

COURSE CONTENTS



Course Agenda

Node JS

Duration – 3 Days

Adding
Value 

www.vinsys.in



Description

Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. As an asynchronous event driven JavaScript runtime, Node is designed to build scalable network applications. HTTP is a first class citizen in Node, designed with streaming and low latency in mind. This makes Node well suited for the foundation of a web library or framework. Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications. Mocha is a feature-rich JavaScript test framework running on Node.js. Chai is an assertion library for node.

Intended Audience

This course is intended for:

- Developers
- Tech Leads
- Architects

Course Objectives

In this course, you will learn how to:

- Create basic web applications with Node.js
- Build an HTTP server using the core modules in Node.js
- Test the reliability of the application using Mocha and Chai
- Create web application using express framework
- Build REST API
- Interface to a MongoDB database and MySQL

Prerequisites

We recommend that attendees of this course have the following prerequisites:

- Knowledge of HTML and JavaScript
- Basic Programming Knowledge

Hands-On Activity

This course allows you to test new skills and apply knowledge to your working environment through a variety of practical exercises.



Course Outline

This course covers the following concepts:

- Introduction
 - Synchronous Programming
 - Asynchronous Programming
 - Single Threaded vs. Multithreaded Applications
 - Understanding Callbacks
- ES6 Fundamentals
 - Default Parameters
 - Arrow Functions
 - Template Literals
 - Multiline String
 - Scoping using Let and Const Keywords
 - Spread Syntax and Rest Parameters
 - Destructuring
 - Promises, resolve(),reject()
 - Async / Await
 - Classes and Objects
- Node Basics
 - The Node.js framework
 - Why Node JS?
 - Advantages
 - Installing Node.js
 - Node.js REPL
 - Language Basics
 - Using Node.js to execute scripts
 - Parsing command line arguments
- Architecture
 - Chrome V8 Engine
 - Node.js Call Stack
 - Node.js Event Loop
 - Event Queue
- Node Projects
 - The Node Package Manager (NPM)
 - Creating a project
 - The package.json configuration file
 - Global vs. local package installation
- Node Modules
 - Built-in Modules
 - External Modules
 - Creating your own Modules
- Node Events



- Event Module
- EventEmitter Object
- File System, Buffers and Streams
 - Synchronous vs. Asynchronous I/O
 - Path and directory operations
 - Asynchronous file reads and writes
 - Using buffers for binary data
 - Streaming I/O from files and other sources
 - Processing streams asynchronously
- Debugging Node.js
 - Enable Inspector
 - Chrome Dev Tools
 - Chrome://inspect
 - Break Points
- URL Module
 - Parse URL
- HTTP Module
 - The HTTP protocol
 - Building an HTTP server
 - Rendering a response
 - Processing query strings
 - Using Representational State Transfer
- Mocha
 - Introduction
 - Installing Mocha
 - Using describe() and it()
- Chai
 - Introduction
 - Assertion library
 - Using expect, should and assert
- Express JS
 - Express Web Application Framework
 - Installation
 - Express Server
 - Routing
 - Route Parameters
 - Route Handlers
 - Response Methods
 - Serving Static Content
 - Middleware
 - app.use()
 - Application level middleware
 - Router-level middleware



- Error-handling middleware
- Built-in middleware
- Third-party middleware
- Template Engine
- Configuring Express to use Handlebars
- Data Sources
 - How Node.js connects to databases
 - RDBMS databases and NoSQL databases
 - Connecting to RDBMS and NoSQL databases
 - Performing CRUD operations
 - Using Mongoose

Related Training

Recommended Related Training Courses:

- Angular
- React

Lab Setup

- Microsoft Visual Code IDE
- Node.js
- Google Chrome Browser
- Desktop PC / Laptop with Windows O.S
- Internet connection
- MySQL
- MongoDB