# Java Questions for Deloitte Full Stack Java Developer

1. How does spring framework function?
   1. Spring’s core logic is dependency injection. Dependency injection is a programming pattern that allows developers to build more decoupled architectures. Dependency injection means that Spring understands the different Java annotations that a developer puts on top of classes. Spring knows that the developer wants to create an instance of a class and that Spring should manage it. Spring also understands the dependency and makes sure that all instances created have properly populated dependencies.
   2. A web application (layered architecture) commonly includes three layers:
      1. Presentation/view layer (UI) - This is the outermost layer which handles the presentation of content and interaction with the user.
      2. Business logic layer - The central layer that deals with the logic of a program.
      3. Data access layer - The deep layer that deals with data retrieval from sources.
2. Java Basics, Hibernate Framework, Use of lambda expression in Java
   1. Hibernate Framework: Object Relational Mapping (ORM) tool for Java to map object to RDBMS.
   2. Lambda: Anonymous functions;
      1. Expressions are limited. They have to immediately return a value, and they cannot contain variables, assignments or statements
      2. Curly blocks
      3. Lambda expressions can be stored in *functional interface or SAM Interfaces (single abstract method interface)*
3. What was the most challenging technical project I worked on and why
4. What is functional interface in Java
5. Java, Angular, Database and a few algorithm questions.
6. Junit code coverage
7. How to create end point in spring boot polymorphism scenerio based questions
8. J2SE, Spring Frameworking, Data Structure like Sorting and SQL
9. A login page is validating user id and password by getting details from the database. But as the database has billion of rows and it is getting a long time to fetch the data from the database hence throwing errors. What should be the short-term and long-term solutions for this issue?
10. *Can we use the static variables inside static methods?*
    1. A local variable can not be static :
    2. You cannot use static variable in the static method because JVM tries to allocate space for the static variable which he already allocated to the method inside which the static variable is declared.
11. *Can we use the final keyword with the constructor?*
    1. *A final method cannot be over-riden at all, while overriding is a very useful feature that a constructor has.*
    2. *We can not override constructor of any class. And final method is used to prevent a method from overriding.*
12. *Difference between List and arraylist?*
    1. *List is an interface whereas arraylist is implementation of List interface by array.*
13. *What are the different bean scopes in Spring?*
    1. *Singleton*
    2. *Prototype*
    3. *Proxy*
    4. *Session*
    5. *Request*
14. *Difference between stringbuffer and stringbuilder?*
    1. *Stringbuffer operations are threadsafe and synchronized whereas stringbuilder is the opposite.*
    2. *Stringbuffer is used when multiple threads are working on same string where as stringbuilder is in single-threaded environment..*
15. *Difference between Post and Put in springrest.*
    1. *Post is for creating a new resource where as put is for updating the resource.*
16. *How to get the second highest salary from the table?*
    1. *??*
17. *Difference between Inner Join and Outer join.*
    1. *An INNER JOIN and an OUTER JOIN is that the inner join will keep only the information from both tables that's related to each other (in the resulting table). An Outer Join, on the other hand, will also keep information that is not related to the other table in the resulting table*
18. *Code to sort an array without using any predefined class and methods.*
19. *Explain MVC design pattern with diagram.*
20. *Difference between Application Server and Web Server*
    1. *A web server accepts and fulfills requests from clients for static content (i.e., HTML pages, files, images, and videos) from a website. Web servers handle HTTP requests and responses only. An application server exposes business logic to the clients, which generates dynamic content.*
21. About binary search
22. First round was an online test consisting mcqs related to Core Java, Java EE, Multi threading, Hibernate and so on.
23. Core Java, OOPS, Exception handling, Finding a palindrome etc.
24. How to reverse linked list
25. Online test with MCQ and code implementation 2. Technical F2F introduce, explain all projects test(int a){};test(byte a); test(1); which will get call? Why ? Expalin OOPS concepts, iterate hashmap diff b/w soap & rest, why rest over soap
26. Basic Java, spring MVC, Hibernate, SQL
27. Basic questions about SSH(Struts 2, Spring, Hibernate)
28. Describe architecture of your project, which version of java you use, which functionalities of the particular version are used. Difference between final, finally and finalize, how is exception handling is used in your project, what is throwable.
    1. Final is a keyword and used if you dont want to change its value. Finally block for executing code irrespective of other try n carch and finalize for checking garbage collections and release dara.
    2. We are using exception handling chains and we also write custom exceptions in order to make code more exception readable.
29. What is your Project Architecture , followed by question based on that like , which version of Spring your are currently using?
30. Difference between Component-Scan and @Component?
    1. Component-scan: can be added in XML configuration to tell spring to look at this package for components in it. Basically, it is the first step for annotation. @componet: by adding @componet we are telling spring to create a bean.
31. Which Data Structures(Collection) you have used in your project?
32. Difference between hashmap and Concurrent hashmap?
33. What are the real time problems with hashmap?
34. Haven't you seen concurrent modification exception ever?
35. Difference between Rest and Soap?
36. Did you use microservices?
37. Multithread concepts, comparable and comparator differences, bean lifecycle, how to make a class immutable, where do you define servlet, which is the starting point for a spring framework. Explain mvc framework.hashcode.equals functionality. How hashmap stores work internally.
38. In first technical round string reversing, sql queries, spring mvc working, immutable classes, about project and i had cleared the round. Then they had called my for 2nd round. In that they had asked me about my project, hibernate, spring, cache. I had answered all the questions. But hibernate was not used in our project. Then, he had asked me about hibernate configuration file. How you will write config file and what are all the things will you write over there. I had not cleared the 2nd round as our previous project is not so suitable to the deloitte projects. But, if you prepare well, you can cleared the process as the interview questions were very easy.
39. Caching, multithreading, hashmap working, rest api, string
40. What is ajax
41. What are the default modifiers in interface
42. *What is dependency injection.*
43. Write a code of cache mechanism in core java without any framework support.
44. Reverse a string. Print a pyramid pattern from a string. Collections.
45. What is inheritance..! 2. What is interface..!
46. Java collections framework, java constructors, abstract classes and interfaces etc.
47. Multithreading and concurrent modelling,
48. How do you declare a bean in spring, how do you write hibernate configuration file, why do you write an sql procedure and what is a cursor in sql and why do you need it.
49. Write a method to remove even numbers from list without using any other collection.
    1. Use iterator()
50. Print a star pattern
51. Thread vs runnable
52. Immutable class
53. How concurrent hashmap works?
54. How to apply security on restful web services
55. If application has more write operations and less read operations. What will you use, hibernate or jdbc? Why?
56. Why wait and notify should be called from synchronized block?
57. How to implement second level cache in hibernate?
58. If your application has multiple databases, how many changes you have to made in hibernate?
59. If soap is more secure than rest then why many banking application still uses rest?
60. What is polymorphism ? Give a real life example.
    1. Polymorphism - when one task is performed by different ways i.e. Known as polymorphism.
    2. Real life example for polymorphism (object oriented) explained better :
       1. Father : son, go and get red label.
          1. Son : 750 ml or 1 l ? ..
       2. Mother : son, go and get red label.
          1. Son : 500g or 1 kg ?
61. *What is the difference between arraylist and vector?*
    1. Arraylist is not synchronized while the vector is synchronized
    2. Arraylist is fast because it is synchronized , on the other hand vector is slow
    3. Arraylist uses iterator to traverse while vector uses enumeration
    4. Arraylist is not legacy and is introduced in jdk1.2 while vector is legacy.
62. *What is the difference between hashmap and hashtable?*
    1. Hashmap is not-synchronized while hashtable is synchronized
    2. Hashmap is fast and hashtable is slow because it is synchronized
    3. Hashmap is traversed through iterator while hashtable through enumerator
    4. Hashmap is fail-fast while hashtable is fail-safe(weakly consistent)
    5. Hashmap can have one null key while hashtable cannot have null key
63. *Types of ioc container*
    1. Two types of ioc container is there .
       1. Beanfactory(old -lazy loading)
       2. Applicationcontext (new- eager loading)
64. Why we have 2 representation of string in java?
65. Execution sequence of static blocks ?
66. Hash code and hashmap?
67. Difference between abstract class and interface?
68. What is the business need of run time polymorphism in your project?
69. What is dialect in hibernate?
70. Genericservlet vs httpservlet
71. Given multiple catch blocks, which catch block would be executed. How to display error in jsp page using jsp tags
72. Define encapsulation and abstraction. In what scenarios would you use them?
73. Explain the internal working of hashmap.
74. Explain the design patterns. Explain singleton pattern.
75. Questions about collection framework.
76. *Difference between ddl and dml*
    1. Ddl (data definition language) is used to specify the database schema database structure. On the other hand, dml (data manipulation language) is used to access, modify or retrieve the data from the database.
    2. Ddl commands - create, alter, rename, drop
    3. Dml commands - insert, delete, select,update
77. *Steps for jdbc connection*
    1. There are 5 steps for jdbc connection
       1. Register the driver class
       2. Creating connection
       3. Creating statement
       4. Execution queries
       5. Closing the connection
78. *How to sort objects based on one of it's variables*
    1. Using comparator interface of collection.
79. What are generics
80. Different type of for loop
    1. Simple: for int i …
    2. Complex: for obj : coll, labeled
81. Servlet life cycle
82. Internal working if hashmap
83. Oops concepts, multithreading, garbage collection in java, hibernate mapping and sql server related questions on joins, cursor and trigger.
84. They asked deep spring mvc interactions with hibernate on application servers. Transaction management systems. Spring core container classes. Etc
85. How can you access private functions using reflection
    1. //file a.java  
       Public class a {

Private void message(){system.out.println("hello java"); }

}

//file methodcall.java

Import java.lang.reflect.method;

Public class methodcall{

Public static void main(string[] args)throws exception{

Class c = class.forname("a");

Object o= c.newinstance();

Method m =c.getdeclaredmethod("message", null);

M.setaccessible(true);

M.invoke(o, null);

}

}

1. Write a program to check if a string is palindrome or not.
2. Difference between == and equals method
3. Write a program to write fibonacci series.
4. What is polymorphism? What is runtime and compile time polymorphism.
5. What is concurrent modification exception? How to prevent it.
6. *How do you create jvm ?*
   1. *Its packaged in jdk*
7. Security, unit test, ui design, exception handling as a java web developer
8. Can muliple inheritance possible in java using abstract class
9. What is o/p of below code

Interface a{

Int i=10}

Interface b{

Int i=5}

Class c a,b{

Sop(i)

}

1. Write code for jdbc connection in java
2. How hashmap internally work
3. Write code to retrieve elements from hashtable
4. Diff between vector and arraylist
5. What is synchronization
6. What ibm websphere?Is it application server or web server?Why you use it?
7. Diff between application server and web server
8. What is o/p of below code

Class a

{

Int i=10}

Class b

{

Int i=5}

Public class c extends a,b{

Psvm{

Sop(i)

}

}

* 1. It will produce compile time error as a class can only extend one class. It can implement many interface it will produce compile time error as a class can only extend one class.

1. Whats is multiple inheritance in java
2. *Is abstraction and abstract class is same ?*
   1. No abstraction is the mechanism which can be achieved from abstract class or from interface.
3. *What is diff between procedural oriented language and object oriented language*
4. They will check your logical understanding by asking some algo e.g. Sorting, searching algos and difficulty will depend on your exp. In my case array was given and interviewer asked me to do sorting with any algo.
5. *Abstraction and diff between abstract class and interface.*
6. Abstract classes provide very strong is-a relationship and interfaces can be seen as capabilities. When one need to give abstraction with some pre-defined functionalities we can should use abstract classes else interfaces can be used. So when there are similar kinds like we use abstract classes and if they also require different kind of capabilities(functionalities) we can provide it by interfaces. And sooo on.... Explained them java specific rules of creating abstract classes and interfaces
7. *I was asked about singleton, factory and dao design pattern as they were on my cv.*
8. *Javascript basic knowledge like diff between == and ===..*
9. Structure of a web application
10. Inner implementation of hash mapuse of synchronized keyword and various combination of object along with minor changes in the constructor.