"Placement assurance for all, Minimum 50 learners get 100% Placement or Full fee refund."

" IITians-Founded Coder Pathshala "

Expert **MERN** career support with **IIT alumni** and top professionals from **Google, Reliance-Jio**, **IISc-Bangalore** & leading research institutes.

Programming in C (Semester 1)

Module 1: Introduction to Programming & C Basics

- Problem-solving techniques
- Algorithms and flowcharts
- Structure of a C program
- Compilation and execution process

Module 2: Data Types & Operators

- Variables and constants
- Data types and type conversion
- Arithmetic, relational, logical operators
- Expressions and precedence

Module 3: Control Structures

- Conditional statements
- Looping constructs
- Nested loops
- Break and continue

Module 4: Arrays & Strings

- One-dimensional and two-dimensional arrays
- Character arrays and string functions
- Array applications

Module 5: Functions & Pointers

- Function declaration and definition
- Call by value and call by reference
- Pointer basics and pointer arithmetic
- Dynamic memory allocation

Module 6: Structures & File Handling

- Structures and unions
- File operations
- Command-line arguments
- Simple applications

Python Programming (Semester 1)

Module 1: Python Fundamentals

- Introduction to Python
- Syntax and indentation
- Variables and input/output

Module 2: Control Flow & Functions

- Conditional statements
- Loops
- Functions and parameters

Module 3: Python Data Structures

- Lists and tuples
- Sets and dictionaries
- Data manipulation techniques

Module 4: Strings, Files & Exceptions

- String operations and formatting
- File read/write
- Exception handling

Module 5: Modules & Libraries

- Importing modules
- Standard libraries
- Introduction to NumPy

Module 6: Problem Solving with Python

- Algorithmic thinking
- Logical problem solving
- Mini-project using Python

Object Oriented Programming using C++ (Semester 2)

Module 1: C++ Fundamentals

- C vs C++
- Program structure
- Input/output streams

Module 2: Classes & Objects

- Class definition
- Objects and access specifiers
- Constructors and destructors

Module 3: Inheritance

- Types of inheritance
- Method overriding
- Virtual base classes

Module 4: Polymorphism

- Function overloading
- Operator overloading
- Virtual functions

Module 5: Templates & STL

- Function and class templates
- STL containers
- Iterators and algorithms

Module 6: Exception & File Handling

- Exception handling mechanism
- File input/output
- Applications

Programming in Java (Semester 2)

Module 1: Java Basics & JVM

- Java features
- JVM, JDK, JRE
- Java program structure

Module 2: OOP Concepts in Java

- Classes and objects
- Constructors
- Inheritance

Module 3: Polymorphism & Abstraction

- Method overriding
- Abstract classes
- Interfaces

Module 4: Packages & Exception Handling

- Built-in packages
- User-defined packages
- Exception hierarchy

Module 5: Multithreading

- Thread lifecycle
- Synchronization
- Inter-thread communication

Module 6: File Handling & Collections

- File I/O streams
- Serialization
- Java Collection Framework

Semester 3 – Modern Web Foundations with React

Module 1: Fundamentals of Web Development

- How the web works
- Client–server architecture
- Frontend vs backend
- Web technology overview
- Developer tools

Module 2: HTML Foundations

- HTML structure
- Semantic elements
- Forms & inputs
- Accessibility basics
- Page structuring

Module 3: CSS Basics

- Selectors & properties
- Box model
- Layout & positioning
- Flexbox basics
- Typography & colors

Module 4: Advanced CSS & Tailwind

- Responsive design
- CSS Grid
- Animations
- Tailwind utility classes
- Mobile-first layouts

Module 5: Git & GitHub

- Version control basics
- Git workflow
- Branching & merging
- GitHub repositories
- Pull requests

Module 6: JavaScript Fundamentals

- Variables & data types
- Functions & scope
- Loops & conditionals
- DOM basics
- Event handling

Module 7: Advanced JavaScript

- Promises & async/await
- Callbacks
- Error handling
- API consumption
- Modern ES6 features

Module 8: React Basics

- SPA concepts
- JSX
- Components
- Props & state
- Basic hooks

Module 9: State Management with Redux

- Redux fundamentals
- Redux Toolkit
- Store & reducers
- Actions & slices
- React-Redux integration

Semester 4– Full-Stack Development with Node & MongoDB

Dive deeper into full-stack development by mastering advanced React techniques and building robust backends with Node.js and MongoDB.

Module 10: Advanced React Concepts

- Component lifecycle
- Error boundaries
- Strict mode
- Code organization
- Best practices

Module 11: React Hooks & Routing

- useState & useEffect
- Custom hooks
- Context API
- React Router
- Dynamic routes

Module 12: React Performance Optimization

- Memoization
- Lazy loading
- Code splitting
- useCallback & useMemo
- Optimization patterns

Module 13: Node.js Fundamentals

- Event-driven architecture
- Modules & NPM
- File system
- Async programming
- CLI apps

Module 14: Express.js & REST APIs

- Express setup
- Routing & middleware
- REST principles
- Error handling
- API structure

Module 15: MongoDB Fundamentals

- NoSQL concepts
- Collections & documents
- CRUD operations
- MongoDB Atlas
- Compass usage

Module 16: Mongoose & Backend Integration

- Schema design
- Models & validation
- Relationships
- Express + MongoDB
- Backend architecture

Semester 5 – Secure & Scalable Applications

Master the critical aspects of building robust, secure, and scalable web applications from frontend to backend.

Module 17: Authentication & Authorization

- Auth vs authz principles
- Sessions vs tokens
- JWT structure
- Secure login flows
- Token validation and refresh

Module 18: Password & Role Security

- Hashing & salting with bcrypt
- Role-Based Access Control (RBAC)
- Authentication middleware
- Secure credential management
- Protecting sensitive data

Module 19: Advanced MongoDB

- Aggregation framework
- Indexing strategies
- Transactions for data consistency
- Performance tuning techniques
- Complex query optimization

Module 20: Frontend–Backend Integration

- Efficient API consumption
- Axios & fetch for HTTP requests
- Token handling in frontend
- Implementing protected routes
- State synchronization techniques

Module 21: File Uploads & Media Handling

- Multer basics for file uploads
- Cloud storage integration (e.g., AWS S3)
- Image upload and manipulation
- Secure file handling practices
- Media optimization for performance

Module 22: Testing & Error Handling

- Jest fundamentals for unit testing
- Supertest for API testing
- Comprehensive API test suites
- Global error handling strategies
- Effective logging for debugging

Module 23: Deployment & Production Readiness

- Environment variables management
- Frontend deployment best practices
- Backend hosting options
- MongoDB Atlas for production DBs
- Production configuration and monitoring

Semester 6 – Enterprise-Grade MERN & Capstone

Build and deploy robust, real-time, and scalable full-stack applications, culminating in an industry-grade capstone project.

Module 24: Real-Time Applications

- WebSockets
- Socket.IO
- Event-based communication
- Live updates
- Chat systems

Module 25: GraphQL & Apollo

- GraphQL fundamentals
- Queries & mutations
- Apollo Server
- Apollo Client
- GraphQL caching

Module 26: Microservices Architecture

- Monolith vs microservices
- Service communication
- API gateways
- Scalability concepts
- Service isolation

Module 27: Docker & Kubernetes

- Docker images
- Docker Compose
- Containers
- Kubernetes basics
- App scaling

Module 28: CI/CD Pipelines

- DevOps concepts
- GitHub Actions
- Automated testing
- Build pipelines
- Deployment automation

Module 29: Security & Payments

- API security hardening
- Rate limiting
- Secure headers
- Payment gateways
- Transaction handling

Module 30: Performance & Capstone

- Caching with Redis
- Load optimization
- Scalability strategies
- Final MERN application
- Industry-grade delivery