

Starts @ 10:10 am

Introduction

Session

→ Boot camp → 10 months

↳ M1 → Basic Java Programming

↳ M2 → Data Structure and Algorithms

↳ M3 → Role based preparation and aptitude, logical reasoning

↳ Projects → 5-7 projects on Data Structure

↳ Module 1 (M1) → Basics of Java Programming Language.

- ↳ How to print in Java.
 - ↳ Operators
 - ↳ Conditional Statements
 - ↳ Return / break
 - ↳ Loops → for, while, do while
 - ↳ Functions
- ↳ Importance of Java Programming Language.
- ↳
-

↳ Module -02 (M2) → Data Structure and Algorithms

↳ Array

↳ String and String Builder

↳ Stack

↳ Queue

↳ Linked List

↳ Binary Tree

→ Binary Search Tree

→ Graph.

→ Binary Search.

→ Sliding Window

→ Two pointers

→ Bit Manipulation

→ Recursion / Backtracking

→ ODP's //

→ Priority Queue

→ Hash map and Hashset

→ Dynamic Programming

optional
→ Trie.

→ Sorting Algorithms

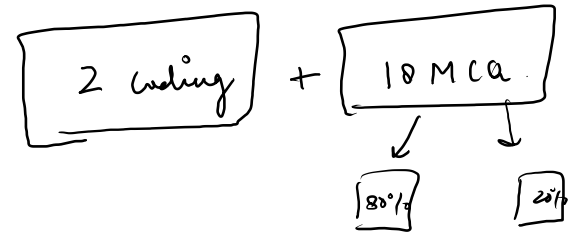
→ Time Complexity and
space complexity.

↳ Module 03 (M3)

↳ Role based Knowledge:- (80%) and Data Structure Theoretical Question

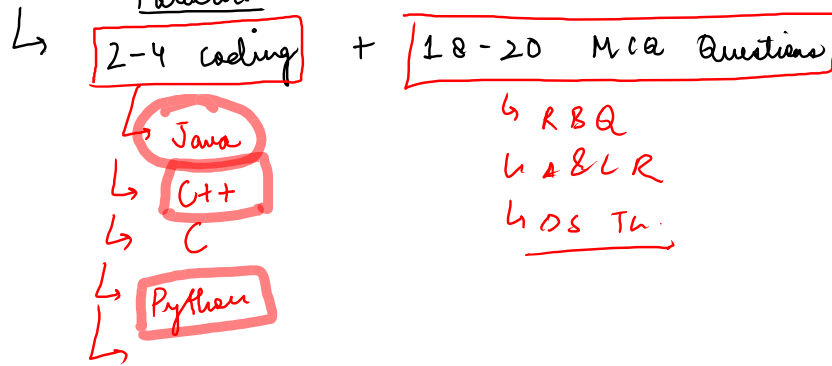
- Software Development Engineer (SDE)
- Data Engineer
- Support Engineer
- Quality Assurance

↳ Aptitude and Logical Reasoning (20%)



Tests

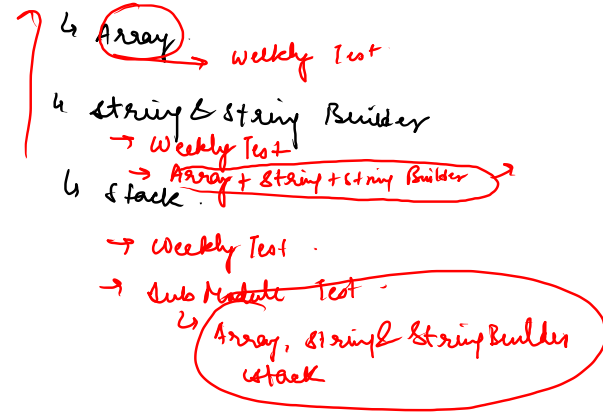
Pattern



↳ Weekly Test

↳ Module Test → M1, M2,
↓ ↓

M2



↳ Sub Module Test

↳ MCQ Test → 1 week deadline -
→ Google form (30-40)

→ Hacker Rank →

→ Classwork Question Link → 2-5 Question

→ Homework Question Link → 4-5 Questions

→ Hacker Rank

→ Gfg

→ LeetCode

→ CSOS

→ Codechef

→ Codeforce

↳ 120 minutes

↳ Join within 5 minutes.

↳

↳ 10 - 12

11)

10 break

→ ~~test code contest~~

4 questions

→ 1 Easy

→ 2 Medium

→ 1 Hard

→ 2 hours

Saturday

↳ 8:30 - 10:30 pm

Sunday

↳ 8 - 10 am

→ ~~codechef contest~~

→ ~~codeforces contest~~

→ Gfg (Geeks for Geeks) → select any random topic

Friday → 7-9 pm

Sat/Sunday → 10-12 morning

Open board

→ Cont

→ cin

→ for, while

↳ for (int i = 0; i < n; i++)

→

→ return, break

→ if, else if, else

→ STL →

(stack

→ system.out.println()

→ Scanner class.

→ for, while

→ for (int i = 0; i < n; i++)

{

}

→ return, break.

→ "if, else if else.

→ Classes

Stack < Integer →