

Algorithm

How to traverse BFS

I will use queue data structure and i will push(offer)the entire level of node to the queue first and then get the size of the current queue, so in order to retrieve the number of node in that level, and then i will iterate throw those nodes to get the next level, but meanwhile i have stored the temporary number of node in that level and once i have pushed next level of the node, i can pop one by one to traverse the first level of the node. And we use the same method to iterate through all the level.

Java:

1. **ArrayList vs HashMap**, hash code/equal, hashmap/hashtable

ArrayList	HashMap
Implements List interface	Implements Map interface
Object in order for inserting	No order guarantee
Only stores 1 object	Stores 2 objects (key & value)
Allows duplicates	Doesn't allow duplicate key, but allows duplicate value
get(index) always gives O(1) performance	get(key) best case O(1), worst case O(n)

hashCode()/ equals()

If two Objects are equal, their return same hashCode();

If two Objects have same hashCode(), doesn't mean they are equal. Unless they have the perfect hash function.

Usually use hashCode() to find non equal object, so doesn't have to use equal() function.

HashMap/ Hash Table

What is the difference between HashMap and Hashtable?

Both the HashMap and Hashtable classes implement the Map interface and thus, have very similar characteristics. However, they differ in the following features:

- * A HashMap allows the existence of null keys and values, while a Hashtable doesn't allow neither null keys, nor null values.
- * A Hashtable is synchronized, while a HashMap is not. **Hash table is thread safe and hash map is not. Thus, HashMap is preferred in single-threaded environments, while a Hashtable is suitable for multi-threaded environments.**
- * A HashMap provides its set of keys and a Java application can iterate over them. Thus, a HashMap is fail-fast. On the other hand, a Hashtable provides an Enumeration of its keys.
- * The Hashtable class is considered to be a legacy class.

How does a HashMap work? What about hashCode?

A HashMap in Java stores key-value pairs. The HashMap requires a hash function and uses hashCode and equals methods, in order to put and retrieve elements to and from the collection respectively. When the put method is invoked, the HashMap calculates the hash value of the key and stores the pair in the appropriate index inside the collection. If the key exists, its value is updated with the new value. Some important characteristics of a HashMap are its capacity, its load factor and the threshold resizing.

When we 'get' from hashMap, first it finds the hashCode of the key, using hashCode, finds the location/bucket of all keys having same hashCode. Then it uses the equals method of the key to identify which key value pair to be fetched from the bucket.

HashMap uses hashCode(), == and equals() for entry lookup. The lookup sequence for a given key k is as follows:

2. Use k.hashCode() to determine which bucket the entry is stored, if any
3. If found, for each entry's key k1 in that bucket, if k == k1 || k.equals(k1), then return k1's entry
4. Any other outcomes, no corresponding entry

5. Java version difference 1.7 and 1.8

1. Lambda Expressions
2. Pipelines and Streams
3. Date and Time API
4. Default Methods

5. Type Annotations

6. Interface / Abstract class

For interface, we normally only define the method, it's like a standard, it doesn't have any implementation, so if a class implement an interface, you have to implement all the method of the interface in order to fulfill the interface standard.

Abstract class on the other hand is a little different. Abstract class is still a class, so you can only inherit, you should not implement, secondly, an abstract class can have a method with the implementation, also it can have a method that without implementation as well, though we have to add a signature called abstract method. So when the class inherit the abstract class, it has to implement the abstract method only. But also we can override existing method with implementation. We can implement multiple interface, but we can only inherit one abstract class. Sometimes we use abstract class, because we can provide some existing implementation and save us time of rewrite different code. Other times, if we need to implement multiple method from different interface, then we have to choose interface to implement.

7. Final / Finalize / Finally

Final: apply restrictions on class, method, and variable

- **Final class:** can't be inherited
- **Final method:** can't be overridden
- **Final variable value:** can't be changed

Finalize: perform clean up processing before object is garbage collected

Finally: place important code, will be executed whether exception is handled or not

8. ArrayList / LinkedList

advantages/disadvantages between the two—when you'd use one or other

ArrayList	LinkedList
Implemented dynamic array	Implemented doubly linked list
Insertions/removal $O(n)$ (resizing array & copy to new array if full)	Insertions/removal $O(1)$ (easy and fast)

& Update its index if insertion)	
(each index only holds actual object (data))	Has more memory overhead (each node holds both data and address of next and previous node)
Search O(n) faster (if sorted, use Binary Search O(Log n) & Get() random access)	Search O(n) (checking until end)

Detail explanation:

- ArrayList implemented with the concept of dynamic array, LinkedList implemented with the concept of doubly linked list
- Insertions are easy and fast in LinkedList as compared to ArrayList because there is **no risk of resizing array** and **copying content to new array** if array gets full which makes adding into ArrayList of O(n) in worst case, while adding is O(1) operation in LinkedList in Java. ArrayList also needs to be **update its index** if you insert something anywhere except at the end of array.
- Removal also better in LinkedList than ArrayList due to same reasons as insertion.
- LinkedList has more memory overhead than ArrayList because in ArrayList each **index only holds actual object** (data) but in case of LinkedList **each node holds both data and address** of next and previous node.
- Both LinkedList and ArrayList require O(n) time to find if an element is present or not. However we can do Binary Search on ArrayList if it is sorted and therefore can search in O(Log n) time.

<https://www.geeksforgeeks.org/arraylist-vs-linkedlist-java/>

9. Iterator

What is iterator?

It is a universal iterator as we **can apply it to any Collection object**. By using Iterator, we **can perform both read and remove operations**. It is improved version of Enumeration with additional functionality of remove-ability of a element.

Iterator must be used whenever we want to enumerate elements in all Collection framework implemented interfaces like Set, List, Queue, Deque and also in all implemented classes of Map interface. Iterator **is the only cursor available for entire collection framework**.

<https://www.geeksforgeeks.org/iterators-in-java/>
<https://www.geeksforgeeks.org/how-to-use-iterator-in-java/>

10. Pass by reference / pass by value

Does java pass by reference or pass by value?

Java is Strictly Pass by Value!

- Primitive type is passed by Value
- References also passed by value
- We can also change members using reference if we do not change the reference itself

Example:

<https://www.geeksforgeeks.org/g-fact-31-java-is-strictly-pass-by-value/>

(For primitive types, it pass by value, others pass by reference.)

There are 8 primitive types:

Byte | short | int | long | char | float | double | boolean

11. Example program:

```
int a = 10;  
int b = a;  
a = 14;  
System.out.println(b);
```

What is output 14 or 10?

Output is 10.

12. static / non-static variables

The instance method is where we have to construct instance from the class and invoke from the instance. 每个房间有一个toolbox

Static method is where we can invoke directly from a class. 一个楼公用一个, 跟着class, behavior goes along with whole class

<https://medium.com/@gabriellamedas/differences-between-static-and-non-static-methods-in-java-b93156be75c6>

13. Sort comparable/Comparator

Comparable is implemented in object, the object itself can be compared with another external instance, so when we use a `collection.sort` or `list.sort`, without supply any other argument, first look at the object itself, has it implemented the comparable, so it can be compared

Comparator is a little different, it is an external function interface, and you can implement it multiple times with different compare logic, when use `list.sort` or `collection.sort`, we can supply external comparator logic to sort it in different ways.

Comparable and comparator shares the same internal method logic, that is returning integer, if it is negative it is less than, if it is positive it is greater than or if it is zero, it means equal

14. Java Modifier

Access Levels

Modifier	Class	Package	Subclass	World
public	Y	Y	Y	Y
protected	Y	Y	Y	N
no modifier	Y	Y	N	N
private	Y	N	N	N

Diff of protected and private: the most commonly seen difference is that when declared private, it can not be accessed by its subclass even though subclass inherit from original class, on the other hand when declared protected it can still be accessed by subclass.

15. Polymorphism and Encapsulation

Polymorphism for example, a deer can be a vegetarian and also an animal. An animal is an interface, if it has a tail, four feet, we call it animal, vegetarian can be another interface. So a deer has the attribute for both interfaces. In java, it is object oriented programming and it is a single inheritance model, so we can only inherit from one class, in order to achieve polymorphism, we can inherit from one class and implement multiple interfaces

Encapsulation: one way to do encapsulation is the get() set() method, so we can set the attribute to private and only expose a get set method, and this is encapsulation because other people or external program or other package does not have to know any logic inside the getter setter method

16. OOP

Object oriented programming is to abstract each object in the real time into a class in java. And from the class we can create multiple instances with different properties. For example, we can define wheel shape, wheel number in class bike and the motorcycle can inherit from bike and initialize wheel number and wheel shape.

17. GIT

Yes, i use git in my day to day work and we use git to merge our feature branch after code is reviewed by supervisor with a standard merging process in git branch so we won't mess up the master branch

MVC SPRING

1. What is spring?

It's application development framework for enterprise applications in Java.

The goal of Spring is to make J2EE development easier to use and promote good programming practice.

2. Why MVC:

MVC: Model View Controller

It's a design pattern, we use mvc and event driven in our framework.

Typically, in our spring framework, we have a controller, we have a model, we have a service, but we don't have a view, because view has been taken care of by front end and we use a REST API to communicate between front end and back end.

In the controller, it is where we write our API. The model, is where we structure our models in order to map to the database.

We use Hibernate to do object relationship mapping to the database table.

Service is where we provide different services. It's a basic wrap of the repository layer. So any time we change any data, we can use service layer to change the data and then save to the database by using repository. In this way, the logic has been coded in the service layer and we don't have to change anywhere besides service itself. So the controller may invoke service and later on if there is any logic change we just need to touch the service layer and we don't need to rewrite the controller at all.

Its component of Spring that enables implementation of MVC design pattern.

1.Handler Mapping > 2.Controller > 3. View Resolver > 4.View

>DispatcherServlet<

HTTP request or HTTP response

Explain / Advantages of Spring

*Lightweight - framework doesn't need to be completely loaded to be useful, so spring framework is very large, so there may be a point in which you are only working with one component of the spring framework and you don't need to work with any of other components which makes it lightweight you know so you don't have to use all of you can only use one you have the ability to use one framework without using another one.

*Flexible - Spring is a framework that is primarily based on dependency injection. Configuration by XML or annotation based style

*Loosely coupled - Spring allows loose coupling through Dependency Injection and interface orientation

*easy to test - provides testing modules that make testing easier.

3. Spring vs. SpringBoot

Spring: is an Xml based configuration; Application creation or execution takes more time; More codes

Spring Boot : based on Annotation; Application creation / execution is quicker; reduced amount to boiler plate code.

4. Autowiring

It is the process of injecting object dependencies; Object can be injected in setter or constructor of a class;

Two types of Autowiring: XML Autowiring and Annotation Autowiring.

5. IoC container

It is a framework used to create and inject dependencies automatically;

There are two types of IoC containers:

BeanFactory - Simple container used for DI;

ApplicationContext - advanced container used for DI;

Types of autowiring modes

byName - auto wire by property name;

byType - auto wire by datatype

Constructor - auto wire by constructor

Bean scopes

It determines the type of bean instance that should be returned

Types of Scopes:

Singleton - return single bean instance per Spring IoC container

Prototype - return new bean instance each time when requested

Request - return a single bean instance per HTTP request

Session - return a single bean instance per HTTP session

Global session - return a single bean instance per global HTTP session

Bean?

Object managed by spring IoC container

How to use annotation (@Component, @Autowire, @RestController, @Controller, @RequestMapping)

Component - Marks a class that will be registered in context as a bean

Autowire - Spring auto wire other beans in your class

RestController - mark a class as controller when developing restful web services

Controller - mark a class as controller in Spring MVC

RequestMapping - map urls to a class or a handler method when developing rest web service applications

Different modules in Spring

Spring core - provide fundamental parts of framework; IoC and DI features

Data Access - Consist of JDBC, ORM, OXM, JMS, and Transaction Modules

Web - Module related to web application development; web services, web Servlet Module

AOP/Aspect/Instrumentation - Module dedicated to Aspect Oriented Programming

Test - Supports unit testing using Unit and TestNG; Mock Objects (test code in isolation)

Dependency Injection

Instead of creating the dependencies yourself, ask a third party object or framework to do so.

6. What is docker?

Docker has the ability to reduce the size of development by providing a smaller footprint of the operating system via containers.

With containers, it becomes easier for teams across different units, such as development, QA and Operations to work seamlessly across applications.

Difference Docker Image and Container

Docker container is the runtime instance of docker image; Docker Image doesn't
Tomcat is a Servlet container

7. What is servlet?

The servlet is a Java programming language class used to process client requests and generate dynamic web content. Servlets are mostly used to process or store data submitted by an HTML form, provide dynamic content and manage state information that does not exist in the stateless HTTP protocol.

Explain the architecture of a Servlet.

The core abstraction that must be implemented by all servlets is the `javax.servlet.Servlet` interface. Each servlet must implement it either directly or indirectly, either by extending `javax.servlet.GenericServlet` or `javax.servlet.http.HttpServlet`. Finally, each servlet is able to serve multiple requests in parallel using multithreading.

What is the difference between an Applet and a Servlet?

An Applet is a client side java program that runs within a Web browser on the client machine. On the other hand, a servlet is a server side component that runs on the web server. An applet can use the user interface classes, while a servlet does not have a user interface. Instead, a servlet waits for client's HTTP requests and generates a response in every request.

What is the difference between `GenericServlet` and `HttpServlet`?

`GenericServlet` is a generalized and protocol-independent servlet that implements the `Servlet` and `ServletConfig` interfaces. Those servlets extending the `GenericServlet` class shall override the `service` method. Finally, in order to develop an HTTP servlet for use on the Web that serves

requests using the HTTP protocol, your servlet must extend the HttpServlet instead. Check Servlet examples here.

Explain the life cycle of a Servlet.

On every client's request, the Servlet Engine loads the servlets and invokes its init methods, in order for the servlet to be initialized. Then, the Servlet object handles all subsequent requests coming from that client, by invoking the service method for each request separately. Finally, the servlet is removed by calling the server's destroy method.

8. JWT Token

Jwt token has 3 paragraph. The first paragraph is Header. Header indicated algorithm/Token type. The second paragraph is the Payload. Payload indicated any information that you would like to see in the token. The third paragraph is verify signature . It actually compiles header and payload encrypt with a secret key.

The application server receives the secret key from the authentication server when the application sets up its authentication process. Since the application knows the secret key, when the user makes a JWT-attached API call to the application, the application can perform the same signature algorithm as in Step 3 on the JWT. The application can then verify that the signature obtained from it's own hashing operation matches the signature on the JWT itself. If the signatures match, then that means the JWT is valid which indicates that the API call is coming from an authentic source.

9. Restful API

Parameter

Payload

Difference between Soap and Restful web service

<https://stackoverflow.com/questions/19884295/soap-vs-rest-differences>

REST is an architectural style. It doesn't define so many standards like SOAP.

REST is for exposing Public APIs(i.e. Facebook API, Google Maps API) over the internet to handle CRUD operations on data.

SOAP brings its own protocol and focuses on exposing pieces of application logic (not data) as services.

- Since REST uses standard HTTP it is much simpler in just about every way.
- REST is easier to implement, requires less bandwidth and resources.
- REST permits many different data formats where as SOAP only permits XML.

- REST allows better support for browser clients due to its support for JSON.
- If we need Stateless CRUD operations then go with REST.

What are RESTful web services and the basic 4 verbs/methods?

https://www.w3schools.com/tags/ref_httpmethods.asp(Comprehensive Of HTTP request method.)

GET, PUT, POST, DELETE, PATCH

GET is used to request data from a specified resource, so it is to retrieve data

POST is used to send data to a server to create/update a resource, which is create data

PUT is to replace the whole object, so we can modify many attribute

PATCH is to modify one attribute, such a change a family name

The difference between POST and PUT is that PUT requests are idempotent.

That is, calling the same PUT request multiple times will always produce the same result. In contrast, calling a POST request repeatedly have side effects of creating the same resource multiple times.

DELETE is to delete the specified resource.

When inserting data into a REST call, is it doing an update command on a table or an add command?

ADD, is called POST, update command is put.

How do you define synchronous and a-synchronous web services? In REST space, what are the underlying API being used?

"Synchronous" or "Asynchronous" is the behaviour of the client that is requesting the resource.

It has

nothing to do with REST webservice, its structure, or the supporting server.

Synchronous behaviour:

- Client constructs an HTTP structure, sends over the socket connection.
- Waits for the response HTTP.

Asynchronous behaviour:

- Client constructs HTTP structure, sends the request, and moves on.
- There's another thread that is waiting on the socket for the response. Once response arrives, the

original sender is notified (usually, using a callback like structure).

how do you handle changes in web service definition for restful web services?

....

@Deprecated

REST API: If got a code request that didn't go through and got an error code, how do you

recognize if a request was successful in REST API?

1. It is gonna show the code : 200
2. Using API test tool like postman, swagger, if you got what you want.

REST API: Familiar with put, patches? Write a java function to find the second highest number

1. The existing HTTP PUT method only allows a complete replacement of a document. This proposal adds

a new HTTP method, PATCH, to modify an existing HTTP resource.

1. Sort, find num[lenth()-2]

What is HTTP versus https?

HTTP VS HTTPS. Hyper Text Transfer Protocol Secure (HTTPS) is the secure version of HTTP, the

protocol over which data is sent between your browser and the website that you are connected to. The 'S'

at the end of HTTPS stands for 'Secure'. It means all communications between your browser and the

website are encrypted.

10. Try catch

Try catch is to catch the exceptions, the reason we throw exception is that certain method has an unexpected result in the runtime. For example, one method in the service layer throw exception, we can throw a nullpointerexception, in that method signature, we can have a throw nullpointerexception. And if any method to invoke that method, we need to either catch that method in the method implementation, or continue to throw that exception from that method until the higher level method to invoke that.

11. Authentication and authorization

Authentication is a basically a login

Authorization is once the user log in we have to determine certain resource can be requested by user by different type of role

AWS

1. Authentication provider chain

provides different layer of authentication and usually in the dev environment and production environment we are using different authentication strategy. Production is handled by deVops, but authentication provider chain provides different ways to authenticate, like in local we use key and secret, but on the production we use IM role which is much secure

1. EC2

Amazon Elastic Compute Cloud, launching, connecting a Linux instance (virtual server) in the AWS cloud. deployed tomcat at ec2 for dev environment

2.S3

Amazon Simple Storage Service is a storage service for the internet, you can store and retrieve any amount of data at any time from anywhere on the web. It is a more reliable solution and provides centralized storage. And we can integrate existing feature without writing a lot of code. We can store our file in S3, and whenever we need it, we can retrieve from it. And they also provided a very simple SDK integration tools, we can grab the library from maven and provide authentication provider chain in order for us to authenticate it and act with S3 service. Also it is very economic cost wise.

3. Sqs

SQS is a messaging service. I use SQS to make my server asynchronous. I store all the events that I can execute later in the SQS, and use spring boot to deal with those events later. It can save a lot of time of the main server and make the whole project more scalable.

4. How do we send request to Amazon S3?

You can send request by using the REST API or the AWS SDK wrapper libraries that wrap the underlying Amazon S3 REST API
REST API

SQL:

1. find only managers from Employee

Sql query to find only managers from Employee table?

```
SELECT e.* FROM Employee e  
Join Manager m  
On e.empID = m.empID;
```

Sample answer:

```
SELECT E.* FROM Employee E, Manager M  
WHERE E.empID = M.empID;
```

2. find highest salary

Sql query to find highest one salary employee?

Select empID, **MAX**(salary) **as** salary **from** Employee

3. 2nd highest salary

How would you find the second highest employee salary in the Employee table?

With cte as

(select *, **Dense_rank()** **over**(**order by** salary **desc**) **as** Rnk **from** Employee)

Select name **from** cte

Where Rnk = 2;

<https://www.geeksforgeeks.org/sql-query-to-find-second-largest-salary/>

Notes:

ROW_NUMBER(), RANK(), DENSE_RANK() difference:

code	model	color	type	price	num	rnk
2	1433	y	Jet	270.00	1	1
3	1434	y	Jet	290.00	2	1
1	1276	n	Laser	400.00	3	3
6	1288	n	Laser	400.00	4	3
4	1401	n	Matrix	150.00	5	5
5	1408	n	Matrix	270.00	6	5

code	model	color	type	price	rnk	rnk_dense
2	1433	y	Jet	270.00	1	1
3	1434	y	Jet	290.00	1	1
1	1276	n	Laser	400.00	3	2
6	1288	n	Laser	400.00	3	2
4	1401	n	Matrix	150.00	5	3
5	1408	n	Matrix	270.00	5	3

http://www.sql-tutorial.ru/en/book_rank_dense_rank_functions.html

4. Count # of Employee

Count number of employees in department

```
Select count(Name) as 'cnt' from Employee
```

5. Count datetime

One column table contains dd/mm/yyyy, count number

```
SELECT COUNT(*) AS Total,  
      (CAST(DAY(ApplicationDate) AS VARCHAR(2)) + '-' +  
      (CAST(MONTH(ApplicationDate) AS VARCHAR(2)) + '-' +  
      CAST(YEAR(ApplicationDate) AS VARCHAR(4))) as ApplicationDate  
FROM [Jobs]  
GROUP BY  
      (CAST(DAY(ApplicationDate) AS VARCHAR(2)) + '-' +  
      (CAST(MONTH(ApplicationDate) AS VARCHAR(2)) + '-' +  
      CAST(YEAR(ApplicationDate) AS VARCHAR(4)))  
ORDER BY  
      ApplicationDate
```

Or use cte:

```
;WITH GroupedData AS  
(  
  SELECT  
    DateMonth = MONTH(ApplicationDate),  
    DateYear = YEAR(ApplicationDate),  
    JobsCount = COUNT(*)  
  FROM  
    dbo.[Jobs]  
  GROUP BY  
    YEAR(ApplicationDate),  
    MONTH(ApplicationDate)  
)
```



```

SELECT
    RIGHT('00' + CAST(DateMonth AS VARCHAR(2)), 2) + '-' +
    CAST(DateYear AS VARCHAR(4)),
    JobsCount
FROM
    GroupedData
ORDER BY
    DateMonth, DateYear

```

<https://stackoverflow.com/questions/20536422/sql-query-to-count-based-on-mm-yyyy-part-of-date>

6. Find missing record in another table

Assume that some of manager's records are missing in Employee table, using sql query how do you find those records?

```

Select * from Manager m
Where m.empID NOT IN (select e.empID from Employee e)

or:

Select m.* from Manager m
Left join Employee e on m.empID = e.empID
Where e.empID is NULL

```

Sample answer:

```

SELECT * FROM Employee E
WHERE E.empID NOT IN (SELECT M.empID FROM Manager M);

```

7. Find difference between 2 tables

If you have two identical tables with same columns and those tables has thousands of records and how do you find difference between those tables data?

```

SELECT A.*, B.* FROM A
FULL JOIN B ON (A.C = B.C)
WHERE A.C IS NULL OR B.C IS NULL

```

Or:

```
SELECT 'TABLE1-ONLY' AS SRC, T1.*
FROM (
    SELECT * FROM Table1
    EXCEPT
    SELECT * FROM Table2
) AS T1
UNION ALL
SELECT 'TABLE2-ONLY' AS SRC, T2.*
FROM (
    SELECT * FROM Table2
    EXCEPT
    SELECT * FROM Table1
) AS T2
;
```

<https://stackoverflow.com/questions/2077807/sql-query-to-return-differences-between-two-tables>

8. Check 2 tables are same

How to check two large tables are same?

Using Left join:

```
select *
from dbtest02.dbo.article d2
left join dbtest01.dbo.article d1 on d2.id=d1.id
```

Using except:

```
select * from dbtest02.dbo.article
except
select * from dbtest01.dbo.article
```

The except returns the difference between the tables from dbtest02 and dbtest01.

<https://www.mssqltips.com/sqlservertip/2779/ways-to-compare-and-find-differences-for-sql-server-tables-and-data/>

9. Find duplicates in table

DB/SQL How do you know if there are any duplicate values in table?

```
SELECT name, email, COUNT(*) FROM Users  
GROUP BY name, email  
HAVING COUNT(*) > 1
```

Or:

```
Select COUNT(1) As Total_Rows from TableA  
Select Count(1) As Distinct_Rows from ( Select Distinct * from TableA) abc
```

10. Find duplicates in a column

How do you find that a column in a table has duplicates?

```
SELECT Count(*) AS DistinctCountries  
FROM (SELECT DISTINCT Country FROM Customers);
```

Or:

```
Select Column from TableA  
Group by Column  
Having count(Column) > 1
```

Sample answer:

```
SELECT author_id, author_name  
FROM ( SELECT author_id, author_name , count(*) OVER (PARTITION BY  
author_name) AS ct  
FROM author_data ) sub WHERE ct > 1;
```

11. truncate / delete

DB/SQL: Difference between truncate and delete?

Truncate	Delete
Data cannot be rolled back	Data can be rolled back
faster	Slower (remove row-by-row one at a time and records an entry in transaction logs)
Delete all records	Can delete partial
Reset identity value	Don't reset

12. Where / Having

What is the difference between "where" and "having" clause?

where	having
Use before grouping	Use after grouping
Cannot have aggregate functions	Can have aggregate functions
Can use on select, insert, update	Only on select

13. Joins

What is the difference between outer joins and inner joins?

How do you use inner joins, outer joins?

inner	inner:	Select and return the records that have matched values in both tables
outer	Left outer join:	Return all the records from the left table, and matching results from the right table (starting with the left table and then any matching table records from right)
	Right outer join:	Same as left outer join

	Full outer join:	Return all records from both tables whether the other table matches or not
self	self	A table that is being joined to itself (usually used on left join itself) referencing the same table

The most common case for this is when you're matching the foreign key of one table to the primary key of another, such as when using an ID to lookup a value.

14. How can you do a join statement?

```
SELECT column_name(s)
FROM table1
LEFT JOIN table2 ON table1.column_name = table2.column_name;
```

15. Update / Insert

*When do you use an **update**; when do you use an **insert**?*

Update: update an existing record

Insert: insert a new record

16. Soccer team

soccer league, team and player how to build table

Players p

Team t

Score s

P - t: M to 1

T - s: M to M → junction table: Matches m

query to retrieve players with team

Select

count player number for each team

Schedule all matches between Teams:

<https://stackoverflow.com/questions/46262575/query-to-schedule-a-match-between-each-team?rq=1>

Self join team1 <> team2

17.

- Table relationships
- Questions on set operators
- Using set key how could we get data fields from data tables
- GroupBy scenario
- Test Scenario
- Check if a number has a decimal place/is a whole number
- Best scenario to get positive numbers

Command line, How do you decide if the info in Table A has been copied into Table B correctly?

TESTING PROBLEM

In purple are testing related questions that may help you understand better the concept of the type of TDD Java developers they like.

You are testing code which is different from QA alongside coding. You don't need to feel intimidated by the questions. You don't need to know them all but these are great for helping you understand what TDD means.

1. *Testing problem - If numbers range 1-10, what is most efficient test numbers to use?*
1, 10.(边界) 0,11(Excpction) 5(Normal). Testing – structure of the unit test
2. *Are you a full stack engineer and do you take pride in your testing?*
3. *functional and non-functional testing?*
All is functional test for our project, i have notGUI non-functional testing...
4. *How do you minimize testing to maximize results?*

For each commit CI/CD

5. *what are the boundary values you used to test? boundary value testing?*
6. *best way to get test coverage? how do you get 100% test coverage? how much coverage do you have for unit testing?*
80%
7. *when you do java class with say 10 methods, how many Junit tests do you typically write?*
Usually i write the significant method for testing, if i have time, i will write the public method, access private method by access the public method which involks the private one.
8. *do you write tests for pos. as well as neg. Calculations?*
9. *Knowledge on TDD, automated testing. What kind of testing tools you use?*
Good to read up conceptually on tools like Selenium

Previous question:

1. *what is lazy loading in hibernate ?*
what is eager loading ?

<https://stackoverflow.com/questions/2990799/difference-between-fetchtype-lazy-and-eager-in-java-persistence-api>



878



Sometimes you have two entities and there's a relationship between them. For example, you might have an entity called University and another entity called Student.

The University entity might have some basic properties such as id, name, address, etc. as well as a property called students:

```
public class University {
    private String id;
    private String name;
    private String address;
    private List<Student> students;

    // setters and getters
}
```

Now when you load a University from the database, JPA loads its id, name, and address fields for you. But you have two options for students: to load it together with the rest of the fields (i.e. eagerly) or to load it on-demand (i.e. lazily) when you call the university's getStudents() method.

When a university has many students it is not efficient to load all of its students with it when they are not needed. So in suchlike cases, you can declare that you want students to be loaded when they are actually needed. This is called lazy loading.

2. *input string array, how to count frequency*

<https://www.geeksforgeeks.org/frequency-of-a-string-in-an-array-of-strings/>

3. *check is number a prime number*

<https://www.geeksforgeeks.org/prime-numbers/>

4. *in array move positive number to array left, move negative to right*

<https://www.geeksforgeeks.org/move-negative-numbers-beginning-positive-end-constant-extra-space/>

5. *Advantage of Spring*

what is IOC? why do we have IOC

<https://stackoverflow.com/questions/3058/what-is-inversion-of-control>

what is AOP

<https://stackoverflow.com/questions/242177/what-is-aspect-oriented-programming>

6. *advantage of loose coupling in test*

Loose Coupling is when two objects can interact with each other but have very limited knowledge about what the other object can do.

While this concept is relatively simple to understand, it's difficult to implement. Sometimes you may be under the gun to get a project out the door (hey, you gotta ship it, right?).

Some advantages of making your code loosely coupled include:

- **Better Testability**

Because your code isn't dependent on other objects and they are just passed in, this makes your unit tests easier to write.

- **Easy-to-understand code**

When your code is decoupled from other objects, they are usually passed in or dependency injected into the code. Your code provides a self-documenting service to your users.

- **Swappable components**

While most developers don't think about a plug-in architecture, this is ultimately what developers strive to achieve. If you want to swap out the Oracle database component with a SQL Server component, if developed properly, it can be done easily.

- **Scalability**

As your system grows, you can provide a diverse number of components to plug into your application, making it more scalable. There is a term I use when a system can't scale properly. It's called "painting yourself into a corner" where you need to re-evaluate your design.

- **Isolated Code/Features**

Adding new features to a system means that you can write additional code without breaking existing functionality and feel safe writing it.

But how do you know when a code base has been "tightly wound?"

<https://dzone.com/articles/what-is-loose-coupling>

<https://8thlight.com/blog/chris-jordan/2015/08/26/loosely-coupled-testing.html>

7. Ransom:{"we","is","are","is"}
Newspaper:{"we","are","good","car","is","nice","it","is","new"}
Boolean if all the words in ransom are in the newspaper

<https://codereview.stackexchange.com/questions/133442/validate-whether-a-ransom-note-can-be-made-from-a-dictionary-of-words>

8. Diff between authorization and authentication

9. Sql injection, how to deal with it

<https://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks.html>

10. How to create document - swagger

<https://www.3pillarglobal.com/insights/restful-api-documentation-using-swagger-and-spring-mvc>

11. 1) finding island, 1 stands for island, 0 stands for sea

00000

11001

10011

00011

Return 2, only check left, right, up and down

<https://www.geeksforgeeks.org/find-number-of-islands/>

12. 2) Class car {

Private String model;

}

Map<Car, int> map

Car car1 = new Car();

Car car2 = new Car();

map.put(car1, 1000);

car1.setName();

car1.setMaker();

...

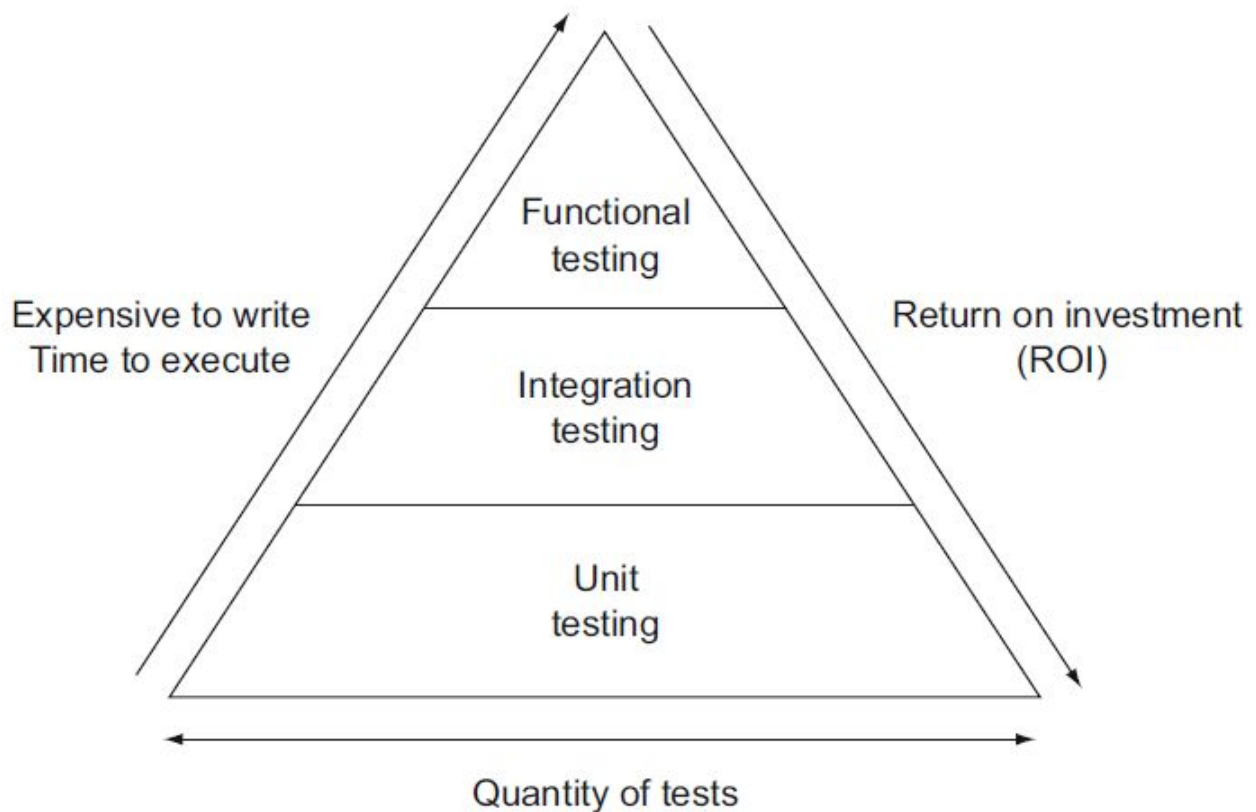
Return what map.get(car2);

13. 3) How hashCode(), equals() works in Map, how put and get of Map work

Apart from Unit Test, any other types of test you used in previous work? Mockito

where you are using testing, which project?

Any functional, integration testing?



<https://codeutopia.net/blog/2015/04/11/what-are-unit-testing-integration-testing-and-functional-testing/>

How do you ensure your method return?

How do you know what tests are good?

how do you know test code coverage?

Code coverage is a measurement of how many lines/blocks/arcs of your code are executed while the automated tests are running.

Code coverage is collected by using a specialized tool to instrument the binaries to add tracing calls and run a full set of automated tests against the instrumented product. A good tool will give you not only the percentage of the code that is executed, but also will allow you to drill into the data and see exactly which lines of code were executed during a particular test.

Our team uses [Magellan](#) - an in-house set of code coverage tools. If you are a .NET shop, Visual Studio has integrated tools to collect code coverage. You can also roll some custom tools, like [this article](#) describes.

If you are a C++ shop, Intel has some [tools](#) that run for Windows and Linux, though I haven't used them. I've also heard there's the gcov tool for GCC, but I don't know anything about it and can't give you a link.

As to how we use it - code coverage is one of our exit criteria for each milestone. We have actually three code coverage metrics - coverage from unit tests (from the development team), scenario tests (from the test team) and combined coverage.

BTW, while code coverage is a good metric of how much testing you are doing, it is not necessarily a good metric of how well you are testing your product. There are other metrics you should use along with code coverage to ensure the quality.

<https://stackoverflow.com/questions/195008/what-is-code-coverage-and-how-do-you-measure-it>

Did you use any code quality tools?

what test cases you will write for your rest web services?

Postman is used for functional test right? Yes

How many users/how many calls for your rest web services per day?

What is dependency injection?

<https://medium.freecodecamp.org/a-quick-intro-to-dependency-injection-what-it-is-and-when-to-use-it-7578c84fa88f>

what tools you used for your user interface testing?

What is Security testing?

What do you know about selenium, cucumber, protractor

<https://semaphoreci.com/community/tutorials/getting-started-with-protractor-and-cucumber>

Test-driven development (TDD)

Related to Resume:

- Java, Spring stack (hibernate, security, etc), Unit testing technologies, Agile – and can show this in a discussion/interview.
- Angular or later JS framework, working with REST apis, always a plus
- AWS Cloud - or something else along these lines always a plus
- Javascript
- ability to talk in depth about any of your experiences
- use of modern practices
- Cloud/AWS experience or conceptual knowledge (not required but always a bonus)
- XML – parsers
- Knowledge on continuous integration, different types of testing frameworks
- Agile/sprint knowledge. JIRA story and its points

Brainteasers Questions:

key isn't getting a right answer; it's to gauge how analytical and problem-solving you are; how comfortable exploring different approaches to the unknown, & NOT being afraid to fail.

- One ball is slightly heavier out of 8 balls. How many times do you need to use a scale to determine which is heavier?

<https://www.youtube.com/watch?v=uBnI5FIJsxQ>

- Bacteria multiplies every minute. You place a bacterium into an empty cup at 12. By 1pm, it's full. At what time was it half full? (think EXPONENTIAL concept)
- Two sets of numbers: IDs of employees and IDs of managers. Write a code function to get all the employees that were not managers.
- If you have a group of people and they all left a handshake with each other, what formula would you use to find total # of handshakes? Summation formula? If product costs from \$1 to \$10, which range of number used to test the code?
- There are 8 bottles and 3 rats. One bottle is poisoned. Each rat, once poisoned, has 5 minutes to live. Within 5 min, how can you use all 3 rats to figure out which bottle was poisoned?
- There is a cake with a hole somewhere random in the cake. How can we divide so it's evenly in half? Different ways? There are 2 cars at a railroad track 100miles. Each car is moving at 10mph and there is a flying bird that flies back and forth 20mph and turns around whenever it hits a car. How far did the car travel before it hit both trains?
- There are 10 light switches. How many different combinations can you have of on/off? (probability q)
- Flip a coin heads/tails. If first 3 tosses was tails and six were heads, 10th time - What are the chances last was heads or tails?
- There are 5 lanes. 25 horses. How many times do you need to race to get the top 5 horses?

About Finra:

www.finra.org

Explains who the company is FINRA Technology

<http://technology.finra.org>

Please review this website. They will ask during the interview what you thought of it.
FINRA

named among Best places to work for 5th consecutive year FINRA Videos

:

<https://www.youtube.com/watch?v=SgHyXF-zWfM>

<https://www.youtube.com/watch?v=xbQQnU1OfGY>

<https://www.youtube.com/watch?v=rHUQQzYoRtE>

<http://www.businesswire.com/news/home/20160614005839/en/Computerworld-Names-FINRA-Places-Work-5th-Consecutive>

<https://vision.cloudera.com/managing-big-data-for-audit-compliance-and-business-intelligence/>

<http://marketsmedia.com/regulations-drive-big-data%E2%80%94part-ii/>

<http://www.fiercefinanceit.com/story/finra-leverages-cloud-and-hadoop-its-consolidated-audit-trail-proposal/2014-10-16>

<http://aws.amazon.com/solutions/case-studies/finra/>

<http://www.investmentnews.com/article/20141026/REG/310269999/finras-cards-plan-is-like-ly-to-be-a-game-changer>
Company:

independent securities firm worth over \$2 billion in assets. Regulates U.S. securities industry, for New York Stock Exchange LLC (NYSE), NASDAQ groups and others. High-impact/visibility, long-term projects. Just GOOGLE them in News. FINRA is notorious for caliber of tech talent and techs (Java,

Python, noSQL, real-time data analytics, Big Data, Hadoop Ecosystem, cloud/AWS, trading algorithms —

they are one of the FEW select partners with Amazon Web Services). I've observed for years that people

with FINRA on resume are HIGHLY sought after. It's an awesome place to start foundation of your

career as a developer. You will have amazing growth opportunities and work with sharp, bright developers who are engaged problem solvers. As mentioned in video, FINRA is here "to bend the curve

towards the future" and "help evolve the industry." Director Raghu Raman talks about breaking into

new collaborative modern settings and how "we are technologists to the core and it bounds us together

and that makes me want to keep coming back every day.

Past

Always hit the point:

- FINRA looks at someone's experience.
- take with grain of salt. They tailor questions to how experienced you are
- Algorithmic brain teasers Google-style - NOT meant to trick people or make them race for the right answer- more about who's comfortable with the unknown, viewing the unknown as a challenge and opportunity to dig into problem solving skills and be okay with being wrong and pivoting