**NOTES:**

[A]: Basic knowledge, need to know. (Junior)

[B]: Advanced knowledge, should know. (Senior)

[C]: High-level knowledge, good to know. (TA/SA)

[D]: Optional knowledge. You’re will have more advantages when knowing. If don’t know, dismiss.

# PART 0. INTRODUCTION

## [A] Could you please introduce yourself and tell us more about working experience?

You need to outline your ideas into some items:

* Part 1: Name (short name)
* Part 2: Number of year experience
* Part 3: Experiences (framework, technologies, techniques, difficult solved-problem).
* Part 4: The most exciting projects (choose the best one). Tell about techniques and solution for this project.
* Part 5: Motivation.

## [A] Introduce about your recent projects, role, and responsibility?

Introduce about “difficult”, “challenging” project first, **don’t care timeline/order.**  
For each project, need to introduce:

* + Small description about project’s business, don’t care project name (because as usual project name is just an acronyms or doesn’t make sense).
  + Technologies was used in the project.
  + Your responsibilities (not manager, need to be a key person)

# PART 1. BASIC

## [A] What equals() and hashCode() method respond for? How and when override them?

* Open equals and hashcode function in Object.java source code.
* Understand how hashcode and equals are used in HashMap by opening HashMap source code.
* Understand contract between equals and hashcode function.

## [A] When you override hashCode, which method do you override?

* Equals, why? 🡺 refer to contract between hashcode and equals function (open source code of HashMap to view)

## [A] If two objects are not equals by equals() method 🡪 Is it require that hashcode must be the same?

* Possible, hashcode can be the same eventhough equals = false. (base on contract between hashcode and equals function)

## [A] What are differences between Deep copy and Shallow copy?

* Deep copy: clone object and related child object
* Shalow copy: just copy referfence address of an instance.

## [A] Is String mutable? Why?

* String is immutable. Because String need to be a thread-safe object in order to process for many threads and considered as a constant for many formulas.

## [A] What finally() block use for?

* To release/clean up resource after using or after throwing exception.

## [A] What are pass by reference and pass by value?

* Pass by reference means the called functions parameter and the callers passed argument are the same.
* Pass by value means the called functions parameter is a copy of the callers passed argument.
* In Java, It is pass by value. Java passes objects as references and those references are passed by value.

## [A] What is the difference between a constructor and a method?

* Constructor, member function, used to create objects of that class.
* Method, ordinary member function of a class, has own name, a return type

## [A] What if the main method is declared as private?

* Will compile, but it will not run.

## [A] What if the static modifier is removed from the signature of the main method?

* Will compile, but it will not run.

## [A] What if I write “static public void” instead of “public static void”?

* Will compile, and will run.

## [A] What if I don’t provide the String array as the argument to the method?

* Will compile, but it will not run.

## [A] What is the first argument of the String array in main method?

* Will compile, but it will not run.

## [A] If I do not provide any arguments on the command line, then the String array of Main method will be empty or null?

* Empty

## [B] How to print “Hello World” with only one line of code in Java?

## [A] What environment variables do I need to set on my machine in order to be able to run Java program?

## [A] Can an application have multiple classess having main method?

* Yes

## [A] Can I have multiple main methods in the same class?

* No

## [B] Do I need to import java.lang package any time? Why?

* No

## [A] Can I import same package/class twice?Will the JVM load the package/class twice at runtime?

* Yes and No

## [A] Does importing a package imports the subpackages as well? E.g. Does importing com.MyTest.\* also import com.MyTest.UnitTests.\*?

* No, only classes on same package. Subpackage NO.

## [A] What is the difference between declaring a variable and defining a variable?

* String str; <- Declaring
* String str = "Hello"; Declaring and Defining.

## [A] What is the default value of an object reference declared as an instance variable?

* null

## [A] Can a top level class be private or protected?

* public, abstract & final are allowed modifiers for a class.

## [A] What type of parameter passing does Java support?

* Pass By Value

## [A] Primitive data types are passed by reference or pass by value?

* Pass By Value

## [A] Object are passed by value or by reference?

* Pass By Value

## [A] Give a simplest way to find out the time a method takes for execution without using any profiling tool?

* Get time difference.

## [A] What are wrapper classes? Why do we need wrapper classes?

* A class that wraps primitive

## [A] What is the difference between the instanceof and getClass, these two are same or not?

* instanceOf, operator
* getClass(), method

## [A] What is a static block?

* Executed first when an object is instantiated

# PART 2. EXCEPTION HANDLING

## [A] Is it necessary that each try block must be followed by a catch block?

* No. Either Catch, Finally or Both.

## [B] What are Checked and UnChecked Exception?

* Checked, compile time
* Unchecked, runtime

## [A] What are runtime exceptions?

* Exceptions occurred at runtime

## [B] What is the difference between error and an exception?

* Error - Unrecoverable
* Exception - Recoverable

## [A] How to create custom exception?

* extends Exception

## [A] What are the different ways to handle exceptions? When should you use a particular exception handling between the one you give in your answer?

* Try, Catch, Throws

1. **What is the basic difference between the 2 approaches to exception handling.**  
   1> try catch block and (You handle the problem)  
   2> specifying the candidate exceptions in the throws clause? (You let them handle the problem)  
   When should you use which approach? Try - owning responsibility. Throws when making libraries or let them own the responsibility.

## [A] If I write return at the end of the try block, will the finally block still execute?

* Yes

## [A] If I write System.exit (0); at the end of the try block, will the finally block still execute?

* No

# PART 3. JAVA OOP

## [A] Compare Object vs Class vs Instance?

* An object is a software bundle of related state and behavior.
* A class is a blueprint/template or prototype from which objects are created.
* An instance is a single and unique copy of a class that representing an Object.

## [A/B] Describe the principles of OOP? Explain for each principle?

* Encapsulation:
  + encapsulate states, behaviors in a class, abstract class or interface;
  + restrict access by access modifiers.
* Abstraction: abstract states/behaviors of an object by using abstract class, interface, extends, super, this.
* Inheritance: extends states, behaviors of a super class.
* Polymorphism: abstract class, interface, overriding, overloading.

## [A] How many way to implement inheritance in Java?

* Extends: ChildClass extends ParentClass.
* Delegation: ChildClass contain one attribute is an instance of ParentClass.

## [A] Why multiple inheritance is not supported in Java?

* Ambiguity around Diamond problem

## [A] Expplain the different forms of Polymorphism.

* Runtime Polymorhism( or Dynamic polymorphism): Overriding
* Compile time Polymorhism( or Static polymorphism): Overloading

## [A] What is an abstract class?

## [A] What is an Interface?

## [A] What is the ‘static’ keyword in java?

## [A] What is final in Java? How is it used?

## [A] What is method overloading, method overriding? What are the differences?

## [A] How to prevent a method from being overridden?

* final

## [A] Do interface have member variables?

* No. Why?

## [A] Which modifiers are allowed for methods in an Interface?

* public, abstract

## [A] How to override the main method?

* No

## [A] How to invoke a superclass version of an overridden method?

* Use super.method()

## [A] Compare Composition, Aggregation, Association vs Inheritance?

* Inheritance: Manager is an employee of XYZ limited corporation.
* Association: Manager uses a swipe card to enter XYZ premises.
* Aggregation: Manager has workers who work under him.
* Composition: Manager has the responsibility of ensuring that the project is successful.
* Composition: Manager's salary will be judged based on project success.

## [A/B][IMPORTANT] What is the difference betweeen an Interface and an Abstract class?

* Different in syntax:…
* Different in usage:
  + Interface: for optional behavior, API definition.
  + Abstract class: for some common behavior, all sub-class need to have.

## [A/B] When should I use Abstract Class, when shoud I use Interface?

* Give 3 examples base on answer in previous questions.
  + Example about Abstract Class
  + Example about Interface
  + Example about combination of Abstract class and Interface.

## [A] Give an example of Interface in real life?

## [A] Give an example of Abstract Class in real life?

## [B] Give an example of Abstract Class combine with Interface?

## [A] State the significance of public, private, protected, default modifiers?

## [A] What is Overriding?

## [A] What are different types of inner classes?

## [B] What is a marker Interface?

* Interface doesn’t have anything

## [A] Can we declare an abstract method in a normal class?

* Impossible. Why? (what happen if people invoke the abstract method of the instance which instantiate from this normal class)

## [A] How does the Java default constructor be provided?

* If you don’t define your own constructor.

## [A] Can constructor be inherited?

* Yes, use super(…)

## [A] What are the differences between Constructors and Methods?

* Syntax: constructor doesn’t have return type, name = class name.
* Usage: constructor is used to instantiate an object.

## [A] How are this() and super() used with constructors?

## [A] What are the diffrences between Class Methods and Instance Methods?

* Class method: static method
* Instance method: normal method

# PART 4. SERIALIZATION

## [A] What is serialization and Why serialize?

* Transformation of objects to bytes. Save to file, database or send it over the network.

## [A] How do I serialize an object to a file?

* Implement Serializable. Use ObjectOutputStream.

## [A] Which methods of Serializable interface should I implement?

* None

## [B] How can I customize the serialization process? i.e how can one have a control over the serialization process?

* Externalizable

## [A] What is the common usage of serialization?

* Data transfer

## [B] What is Externalizable interface?

* Custom serialization process

## [A] When you serialize an object, what happens to the object references included in the object?

## [A] What one should take care of while serializing the object?

* Make sure that objects included are also serializable

## [A] What happens to the static fields of a class during serialization?

* Not included.

## [B] What are serialization and externalization?

## [A] What version Id when serializing use for?

* serialVersionUID

## [B] What is reflection?

* To describe code which is able to inspect other code in the same system

## [A] What is transient variable?

* Not include during serialization process

# PART 5. COLLECTIONS

## [A] What is Collection API?

* Set of classes and interface that allows you to store objects in a collection.

## [A] How do you traverse through a collection?

* Loops, Iterator

## [A] What is the List interface?

* Ordered collection according to insertion

## [A] What are the main implementations of the List interface?

* ArrayList, LinkedList, Vector

## [A] What is the Set interface?

* No duplicates

## [A] How to remove the duplication of the Set Collection?

* Nothing to remove.

## [A] How to remove the duplication of the List Collection?

* Copy to Set then copy back to List.

## [A] What are the main implementations of the Set interface?

* HashSet, TreeSet, LinkedHashSet

## [A/B] What are differences between HashSet and TreeSet?

* Hashset = not sync, unordered, allow NULL.
* Treeset = not sync, natural order, faster

## [A] What is a Map?

* Key Value Pair

## [A] What are the main implementations of the Map interface?

* HashMap, TreeMap, Hashtable

## [A] How do you sorting a list of user-defined objects?

* Comparable, Comparator

## [A] What are the differences between the Comparable and Comparator interfaces?

* Comparable, compareTo : Comparator, compare

## [A] What is difference between Arrays and ArrayList ?

* Arrays:FixedSize, primitives, Objects, multidimensional. ArrayList:Growable, Objects only, single dimension

## [A] What are the advantages of ArrayList over arrays?

* Lot of helper methods and dynamic.

## [A] What are differences between ArrayList and Vector?

* Unsynchronized and Synchronized.

## [A] What are differences between HashMap and HashTable?

* Hashmap allows NULL while HashTable does not, and HashMap is not synchronize while Hashtable is synchronized.

## [A] What are differences between LinkedList and ArrayList?

* ArrayList uses index to access elements, faster than LinkedList, it is more lighter because memory address only contains the value. LinkedList is using pointers to next and previous elements

## [A/B] How do you decide when to use HashMap and when to use TreeMap?

* Use HashMap if you don't care of the order of the elements, and use TreeMap if you want your elements to be sorted

## [A] What are differences between HashMap and TreeMap?

* Hashmap is implemented as a hash table, and unordered while TreeMap is ordered

## [A] What is Set interface? TreeSet?

## [A] What is Iterator? How to use it? When you use For loop, when you use Iterator?

## [B] Why are Iterators returned by ArrayList called Fail Fast?

## [A] What are differences between List and Set?

## [A] Do List & Set have common parent?

* Yes. Collection.

## [A] Is Iterator a Class or Interface? What is its use?

* Interface, it is used to traverse the collection.

## [B] What is Collection.synchronizeList()?

* make List synchronized. It is use to make non-thread safe collection to be thread-safe

# PART 6. GARBAGE COLLECTION

## [A] What is the purpose of garbage collection in Java, and when is it used?

* To removed unreferenced Objects.

## [B] How to do GC tunning?

## [B] How many type of memory in JVM?

* Heap Memory - storage for Java Objects
* Non-Heap Memory - stores loaded class and other metadata

## [B] What is young-generation, old-generation memory?

* Young-generation – composed of one eden space and two survivor spaces. The place where all new objects are created.
* Old-generation – contains the objects that are long lived and survived many rounds of Minor GC.

## [B] What is isolated-island in term of GC?

* Describes one or more objects having no reference to them from active parts of an application.  
  When Object A references Object B and Object B references A but no other Objects references both.

## [B] When object will be remove by GC?

* If the object cannot be reached by any live thread.

## [B] Can you force GC to remove an object? Explain in detail.

* No

# PART 7. MULTITHREADING

## [B] Describe synchronization in respect to multithreading

## [A] Explain different way of using thread?

* Extends Thread
* Implements Runnable

## [A] How to implement Java thread?

* Override run() method

## [B] What is thread safe?

* Take time to read this: <https://en.wikipedia.org/wiki/Thread_safety>

## [B] How to implement thread safe?

* Immutable class
* Semaphore
* Synchronization

## [B] What is deadlock?

* Circular waiting for resource.

## [B] What is thread monitor?

* Use some tool like: Jconsole, JvisualVM to monitor Thread.

## [B] What are Critical sections, Mutex (Mutual Exclusion) and Semaphore?

* Critical sections: road intersections. Vehicles move in different directions(different threads) so these intersections need traffic lights(synchronize) to avoid accidents.
* Mutual exclusion: If in your house there is only one restroom, once someone uses it, the others must wait for you to finish your business there before using it.
* Semaphore: You are in a restaurant with limited number of tables. The reception lady will act as a semaphore. You will only be accommodated when there is an empty table. if there's none, you have to wait after others will finish eating.

## [B] Java concurrency framework if you know?

# PART 8. PERFORMANCE TUNING

## [B] How to monitor JVM performance?

* Use Jconsol, jVisualVM, logging CPU+Memory+TCP open connection, GC (overhead?)

## [B] How to do JVM tuning?

* Increase headsize
* Increase perm size
* Configure how long for GC to collect unused variables.

## [B][IMPORTANT] Have you ever faced to performance issues and how to solve them?

* State your own problem and your solution. If you don’t have, skip this question. But, this is IMPORTANT question. Your assessment is very high if you can answer this question.

# PART 9. SPRING

## [A] How many modules in Spring Framework?

## [A] How many modules do you work with?

## [A] What is IOC ? How to inject in spring bean?

* 1. (discuss inversion of control)

## [A] Give me 5 reasons to use spring.

* 1. (keywords: easier, dependencies, modularity, Flexible)

## [B] What is IoC vs DI?

* 1. (discuss difference of IoC and dependency injection)
  2. (also check about the other forms of IoC)

## [B] What is AOP?

* 1. (discuss aspect-oriented programming)
  2. (discuss how to implement)

## [B] What are the different modules in Spring framework?

## [B] What are important ApplicationContext implementations in Spring framework?

## [B] What is BeanFactory interface?

## [B] Do I need to instantiate the container more than once? Why?

## [B] Can you integrate multiple bean configuration xml?

## [B] How many are the different types of AutoProxying, What are they?

## [B] What are the 5 types of AOP Advice?

* 1. (also provide examples of practical uses)

## [B] What is the difference between singleton and prototype bean?

* 1. (also research about bean scopes)

## [B] What kind of exceptions does spring DAO classes throw?

## [B] How many methods to provide configuration metadata to the Spring Container?

* 1. (also know how to implement)

## [B] What is an Aspect?

## [B] What is a Jointpoint v/s Advice v/s Pointcut?

# PART 10. REST – WEB SERVICE

## [A] How many year of restful web service experience do you have?

## [A] How do you use a web service?

* 1. (include discussion about input/output, content format)

## [A] Have you worked with web services before? (should explain SOAP or REST)

## [A] How would you implement a rest web service? (should explain SOAP or REST)

* 1. use the following keywords in your discussion
     1. Spring Web MVC
     2. DispatcherServlet
     3. Presentation Layer
     4. Service Layer
     5. Persistent Layer
     6. Annotations

## [A] How can we parse Json and XML with REST?

* 1. (discuss Jackson and JAXB)

## [A] Can you explain how to implement Restful by using Spring framework?

* 1. (cite the libraries needed, discuss the components you need to create)

## [A] How can implement the restful web service to return the json , xml , you can use any framework?

* 1. (discuss Content-type Negotiation)

<http://theblasfrompas.blogspot.com/2013/10/spring-mvc-rest-content-negotiation.html>

## [A] If I want a json response or xml response, what’s difference between requests?

## [A] Could you give me some steps to create a restful Web Service?

* 1. use the following keywords in your discussion
     1. Spring Web MVC
     2. DispatcherServlet
     3. Presentation Layer
     4. Service Layer
     5. Persistent Layer
     6. Annotations

## [A] What are differences between GET and POST method? How about other method?

* 1. (mention the supported length, data type, resource, cache)

## [A] What is REST and RESTful web services ?

* 1. (discuss in terms of communication, state, cache)

## [B] What is differences between RESTful web services and SOAP web services ?

* 1. (discuss in terms of protocol, uses, format, performance, cache)

## [D] What is Restlet framework ?

## [A] What is Resource in REST framework ?

## [D] Can you use Restlet without any web-container ?

## [D] What is difference between Restlets and Jersey ?

## [D] What is RESTEasy ?

## [A] What are the tools used for creating RESTFull web services ?

## [A] How to display custom error pages using RestFull web services ?

## [A] Which HTTP methods are supported by RestFull web services ?

## [B] What is difference between top-down and bottom-up approach of developing web services ?

## [B] What happens if RestFull resources are accessed by multiple clients ? do you need to make it thread-safe?

## [B] What differences between synchronous vs asynchronous web services?

## [A] What is WSDL?

(keywords: web service, operations, parameters)

## [B] What is JAX-WS and JAX-RS?

## [A] What are the underlying protocol over which you can send SOAP request?

## [A] What are the different formats supported by REST API?

## [D] What is use of JAX-RPC?

## [B] How many types of JAXM messaging models?

## [D] Explain WSDL types?

## [B] REST vs SOAP?

## [D] What is UDDI?

## [A] @Service, @Controller, @Repository, @Component?

## [B] What kind of design pattern used in Spring?

## [A] What is used for parsing XML and JSON in Spring?

## [A] What is JAXB?

## [A] What are differences between DOM parser and SAX parser?

## [A] What Maven used for?

* 1. (keywords: build automation)

## [A] What is POM file?

* 1. (keywords: configurations, dependencies)

## [A] Do you use maven? Multi modules or single module?

# PART 11. DATABASE – JDBC – HIBERNATE

## [A] What is outer join, left join, inner join?

## [B] What is index? When should you use index?

## [B] What is transaction? Why do you use transaction?

## [C] Explain the new Features of JDBC 2.0 Core API?

## [A] What is JPA framework?

## [A] Mapping one-to-may , many-to-one. where to use this annotation

## [A] What should you do when mapping list?

## [B] Differentiate local and JTA transaction managing.

## [A] Which provider do you you use?

# PART 12. DESIGN PATTERN

## [A] What kind of design pattern you know?

* Singleton
* Factory
* Visitor
* Observer
* Façade
* Template
* Service Locator
* Front Controller
* IoC
* MVC
* Iterator
* Comparable vs Comparator

## [A] What façade pattern, factory pattern, singleton, observer, IoC, Dependency Injection, AOP pattern? When you use them?

## [B] What is service locator pattern?

## [A] What is an Iterator?

## [A] Dynamic proxy pattern?

## [A] Builder pattern?

# PART 14. DESIGN

## [A] Which tool do you use to draw diagram? Which kind of diagram do you use?

* Sequence diagram: Visio, Rational Rose.
* Activity diagram: Visio
* System Integration diagram: Visio, Power Point, Excel, Word

# PART 15. XML

## [A] What is XML Schema?

* 1. keyword: structure

## [A] What is JAXP, JAXB?

* 1. also discuss differences

## [A] If we have convert objects from Java to XML, what the framework you will use?

# PART 16. TESTING

## [A] How to write a test case?

## [A] Have you used unit testing before?

## [A] When to use @Before and when to use @After in unit tests?

## [A] What test framework do you use in your project?

## [A] What junit life circle?

* 1. (also mention the annotations)

## [A] How to implement the method test?

## [B] Do you know some automation test tool ?

## [A] How to implement the test the restful web service ?

## [A] Could you give me the life-cycle of unit test?

## [A] Do you have experience about Mockito & Power Mockito?

<http://www.vogella.com/tutorials/Mockito/article.html>

<http://www.rapidprogramming.com/questions-answers/difference-between-mockito-and-powermock-mockito-vs-powermock-1504>

## [A] How to test web service with the data will be updated everyday?

* 1. (discuss the test case for this)

## [A] Can you give me an example when we use before and after?

## [A] If we have to write a test case to test the method convert Java object to XML (ex: Employee object) using JAXB, how do we implement/arrange the test case?

## [A] If we have to write a test case to test a restful web service, how do you write in java?

# PART 17. INTERNALIZATION

## How to store localization in db. There's a form saving movie titles in multi-languages. How to organize saving in db

# PART 18. SECURITY (JAVA SECURITY)

## [A] Tell your experience about security in pragramming

## [A] What is JAAS?

## [A] How do you organize authorization vs authentication?

## [A] How to customize user information object after validating successfully?

# PART 19. OSGi

## Have you ever worked with GIT and OSGi

# PART 19. CONCLUSION

## Do you have any question for us? (IMPORTANT, LAST IMPRESSION)

* What technologies will be used in your project?
* Do I need to research about some technologies?
* Which kind of caching methodology that you prefer?
* What can I achieve while working with your project? (need to confidence)