

NodeJS (NPM, Express)

Duration: 3 days

Objectives:

- Learn why server-side JavaScript is useful
- Install Node.js
- Learn the core concepts of Node, explore and use the built-in API's
- Learn how Node.js is architected to allow high scalability with asynchronous code
- Learn the Node Package Manager concepts
- Build RESTful API using Node and Express
- Create real-time applications using SocketIO
- Integrate with Databases like MongoDB and MySQL

Prerequisites:

- All participants should have experience developing Web Applications
- Participants should have experience in JavaScript Development
- Understanding of Web Services would be an added advantage

Prerequisites tutorial

<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

Course Summary : Node.js is a powerful tool for backend JavaScript development. In this course we will cover all topics required for creating a server-side web application exposing a set of REST endpoints using Node, Express, SocketIO, MongoDB and modern JavaScript. The course starts with the Node concepts, built-in libraries, using the Node Package Manager and moves on to understanding asynchronous programming for building scalable applications. Later we cover creating web applications using Node Express, building real-time applications using WebSockets using SocketIO and working with databases like MongoDB and MySQL. We finally look into the aspects of security and deployment.

Lab Setup Details.

Software Requirements

1. Node 12.16.3

2. Visual Studio Code (<https://code.visualstudio.com/>)
3. MongoDB
4. WinRar or 7-zip
5. Acrobat Reader
6. Latest Browsers: Chrome, Firefox

Hardware

1. 8 GB RAM Minimum
2. 40GB Free HD Minimum
3. LCD Projector connected to trainer machine
4. **All the machines with internet connections with at least 20mpbs speed**

Note: If the company has a private NPM registry or Proxy to access the internet, ensure the details are updated in the .npmrc file.

Please contact the L & D execute or respective SME for details.

Contents

Introduction

- The Node.js framework
- Installing Node.js
- Using Node.js to execute scripts
- Node Architecture
- Global Object and Process
- Modules
- Using the require() function

Node Projects

- The Node Package Manager
- Creating a project
- The package.json configuration file
- Global vs. local package installation

Built-In Libraries

- Working with the Operating System
- Working with the File System
- Console and Utilities
- Debugging

Interacting with File System

- Listing and reading directories and files
- Writing to files
- Reading and writing from file stream
- Using buffers for binary data
- Flowing vs. non-flowing streams
- Streaming I/O from files and other sources
- Processing streams asynchronously
- Configuring event handlers

Concurrency and Events

- I/O Operations
- The Event Loop
- Call Stack
- Callbacks, Promises
- Event Emitter

NPM

- What Is a Package
- Typical NPM Usage
- NPM Help
- Creating a Package.json
- Setting Defaults
- Installing Packages
- Listing Installed Packages
- Installing Global Packages
- Removing a Package
- Installing Specific Versions
- The NPM Registry
- Searching for Packages
- Pruning
- Using Simple Scripts
- Publishing Your Package

HTTP

- The HTTP protocol
- HTTP Module
- Making a request and response mechanism
- Build a web server
- Serving to files
- Serving to Json Data
- Collection post data from web pages

- Scaling the application

Working with Web Sockets

- Creating a web socket server
- Broadcasting message with web sockets
- Creating web sockets with Sockets.io.
- Realtime interaction with Socket.io

Express

- The model-view-controller pattern
- Defining Jade and Handlebars templates
- Building a front-end controller
- Defining routes
- Creating actions

REST

- Understanding the principles of REST
- Richardson Maturity Model
- Creating a RESTful Web Service (CRUD based) using Node Express
- State management options in REST
- Securing REST services

Data Sources

- How Node.js connects to databases
- RDBMS databases and NoSQL databases
- Connecting to RDBMS and NoSQL databases
- Performing CRUD operations

Deployment

- Cloud Deployment