



# Nodejs with Microservices

**Duration: 5 Days**

**Prerequisite:**

- Must have strong knowledge on java script – ES5,& ES6
- Must have knowledge on Microservices architectures
- Must have above 2 years programming experience

**Level : Intermediate to Expert**

**Software Requirement and Hardware Requirement Hardware Requirement:**

8 GB RAM Minimum

Windows 10

**Internet Connection**

Admin rights for installing node modules

Note : if you face problem during npm install phase, then you have to set admin rights and also proxy must be enabled.

**Proxy setup:**

Open an command prompt or terminal session and run the following commands to configure npm to work with your web proxy. The commands use proxy.company.com as the address and 8080 as the port.

npm config set proxy http://proxy.company.com:8080

**IDE**

1.Visual Studio Code

**Runtime**

Node - latest

NPM Modules:

Can be installed during Session Only.

**Web Browsers - Any one Browser - Recommended Chrome**

Chrome

**Tools -**

Chrome Plugin -POST Man

Docker

Docker Compose



## Day 01

### ES 6

- ✚ Variables , literals, types
- ✚ ES 5 vs ES6
- ✚ Es 6 Overview
- ✚ Let ,const, block scope
- ✚ String template literals
- ✚ Arrow functions
- ✚ Classes
- ✚ inheritance
- ✚ Instance variables, methods
- ✚ Object Destructuring
- ✚ Async programming and implementations
- ✚ Call back hell
- ✚ Promises
- ✚ Async and await
- ✚ ES 6 modules

### Typescript Getting Started

- ✚ What is typescript, Why Typescript
- ✚ Typescript compiler and Project Setup
- ✚ Static Typing vs Dynamic Typing
- ✚ Primitive types
- ✚ Object types
- ✚ Encapsulation
- ✚ Interfaces, abstract classes
- ✚ Decorators over view



## Node JS Architecture

### What is Node.js

- ✚ Node.js Architecture
- ✚ IO Demystification
- ✚ OS Kernal
- ✚ File Descriptor
- ✚ Blocking API
- ✚ Concurrency and Multi-Threading
- ✚ C10k Problems
- ✚ Nodejs Low level architecture
- ✚ Libuv
- ✚ Event Loop threads
- ✚ Workpool threads

### Nodejs Async Implementation

- ✚ Timers
- ✚ Callbacks
- ✚ Callback hell
- ✚ Promises
- ✚ Async await

## Day 02

### Nodejs Modules

- ✚ Module design patterns
- ✚ Common Js
- ✚ Require,exports,module,exports
- ✚ Npm
- ✚ Node\_modules
- Node built in modules



## **Node core**

- ✚ **Modules** Event emitter
- ✚ Operating system
- ✚ Process module
- Global variables

## **Node IO fs**

- ✚ File System IO
- ✚ Sync and async apis
- ✚ Streaming and Non Streaming api
- ✚ Back pressure

## **Http Programming**

- ✚ Http module
- ✚ Request, response objects
- ✚ Server Objects
- ✚ Request Handling
- ✚ JSON response

## **Express.js**

- ✚ What is Express js & Why Express js
- ✚ Setup Express app
- ✚ Understanding application object
- ✚ Working with Basic routes
- ✚ Sub Routers and Modularization
- ✚ Building REST Endpoints
- ✚ Middlewares
- ✚ Types of Middlewares
- ✚ Built in Middlewares
- ✚ Database Integrations- Mongo DB

## **SOAP Web Service Integration**

- ✚ What is SOAP Web Service
- ✚ Consuming SOAP Service using easy-soap-request

## **Unit Testing Node apps**

- ✚ Introduction to unit Testing
- ✚ Chai and mocha



✚ Writing test cases using chai and mocha

## Deployment and production node apps

- ✚ How to include production dependencies and exclude
- ✚ Application deployment on deployment.

## Day 03

### Micro services on Node.js using Molecular

- ✚ **Micro Services** What is Micro Service?
- ✚ Why Micro Services
- ✚ Distributed Application Architecture
- ✚ Domain Driven Design
- ✚ Monolithic to Micro services
- ✚ Advantages and Disadvantages
- ✚ Patterns
- ✚ **Molecular Getting Started** What is Molecular?
- ✚ Installing Molecular
- ✚ Creating Molecular Project

### Molecular Core Concepts











- ✚ Service
- ✚ Node
- ✚ Local Services
- ✚ Remote Services
- ✚ Service Broker
- ✚ Transporter
- ✚ Gateway
- ✚ Overall View
- ✚ Architecture
- ✚ Implementation
- ✚ Micro Service
- ✚ Implementation Create first Service
- ✚ Understanding Service broker
- ✚ Starting service broker
- ✚ Promises and broker









Async and await way of starting services  
Invoking services

## Day 04







### Advanced Service

-  **Creation** Service methods
-  Grpc Service
-  Types of methods
-  Public and private methods
-  Life cycle methods
-  Parameters
-  Parameter validation
-  Internal services
-  Monitoring internal services
-  Single service
- Multi Services
- Calling multiple services

### Service Interaction: Networking

-  Local Service
-  Remote Service
-  Remote Service calls
-  Networking configuration
-  TCP transporter
-  NATS Transporter
- Redis Transporter










### Service Registry & Discovery

-  Dynamic service discovery
-  Local
-  Redis
-  etcd3
-  Customization
-  Built-in Service Registry














## Day 05








### Load Balancing

-  Built-in strategies
  -  RoundRobin strategy
  -  Random strategy
  -  CPU usage-based strategy
  -  Latency-based strategy
  -  Sharding strategy
  -  Overwrite global options
  -  Custom strategy
  -  Create custom strategy
- Use custom strategy



### Fault tolerance

-  Circuit Breaker
  -  Settings
  -  Retry
  -  Settings
  -  Timeout
  -  Distributed timeouts
  -  Bulkhead
  -  Global Settings
  -  Action Settings
  -  Events Settings
  -  Fallback
- Fallback in action definition

### Metrics

-  Metrics Reporters
  -  Console
  -  CSV
  -  Event
  -  Datadog
  -  Prometheus
  -  StatsD
- Supported Metric Types

### Deploying

-  Docker
- 

Kubernetes