



JSM Infotech
A Step Towards Digital India

Logic Building (Complete Practical Approach)



आमच्या इथे स्वच्छ, ताजे, सात्विक, घरगुती पध्दतीने **Logic Build** करून मिळेल

Boost Your Programming Skills with Our Logic Building Batch!

Master the fundamentals of programming with our hands-on batch in C, C++, and Java. Whether you're a beginner or looking to strengthen your problem-solving skills, this course will help you build a solid foundation in logic development, algorithm design, and coding. Start your journey towards becoming a skilled programmer today!



- **Master the Basics:** Strengthen your understanding of core programming concepts with C, C++, and Java.
- **Hands-on Learning:** Engage in practical coding exercises to sharpen your problem-solving abilities.
- **Build Strong Logic:** Learn how to develop efficient algorithms and write clean, optimized code.
- **Ideal for Beginners & Intermediates:** Perfect for anyone looking to build a solid foundation in programming.
- **Comprehensive Curriculum:** Dive deep into data structures, algorithms, and logical problem-solving techniques.
- **Career Readiness:** Enhance your technical skills to stand out in job interviews and technical assessments.
- **Expert Guidance:** Learn from experienced instructors with a deep understanding of programming and real-world applications.
- **Real-World Applications:** Apply your logic-building skills to real-world programming challenges and projects.
- **Time-Efficient:** Achieve maximum learning in a structured, time-efficient format designed to fit your schedule.
- **Interactive Sessions:** Participate in live coding sessions, peer discussions, and Q&A to clarify doubts and improve understanding.
- **Increase Efficiency:** Learn best practices to write efficient, scalable, and maintainable code.
- **Problem-Solving Focus:** Tackle a wide variety of problems to improve critical thinking and enhance analytical skills.
- **Industry-Recognized Skills:** Gain programming skills that are highly valued across industries in tech, finance, and more.
- **Personalized Attention:** Small batch sizes ensure you get the individual attention you need to succeed.

In this batch we cover below topics



- ❖ **Programming Focused on Core Algorithmic Strategies:** Build strong problem-solving skills by mastering fundamental algorithmic approaches for efficient coding.
- ❖ **Programming Using Selection Techniques:** Enhance your problem-solving skills by mastering selection algorithms for efficient decision-making in coding.
- ❖ **Programming Using Iteration Techniques:** Strengthen your coding skills by mastering iterative methods for solving problems efficiently through repetition and loops.
- ❖ **Programming Using a Functional Approach:** Develop your coding skills by applying functional programming principles to create clean, modular, and efficient code.
- ❖ **Programming Focused on Numerical Methods:** Enhance your problem-solving abilities by mastering techniques and algorithms centered around numerical computations and operations.
- ❖ **Programming with Arrays and Their Manipulations:** Strengthen your coding skills by mastering array operations and techniques for efficient data handling and manipulation.
- ❖ **Programming with Dynamic Memory Allocation:** Enhance your programming expertise by mastering techniques for efficient memory management and dynamic memory allocation.
- ❖ **Programming Focused on Digits and Number Manipulation:** Develop your coding skills by working with digits and implementing techniques for effective number manipulation and processing.
- ❖ **Programming with Input Validation Techniques:** Strengthen your coding skills by implementing robust input validation methods to ensure data integrity and prevent errors.



- ❖ **Programming with Series and Sequences:** Enhance your coding skills by working with different types of series and sequences, and applying techniques to generate and manipulate them effectively.
- ❖ **Programming with Brute Force Algorithms:** Develop problem-solving skills by implementing brute force techniques to solve problems through exhaustive search and direct computation.
- ❖ **Programming Using Divide and Conquer Strategy:** Strengthen your problem-solving abilities by applying the divide and conquer approach to break down complex problems into simpler sub-problems for efficient solutions.
- ❖ **Programming with Pattern Printing:** Enhance your coding skills by mastering techniques to generate and print various patterns using loops and logic.
- ❖ **Programming with Numbering Systems:** Strengthen your coding abilities by working with various numbering systems, such as binary, decimal, octal, and hexadecimal, and converting between them efficiently.
- ❖ **Programming with Strings:** Enhance your coding skills by mastering string manipulation techniques, including searching, modifying, and formatting string data efficiently.
- ❖ **Programming with Multiple Strings:** Strengthen your skills by working with multiple strings, focusing on operations like concatenation, comparison, and manipulation of string arrays or collections.
- ❖ **Programming with Matrices (Multidimensional Arrays):** Enhance your problem-solving skills by working with matrices, performing operations like traversal, manipulation, and matrix-based calculations in multidimensional arrays.
- ❖ **Programming with Bitwise Operators:** Strengthen your coding skills by mastering bitwise operations, including AND, OR, XOR, and shifts, to efficiently manipulate data at the bit level.



- ❖ **Programming with Mask Designing:** Enhance your skills by working with mask design techniques to manipulate specific bits in data for tasks like filtering, encoding, and processing.
- ❖ **Programming with Structures and Unions:** Strengthen your coding skills by mastering the use of structures and unions to efficiently store and manage different data types in memory.
- ❖ **Programming with Structure Design Strategy:** Enhance your skills by applying structure design techniques to organize and manage complex data efficiently in memory.
- ❖ **Programming with Searching Algorithms:** Strengthen your problem-solving skills by mastering various searching techniques, such as linear search and binary search, for efficient data retrieval.
- ❖ **Programming with Sorting Algorithms:** Enhance your coding skills by mastering sorting techniques like bubble sort, quicksort, and merge sort to efficiently arrange data in a desired order.
- ❖ **Programming with Recursion vs Iteration:** Strengthen your problem-solving skills by understanding the differences and applications of recursion and iteration for efficient algorithm design.
- ❖ **Programming with Java Libraries:** Enhance your coding efficiency by leveraging Java's built-in libraries for tasks like data manipulation, file handling, and advanced operations.
- ❖ **Programming with Collection Frameworks:** Strengthen your skills by mastering Java's collection framework, including lists, sets, maps, and queues, to efficiently store, retrieve, and manipulate data.
- ❖ **Programming with Data Structures:** Enhance your coding skills by mastering fundamental data structures like arrays, linked lists, stacks, queues, and trees for efficient data organization and manipulation.



- ❖ **Programming with Complex Data Structures:** Strengthen your skills by working with advanced data structures such as graphs, hash maps, heaps, and tries to solve complex problems efficiently.
- ❖ **Programming with Object-Oriented Thinking:** Enhance your coding skills by applying object-oriented principles like encapsulation, inheritance, polymorphism, and abstraction to design efficient and scalable software.
- ❖ **Programming with Class Design:** Strengthen your skills by mastering the design and structuring of classes to create modular, maintainable, and reusable code in object-oriented programming.
- ❖ **Programming with Polymorphism:** Enhance your coding skills by applying polymorphism techniques to allow objects to be treated as instances of their parent class, enabling flexible and scalable code design.
- ❖ **Programming with Inheritance and Reusability:** Strengthen your skills by leveraging inheritance to create reusable code and establish relationships between classes for efficient software development.
- ❖ **Programming with Generic Programming Strategy:** Enhance your coding skills by using generics to write flexible, reusable, and type-safe code that works with any data type.
- ❖ **Programming with File Handling:** Strengthen your skills by mastering techniques for reading, writing, and manipulating files, ensuring efficient data storage and retrieval in your programs.
- ❖ **Programming with Linked List Data Structure:** Enhance your skills by working with linked lists to efficiently manage dynamic data and perform operations like insertion, deletion, and traversal.
- ❖ **Programming with Queue Data Structure:** Strengthen your skills by utilizing queues to manage data in a first-in, first-out (FIFO) order, ideal for tasks like scheduling and resource management.

- ❖ **Programming with Stack Data Structure:** Enhance your skills by working with stacks to manage data in a last-in, first-out (LIFO) order, useful for tasks like function calls and expression evaluation.
- ❖ **Programming with Tree Data Structure:** Strengthen your skills by mastering tree structures to efficiently organize and manage hierarchical data, enabling fast search, insertion, and deletion operations.
- ❖ **Programming with Advanced Linked List Applications:** Enhance your skills by working on complex linked list problems and applications, such as circular lists, doubly linked lists, and multi-level linked structures.
- ❖ **Programming with Generic Data Structure Implementations:** Strengthen your skills by implementing flexible and reusable data structures using generics, enabling type safety and adaptability across different data types.

After all we will create our own Industrial Project



For Registration: [Registration Form Link](#)