1. What is Immutable object and how to make object immutable?
2. Are Immutable objects thread-safe?
3. What is WeakHashmap?
4. Time complexity in Hashmap?
5. Have you worked on garbage collector? How does it work?
6. What is the difference between Hashmap vs Hashtable?
7. What is synchronization?
8. In Hashtable , what is synchronized, methods or object?
9. How you will scan the custom Annotations?
10. Which modules have you used in spring?
11. How you will get JSON output from rest services?
12. Are there any other ways to implement security except SSL?
13. How to delete record from a table whose primary key is foreign key in other table?
14. Describe the steps that you need to take care of for implementing Singleton
15. Which is the best project that you’ve worked on and why?
16. Which is the worst project that you’ve worked on and why?
17. Can you tell me about the worst bug that you’ve fixed and what was the fix?
18. Have you worked on XML parsing?
19. What is the difference between SAX and DOM Parser?
20. Can you give an instance where there was a synchronization issue in your project and you resolved it?
21. What happens in Left and right joins?
22. Can you explain about the development methodologies that you have worked on? (Kanban, Agile Scrum or Waterfall)
23. Do you have any UI experience?
24. Was your application a Single Page application?
25. Have you worked on Akka framework?
26. Have you worked on Apache Solr?
27. Have you worked on Drupal?
28. Have you worked on GIT?
29. What is CloudFoundry (if applicable)?
30. What is MongoDB (if applicable)?
31. What tools have you used for Performance capturing?
32. How do you debug network issues?
33. What is the difference between TCP and UDP?
34. Which features of Java 7 have you used?
35. Why did you choose a particular framework in a project?
36. Which healthcare workflows are you aware of?
37. Which building tools have you used?
38. Which tool have you used for versioning?
39. Which code review tools used?
40. Which web services have you used? SOAP or REST?
41. What areas do you want to learn more or explore?
42. Which web sites do you use to learn or explore new technologies / areas?
43. Highlights of your resume?
44. EDI to HL7, what are the challenges and how did you overcome it?
45. EDI to Object what framework are you using?
46. Was there an existing framework to tie up the entire workflow together?
47. Is this a batch process or web process?
48. What is message input and output?
49. Convert to JSON what framework?
50. Is the java object user defined or open source based?
51. Did you have to worry about performance?
52. What do you mean by chunks?
53. Did you use annotations for data binding?
54. Data mining was it a lot of sql or code?
55. Is this all done in application side or is there sql involved?
56. Is the SQl mainly just inserting updating or are there complicated sql statements?
57. What SQL DB did you use?
58. How familiar are you with converting a project to hibernate?
59. Most of the projects you worked on already had hibernate?
60. What's the main performance issue with Hibernate and what are the tips and tricks to fix it?
61. What version of Spring are you using?
62. What interests you the most in work? What kind of problems do you like solving like algorithmic, DB optimization, performance, instant gratification?
63. What did you do for performance?
64. Did you use hash for like queries or equal queries?
65. What projects have you worked on in the past year?
66. Difference between Inheritance & Composition? Examples of each? When would you use one over the other?
67. Different ways to create Thread in Java?
68. Have you used ExecutorFramework?
69. Difference between HashMap & HashTable?
70. Difference between HashSet & TreeSet?
71. Time complexity of adding and deleting from both?
72. Difference between comparable & comparator?
73. When will I use Comparable & when comparator?
74. Difference between ArrayList & LinkedList?
75. Which Java Design Patterns have you used?
76. Write a Singleton Class?
77. What kind of automation framework have you worked on?
78. Do you have experience with CI?
79. Tell me something about REST Assured?
80. What benefits does it have?
81. Have you used any Web based REST testing frameworks?
82. What Spring version did you use?
83. Do you have experience with Hibernate?
84. Did you use Spring JDBC in project?
85. Do you have node.js experience?
86. How comfortable are you on Java vs Javascript?
87. What is Mockito? Where did you use it?
88. How does SonarCube define 80% coverage?
89. Did you create a new project using Maven or was it already configured?
90. Which code repository did you use?
91. How were different GIT branches getting tested on environments? Did they spin a new box for each branch?
92. What is difference between Inheritance vs Composition? When will you prefer one over the other?
93. Is Java pass by valur or pass by reference? Substantiate with an example?
94. Different ways to create ThreadPool in Java?
95. Difference between Enumeration and Iterator?
96. Difference between HashSet and LinkedHashSet?
97. Difference between List and Set?
98. What is time complexity of insertion, search and deletion in ArrayList and LinkedList?
99. Did you write any server side code in Scala in Play framework?
100. What Spring version was project using?
101. Did you use Spring MVC?
102. Which UI framework did you use for Single Page Applications?
103. Did you use AngularJS?
104. Did you use Spring JDBC or Hibernate?
105. How was the build process?
106. Which version control did you use?
107. Can you explain the workflow from code checkin to deployment?
108. How was code coverage measured?
109. Do we need to implement all methods when implementing an Interface using Abstract Class?
110. What is deadlock? Example?
111. How would you create your own HashMap? Internal Working of HashMap?
112. Difference between Iterator and ListIterator?
113. Difference between Complete Binary Tree and Perfect Binary Tree?
114. What deployment software did you use?
115. What components of Spring are you using?
116. Have you built projects from scratch using Maven?
117. How were you processing HL7 messages?
118. Have you worked on batch processing?
119. Serialization and Deserialization
120. Difference between Encapsulation and Abstraction
121. Explain any algorithm to check if a number is prime number or not
122. Sort 1000GB of data locally – (example on Patient Name)
123. Logic to swap two numbers without using temporary variable
124. Find sum of digits of a number using recursion
125. What is yield in multithreading
126. Find the equilibrium point of an array
127. Find the missing number from an array containing 1 to N values
128. What are your core skill sets?
129. Have you worked on multi-threading?
130. What's the difference between sleep and wait methods?
131. If you have n number of resources and x number of threads, how do you prevent deadlocking?
132. Arraylist and hashmap - when to use what?
133. What's the difference between arraylist and vector?
134. If you have 100 GB data on prod and local machine has 50GB space, how do you manage to get that data on local?
135. What's the K merge sort algorithm?
136. An array has numbers 5 6 27 16 32 - how to find third largest number?
137. How to determine if a string is a palindrome?
138. How to find out if a string is an anagram?
139. Have you worked on big data?
140. What's the Mapreduce algorithm?
141. How to swap two integers without temp variable?
142. How can we prevent threads from having deadlock?
143. Difference between implements, runnable & extends thread?
144. Difference between factory pattern and abstract factory pattern?
145. What is unsupported operation exception?
146. What is Blocking Queue?
147. How will you download a file using Rest webservice
148. @path annotation significance
149. How will you prevent SQL injection
150. Examples of Rest webservice Implementation
151. How will you sort hugeeee set of data . Which algorithm? Map Reduce?

Algorithms

1. Given a string “HELLO”. Write logic to avoid repetition of characters in string?

* Using set
* Without using any native Java function

1. You have an array {1, 3, 4, 5, 7}. How to find nth-largest element (e.g. second-largest, third-largest, etc.)?
2. How do you find out if a number is prime or not?
3. How would you reverse a string without using reverse method?

Coding exercise

1. Palindrome - can you write a method and also JUNIT test to check a palindrome?

Do not use the reverse method?

1. Write a program which checks if an input can be made a palindrome? It must just return true or false and not create a palindrome itself. E.g. tnnii can be made a plaindrome
2. Count frequency of words in a file.
   1. List the words in order of their frequency with the highest first.
3. From a list of numbers find out which two pair form the desired sum? Example Input : list = {1,2,3,4,5,7,9}, sum = {8} Output: {3,5}
4. Singleton Pattern – Lazy loading and thread safe
5. Factory Pattern
6. Given two sorted arrays merge the same with no duplicates and present the resulting array?
7. Given an array find the maximum sum of consequent elements
8. Write a method to check if two strings are anagram? Who - how

Apart from the above questions, they may ask you anything on the technologies, tech spec and architecture of projects or roles/responsibilities that you’ve mentioned in the CV.

UI

1. Why did you choose AngularJS?
2. Which other frameworks do you know?
3. Inheritance in javascript and types?
4. What is prototype, function, prototype chain?
5. What is closure?
6. Can you pass parameter in closure?
7. Private methods?
8. Encapsulation using prototype?
9. Scoping in javascript?
10. Hoisting
11. Block level scoping & function level scoping?
12. Which scoping does Ecma6 script provide?
13. What do you know about Ecmascript6?
14. How will you define Ecmascript? (It’s a javascript Standard)
15. Biggest features introduced in Ecmascript 6?
16. What is the position property in CSS? What are the different types and what does each do?
17. Basic datatypes in Javascript?
18. Is undefined a datatype?
19. How would you implement a function inside a for loop and print all the values in the array?
20. CSS specificity?
21. Primitive types in javascript?
22. Difference between call and apply?

**4/08/2016**

1. Explain Event bubbling and how to prevent it.
2. Responsive design.

Coding exercise

Create a unique character string from a given string? What data structure will you use?

Implementation of Inheritance.

**4/08/2016**

Creating a unique string from a given string using dictionary concept (using object)

Solution:

function getUniqueStr( str ) {

var chars = {},

uniqueStr = '';

for (var i = 0; i < str.length; ++i) {

if (!(str [i] in chars)) {

chars[str [i]] = 1;

uniqueStr += str [i];

}

}

return uniqueStr;

}

###UI questions asked by Irien:

1. Tell me about your technical experience

2. What is CSS Box Model?

3. What is an Event Loop?

4. What are promises in Javascript?

5. What are the various states of a promise?

6. What are controllers in AngularJS?

7. What are services in AngularJS?

8. Are Services singleton in AngularJS?

9. Difference between service methods and factory methods in AngularJS?

10. What are directives in AngularJS and what are their types?

11. What is scope in AngularJS?

12. Write a sample code with controllers and services to show a sample AngularJS app running with $scope and $rootScope?

13. Find array [2,2] in list1=[4,2,3,2] and list2=[2,3,2]