

Node.js Training Objectives

All students will:

- Learn why server-side JavaScript is useful
- Install Node.js
- Learn how Node.js is architected to allow high scalability with asynchronous code
- Create basic web applications with Node.js
- Automate tasks with Gulp
- Build an HTTP server using the core modules in Node.js
- Use stream I/O to efficiently serve the web pages
- Create modules to organize the server
- Test the reliability of the application with unit tests
- Convert the application to an MVC framework using Express
- Interface to a MongoDB database and a web service

Node.js Training Outline

DAY 1

- Introduction
- Foundation
 - The Node.js framework
 - Installing Node.js
 - Using Node.js to execute scripts
- Node Projects
 - The Node Package Manager
 - Creating a project
 - The package.json configuration file
 - Global vs. local package installation
- HTTP
 - The HTTP protocol
 - Building an HTTP server
 - Rendering a response
 - Processing query strings
 - Using Representational State Transfer
 - Configuring TLS
- File System
 - Synchronous vs. asynchronous I/O
 - Path and directory operations
 - __dirname and __filename
 - Asynchronous file reads and writes
- Buffers, Streams, and Events
 - Using buffers for binary data
 - Flowing vs. non-flowing streams
 - Streaming I/O from files and other sources
 - Processing streams asynchronously
 - Configuring event handlers

DAY II :

- Modules and Unit Testing
 - Defining modules with exports
 - Modules are singletons
 - Creating a package
 - Module scope and construction
 - Unit testing frameworks
 - What to test and how to test it
 - Unit Testing (TDD/BDD)
 - Introduction to Testing
 - Writing Test Suites & Test cases
 - Unit Testing using Karma & Jasmine
- Working With Web sockets
 - Getting Started
 - Adding Socket.io To Your App
 - Exploring The Front-end
 - Sending Live Data Back & Forth
 - Creating The Front-end UI
 - Showing Messages In App

DAY III:

- Express
 - The model-view-controller pattern
 - Defining Jade and Handlebars templates
 - Building a front-end controller
 - Defining routes
 - Creating actions
 - Using REST
 - Reading POST data
- Authentication in NodeJS
 - Using Passport.js
 - Overview of using OAuth2
- Data Sources
 - Installing MongoDB
 - NoSQL Databases
 - How Node.js connects to MONGO DB databases
 - Connecting To MongoDB
 - Performing CRUD Operations
 - Moongose
- Building Microservices using NodeJS
 - Deploying it using Docker/Container
 - Using Different types of Load Balancing
 - Client vs Server side
 - CONSUMING SERVICE DISCOVERY INFORMATION
 - Overview of Using Ribbon
- Using Gulp
 - Using Gulp to automate tasks
 - Running Default & named tasks
- Conclusion