

Introduction: Node.js is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine). Node.js was developed by Ryan Dahl in 2009 and its latest version is v0.10.36. The definition of Node.js as supplied by its official documentation is as follows –

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Course Objectives

- Understand the JavaScript and technical concepts behind Node JS
- Structure a Node application in modules
- Understand and use the Event Emitter
- Understand Buffers, Streams, and Pipes
- Build a Web Server in Node and understand how it really works
- Use npm and manage node packages
- Build a web application and API more easily using Express
- Connect to a SQL or Mongo database in Node
- Understand how the MEAN stack works

Pre-requisite / Target Audience:

- Basic Knowledge of JavaScript and OOPS
- Knowledge in async programming will be added advantage

Module 1:- Introduction to Node JS In this module, you learn What is Node JS and advantages of Node JS and How Node JS Works and the difference between the traditional web server and what are the limitations of the traditional web server modal.

- Introduction
- What is Node JS?
- Advantages of Node JS
- Traditional Web Server Model
- Node.js Process Model

Module 2:- Setup Dev Environment In this chapter, you will learn about the tools required and steps to setup development environment to develop a Node.js application and to work with the REPL to work with the Console.

- Install Node.js on Windows 10
- Installing in mac os

- Working in REPL
- Node JS Console

Module 3:- Node JS Modules In this chapter, you will know what is a module, functions and examples. And achieving modularity and separation of concern with the NodeJS Modules.

- Functions
- Buffer
- Module
- Module Types
- Core Modules
- Local Modules
- Module.Exports

Module 4:- Node Package Manager This deals with working with npm to install and update packages and Updating your package to the NPM and managing and updating your current Packages.

- What is NPM
- Installing Packages Locally
- Adding dependency in package.json
- Installing packages globally
- Updating packages

Module 5:- Creating Web server In this chapter we will learn creating web server, handling GET, POST,PUT and Delete requests and listening to certain port numbers and handling routing with basic web server.

- Creating web server
- Handling http requests
- Sending requests

Module 6:- File System In this module, we will work with files, reading, writing, updating files, and the concept of chunks, buffers, and uploading files synchronously and asynchronously.

- Fs.readFile
- Writing a File
- Writing a file asynchronously
- Opening a file
- Deleting a file
- Other IO Operations

Module 7:- Debugging Node JS Application In this chapter you will learn how to debug node js application. Debugging is a process of tracing the bugs and performance issues to optimize your code.

- Core Node JS debugger
- Debugging with Visual Studio

Module 8:- Events In this chapter you work with the events in Node JS, and the significance of the events, writing your own events because Node is event driven framework.

- EventEmitter class
- Returning event emitter
- Inhering events

Module 9:- Express.JS In this chapter you will learn how to use express framework to create web applications. Express is the most popular framework to build efficient web applications with minimum coding.

- Configuring routes
- Working with express

Module 10:- Serving Static Resources In this chapter you will learn how to serve static html pages to the browser, and serving other file formats and restricting certain files.

- Serving static files
- Working with middle ware

Module 11:- Database connectivity In this chapter you will learn how to connect to SQL Server and perform CRUD operations. It is like ADO.net for MS.NET applications and JDBC for java. We will use different Node JS modules to connect with database.

- Connection string
- Configuring
- Working with select command and Updating records
- Deleting records

Module 12:- Template Engines In this chapter you learn how use template engines to perform 2 way databinding and appending dynamic data to the webpage and different view engines and their syntax.

- Why Template Engine
- What is Jade
- What is vash
- Example

Module 13:-

Project involving most of the above concepts with following will be provided

- Product Abstract Document
- Requirement Specification Document
- Step-by-Step procedure for building the project from ground up
- Complete Source Code
- Database Script with Sample data
- Instructions to Setup the Project on a Development box
- Instruction to Deploy the project on Production Box / Microsoft Azure