



ASKME
— Training —



MASTERS IN MERN FULL STACK DEVELOPMENT

Chase your Dream Companies

THE FUTURE IS TECHNOLOGY

www.askmetraining.in |
info@askmetraining.in

The Program in MERN Full Stack Development covers many important subjects, provides you with live industry-based projects, and offers valuable practical experience. This in-depth MERN Full Stack Development course will help you master all the key principles of MERN Full Stack Development and develop all the essential skills required to become a professional. After joining this program, you will get all the required course material that will help you understand the core concepts of MERN Full Stack Development. Moreover, you will get the best learning experience equipped with live virtual classrooms and professional trainers. Therefore, by joining this program, you will get a chance to learn about MERN Full Stack Development technology and implement effective solutions to enhance the performance of an organization.



Career Services
By Askme



Duration
180 Hours



Learning Format
Online

Hottest **Job of 21st Century**



1.1 Million Job Postings

There is a global estimate of millions job postings for MERN Full Stack Development roles.



Skill Development

MERN Full Stack Development professionals are equipped with various relevant skills for lucrative job offers



Growing MERN Full Stack Industry

35.5% CAGR in the global MERN Full Stack Development industry.



Future- Oriented Career

MERN Full Stack Development is a budding field; a head start will prove to be beneficial.



Popular Degree

43% of MERN Full Stack Development professionals have a Master's degree.



High Demand

By 2025, India and US will face a demand supply gap of 460,000 MERN Full Stack Development professionals.

Program **FEATURES**



Key **Highlights**

- ✔ Instructor - Led Training
- ✔ Project & Exercises
- ✔ Flexible Schedule
- ✔ Mentor Support
- ✔ 1 - 1 with industry mentors
- ✔ Training Certification
- ✔ Job Assistance
- ✔ Life Time Free Upgrade
- ✔ Guaranteed Interviews

Top Skills & **Tools Covered**



Learning CURVE



Who Should Enroll in **this Program?**

Any technical degree or equivalents such as B.tech, M.tech, a degree in engineering, bachelor/master's in computer science, and basic programming knowledge.

- ✔ This program caters to a wide audience, from those who are hoping to enter the industry
- ✔ Fresh graduates who are intent on taking the plunge into the job market
- ✔ Developers who are working in one of the functional roles of front-end or back development and want to shift to full stack development
- ✔ Test engineers, system engineers, and others who want to make a career shift to

MERN Full Stack **Training Curriculum**

✔ **Module 1 : HTML**

- **HTML**
 - What is HTML?
 - What is a Web Browser?
 - What are Versions of HTML?
 - What can you Do with HTML?
 - HTML Development Environments
 - Writing Code with a Text Editor
- **Review of HTML Elements**
 - Rules of Syntax
 - Making your Code Readable
 - Building a Document
 - Using Colors
 - Adding Color to your Page
 - Using Headings
 - Using Paragraphs
 - Aligning Block–Level Elements
- **Inserting Spaces and Line Breaks**
 - Displaying Preformatted Text
 - Formatting with Inline Elements
 - Controlling Fonts
 - Introducing List Elements
 - Creating Unordered Lists
 - Creating Ordered
- **What is an HTML Table?**
 - Building a Table
 - Cell Padding and Cell Spacing
 - Controlling Table and Cell Width
 - Aligning a Table on the Page
 - Aligning Tables and Text
 - Aligning Table Data
 - Spanning Columns and Rows

- **Creating a Hyperlink**
 - Understanding and Using URLs
 - Linking to a Web Document
 - Linking to a Local Document
 - Linking to Anchors
 - Opening a New Browser Window
- **Image Formats**
 - Inserting Inline Images
 - Aligning Images
 - Using Images to Anchor Links
 - Sizing Images
 - Using Transparent Images
 - Using GIF Animation
- **Forms and Controls**
 - Forms and Form Elements
 - Form Actions, Form Methods, Form Design

HTML 5

- **Introduction**
 - Laying out a page with HTML5
 - Page Structure
 - New HTML5 Structural
 - Page Simplification
- **HTML 5 – How we got here?**
 - New Features of HTML5
 - The HTML5 Semantic Element
 - Current State of Browser Support
- **SECTIONS AND ARTICLES**
 - The section Tag
 - The article Tag
 - The header Tag
 - The Footer Tag

- HTML5 AUDIO AND VIDEO
 - Supported Media Types
 - The audio Element
 - The video Element
- HTML5 FORMS
 - New Input Types
- HTML5 NEW FORM ATTRIBUTES
 - autocomplete
 - no validate
- HTML5 NEW FORM FIELD ATTRIBUTES
 - required
 - placeholder
 - autofocus
 - autocomplete
 - form
 - pattern
- HTML 5 – How we got here?
 - New Features of HTML5
 - The HTML5 Semantic Element
 - Current State of Browser Support
- SECTIONS AND ARTICLES
 - The section Tag
 - The article Tag
 - The header Tag
 - The Footer Tag

✔ **Module 2 : CSS 2.0**

- Introduction of CSS
- CSS Syntax
- CSS Comments
- CSS Types
 - Inline
 - Internal
 - External

- **CSS Selector**
 - ID
 - Class
 - Attribute
 - Grouping
 - Universal

- **CSS Color**
 - RGB Value
 - Hex Value
 - Color Name

- **Background**
 - background-color
 - background-repeat
 - background-attachment
 - background position
 - background-size
 - background-image

- **CSS Margin**
 - Margin-top
 - Margin-bottom
 - Margin-left
 - Margin-Right

- **CSS Padding**
 - Padding -top
 - Padding -bottom
 - Padding -left
 - Padding -Right

- **Outline**
 - Outline-style
 - Outline-color
 - Outline Width
 - Outline-Offset
 - Outline Shorthand Property

- CSS Height and Width
- CSS Display properties
- CSS Position Properties
- CSS Overflow
- CSS Float and Clear
- Pseudo Class and Element

CSS 3.0

- Introduction to CSS 3
 - Border
 - border-radius
- CSS Shadows
 - Text-shadow
 - Box-shadow
- Transitions
 - transition
 - transition-delay
 - transition-duration
 - transition-property
- 2D Transforms
 - transform
 - matrix ()
 - translate (x, y)
 - scale (x, y)
 - rotate(angle)
 - skew (x-angle, y-angle)
- Animations
 - @keyframes
 - animation
 - animation-direction
 - animation-duration
 - animation-name
- Selectors
 - CSS combinations
 - Pseudo Elements

- **Gradients**
 - Linear Gradients
 - Radial Gradients
- **User Interface**
 - `resize`
 - `box-sizing`
 - `outline-offset`
- **CSS Filters**
 - Blur
 - Opacity
- **Media Query**
 - What is Responsive Web Design
 - Intro to the Viewport
 - The Viewport Tag
 - Media Queries
 - Tablet Styles
 - Mobile Styles
 - Making a Mobile Drop-down Menu
- **Web Fonts**
 - `@font-face`
 - `font-family`
 - `src`
 - `font-stretch`
 - `font-style`
 - `font-weight`
- **Flexbox**
 - `flex-grow`
 - `flex-shrink`
 - `flex-basis`
 - `flex`
 - `flex-wrap`
 - `flex-direction`
 - `flex-flow`
 - `justify-content`
 - `align-items`
 - `order`

✔ **Module 3 : Bootstrap Framework Latest Version (HTML, CSS, and JS Library)**

- **Introduction**
 - Introduction of Bootstrap
 - What needs for Bootstrap
 - Our First Web Page
 - What's Inside?
 - Scripts and Styles
 - Bootstrap CDN Link
 - Summary
- **Layout with Bootstrap**
 - Introduction
 - Grid Layouts
 - Simple Layout
 - Fixed Grids
 - Responsive Design
 - Responsive Utilities
 - Summary
- **Transitions**
 - transition
 - transition-delay
 - transition-duration
 - transition-property
- **Everyday Bootstrap**
 - Introduction
 - Typography
 - Tables
 - Forms
 - Buttons
 - Images and Icons
 - Summary
- **Bootstrap Component**
 - Introductions
 - Drop down Menu

- Buttons with Menus
- Tabs and Pill
- The Navbars
- Badges
- Summary
- **Bootstrap and JavaScript**
 - Introduction
 - More Buttons
 - Modals
 - Tooltips and Popovers
 - Carousel
 - Summary

✔ **Module 4 : JavaScript**

- History of JavaScript
- Introduction to JavaScript
 - Syntax
 - Statements
 - Comments
- Enabling in various browsers Popup Boxes
 - Alert
 - Confirm
 - Prompt
- Variables
- Operators
 - Arithmetic
 - Assignment
 - Comparison
 - Logical
 - Relational
 - Ternary
- Conditional Statements
 - If else
 - if...else if...else
 - nested if
 - Switch

- Loops
- While
- do...while
- for
- for...in Statement
- Break
- Continue
- **Functions**
 - User-defined Functions
 - Function Syntax
 - Function with Arguments
 - Returning Values from Functions
 - Built-in Functions
- **Events**
 - Introduction
 - Mouse
 - Events
 - Keyboard Events
 - Form Events
 - Document/Window Events
- **Built-in Objects**
 - Number
 - Strings
 - Math
 - Array
 - Date
 - Reg exp use in form validation

JavaScript with ECMAScript

- **Introduction to JavaScript Framework**
 - What is ES6 (ECMAScript 6/JavaScript 6)
 - ES6 Module System
 - A Word on Babel
 - Block Scope, Let & Const
 - Template Literals
 - Arrow Functions

- Spread and Rest Operators
- Object Literal Improvements
- De-structuring
- Classes
- Inheritance
- Static Properties and Methods
- Promises
- Iterators and Iterables
- Generators
- Modules
- New Features in ES6
- JavaScript let
- JavaScript const
- JavaScript Arrow Functions
- JavaScript Classes
- Default parameter values
- Array.find()
- Array.findIndex()
- Exponentiation (**) (ECMA Script 2016)

✔ **Module 5: JQuery**

- Getting Started with jQuery
- Selecting Elements
- Manipulating the Page
- Traversing the DOM and Chaining
- Handling Events
- Enhancing with Animation Effects
- JQuery Hide/show
- jQuery toggle()
- jQuery Slide method
- jQuery Fade method
- jQuery stop()
- jQuery Callback and Chaining
- jQuery text(),html() and val()
- jQuery css()
- jQuery addClass(),removeClass(),toggleClass()
- jQuery Best Practices

✔ **Module 6: React.JS** (An Open-Source JavaScript library for Building User Interfaces)

- **Introduction of React.JS**
 - Audience
 - Pre-requisites
 - About React
 - Features
 - Advantages
 - Limitations
- **Environment Setup**
 - Create of Root Folder
 - Install Global Packages
 - Add Dependencies and Plugins
 - Create the Files
 - Set Compiler, Server and Loaders
 - html
 - JSX and Main.js
 - Running the Server
- **JSX**
 - What is JSX?
 - Using JSX
 - Nested Elements
 - Attributes
 - JavaScript Expressions
 - Styling
 - Components
 - Naming Convention
- **Components**
 - Stateless
 - State full
- **State**
 - What is State?
 - Props

- **Props Overview**
 - Using Props
 - Default Props
 - State and Props
- **Props Validation**
 - Validating Props
- **Component API**
 - Set State
 - Force Update
 - Find DOM Node
- **Component Life Cycle**
 - Lifecycle Methods
- **Forms**
 - Simple
 - Complex
- **Events**
 - Simple
 - Child
- **Refs**
 - What is Refs?
 - Using Refs
- **Keys**
 - What are Keys?
 - Using Keys
- **Router**
 - Install a React Router
 - Add a Router
 - Create Components
- **Flux Concept**
 - What is Flux?
 - Flux Elements
 - Flux Props

- **Using Flux and REDUX**
 - Install REDUX
 - Creating Components
 - Working with States
 - Composing Components
 - Actions
 - Event Handlers
 - Reducers
 - Store
 - Root Component
 - Other Components
 - Unit Testing – Tools, React, REDUX

- **Animations**
 - Install React CSS Transitions Group
 - Add a CSS File
 - Appear Animation
 - Enter and Leave Animations

- **Higher-Order Components**
 - What is Higher-Order Component?

✔ **Module 7 : Node.JS: (Node.JS Coding Standards & Best Practices)**

- Introduction of CSS
 - Audience
 - Pre-requisites
 - About Node
 - Execute Node
 - Features
 - Who use Node
 - Concepts
 - Where to use
 - Where not to use
- **Environment Setup**
 - Text Editor
 - NodeJs Run Time
 - Download NodeJs
 - Installation
 - Executing
- **First Application**
 - Creating a NodeJs Application
 - Make a request to NodeJs Server
- **REPL Terminal**
 - What is REPL?
 - Starting REPL
 - REPL Commands
 - Stopping REPL
- **Packager Manager (NPM)**
 - Installing Modules using NPM
 - Global vs Local Installation
 - Using packages.json
 - Attributes of packages.json
 - Uninstalling Module
 - Updating Module
 - Searching Module
 - Create a Module

- **Call backs Concept**
 - Using Props
 - Default Props
 - State and Props
- **Event Loop**
 - Validating Props
- **Event Emitter**
 - Set State
 - Force Update
 - Find DOM Node
- **Buffers**
 - Lifecycle Methods
- **Streams**
 - Simple
 - Complex
- **File System**
 - Simple
 - Child
- **Global Objects**
 - What is Refs
 - Using Refs
- **Utility Modules**
 - What is Keys
 - Using Keys
- **Web Modules**
 - Install a React Router
 - Add a Router
 - Create Components
- **Express Framework**
 - What is Flux
 - Flux Elements
 - Flux Props

- Restful API
 - Install REDUX
 - Create Files and Folders
 - Actions
 - Reducers
 - Store
 - Root Component
 - Other Components
- Scaling Application
 - Install React CSS Transitions Group
 - Add a CSS File
 - Appear Animation
 - Enter and Leave Animations
- MongoDB
 - Connecting Node and MongoDB
 - Database Creation, Drop
 - Collection Operations
 - Documents Operations
- Application
 - Node and MongoDB Application

✔ **Module 8: Express JS – Web Application Framework**

- Express.JS Basics
 - Introduction, Advantages
 - Environment Setup
 - Basic Application
- Advanced Concepts of Express.JS
 - Request
 - Response
 - GET
 - POST
 - Routing
 - HTTP Methods
 - URL Building
 - Middleware

- Templates
- Static Files
- Form Data
- Database
- Cookies
- Sessions
- File Upload
- Authentication
- REST FUL API's
- Scaffolding
- Debugging

✔ **Module 9: MongoDB – No SQL Database**

- **MongoDB Basics**
 - Introduction, Advantages
 - History, Features
 - No SQL Databases
 - Advantages over RDBMS
- **Environment Setup**
 - Install MongoDB
 - MongoDB Shell
 - MongoDB Data Model
 - MongoDB Datatypes
- **Database**
 - Create Database
 - Drop Database
- **Collection**
 - Create Collection
 - Drop Collection
- **CRUD Documents**
 - Insert Documents
 - Update Documents
 - Delete Documents
 - Query Documents

- **Methods**
 - Limit ()
 - Sort ()
 - Skip ()

 **Module 10: Live Project**



REACH US:

Askme Training

G-01, Block H, Sector 63, Noida, Uttar Pradesh – 201301

+91-9582988773, 9999385943 | +91-0120-4198251

✉ info@askmetraining.in | 🌐 www.askmetraining.in



EXCELLENCE