Master in JavaScript with AngularJS and NodeJS

About
DevOpsSchool

DevOpsSchool is a unit of "Cotocus PVT ltd" and a leading platform which helps IT organizations and professionals to learn all the emerging technologies and trend which helps them to learn and embrace all the skills, intelligence, innovation and transformation which requires to achieve the end result, quickly and efficiently. We provide over 40 specialized programs on DevOps, Cloud, Containers, Security, AI, ML and on Big data that are focused on industry requirement and each curriculum is developed and delivered by leading experts in each domain and aligned with the industry standards.

About Course

JavaScript is the most popular programming language in the world. It powers the entire modern web. It provides millions of high-paying jobs all over the world. JavaScript's abilities in the browser are limited for the sake of the user's safety. The aim is to prevent an evil webpage from accessing private information or harming the user's data. JavaScript can easily communicate over the net to the server where the current page came from. But its ability to receive data from other sites/domains is crippled. Though possible, it requires explicit agreement (expressed in HTTP headers) from the remote side. Once again, that's a safety limitation.

This course was developed to teach you the key JavaScript syntax and features you'll need to learn as a successful web developer. Some of the major topics that we will cover include the latest JavaScript syntax improvements, organizing your code into classes and modules, communicating with web services, error handling and security, and much more. By the end of this course, you'll know how to set up a JavaScript development environment and create great JavaScript applications for the web.

Co-coordinator – Akanksha Kumari

Call/WhatsApp: - +91 1800 889 7977

Mail Address: -

contact@DevOpsSchool.com

Secondary contact – Patrick

Call/WhatsApp: - +91 7004 215 841

Mail Address: -contact@DevOpsSchool.com

Duration	80 Hours
Mode	Online (Instructor-led, live & Interactive)
Projects (Real time scenario based)	1

FEATURES	DEVOPSSCHOOL	OTHERS
Faculty Profile Check	✔	✖
Lifetime Technical Support	✔	✖
Lifetime LMS access	✔	✖
Top 25 Tools	✔	✖
Interviews Kit	✔	✖
Training Notes	✔	✖
Step by Step Web Based Tutorials	✔	✖
Training Slides	✔	✖
Training + Additional Videos	✔	✖

AGENDA OF THE MASTER IN JAVASCRIPT WITH ANGULARJS AND NODEJS

Introduction - JavaScript

- Why Study JavaScript
- Application and Feature of Java Script
- Java Script Code Editor with Example
- Java Script Output
- Java Script Placement

Java Script Variable

- Variable Declaration Syntax
- Expression, Keyword. Identifier
- Operator:Arithmetic,Assignment,String,Comparision,Logical,Type,Bitwise
- Ternary Operator.
- Data Type:String,Number,Boolean,Array,Typeof

Decision Control and Loop Structure

- Condition Statement
- Switch Statement
- Loop Structure: While, do-while, For.
- Control Flow: Break,Continue,goto Statement

Function

- Function Definition, Prototype and Types.
- Anonymous and Callback Function
- Call by Value and Call by Reference
- Apply and Bind Method()
- Function Scope: local and global
- Recursive Function

Array and String

- Creating and Accessing Array Elements
- Array Type: Single, Associative, Multidimensional.
- Array Method For Sorting and Reversing:
- concat(),filter(),foreach(),reverse(),join(),splice(),sort(),toString().
- String Properties and Method

Java Script Object and Class

- Object Properties and Method
- Accessing and Displaying the Object
- Primitive and Reference Type and Primitive Wrapper Type
- Class ,Constructor and new Operator
- Inheritance
- Getter and Setter Method
- Static Class

Interfaces

- Using Interfaces
- Demo: Using Interfaces
- Extending an Interface
- Demo: Extending an Interface

Modules

- Overview
- Identifying a Module
- Creating an Internal Module
- Internal Module Accessibility and IIFE
- Named Modules
- Extending Modules and Importing Shortcuts
- Organizing Internal Modules
- Separating Internal Modules
- External Modules and Dependency Resolution
- Module Dependencies
- Importing External Modules Using AMD
- Importing 3rd Party Libraries Using AMD
- Modules Recap

Java Script Regular Expression

- Object Properties and Method
- Accessing and Displaying the Object
- Primitive and Reference Type and Primitive Wrapper Type
- Class ,Constructor and new Operator
- Inheritance
- Getter and Setter Method
- Static Class

Java Script Error

- Handling Error Using Try and Catch Statement
- Error Object Properties and Error Name value.
- Regular Expression and Pattern.

JS Dom Method and Element

- Dom Structure.
- Dom method and properties with Programming Interface
- Finding Html Element By: id, tag name, classname, css selector, object collection.
- Changing Html Content, value of attribute and Style

Java Script DOM Event and Event Listener

- Event Type: Keyboard, Form, Document, Click Event
- AddEventListener Method and RemoveEventListener
- Adding Single and Many Event handler to element and window object

Java Script DOM Nodes and Navigation

- Retrieving Node Element Using Node Properties
- Node Value and Type Properties
- Creating, Removing, Replacing HTML Node Element:append, remove and replace method

Java Script DOM Collection and Node List

- HTML Collection Object and Length
- HTML NodeList Object
- Difference between an HTMLCollection and a Node List

Web ApI Introduction and Web History API

- What is Web API
- Browser API and Third-Party API
- History Object and Properties:back, forward and go method();

Web Storage API

- Web Storage Concept and Usage
- Local Storage Object : (i)How to saving data to Local Storage,(ii)How to reading and removing data from local Storage
- Session Storage Object : (i)How to saving data to Session Storage,(ii)How to reading and removing data from Session Storage
- Storage Object Properties and Method

Web Fetch API

- Sending a Request: uploading single and Multiple File
- Reading Response
- Handling Status Code of Response
- Java script Fetch API Example

Geolocation Web API

- Concept and Usage
- Interfaces
- Geolocation Object Properties and Method

Introduction – AngularJS

- Course Introduction
- Angular Architecture
- Demo: Hello World in Angular
- The Angular Event Reg Application
- Angular Version
- Tools Used
- Summary

Angular Controllers & Markup

- Introduction
- Demo: Creating the Node Server
- Demo: Creating the ASP.NET Server
- Angular Controllers & Scope
- Demo: Controllers
- Demo Handling Events
- Built-in Directives
- Event Directives
- Other Directives - Part 1
- Other Directives - Part 2
- IE Restrictions with Angular
- Expressions
- Filters
- Built-in Filters
- Writing Custom Filters
- Two Way Binding
- Demo: Two Way Binding
- Validation

Creating and Using Angular Services

- Introduction to Angular Services
- Creating Your First Custom Service
- Another Custom Service Example
- Introduction to Built-In Angular Services
- Using Angular's $http Service
- Updating a Node Server for JSON Requests
- Updating an IIS Server for JSON Requests
- Using $http Promises
- Using Angular's $resource Service
- Saving Data with $resource
- Using Angular's $anchorScroll Service
- Using Angular's $cacheFactory Service
- Using Angular's $compile Service
- Using Angular's $parse Service
- Using Angular's $locale Service
- Using Angular's $timeout Service
- Using Angular's $exceptionHandler Service
- Using Angular's $filter Service
- Using Angular's $cookieStore Service
- Overview of Less Common Angular Services

Angular Routing

- Introduction to Angular Routing
- Websites of Yore
- Single Page Applications
- Adding Your First Angular Route
- More Routing and Browser History
- Creating a Default Route
- Accessing Parameters from the Route
- Using Angular's $route Service
- Enabling HTML5 Routing
- Template and Resolve Properties
- Using Angular's $location Service

Creating Custom Angular Directives

- Introduction to Angular Directives
- Creating Your First Angular Directive
- Domain-Specific Language via Custom Elements
- Isolating Directive Scope
- Exploring Isolate Scope Bindings
- Handling Events with Directives
- Observing and Responding to Changes
- Using Controllers within Directives
- Sharing Directive Controllers via Require
- Directive Priority and using Terminal
- Using Require with Nested Directives
- Understanding Transclusion
- Using Compile to Transform the DOM
- Making jQuery More Explicit with Directives
- Summary

Testing Angular

- Introduction
- Installing Karma
- Karma with Webstorm
- Testing Controllers in Angular
- Testing Controllers in Angular Part 2
- Testing Services in Angular
- Testing Services with Dependencies
- Testing AJAX Services in Angular
- Testing AJAX Services in Angular Part 2
- Testing Filters in Angular
- Testing Directives - Overview
- Setting up Karma for testing Directives
- Testing Directives in Angular
- End to End testing - Overview
- Setting up Protractor
- Setting up Protractor with Webstorm
- Angular End to End testing - Part 1
- Angular End to End testing - Part 2
- Summary

Introduction - NodeJS

- Course Introduction
- What Is Node?
- Why Node?
- Some Analogies About Node
- What You Get When You Install Node
- Example Files

Getting Started with Node

- Node's REPL Mode
- TAB and Underscore
- Executing Scripts
- Working with Timers
- Node's Command Line Interface
- The "process" Object
- Wrap Up

Modern JavaScript

- EcmaScript and TC39
- Variables and Block Scopes
- Arrow Functions
- Object Literals
- Destructuring and Rest/Spread
- Template Strings
- Classes
- Promises and Async/Await

NPM: Node Package Manager

- What Exactly Is NPM?
- The NPM Command
- The package.json and package-lock.json Files
- Semantic Versioning (SemVer)
- Installing and Using NPM Packages
- Creating and Publishing an NPM Package
- NPX and the NPM Run Scripts
- Updating NPM Packages

Modules and Concurrency

- Introduction
- Defining and Using Node Modules
- Examples of Module APIs
- Node's Global Object
- The Event Loop
- Errors vs. Exceptions
- Node Clusters
- Node's Asynchronous Patterns
- Event Emitters

Working with Web Servers

- Hello World... The Node's Version
- Monitoring Files for Changes
- The "req" and "res" Objects
- Node Web Frameworks
- Using Template Languages

Working with the Operating System

- Introduction
- The os Module
- The fs Module
- The child_process Module
- Debugging Node Applications

Thank you!

Connect with us for more info

Call/WhatsApp: - +91 968 682 9970

Mail: -

contact@DevOpsSchool.com

www.DevOpsSchool.com