1. Enable HTTPS for localhost in NodeJs
2. Types of API
   1. Private API
   2. Public API
   3. Partner API

API Architectural Style – SOA , Microservices

1. Twitter API with Java
2. Hygraph – self blog creation
3. Elastic Search – Java
4. Defensive programming
5. **How to calculate factorial using recursion in Java.** ([solution](http://www.java67.com/2015/09/factorial-in-java-using-recursion-and-loop.html?ref=faun))
6. How to Print Fibonacci Series in Java using Recursion. ([solution](https://www.java67.com/2016/05/fibonacci-series-in-java-using-recursion.html?ref=faun))
7. **How to reverse String in Java using Recursion.** ([solution](https://www.java67.com/2012/12/how-to-reverse-string-in-java-stringbuffer-stringbuilder.html?ref=faun))
8. Write a countDown(int number) method in Java using Recursion which prints countdown till zero to console, like count(3) should print 3 2 1 0
9. **How to reverse Linked List using Recursion**. ([solution](https://javarevisited.blogspot.com/2017/03/how-to-reverse-linked-list-in-java-using-iteration-and-recursion.html#axzz6e8hmwujv?ref=faun))
10. How to print digitsToWords(int number) for example digitToWords(321) should print three or two ones? (solution)
11. **How to reverse a number using Recursion in Java?** ([solution](https://javarevisited.blogspot.com/2012/04/java-program-to-reverse-number-example.html?ref=faun))
12. How to calculate the sum of arithmetic series from 1 to N? (solution)
13. **How to calculate the power of a number like power(int number, int power) like power(2, 3) should return 8?** ([solution](https://www.java67.com/2018/05/how-to-implement-power-function-in-java.html?ref=faun))
14. How to calculate Greatest Common Division GCD using Euclid’s algorithm ([solution](http://www.java67.com/2012/08/java-program-to-find-gcd-of-two-numbers.html?ref=faun))
15. **How to print nodes of trees in the pre-order traversal using recursion?** ([solution](https://javarevisited.blogspot.com/2016/07/binary-tree-preorder-traversal-in-java-using-recursion-iteration-example.html#axzz6dXsEfLvJ?ref=faun))
16. Write a Java program to convert Decimal to binary using recursion. ([solution](http://www.java67.com/2014/03/decimal-to-binary-conversion-in-java.html?ref=faun))
17. **How to code a post-order traversal algorithm using recursion?** ([solution](https://javarevisited.blogspot.com/2016/10/post-order-binary-tree-traversal-in-java-iteration-recursion.html#axzz5ghwj3kuU?ref=faun))
18. How to implement inorder traversal of binary tree using recursion? ([solution](https://javarevisited.blogspot.com/2016/08/inorder-traversal-of-binary-tree-in-java-recursion-iteration-example.html?ref=faun))
19. **How to count the number of leaf nodes in a given binary tree?**([solution](https://javarevisited.blogspot.com/2016/12/how-to-count-number-of-leaf-nodes-in-java-recursive-iterative-algorithm.html#axzz6e8hmwujv?ref=faun))
20. How to implement a binary search algorithm using recursion? ([solution](http://www.java67.com/2016/05/java-program-to-perform-binary-search-without-recursion.html?ref=faun))
21. **How to find all permutations of a given String in Java?** ([solution](https://javarevisited.blogspot.com/2015/08/how-to-find-all-permutations-of-string-java-example.html#axzz5N6yGHupv?ref=faun))
22. How to calculate the sum of digits using recursion in Java? ([solution](http://www.java67.com/2014/06/write-program-to-find-sum-of-digits-in-java.html?ref=faun))
23. **How to solve the Tower of Hanoi problem using recursion?**(solution)
24. How to implement bubble sort algorithms using recursion? ([solution](https://www.java67.com/2012/12/bubble-sort-in-java-program-to-sort-integer-array-example.html?ref=faun))
25. **How to implement Quick sort algorithm using recursion? (**[**solution**](https://www.java67.com/2014/07/quicksort-algorithm-in-java-in-place-example.html)**)**
26. How to implement merge sort algorithm using recursion ([solution](https://javarevisited.blogspot.com/2022/01/how-to-merge-sorted-array-in-java.html))
27. **Solve the Nine Queen problem in chess using Recursion (solution)**
28. Solve Regular Expression matching problem using recursion where you have given an input string s and a pattern p, implement regular expression matching with support for '.' and '\*' where:

* '.' Matches any single character.​​​​
* '\*' Matches zero or more of the preceding element.

The matching should cover the **entire** input string (not partial).

**25. How to merge two sorted linked list using Recursion (solution)**

26. How to calculate power of three using recursion (solution)

27. How to remove a given element form a linked list using recursion (solution)

SurveyJs + React to build internationlizaion

React : How to build dropdown components

Spring boot 3+ Spring framework 6

Spring security : upgrading Web securityConfigureAdapter

Volatile and Atomic Variable

Scanning Java annotation at runtime.