



from  **Campus** **TO**  
**CORPORATE** 

SINCE  
2005

**TRAINING & PLACEMENT  
IN  
YOUR COLLEGE CAMPUS**

100%  
JOB

**TECHNOLOGIES WE COVER**

- > Logical Coding and Data structures with algorithms
- > Angular and Javascript
- > Salesforce Development

**FROM  
11 TH FEB**

**TIME  
2-5 PM**

**DURATION  
2 MONTHS**

**VENUE AT  
YOUR CAMPUS**

**TRAINER**  
**Mr. Ramesh Reddy, CEO & Founder, Lara Technologies**

8, Outer Ring Rd, BTM 2nd Stage, Kuvempu Nagar, Stage 2,  
 BTM Layout, Bengaluru, Karnataka 560076

# INDEX

## Logical Coding & Data Structures

## PAGE NO.

01. Display Patterns.....	01-16
02. Number Systems.....	17-19
03. Arrays and Sorting Algorithms.....	20-22
04. Recursive Algorithm.....	23-23
05. String and File Handling.....	24-27
06. Data Structures.....	28-28

## Angular & Java Script

01. Getting Started.....	29-29
02. Type Script Fundamentals.....	30-30
03. Angular Fundamentals.....	31-31
04. Displaying Data and Handling Events.....	32-32
05. Displaying Data and Handling Events.....	33-33
06. Directives.....	34-34
07. Template Driven Forms.....	35-35
08. Reactive Forms.....	36-36
09. Consuming HTTP Services.....	37-37
10. Routing and Navigation.....	38-38

## Sales Force Development

01. Apex Programming Language.....	39-39
02. SQL.....	40-40
03. Sales Force Development.....	41-41



**Logical Coding**

**&**

**Data Structures**

# 1. DISPLAY PATTERNS

01  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*

02  
1 1 1 1 1  
2 2 2 2 2  
3 3 3 3 3  
4 4 4 4 4  
5 5 5 5 5

03  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5

04  
AAAAA  
BBBBB  
CCCCC  
DDDDD  
EEEEEE

05  
A B C D E  
A B C D E  
A B C D E  
A B C D E  
A B C D E

06  
5 5 5 5 5  
4 4 4 4 4  
3 3 3 3 3  
2 2 2 2 2  
1 1 1 1 1

07  
5 4 3 2 1  
5 4 3 2 1  
5 4 3 2 1  
5 4 3 2 1  
5 4 3 2 1

08  
E EEEEE  
DDDDD  
CCCCC  
BBBBB  
AAAAA

09  
EDCBA  
EDCBA  
EDCBA  
EDCBA  
EDCBA

# 1. DISPLAY PATTERNS

10.

```
*  
* *  
* * *  
* * * *  
* * * * *
```

11.

```
1  
2 2  
3 3 3  
4 4 4 4  
5 5 5 5 5
```

12.

```
1  
1 2  
1 2 3  
1 2 3 4  
1 2 3 4 5
```

13.

```
A  
B B  
C CC  
D DDD  
E EEEE
```

14.

```
A  
A B  
A B C  
A B C D  
A B D C E
```

15.

```
* * * * *  
* * * *  
* * *  
* *  
*
```

# 1. DISPLAY PATTERNS

16  
1 1 1 1 1  
2 2 2 2  
3 3 3  
4 4  
5

17  
1 2 3 4 5  
1 2 3 4  
1 2 3  
1 2  
1

18  
A A A A A  
B B B B  
C C C  
D D  
E

19  
A B C D E  
A B C D  
A B C  
A B  
A

20  
5 5 5 5 5  
4 4 4 4  
3 3 3  
2 2  
1

21  
5 4 3 2 1  
5 4 3 2  
5 4 3  
5 4  
5

22  
E E E E E  
D D D D  
C C C  
B B  
A

23  
E D C B A  
E D C B  
E D C  
E D  
E

# 1. DISPLAY PATTERNS

24  
\*  
\* \*  
\* \* \*  
\* \* \* \*  
\* \* \* \* \*

25  
1  
2 2  
3 3 3  
4 4 4 4  
5 5 5 5 5

26  
1  
1 2  
1 2 3  
1 2 3 4  
1 2 3 4 5

27  
A  
B B  
C C C  
D D D D  
E E E E E

28  
A  
A B  
A B C  
A B C D  
A B C D E

29  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*

30  
5 5 5 5 5  
4 4 4 4  
3 3 3  
2 2  
1

31  
1 2 3 4 5  
1 2 3 4  
1 2 3  
1 2  
1

32  
E E E E E  
D D D D  
C C C  
B B  
A

# 1. DISPLAY PATTERNS

33  
A B C D E  
A B C D  
A B C  
A B  
A

34  
\*  
\* \* \*  
\* \* \* \* \*  
\* \* \* \* \* \* \*  
\* \* \* \* \* \* \* \* \*

35  
1  
2 2 2  
3 3 3 3 3  
4 4 4 4 4 4 4  
5 5 5 5 5 5 5 5

36  
1  
3 3 3  
5 5 5 5 5  
7 7 7 7 7 7 7  
9 9 9 9 9 9 9 9

37  
A  
B B B  
C C C C C  
D D D D D D D  
E E E E E E E E E

38  
A  
C C C  
E E E E E  
G G G G G G G G  
I I I I I I I I I



# 1. DISPLAY PATTERNS

39

```
  1
 1 2 3
1 2 3 4 5
1 2 3 4 5 6 7
1 2 3 4 5 6 7 8 9
```

40

```
  1
 3 2 1
 5 4 3 2 1
 7 6 5 4 3 2 1
 9 8 7 6 5 4 3 2 1
```

41

```
  A
  A B C
 A B C D E
A B C D E F G
A B C D E F G H I
```

42

```
  A
  C B A
  E D C B A
  G F E D C B A
 I H G F E D C B A
```

43

```
  0
 1 0 1
 2 1 0 1 2
 3 2 1 0 1 2 3
 4 3 2 1 0 1 2 3 4
```

44

```
  A
  B A B
  C B A B C
  D C B A B C D
  E D C B A B C D E
```

# 1. DISPLAY PATTERNS

45

```
  1
 1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

46

```
  A
  A B A
 A B C A B
A B C D A B C
A B C D E A B C D
```

47

```
* * * * *
* * * *
* * *
*
```

48

```
4 4 4 4 4 4 4
3 3 3 3 3
2 2 2
1
```

49

```
7 7 7 7 7 7 7
5 5 5 5 5
3 3 3
1
```

50

```
1 2 3 4 5 6 7
1 2 3 4 5
1 2 3
1
```

# 1. DISPLAY PATTERNS

51  
D DDDDDDD  
C CCCCC  
B BB  
A

52  
G GGGGGGG  
E EEEEE  
C CC  
A

53  
A B C D E F G  
A B C D E  
A B C  
A

54  
\*  
\* \*  
\* \* \*  
\* \* \* \*  
\* \* \*  
\* \*  
\*

55  
3  
3 2  
3 2 1  
3 2 1 0  
3 2 1  
3 2  
3

# 1. DISPLAY PATTERNS

56  
3  
2 3  
1 2 3  
0 1 2 3  
1 2 3  
2 3  
1

57  
D  
D C  
D C B  
D C B A  
D C B  
D C  
D

58  
D  
C D  
B C D  
A B C D  
B C D  
C D  
D

59  
\*  
\* \*  
\* \* \*  
\* \* \* \*  
\* \* \*  
\* \*  
\*

# 1. DISPLAY PATTERNS

60

```
    3
   3 2
  3 2 1
 3 2 1 0
 3 2 1
   3 2
    3
```

61

```
    3
   2 3
  1 2 3
 0 1 2 3
  1 2 3
   2 3
    3
```

62

```
    D
   C D
  B C D
 A B C D
  B C D
   C D
    D
```

63

```
    D
   D C
  D C B
 D C B A
  D C B
   D C
    D
```

# 1. DISPLAY PATTERNS

64

```
*
* *
* * *
* * * *
* * * * *
```

65

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

66

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

67

```
A
B B
C C C
D D D D
E E E E E
```

68

```
A
A B
A B C
A B C D
A B C D E
```

69

```
* * * * *
* * * *
* * *
* *
*
```

# 1. DISPLAY PATTERNS

70

5 5 5 5 5  
4 4 4 4  
3 3 3  
2 2  
1

71

5 4 3 2 1  
4 3 2 1  
3 2 1  
2 1  
1

72

E E E E E  
D D D D  
C C C  
B B  
A

73

E D C B A  
D C B A  
C B A  
B A  
A

74

A B C D E  
A B C D  
A B C  
A B  
A

# 1. DISPLAY PATTERNS

75

```
      *
     * *
    * * *
   * * * *
  * * * * *
 * * * * *
  * * * *
   * * *
    * *
     *
```

76

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
4 4 4 4
3 3 3
2 2
1
```

77

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

78

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```



# 1. DISPLAY PATTERNS

79

```
      A
     B B
    C C C
   D D D D
  E E E E E
 D D D D
C C C
 B B
  A
```

80

```
  A
 A B
 A B C
 A B C D
 A B C D E
  B C D E
   C D E
    D E
     E
```

81

```
      *
     **
    * *
   * *
  * *
 * *
```

82

```
      1
     2 2
    3 3
   4 4
  5 5
```

# 1. DISPLAY PATTERNS

83

```
5
4 4
3 3
2 2
1 1
```

84

```
A
B B
C C
D D
E E
```

85

```
E
D D
C C
B B
A A
```

86

```
* *
* *
* *
* *
*
```

# 1. DISPLAY PATTERNS

87

```
1 1
2 2
3 3
4 4
5
```

88

```
5 5
4 4
3 3
2 2
1
```

89

```
A A
B B
C C
D D
E
```

90

```
E E
D D
C C
B B
A
```

## 2. NUMBER SYSTEMS

1. Print bigger number from 2 given numbers.
2. Print smaller number from 3 given numbers.
3. Print middle number from 3 given numbers.
4. Print ascending order of 3 given numbers
5. Print true, if sum of any 2 numbers is a 3rd given number
6. Print true, if sum of squares of any 2 numbers is a 3rd given number
7. Find out given number is an even or odd?
8. Find out immediate next 5 multiple of a given number?
9. Print all odd numbers between two given numbers?
10. Find out sum of all even numbers between two given numbers?
11. Print java for multiples of 4 and print j2ee for multiples of 5 and if number is multiple of 4 and also 5, then print lara. continue till to 100.
12. Find out whether digits are in raising order or not in a given number?
13. Swap two int variables without 3rd variable?
14. Find out total digits of a given number?
15. Find out sum of all digits of a given number?
16. Reverse a given number?
17. Find out given number is a palindrome or not?
18. Assume total number of digits in a given number is even. Reverse only first half of the given number
19. Assume given number containing 3 digits. now check it out first two digits sum is 3rd digit or not?
20. Find out factorial value for a given number?

## 2. NUMBER SYSTEMS

21. Develop a program to find out given number is a prime or not?
22. Develop a program to print initial 20 prime numbers?
23. Develop a program to print prime numbers between 50 and 100?
24. Develop a program to print prime numbers between two given numbers?
25. Develop a program to print 20 prime numbers from 100.
26. Develop a program to print sum of initial 30 prime numbers?
27. Develop a program to print sum of prime numbers between two given numbers?
28. Develop a program to print prime numbers in the reverse order from 70 to 20?
29. Develop a program to print prime numbers which are immediately after multiples of 10 and below 200.
30. Develop a program to print prime numbers which are just before multiples of 10 and below 200.
31. Develop a program to print 30 prime numbers which are having digits in the raising order after 10. (Example: 13, 17, 19, 23, 29, 37, 39, 47,.....)
32. Develop a program to print 30 prime numbers which are having digits in the descending order after 10. (Example: 31, 41, 43, 51, 53, 61, 71....)
33. Develop a program to print numbers which should come after 3 non prime numbers. Develop between 10 to 100.
34. Print Fibonacci series till to 100.
35. Print initial 10 numbers from the Fibonacci series.
36. Print Fibonacci series from 100 to 10000?
37. Print Fibonacci series in the reverse order from 5000 to 500?
38. Print immediate next number in the Fibonacci series? Consider till to 5000
39. Print all prime numbers only from the Fibonacci series? Continue till to 5000
40. Find out given number is an Armstrong number or not?

## 2. NUMBER SYSTEMS

41. Find out initial 3 Armstrong numbers (Consider from 10)
42. Find out all Armstrong numbers between 2 given numbers?
43. Find out given number is a perfect number or not?
44. Find out initial 5 perfect numbers (Consider from 10)
45. Find out all perfect numbers between 2 given numbers?
46. Find out given 2 numbers are anagrams or not Anagrams for 123 are  
321, 312, 213, 231, 131
47. Assume given number containing even number of digits. Reverse only  
2nd half of the number?
48. Find out sum of the digits of a given number?  
123 -> 6  
67 -> 13 -> 4  
869 -> 24 -> 6
49. Move first half to second and second half to first?  
(total digits are even) Ex: 123456 convert to 456123, 903512  
convert to 512903
50. Every digit swap with immediate digit. (total digits are even)  
Ex: 123456 convert to 214365

### 3. ARRAYS AND SORTING ALGORITHMS

1. Find the sum of all given elements from an int array?
2. Find the min element from the given int array?
3. Find the max element from the given int array?
4. Find the 2 nd min element from the given int array?
5. Find the 2 nd max element from the given int array?
6. Find the average value of an int array?
7. Find out the sum of all even indexed elements from a given int array?
8. Find out the sum of all odd indexed elements from a given int array?
9. Find out the min value from all even indexed elements from a given int array?
10. Find out the max value from all odd indexed elements from a given int array?
11. Find out the avg value from all even indexed elements from a given int array?
12. Find out the avg value from all odd indexed elements from a given int array?
13. Find out the sum of all elements from a first half of given int array?
14. Find out the sum of all elements from a second half of given int array?
15. Find out the min value from a first half of given int array?
16. Find out the min value from a second half of given int array?
17. Find out the max value from a first half of given int array?
18. Find out the max value from a second half of given int array?
19. Find out the avg value from a first half of given int array?
20. Find out the avg value from a second half of given int array?
21. Read all elements from an array in the reverse order?
22. Read first half of the elements in the reverse direction from an array?
23. Read second half of the elements in the reverse direction from an array?
24. Read only even indexed elements from an array?
25. Read only even indexed elements from an array in the reverse order?
26. Read only odd indexed elements from an array?
27. Read only odd indexed elements from an array in the reverse order?
28. Find out an index of a specified element from a given array?
29. Swap two given indexed elements from the array?
30. Reverse the elements of given array?

### 3. ARRAYS AND SORTING ALGORITHMS

31. Reverse only first half of the elements of given array?
32. Reverse only last half of the elements of given array?
33. Reverse only even indexed of the elements of given array?
34. Reverse only odd indexed of the elements of given array?
35. Swap odd indexed elements with its immediate next even indexed elements of given array?
36. Do right shift by one for elements of given array?
37. Do right shift by two for elements of given array?
38. Do right shift by three for elements of given array?
39. Do left shift by one for elements of given array?
40. Do left shift by two for elements of given array?
41. Do left shift by three for elements of given array?
42. Do right rotate by one for elements of given array?
43. Do right rotate by two for elements of given array?
44. Do right rotate by three for elements of given array?
45. Do left rotate by one for elements of given array?
46. Do left rotate by two for elements of given array?
47. Do left rotate by three for elements of given array?
48. Rotate first half of elements by one?
49. Rotate 2nd half of elements by one?
50. Rotate first half of elements by one and 2 nd half of the elements by one separately?
51. Remove specified indexed element from the given array?
52. Update specified indexed element with a new element from the given array?
53. Remove all occurrences of specified element from the given array?
54. Remove rang of elements from the given array
55. Remove all odd indexed elements from the given array?
56. Remove all even indexed elements from the given array?
57. Remove the duplicates from the given array?
58. Find out missed elements from the given array between min and max element
59. Remove elements of one array from another?
60. Retain one array elements in another array?)



### 3. ARRAYS AND SORTING ALGORITHMS

61. Find out common elements from two given arrays?
62. Find out uncommon elements from two given arrays?
63. Combine two arrays and develop a third array?
64. Find out index of an element which contains left indexed element is same as right indexed element.
65. Find out the elements which are not duplicates in the given array
66. Find out elements which are having minimum one duplicate?
67. Find out element frequency in the given array?
68. Combine two arrays and develop a third array.  
Consider element by element while combining
69. Combine two arrays and develop a third array. Consider element by element while combining and take forward direction from first array and reverse direction from 2 nd array
70. Sorting int elements from an arrays? (use bubble sort)
71. Sorting int elements from an arrays? (use quick sort)
72. Sorting int elements from an arrays? (use merge sort)
73. Sorting int elements from an arrays? (use insertion sort)
74. Sorting int elements from an arrays? (use selection sort)

## 4. RECURSIVE ALGORITHMS

1. Print sequential number from 1 to 100 without any loops
2. Print sequential number from 100 to 1 without any loops
3. Print prime numbers from 50 to 100 without any loops
4. Develop Fibonacci series without loops
5. Reverse a string without any loops
6. Find out sum of all the digits in a given number without loops
7. Reverse a number without any loops?
8. Find out factorial value for a given number without loops?

LARATECHNOLOGIES

## 5. STRING AND FILE HANDLING

1. Write a program to reverse a String
2. Write a program to reverse a String with recursive algorithm?
3. Write a program to reverse first half separately and 2 nd half separately?
4. Write a program to rotate one char in a given string
5. Find out length of the string without length() method of a String?
6. Find out how many words are there in a given string?
7. Write a java program to find the duplicate words and their number of occurrences in a string?
8. Write a program to reverse the given string word wise?
9. Rotate the string word wise by one
10. Write a java program to count the total number of occurrences of a given character in a string?
11. Write a java program to count the number of occurrences of each character in a string?
12. Write a java program to remove all white spaces from a string?
13. Write a program to check whether given string is a palindrome or not?
14. Write a program to check whether given two strings are anagrams?
15. Write a program to check the balance of brackets in the given string?
16. Write a java program to find duplicate characters in a string?
17. Write a java program to check whether one string is a rotation of another?
18. Write a java program to reverse a given string with preserving the position of spaces?
19. Write a java program to reverse each word of a given string?
20. Write a java program to find the percentage of uppercase letters, lowercase letters, digits and special characters in a given string?

## 5. STRING AND FILE HANDLING

21. How do you find longest substring without repeating characters in the given string?
22. How do you swap two string variables without using third or temp variable in java?
23. Write a java program to find all permutations of a string?
24. How do you find first repeated and non-repeated character in the given string in java?
25. How do you find the number of characters, words and lines in the given text file in java?
26. How do you find the most repeated word in a text file in java?
27. How to search a word inside a string?
28. How to remove html tags from a string?
29. Write a program to print all permutations of String?
30. Write a function to find out longest palindrome in a given string?
31. Write a program to validate email format?
32. Write a program to validate date format?
33. Write a program to validate phone number format?
34. Write a program to validate specified username format?
35. Write a program to validate specified password format?
36. Write a program to validate hex code format?
37. Write a program to validate image file extension?
38. Write a program to validate IP Address
39. Write a program to validate 12 hours specified time format?
40. Write a program to validate 24 hours specified time format
41. Find out longest string from the given file?
42. Find out longest sentence from the given file?
43. Find out number of lines in the given file?
44. Find out number of words in a given file?
45. Find out a word which is occurring more times in a file?
46. Find out a word count of all the words from a given files?
47. Find out total number of files and sub directories from a given directory?

## 5. STRING AND FILE HANDLING

48. Given a string, look for a mirror image (backwards) string at both the beginning and end of the given string. In other words, zero or more characters at the very beginning of the given string, and at the very end of the string in reverse order (possibly overlapping). For example, the string "abXYZba" has the mirror end "ab".
- ```
mirrorEnds("abXYZba") → "ab"  
mirrorEnds("abca") → "a"  
mirrorEnds("aba") → "aba"
```
49. Given a string, return the sum of the numbers appearing in the string, ignoring all other characters. A number is a series of 1 or more digit chars in a row. (Note: Character.isDigit(char) tests if a char is one of the chars '0', '1', .. '9'. Integer.parseInt(string) converts a string to an int.)

```
sumNumbers("abc123xyz") → 123  
sumNumbers("aallb33") → 44  
sumNumbers("7 11") → 18
```

50. Given a string, return the sum of the digits 0-9 that appear in the string, ignoring all other characters. Return 0 if there are no digits in the string. (Note: Character.isDigit(char) tests if a char is one of the chars '0', '1', .. '9'. Integer.parseInt(string) converts a string to an int.)

```
sumDigits("aallbc2d3") → 6  
sumDigits("aallb33") → 8  
sumDigits("Chocolate") → 0
```

51. Given a string, compute a new string by moving the first char to come after the next two chars, so "abc" yields "bca". Repeat this process for each subsequent group of 3 chars, so "abcdef" yields "bcaefd". Ignore any group of fewer than 3 chars at the end. oneTwo("abc") → "bca"  
oneTwo("tca") → "cat" oneTwo("tcagdo") → "catdog"

## 5. STRING AND FILE HANDLING

52. Given a string, return a string where for every char in the original, there are two chars.
- ```
doubleChar("The") → "TThhee"  
doubleChar("AAbb") → "AAAAbbbb"  
doubleChar("Hi-There") → "HHii--TThheerree"
```
53. Given two strings, a and b, create a bigger string made of the first char of a, the first char of b, the second char of a, the second char of b, and so on. Any leftover chars go at the end of the result.
- ```
mixString("abc", "xyz") → "axbycz"  
mixString("Hi", "There") → "HTihere"  
mixString("xxxx", "There") → "xTxhxexre"
```
54. Given a string and an int n, return a string made of n repetitions of the last n characters of the string. You may assume that n is between 0 and the length of the string, inclusive.
- ```
repeatEnd("Hello", 3) → "llollo"  
repeatEnd("Hello", 2) → "lolo"  
repeatEnd("Hello", 1) → "o"
```
55. Return a version of the given string, where for every star (\*) in the string the star and the chars immediately to its left and right are gone. So "ab\*cd" yields "ad" and "ab\*\*cd" also yields "ad".
- ```
starOut("ab*cd") → "ad"  
starOut("ab**cd") → "ad"  
starOut("sm*eilly") → "silly"
```
56. Given a string and a non-empty word string, return a version of the original String where all chars have been replaced by pluses ("+"), except for appearances of the word string which are preserved unchanged.
- ```
plusOut("12xy34", "xy") → "++xy++"  
plusOut("12xy34", "1") → "1++++"  
plusOut("12xy34xyabcxy", "xy") → "++xy++xy+++xy"
```

## 6. DATA STRUCTURES

1. LinkedList : Adding an Element
2. LinkedList : Adding array of Elements
3. LinkedList : Insertion of an element in the specified position
4. LinkedList : Insertion of an array of elements in the specified position
5. LinkedList : Iterating all elements
6. LinkedList : updating a specified Element with a new element.
7. LinkedList : Finding out length (size)
8. LinkedList : Removing specified element
9. LinkedList : Removing specified position element
10. LinkedList : Removing all duplicate data nodes.
11. LinkedList : swapping two specified node's data
12. LinkedList : swapping two specified nodes itself
13. LinkedList : Reverse data from the nodes
14. LinkedList : Reverse all nodes itself
15. LinkedList : Sort Data from the nodes
16. LinkedList : Sort Nodes itself
17. Merge two sorted linkedlists
18. Merge Sort for LinkedLists
19. Detect and Remove Loop in a Linked List
20. Rotate a LinkedList
21. Making LinkedList as a circular
22. Split a Circular LinkedList into two halves
23. Developing Doubly LinkedList
24. Delete a node in a Doubly LinkedList
25. Reverse a Doubly LinkedList
26. Making Doubly LinkedList as a circular
27. BinaryTree: Add elements
28. BinaryTree : Reading InOrder
29. BinaryTree : Reading PostOrder
30. BinaryTree : Reading PreOrder
31. BinaryTree : Search
32. BinaryTree : Count Nodes



**ANGULAR**

**&**



**JAVA SCRIPT**



# 1. GETTING STARTED

1. What is Angular
2. Angular Version History
3. Architecture of Angular Apps
4. Setting Up the Development Environment
5. Structure of Angular Projects
6. Your First Angular App

## 2. TYPESCRIPT FUNDAMENTALS

1. Introduction
2. What is TypeScript
3. Your First TypeScript Program
4. Declaring Variables Types
5. Type Assertions
6. Arrow Functions
7. Interfaces
8. Classes
9. Objects
10. Constructors
11. Access Modifiers
12. Access Modifiers in Constructor Parameters
13. Properties
14. Modules

## 3. ANGULAR FUNDAMENTALS

1. Introduction
2. Building Blocks of Angular Apps
3. Dependency Injection
4. Generating Components Using Angular CLI
5. Templates
6. Directives
7. Services
8. Generating Services Using Angular CLI

## 4. DISPLAYING DATA AND HANDLING EVENTS

1. Introduction
2. Property Binding
3. Attribute Binding
4. Adding Bootstrap
5. Class Binding
6. Style Binding
7. Event Binding
8. Event Filtering
9. Template Variables
10. Two-way Binding Pipes
11. Custom Pipes

## 5. BUILDING REUSABLE COMPONENTS

1. Introduction
2. Component API
3. Input Properties
4. Aliasing Input Properties
5. Output Properties
6. Passing Event Data
7. Aliasing Output Properties
8. Templates Styles
9. View Encapsulation
10. ngContent
11. ngContainer

## 6.DIRECTIVES

1. Introduction
2. ngIf
3. Hidden Property
4. The Leading Asterisk
5. ngClass
6. ngStyle
7. ngSwitchCase
8. ngFor
9. ngFor and Change Detection
10. ngFor and track By
11. Safe Traversal Operator
12. Creating Custom Directives

## 7. TEMPLATE-DRIVEN FORMS

1. Introduction
2. Building a Bootstrap Form
3. Types of Forms
4. ngModel
5. Adding Validation
6. Specific Validation Errors
7. Styling Invalid Input Fields
8. Cleaner Templates
9. ngForm
10. ngModelGroup
11. Control Classes and Directives
12. Disabling the Submit Button
13. Working with Check Boxes
14. Working with Drop-down Lists
15. Working with Radio Buttons

## 8. REACTIVE FORMS

1. Introduction
2. Building a Bootstrap Form
3. Creating Controls
4. Programmatically
5. Adding Validation
6. Specific Validation Errors
7. Implementing Custom Validation
8. Asynchronous Operations
9. Asynchronous Validation
10. Showing a Loader Image
11. Validating the Form Upon Submit
12. Nested Form Groups
13. Form Array
14. Form Builder



## 9. CONSUMING HTTP SERVICES

1. Introduction
2. JSON Placeholder
3. Getting Data
4. Creating Data
5. Updating Data
6. Deleting Data
7. OnInit Interface
8. Separation of Concerns
9. Extracting a Service
10. Handling Errors
11. Handling Unexpected Errors
12. Handling Expected Errors
13. Throwing Application- specific Errors
14. Handling Bad Request Errors
15. Importing Observable
16. Operators and Factory Methods
17. Global Error Handling
18. Extracting a Reusable Error
19. Handling Method
20. Extracting a Reusable Data Service
21. The Map Operator
22. Optimistic vs Pessimistic Updates
23. Observables vs Promises

## 10. ROUTING AND NAVIGATION

1. Introduction
2. Routing in a Nutshell
3. Configuring Routes
4. RouterOutlet
5. RouterLink
6. RouterLinkActive
7. Getting the Route Parameters
8. Why Route Parameters Are
9. Observable
10. Routes with Multiple Parameters
11. Query Parameters
12. Subscribing to Multiple Observables
13. Switch Map Operator
14. Programmatic Navigation



# **SALES FORCE DEVELOPMENT**

# 1. APEX OBJECT ORIENTED PROGRAMMING LANGUAGE

1. Creating a Developer's Account
2. Data types in Apex
3. Variables
4. Operators
5. Decision Controls (if, else and switch-case-default)
6. Loops (for, while, do-while, break, continue, inner, infinite and advanced for loop)
7. Static variables and methods
8. Class and object with non static variables and methods
9. Constructors
10. Access levels
11. Inheritance
12. Abstraction
13. Casting
14. Polymorphism
15. Final
16. Wrapper classes
17. Exception Handling (try, catch, finally, throw and custom)
18. String
19. Collection (List, Set and Map)

## 2. SQL

1. Downloading and installation of Oracle DataBase 11g Express Edition
2. Downloading and installation of SQLdeveloper
3. Sql commands for table, sequence and procedures □ DDL, DML and DQL
4. SQL Functions
5. SQL without Constraints
6. SQL with Constraints
7. Not Null
8. Unique
9. Combination of not null and Unique
10. Primary key
11. Composite Key
12. Composite Unique Key
13. Composite Primary Key
14. Foreign Key
15. Foreign key relation ship
16. One-to-one-relationship
17. One to-many-relationship / Many-to-one relationship
18. Many-To-Many relationship
19. SQL Joins
20. Inner-Join
21. Left-Outer-Join
22. Right-Outer-Join
23. Full-Outer-Join
24. Date,
25. TimeStamp
26. RowId
27. RowNum

1. CRM Overview
2. Salesforce Architecture (SaaS, PaaS, On Demand)
3. Salesforce Applications (Marketing, Sales, Service) Overview
4. Developer account
5. Identifying a project and exploring all requirements
6. Creating custom Objects
7. Different types of Fields
8. Creating Junction Object
9. Master-Detail Relationship
10. Schema builder
11. Creating Tabs
12. Creating Applications
13. Inserting/updating/deleting data into the application.
14. Lookup Dialogs
15. Edit Layout
16. Apex API classes
17. Linking Apex Programming task to UI event.
18. Creating Apex classes
19. SOQL and DML
20. Triggers
21. Visualforce pages
22. Standard controllers.
23. Custom controllers
24. Email sending and batch processes
25. Creating Unit Tests



**THANK YOU**

**FOR ANY QUERY FEEL FREE TO CONTACT US**

Our support Hotline is available 24 Hours a day: 0997-200-6654  
Monday – Sunday: 7 am to 9 pm

**ADDRESS**

8, Outer Ring Rd, BTM 2nd Stage, Kuvempu Nagar, Stage 2, BTM  
Layout, Bengaluru, Karnataka 560076

"Lara is the knowledge hub,  
Lets come, get the tremendous knowledge & Serve to the IT industry"