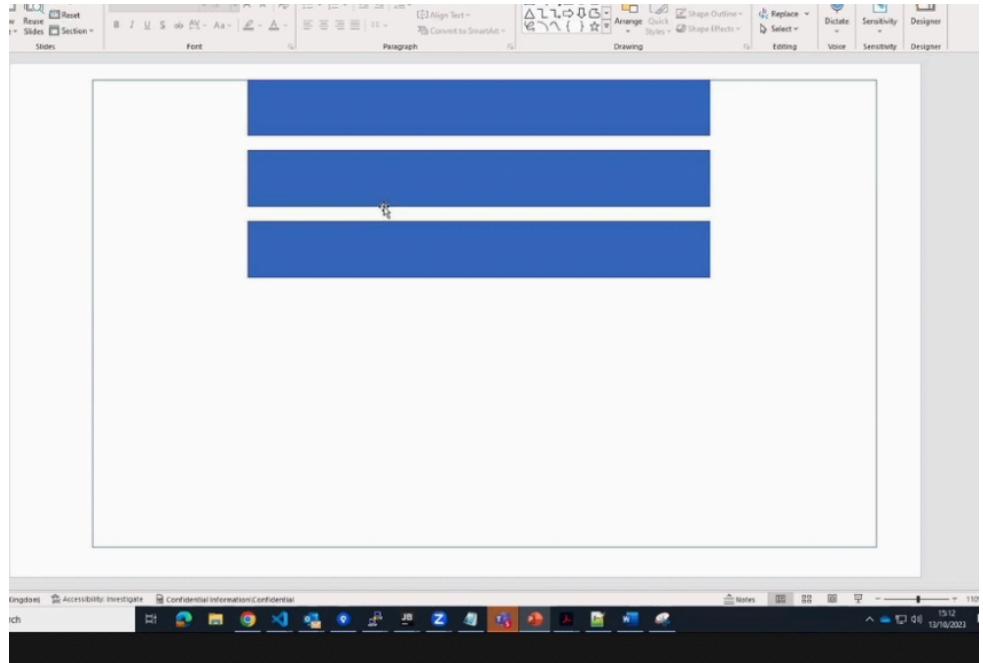
interface ExampleProps {
  count: number;
}

export function Example(props: ExampleProps) {
  const [label, setLabel] = useState<string>('');

  useEffect(() => {
    setLabel(`Counter: ${props.count}`);
  }, [props.count]);

  return <div>
    {label}
  </div>;
}
```

- 19.
20. how to let it render only once
21. optimize it (answer use memo)
22. how to implement this:



23. if we modify the size of screen, the size will change, how to maintain the fix relative size of components
24. margin vs padding
25. what is Spring
26. what IOC in Spring
27. Collection interface
28. LinkedList vs ArrayList
29. how does hashCode method work in java
30. what is hash collision
31. what is hashmap
32. What is a deadlock?
33. what is race condition
34. what is atomicity
35. what is volatile
36. What is the mechanic for volatile in java?

#### Jim Song - 10/12/2023 - BNY Mellon

1. self intro
2. frontend exp
3. agile scrum
4. most proud implementation / feature you done before (dive deep)

#### Jimmy Wang - 10/10/2023 - Sherwin Williams

1. Self intro
2. Introduce the most recent project
3. How did you implement security for this project.
4. OAuth2 implementation
5. How does the front end send the auth token to the back end?
6. How did you implement authorization
7. Why did you use a table to store authorities.
8. CI/CD pipeline
9. Maven goal in CI/CD pipeline
10. Introduce the second project
11. What types of documents can users upload?
12. Can users upload images/videos?
13. Will you create thumbnail for image? How?
14. How did you implement the searching features?
15. Front end exp
16. Spring DI and IOC
17. Spring security

18. How to start a new SpringBoot app.(Spring initializer)
19. How to run a query to search for entities by some fields.
20. Java version
21. All Java 8 new features
22. How to implement a singleton class?
23. SLF4J, how did you add context info to an exception log.
24. React exp
25. selector in JQuery
26. in React, how to return diff view in one component?
27. functional component or class component in react?
28. State vs props
29. Hook in React
30. Design of components
31. How to use Axios with React.
32. US working exp
33. working exp timeline
34. Visa and Sponsorship
35. Any other questions?

#### Hanson Chen - 10/05/2023 - BNY Mellon

1. coding: Given an array of numbers, find the index of the smallest array element (the pivot), for which the sums of all elements to the left and to the right are equal. The array may not be reordered
2. coding: given an entire Angular project within the online OA platform. In Angular, there are two Angular components: USDValue and YenValue. Candidates need to write the code in ts files and html files to achieve this effect: in the web page, there are two textareas, one is for USD, one is for Yen. if the user input USD amount, then Yen should show an auto-converted amount of YEN that equals the user input USD amount.

#### Hanson Chen - 10/05/2023 - BNY Mellon

1. self intro
2. previous experience
3. AWS experience
4. Do you prefer the front end or back end?
5. how do you feel with legacy code
6. what other coding language are you familiar with

#### Max Gu - 10/02/2023 - Google - round 4

1. coding1: implement a method that can take a file as input, and return a modified file.  
Rules: read file line by line, remove all the leading spaces, and also make sure each word only has one space between them, also remove all the ending spaces.
2. what is try with resource
3. coding2: can you write a class which can be used in try with resource
4. coding3: Following Question:  
Suppose your file is really large that you need to process chunk by chunk  
Given four method:  

```
long sizeOfFile() //this will return the size of the file
void readNextChunk(byte[] chunk) //can only read 64mb byte
void writeNextChunk(byte[] chunk) // only write when we have enough byte: 64mb
long sizeOfOutput //call this once at the end of the file
```

#### Max Gu - 09/25/2023 - Google - round 3 - MF

1. what courses you learned in college / master
2. invertible matrix
3. non-singular matrix
4. Normal distribution
5. how does C work
6. dynamic library vs static library
7. dynamic link vs static link
8. memory page

9. computer architecture
10. how does memory part work in computer architecture
11. what if memory is full
12. coding: OOD foosball



Naomi - 09/25/2023 - Google - round 3

1. what courses you learned in college / master
2. details about different courses she mentioned
3. have you learned operating system
4. how does java execute
5. what is bytecode, what is source code, what is java jar
6. how does C run
7. how does python execute code
8. different statistics questions based on her introduction
9. what design patterns do you know and explain each of them
10. difference between interface and abstract class
11. coding: OOD monopoly



Naomi / Max Gu - 09/18/2023 - Google - round 2

1. how to test random generator(generate number from 0 to 1)
2. coding:

## 2. Generate pseudo-random words based on sample text

Implement the Model class, which would be used like this:

```
Model m = new Model();
m.buildModel(new String[] {"ACE", "APE", "APPLE", ...});
System.out.println(m.getRandomWord());
```

The getRandomWord() method should return a similar-sounding word to the ones you fed to the model with the buildModel() call. Specifically, it should do this by keeping track of the probability of a letter following another letter, also known as Markov chains.

For example, if you fed the model the words ("ACE", "APE", "APPLE") it would calculate this:

100% of words start with A.  
A is followed by C 33% of the time.

A is followed by P 66% of the time.  
C is followed by E 100% of the time.  
P is followed by E 33% of the time.  
P is followed by P 33% of the time.  
P is followed by L 33% of the time.  
L is followed by E 100% of the time.  
E always ends the word.

It would then generate new pseudo-random words like "APPE" or "APEL" when you call getRandWord().

## 3. coding:

The farmer owns an N by M meter field and is facing financial challenges. He's considering selling most of the field but wants to keep a 3x3 section while conserving water usage. Each 1x1 plot needs a specific amount of water. Your task is to identify the optimal 3x3 slot for him, considering water efficiency.

(Given a N\*M matrix, find the minimum sum of 3x3 submatrix.)

Example: Given matrix:

```
[  
1,2,3,4,5,4,  
3,1,2,1,0,4,  
2,6,7,0,1,1,  
4,6,7,0,0,1,  
1,2,4,6,2,3,  
2,7,9,8,1,5,  
]  
for the sub-matrix [ x1 = 3, x2 = 5, y1= 1, y2 = 3 ], the sum is 8 and this is the minimum 3*3 cells.
```

### Follow Up Question #1: Empty cells.

Consider a situation where you are given matrix sizes N\*M, but some plots have a value of zero. Your task is to find the minimum 3x3 plot on this grid where all 9 plots have a non-zero value.

Naomi - 09/18/2023 - Google - round 1(vendor)

1. Coding: find all the factors of one number
2. Coding: input matrix, and print matrix with commas  
1, 0, 2, 1  
2, 0, 2, 1  
1, 0, 1, 1
3. Coding: print out

```

1x1= 1
1x2= 2  2x2= 4
1x3= 3  2x3= 6  3x3= 9
1x4= 4  2x4= 8  3x4=12  4x4=16
1x5= 5  2x5=10 3x5=15 4x5=20  5x5=25
1x6= 6  2x6=12 3x6=18 4x6=24  5x6=30  6x6=36
1x7= 7  2x7=14 3x7=21 4x7=28  5x7=35  6x7=42  7x7=49
1x8= 8  2x8=16 3x8=24 4x8=32  5x8=40  6x8=48  7x8=56  8x8=64
1x9= 9  2x9=18 3x9=27 4x9=36  5x9=45  6x9=54  7x9=63  8x9=72  9x9=81

```

$$A = \begin{pmatrix} 1 & 2 & 2 \\ 4 & 5 & 8 \end{pmatrix}, \quad A^T = \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 2 & 8 \end{pmatrix};$$

$$B = (18 \ 6), \quad B^T = \begin{pmatrix} 18 \\ 6 \end{pmatrix}.$$

4. Coding:

#### Noah Cao - 09/18/2023 - Kroger - round 2

1. self intro && previous project(what tools you used)
2. nosql exp
3. java version
4. angular version
5. Spring boot vs Spring MVC
6. your multithreading experience
7. how to prevent dead lock
8. thread pool exp
9. how to make variable thread safe
10. security in your application
11. actuator
12. AOP
13. transaction propagation levels
14. Dependency injection
15. Beanfactory vs Application Context
16. bean scopes
17. diff singleton vs prototype
18. what is binary search
19. coding: Singleton Lazy loading thread safe
20. given : prodid, siteid, date1, date2, date3, date4.....  
how do you design table
21. angular guard
22. angular lifecycle hook
23. angular state management
24. angular dependency injection
25. jit vs aot
26. angular routing
27. what is ng-container
28. angular memory leak

#### Noah Cao - 09/15/2023 - Kroger - round 1

1. self intro && previous project(what tools you used)
2. most proud project recently
3. microservice component what you used before(config, discovery, gateway, circuit breaker, tracing, etc)
4. Design Patterns You Used in microservice
5. disadvantage of index in database

6. what is view in database, view vs table
7. NoSQL database experience
8. you said you know dynamodb, tell me the CAP theory in dynamodb
9. SQL vs NoSQL
10. N+1 problem in database with ORM + solution
11. Spring data JPA + Hibernate experience
12. how to prevent overridden
13. how to prevent Serialization for a field in java pojo
14. how to improve map performance
15. tell a structure in java collection fast in modification and slow in access
16. observable vs promise in Angular
17. component communication in Angular
18. Have you used reactive programming in Angular before, explain
19. kubernetes experience(have you used the cli) YES use kubectl before
20. Kafka experience
21. What is partition in kafka
22. Coding: given an int array, find the distance between two closest numbers. (allow sort API in java) / ask to run so that check candidate debugging capabilities

#### **Watson Peng - 09/14/2023 - Kroger - round 2**

1. self intro && previous project
2. cloud platform experience
3. biggest challenge
4. followup for biggest challenge on index in database (read/write tradeoff)
5. stored procedure vs logic in java web app
6. java version in project
7. child/parent class method overridden with access modifier
8. security in web app
9. documentation in web app
10. N+1 problem in database with ORM + solution
11. observable vs promise in Angular
12. list out Angular lifecycle hook you used
13. NoSQL database experience
14. SQL vs NoSQL
15. TerraForm experience
16. Coding: given an int array, find the distance between two closest numbers. (allow sort API in java) / ask to run so that check candidate debugging capabilities
17. Azure experience (current project is using)
18. Kafka experience
19. What is partition in kafka

#### **Watson Peng - 09/08/2023 - Plymouth Rock Assurance (onsite)**

##### **1st Round**

1. Talk about your most recent project, and the tech stack you used for the project
2. How do you remove duplicate rows in SQL, please explain in actual query commands
3. Difference between RequestMapping and GetMapping
4. Please talk about any framework you use in Spring
5. What is Spring MVC and its architecture
6. What is Spring AOP
7. How do you create a thread with thread pool, and how to handle the result
8. If there is an error, how do you locate and debug it

##### **2nd Round**

1. Talk about your most recent project, and the tech stack you used for the project
2. What is Rest template
3. What is Filter in Spring MVC and how do you use it
4. What is Interceptor in Spring MVC and how do you use it
5. When you receive JSON data from front-end, how do you turn it into a java object in back-end
6. Difference between URI and URL
7. How do you create an AJAX Call
8. How do you add dynamic suffix to the url in the AJAX call

9. How do you add an object at the front of the ArrayList, after adding the object, how is the order changed

### 3rd Round

1. Talk about your most recent project, and the tech stack you used for the project
2. What is your most daunting experience for your former project
3. Please talk about a specific function you design in the front-end
4. What is z-index, when and how do you modify it in CSS
5. What is chaining in jQuery, please give an example
6. What is Overload and Override
7. What is the size of your team, is it agile? What are the roles for the team members?

### 4th Round

1. Talk about your most recent project, and the tech stack you used for the project
2. How do you manage the token when you are visiting different component in Angular
3. How do you handle the token in the back-end with Spring MVC
4. Paper code: use KeyUp() event, only read the user input after 2000ms

### Anna Xu - 09/01/2023 - Apple

1. self intro && previous project
2. Coding1

You are given pairs of student ID and course name, and return the shared course of all students.

Input:

[58, Intro to Computer Science]  
[79, Principles of computing]  
[58, Computer science 101]  
[79, Computational Thinking for Problem Solving]  
[63, Computational Thinking for Problem Solving]  
[49, Computer science 101]

Output:

[58, 49]: Computer science 101  
[63, 79]: Computational Thinking for Problem Solving  
[79, 58]:  
[63, 49]:

3. Coding2

In college, every course has a prerequisite course, you can find the relation of each course in the input, pls return the middle course the student should take. If you have even total courses, return the first middle course.

The first course is the prerequisite of the second course. Take [Intro to Computer Science, Computer science 101] for example, if you want to take Computer science 101, you must finish Intro to Computer Science firstly.

Input:

[Intro to Computer Science, Computer science 101];  
[Data Structures, Algorithms];  
[Programming, Operating Systems];  
[Computer science 101, Data Structures];  
[Algorithms, Programming]

Output:

Data Structures;

### Anna Xu - 08/30/2023 - Apple

1. self intro && previous project
2. If i give you a 5GB file, it contains json data line by line.  
How will you insert data into mongodb  
Tell me all your approaches.
3. coding1: convert [1, 2, 3, 4, 5] to [5, 1, 4, 2, 3]
4. coding2: verify parentheses

**Kelly Gao - 08/28/2023 - Charter**

1. self intro && previous project
2. why discovery service
3. why circuit breaker
4. how to use multithreading in java
5. how to make system robust when you call 3rd party api
6. how to decide thread pool size
7. how to handle exception
8. how to do when application slow
9. what is memory leak
10. why JVM throw out of memory if heap is not full

**Ethan Huang - 08/25/2023 - American Airlines**

1. Self intro
2. Set vs hashmap
3. build a custom class that is a set of airplanes,no duplicate values of airplane
4. Coding: recursive add function

**Jonas Sun - 08/23/2023 - Starbucks (python position)**

1. self intro
2. previous project
3. how to do authentication in django
4. TPS in your project
5. how to do the test in django
6. how to improve database performance
7. where did you deploy your application
8. Is your database public?
9. what is your single web page app experience
10. how do you manage the state in react
11. how do you reuse the code in python
12. how to make sure your code is ready for public
13. Did you use docker for local development?

**Edmond Zhou - 8/22/2023 - CANTON GROUP**

1. self intro
2. AWS service
3. Database Service
4. hibernate
5. agile methodology

Given an array of integers, return the maximum sum for a non-empty subarray (contiguous elements) with at most one element deletion. In other words, you want to choose a subarray and optionally delete one element from it so that there is still at least one element left, and the sum of the remaining elements is maximum possible.

Note that the subarray needs to be non-empty after deleting one element.

Example:

Input: arr = [1,-2,0,3]

Output: 4

Explanation: Because choosing the entire array and deleting the negative gives  $1 + 0 + 3 = 4$ .

Input: arr = [1,-2,-2,3]

Output: 3

Explanation: We just take the last element.

Input: arr = [-1,-1,-1,-1]

Output: -1

6. Explanation: Because choosing a single -1 is better than deleting one of them to get 0.

Max Gu - Aug/18/2023 - Equifax

1. Self intro
2. your recent project
3. what's you looking for next position
4. how did you secure your web service
5. where did you guys run application
6. ci/cd pipeline in your project
7. what do you see the difference between chinese company tech stacks and us company tech stacks

Javey Lu - Aug/18/2023 - Equifax

1. Self intro
2. What is your most recent project and walk me through from the beginning
3. What were your responsibilities
4. Did you deploy your project on aws/gcp?
5. What was your second project?
6. Do you have angular experience, if yes, introduce it?

7. How to secure restful api?
8. What services did you use for your first project?
9. Why did you use gcp for your first project?
10. Introduce your bachelor and master education

**Kelly Gao - Aug/18/2023 - Equifax**

1. Self intro
2. questions about all projects on resume
3. do you have ci/cd exp
4. why you fit this position(we looking for angular, java, gcp)
5. where did you use angular in project
6. which angular version did you use
7. what have you done on frontend side
8. what were your responsibilities in last project
9. give me an example of production support
10. how many years of exp about splunk

**Naomi - Aug/14/2023 - Cotiviti**

1. Self intro
2. put vs patch
3. do you know Cotiviti company
4. why do you think you fit for this role
5. what services you created in last project
6. how did you create api in last project
7. give me an example when you fixed rest api bugs
8. how did you deploy application
9. how to handle exception in java / spring
10. what is jpa
11. how to create query in jpa

**Anna Xu - Aug/10/2023 - Apple**

1. self intro && previous project
2. how many services in your project
3. how do services communicate each other
4. how does kafka work
5. if message is too big, 10MB how do we send message to consumer
6. how did you deploy application
7. ci/cd pipeline in your last project
8. which collection is your favorite
9. diff set and map
10. if you want sorted data, which will you use

**Esther Gong - Aug/08/2023 - BNY**

1. self intro && previous project
2. ORM vs JDBC
3. Hibernate architecture
4. How do you handle security in your web app?
5. design pattern you have used in web app
6. Can you explain how you do the JUnit test?
7. Why use Ribbon instead of native AWS load balancer?
8. What is Eureka for?
9. What Angular Module have you used before?
10. Have you used the Spring portlet?
11. DAO vs DTO vs VO
12. JWT token
13. how to ensure your code is thread safe in java?

14. Where do you deploy your web app?
15. checked vs unchecked exception
16. what is object serialization
17. how to handle transaction exception in hibernate
18. What is dirty read
19. what is CICD
20. Have you used AOP before?
21. How do you create a pointcut in spring boot?
22. What is your exposure in the database?
23. stored procedure vs pagination

#### Anthony Pan - 08/03/2023 - Walmart

1. self intro
2. Spring AOP
3. AOP Advice
4. @Autowired
5. Design Pattern - builder

You are given a list of integers containing both positive and negative numbers. The list is sorted in ascending order, except that some numbers appear out of order. Specifically, there might be pairs of adjacent numbers where the left one is greater than the right one. For example, the list [1, 3, 2, 4, 5, -1, 0, 7, 9] contains 3 such pairs: (3, 2), (5, -1), and (7, 9). Implement a function that takes such a list and returns the index of the first number that violates the order.

For example:

```
find_firstViolation([1, 3, 2, 4, 5, -1, 0, 7, 9]) => 2
```

In this example, the first number that violates the order is 2, which has index 2 in the list (0-based index).

Constraints:

- The list can have at most  $10^6$  elements.
- The function should have  $O(\log n)$  time complexity on average, where  $n$  is the length of the list.

Given a sequence of positive integers in an array, write a program to print the longest subsequence where each consecutive element is greater than its previous element by exactly 1. If there are multiple such subsequences, print any one of them.

For example:

Input: {2, 3, 4, 5, 1, 2, 3, 4, 5, 6, 7, 8}

Output: {1, 2, 3, 4, 5, 6, 7, 8}

6.

#### Kelly Gao - Aug/03/2023 - Charter

1. self intro && previous project
2. how will you design endpoint to search employee by name and status
3. diff post and put
4. diff 4xx and 5xx status code
5. why spring starter
6. how do you configure multiple database connection in spring boot
7. how do you rewrite some properties
8. how do you inject bean into IOC container
9. biggest challenge

#### Tina Ma - Aug/02/2023 - Walmart

1. self intro && previous project

2. Coding: Given a tree node, find out all path from root to leaf
3. Given a list of Strings, find out shortest prefix. For example, input: ["Monday", "Tuesday", "Thursday", "Saturday", "Sunday"] -> output: ["M", "Tu", "Th", "Sa", "Su"]

### Wiki Wang - Aug/01/2023 - Starbucks

1. self intro && previous project
2. diff trigger and function in pl/sql
3. nosql vs sql
4. use case of nosql in your project
5. daily work
6. team size
7. how to do production support
8. give me an example of production support in your last project
9. docker / kubernetes experience
10. your ci/cd automation pipeline
11. How to do customization configuratoin?
12. What are the different scope of beans in spring?
13. Where do you use prototype scope?
14. where do you use request
15. any other languages you used before
16. coding: java 8 stream remove dup from list, and sort result in decreasing order ,then return the new list
17. how do you build docker

### Jason Dong - Aug/01/2023 - Starbucks

1. self intro && previous project
2. Are you a role in a team for the back end or front end ?
3. Can you explain what tech stacks you have used?
4. Where do you deploy your web app?
5. What are the annotations you have used in springboot?
6. Spring vs Spring Boot
7. How to do customization configuratoin?
8. What are the different scope of beans in spring?
9. Where do you use prototype scope?
10. What is @ComponethScan for?
11. Have you done any event-driven system before?
12. What is the consumer group in kafka?
13. checked vs unchecked exception
14. throw vs throws
15. stack vs queue in data structure
16. ArrayList vs LinkedList
17. python experience before
18. pyspark experience
19. cucumber BDD
20. mockito -> how to mock a json return
21. spring security -> experience + what have you done for security in web app
22. @RestController vs @Controller
23. Coding: in java, given a list of integers, can you get the sum of prime numbers from this list with java 8 stream API
24. How do you ensure thread safety
25. How do you test thread safety in Junit?
26. docker + kubernetes experience

### Tina Ma - July/31/2023 - Walmart

1. self intro && previous project
2. why nosql
3. Coding: You're given an array of integers and an integer. Write a function that moves all instances of that integer in the array to the end of that array and returns the array. The function should perform this in place (i.e., it should mutate the input array) and doesn't need to maintain the order of the other integers.

**Jeffery Li - July/31/2023 - Vecna**

1. questions related to your resume
2. business logic / business use cases about project
3. your responsibilities
4. any payment experience

**Steven Chen - July/31/2023 - Navy Federal Credit Union**

1. self intro && previous project
2. if your api / backend is slow how do you tune it up
3. if you need to upload a file how do you save it in your backend
4. how to secure rest api
5. how to communicate between micro services
6. what is auto configuration
7. how to load external components into spring IOC
8. how do you override spring properties with command line
9. what's your agile and scrum
10. what do you do when you develop a new service
11. how did you check code coverage and code analysis
12. explain your jenkins pipeline and tools used in jenkins
13. if your coworker added dependency into maven, but when you update your code, rerun the maven, dependency is not found, how do you solve the issue
14. how to handle exception in spring
15. jpa vs hibernate
16. filter vs interceptor
17. if your intellij crash, how do you modify code without intellij
18. flatmap vs map
19. what is predicate
20. reactive programming exp
21. how do you connect different tables in jpa / hibernate
22. why stream api
23. diff stream api and collections
24. what is functional interface
25. why is rest api good for scaling
26. how did you monitor application
27. how to create microservices
28. frontend exp

**Jeffery - July/28/2023 - Delta Airline**

1. self intro && previous project
2. lambda experience
3. how did you create aws lambda and deploy
4. microservice 12 factors
5. why microservice
6. what is functional programming and why
7. have you used aws cdk
8. cloudformation experience
9. what is the mandatory parameter of cloudformation(resource)
10. if you have diff service and need to deploy in order, how to it
11. how to write singleton
12. what design patterns have you used
13. how do you configure api gateway to sqs
14. what are Dynamodb consistency models
15. why nested classes and when do we need it
16. where did you deploy frontend
17. what was your CI/CD pipeline in your last project

**Allen Li - July/20/2023 - Finra**

1. self intro && previous project

2. Given an API in java, how do you test it without seeing the source code?
3. After verification of this API in java, how do you ensure developers will use it correctly?
4. Why do you wanna be a SDET after being a developer in Apple?
5. How do you feel about coding?
6. Given an API with a number as input in which will check whether the number is prime number or not, how will you test it?
7. Let's say I gave you a very very large number and you passed it into the API, the API said it is a prime. How do you know if this input is a prime or not?
8. If developers in another team ask you to pick up the source code today so that you can test today to ensure release tomorrow, will you think it is reasonable?
9. What was the biggest challenge for you when you worked at Apple?
10. Given an Oracle database, we are migrating into Postgres DB. In Oracle there is a table that contains 1 million records, and after migration, there are 1 million + 1 records in Postgres DB. Do you think it is good to go? What about there are 1 million records in Postgres DB? Do you think that is good to go?

#### **Jeffery Li - July/20/2023 - Vecna**

1. questions related to your resume
2. why nosql
3. how to do nosql
4. how did you use mongodb in your last project
5. how did you do security
6. how did you do single sign on
7. where could we use graph db

#### **Kyle Wu - July/14/2023 - Delta Airline**

1. self intro && previous project
2. whiteboard brainstorm: given a employee, design restful endpoint for CRUD
3. path variable vs request parameter
4. integrate the restful endpoint designed above with Spring data JPA
5. native query vs JPQL
6. sql injection concerns on restful endpoint designed above
7. how to do user input validation
8. Any concerns on this employee class for a restful endpoint? (immutable)
9. shall we turn all the objects to immutable in web app?
10. aws - lambda
11. what language you used to implement lambda
12. business logic for using lambda
13. IAM policy vs Resource policy
14. SQS vs SNS
15. When the current SQS cannot carry on, what should we do?
16. How do you set up your SQS?

#### **Esther Gong - July/10/2023 - Finra**

1. self intro && previous project
2. override vs overload
3. exception handle
4. throws vs try catch finally
5. what is finally
6. how to handle runtime exception
7. Where did you use reflection?
8. what is IoC
9. What is an actuator?
10. how to enable actuator in spring boot
11. how to have an immutable class in java?
12. is immutable class thread safe
13. why we don't have all the class immutable
14. what is factory design pattern
15. sql: intersect vs union vs minus
16. truncate vs delete vs drop
17. code: pair programming with interviewer to regenerate a recent test case interviewee has written recently in project

18. from 1 to 10, how much do you grade your javascript?
19. what is prototype in javascript
20. What is the difference between javascript vs java inheritance?
21. How big is your team size?
22. How many frontends? How many backends?
23. Do you have a scrum master?
24. What about other roles?
25. Do you have a daily stand up meeting?
26. What do you do in daily stand up meetings?
27. CICD experience
28. Have you used the S3 service in AWS in a previous project?
29. json validation
30. json mapping
31. Last round someone jumps in and asks about a scenario: given a xxx.br2 file in which contains json objects. we wanna read the content in this file and validate the content via a rule based design pattern(this is a solution the interviewer spoke out at the end). He wants a design on what are the classes you need, what are the flows you are going through. (Solution he gave contains a FileReader interface, an implementation on this interface, a Rule interface, implementation on this Rule interface)

#### **Allen Li - July/06/2023 - Delta Airline - Sharma Anil**

1. self intro && previous project
2. What are the most commonly used 10 aws services you have used in previous projects?
3. How do you start/set up those aws services?
4. What are the mandatory elements for a cloudformation template?
5. Can we change CloudFormation Template Version?
6. Are you familiar with Lambda function in aws?
7. How do you create your lambda function in aws with which programming language?
8. AWS - API gateway
9. AWS - Lambda
10. dynamoDB + spring boot
11. partition key vs sort key in partition
12. hashmap vs treemap
13. In Java, given a list with no declaration on data type, we have 1,2,3 as well as A, B, C. How to filter out all the numbers?

#### **Esther Gong - 07/05/2023 - Finra**

1. latest project
2. AWS experience
3. Junit 5 vs Junit 4
4. how to connect API in test
5. how do you do the Authentication
6. how does JWT work
7. downside JWT
8. Java 8 new features
9. method reference
10. optional keyword
11. sql aggregation function
12. Jenkins
13. Coding: Count the char frequency in a string
14. Hibernate architecture
15. JDBC
16. lazy loading
17. self join
18. SQL Query: update one field of table
19. SQL Query: find all the orders and customer name from the table
20. when the user story is not clear, how do you do

#### **Wiki - July/05/2023 - Finra**

1. Introduce yourself and project
2. follow TDD to do all questions

3. coding: isPalindrome(int number)
4. coding: sql return emp info who has same salary as other emp
5. coding: find first missing positive number
6. given example : we need to get prices in last 15 min , formula is  $xx - \text{price} / xx > 50\%$  ,  
if  $xx = 102.50$   
cur time = 16:00  
what price will you test  
what time will you test

**Anna Xu - June/26/2023 - Finra**

1. Introduce yourself and project
2. Do you have any frontend experiences?
3. What are annotations you used a lot in your project?
4. How your internal spring services connection with external services
5. How do you interact with database
6. What is jpa
7. Multithreading in java
8. How do you handle global exception
9. If you have try and finally block without catch block, can you execute it
10. What is springbootapplication annotation
11. What is java reflection
12. Where did you use reflection
13. What are reasons you used reflection
14. What are steps to make a immutable class
15. What is left and right join
16. What is functional interface
17. How to implement functional interface
18. What is lambda expression
19. Coding: given a duplicated arraylist, remove all duplicate elements and return a list without duplicate elements.(cannot create a new collection)
20. Using stream api solve above question
21. Find the middle element in arraylist without using size method in arraylist
22. Getting all even numbers in arraylist by using lambda expression

**Arthur Ban - Jun/26/2023 - American Airline**

1. most recent project
2. what you done
3. how to call other service
4. what pattern do you follow when you develop applications
5. User TDD and Parallel programming with interviewer to finish following requirements

```

1 Wrist watch
2
3 Please create a backend class that will serve a wrist-watch application
4 The application mimics a classic typical analog wrist watch
5 Hours are displayed in Roman numerals
6
7 Notes:
8 - Use Test Driven Development - "Driven" as in "Tests First"
9 - Consider this a pairing exercise. Please talk out loud the steps you are taking and especially why
10 - Always find the simplest next step
11
12 The API features are represented by the following user stories
13
14 1 - method convertToRomans
15     - receives a number
16     - returns the corresponding roman numeral as a String
17     - example: 1 returns "I", 2 returns "II", 3 returns "III", 9 returns "IX", and so on
18 2 - method currentSeconds
19     - no parameters
20     - returns the current seconds of the server clock
21         - formatted as a String that should always be exactly 2 characters long
22         - if the number of seconds is less than or equal to 9, pad a leading "0"
23         - example: 1 should be "01", 9 should be "09", 14 should be "14" and so on
24 3 - setAlarm
25     - receives 2 numbers and saves them : hour (in 12h format) and minute
26     - no value returned
27 4 - readAlarm
28     - no parameters
29     - returns 4 character String with the alarm time that has been setAlarm in 12h format
30         - example: if the alarm is set for 9:05am return "0905"
31 5 - resetAlarm
32     - no parameters
33         - If an alarm was previously set, remove it
34         - If not, do nothing
35     - no value returned
36
37

```

### Karthik Mohan Raj - Jun/16/2023 - Cincinnati Financial

1. most recent project
2. Explain your SQL experience that you had separate from using any ORM tool or JPA
3. Explain more about your experience with writing/using Sql stored procedures/views etc.
4. Explain the difference between C# and Java
5. What is the difference between PUT and GET?
6. What tools have you used for version control?
7. What tools do you use to track sprints and bugs?

### Jeffery Li - Jun/14/2023 - CVS

1. java coding:
  - a. giving input string , count freq and output key value pair in string  
replace '' with space , and follow the order of "first appear of dup char"
  - b. input "ab a bf"
  - c. return "a 2, Space 2, b 2"
2. angular coding:
  - a. use command line create service and run server
  - b. fetch api
  - c. write service
  - d. dependency injection
  - e. display result in html use ng for each
3. what is lazy loading in angular
4. what is observable and why use it
5. what is angular storage (ngRx store)
6. and how to use ngRx
7. http vs rest api

8. multi threading exp
9. cicd exp
10. database index disadvantages
11. what will you do if task got stuck
12. when will you ask questions to other developers
13. dependency injection in spring
14. design patterns in spring
15. what is template method
16. what if i have one interface 10 implementations
17. how to solve it without using qualifier
18. docker experience
19. diff docker image and docker container
20. have you wrote dockerfile before
21. what will you check during code review

**Will Chen - Jun/14/2023 - BNY**

1. self intro && previous project
2. thread creation
3. What is data race?
4. how to ensure thread safety
5. how to prevent sql injection
6. what is maven
7. how to import a newer version for dependency in maven
8. What does the install lifecycle do in maven in default?
9. what is OWASP
10. what is security practice in your web app
11. what is partition in kafka
12. what is linux vm and docker instance
13. what are the linux command you have used
14. what is  $O(N^2)$
15. What is big O ?
16. describe merge sort algorithm
17. what is git pull
18. how to submit one change from one git branch to multiple other branches?
19. what is the difference between unit test and integration test
20. what is DI
21. How to do a regression test for different environments?
22. given a list of integers, how to find the largest three in java?
23. how to do that via stream API?
24. given two tables, how to find all records with the same id?
25. what does it mean if log is saying out of memory exception?

**Hansion Qin - Jun/12/2023 - Kroger**

1. self intro && previous project
2. fail fast vs fail safe
3. spring boot version
4. java version
5. IoC and DI
6. cloud platform experience
7. CICD experience
8. support experience

**Jeffery Li - Jun/07/2023 - CVS**

1. questions related to your resume
2. previous exp(before recent one)
3. how to connect to database
4. how to do unit test
5. OOP
6. diff overriding and overloading
7. inheritance in java
8. how to design restful api
9. how to create spring boot project for rest api which connected to database

10. what is dependency injection
11. what is loose coupling
12. design patterns you used
13. why builder
14. angular how to create component
15. angular what is observable
16. angular component lifecycle
17. angular how to write form submit username password

Will Chen - Jun/05/2023 - Plymouth Rock Assurance

1. self intro && previous project
2. Given a List of Customers in which contains List of Accounts as one of fields. Sort all customers base on number of Accounts via stream API
3. Given a mapping relationship between accountId to customer, return all accounts owned by each customer via stream API
4. design restful endpoints for CRUD customer operations
5. What if we wanna allow user to provide first/last name then the fetch all endpoint will fetch customer matches the condition and if user does not provide first/last name, we wanna just fetch all customers

Jeffery Li - Jun/05/2023 - Vecna

1. questions related to your resume
2. very deep
3. draw architecture
4. draw data flow