**NBITS**

**(N Benchmark IT Solutions Pvt. Ltd.)**

**Meanstack Course Content**

**Ph No: 9701000415, 8977702468**

**#102, SK Enclave, SAP Street, Behind Maitrivanam, Ameerpet, Hyderabad.**

**JAVA SCRIPT**

**WHAT IS JAVA SCRIPT?**

* **Programming with Java script.**
* **Loops in Java script and conditional statements.**
* **Ternary operators.**
* **What is Document Object Model**
* **What are the various features of DOM**
* **DOM Selectors**
* **DOM Events**
* **Programming with Java script and DOM**
* **Understanding window object**
* **Lexical Scope and the Global Scope**
* **Best practices with Java script**

**BASIC JS**

* **Variables**
* **Comments**
* **Operators**
* **Functions**
* **Conditional Statements**
* **Events**
* **Control Flow or Error Handling**
* **Loops & Iterations**
* **Number & Dates**
* **Text Formatting**

**ADVANCED JS (OBJECT ORIENTED JS)**

* **Working with Objects in Javascript**
* **Understanding Public Private and Static**
* **Prototypal Inheritance**

**PROTOTYPES**

* **Understanding Inheritance**
* **Static Properties**
* **Public / Private Properties**

**CLOSURES**

* **Immediate Invocable Functions (IIF)**
* **Static Mode**
* **Call Backs**

**J QUERY**

**SELECTORS**

* **Class Based Selector**
* **ID Based**
* **Element Based**
* **Hybrid Selector**

**HANDLING AJAX REQUEST**

* **Get**
* **Post**
* **Calling Rest Full APIs**

**TWITTER BOOTSTRAP**

* **What is Bootstrap**
* **Working with Bootstrap components.**
* **Understanding Grid Layout**
* **Creating Different Form Layouts**
* **Creating Navigation Bars**
* **Understanding glyphicons**

**DESIGN PATTERNS**

* **Understanding MV \* pattern**
* **Understanding dependency injections(DI Pattern)**
* **Solid principles**

**ANGULAR 2.X + TYPESRCIPT**

**GETTING STARTED WITH TYPESCRIPT**

● Installing TypeScript

● Compiling the code

● Building a simple demo.

**UNDERSTANDING CLASSES**

● Building a class

● Adding properties

● Demo of using classes in typescript

**UNDERSTANDING INHERITANCE AND INTERFACES**

● What is Inheritance

● How to inherit the classes

● Understanding interfaces

**WORKING WITH MODULES**

● What are Modules in Typescript

● When to use modules

● Demo on building modules.

**GENERICS**

● What are Modules in Typescript

● When to use generics

● Demo on working with generics.

**INTRODUCTION TO ANGULAR -2.X**

● What is angular 2.X

● Working with our build tool

● Setting up out template

**INTRODUCTION TO THE BASICS**

● Creating a simple component

● Using multiple modules

● Understanding template types

● Displaying data in our templates

● Working with events

● Using properties

● Using two-way data binding

● Adding CSS to our component

**ARCHITECTURE OVERVIEW**

● Components, Bootstrap, and the DOM

● Directives and pipes

● Data binding

● Dependency injection

● Services and other business logic

● Data persistence

● Routing

**CREATING A BASIC MODULE**

● Using more complex data

● Creating a subcomponent

● Using multiple subcomponents

● Cleaning up components

● Filtering content through data pipes

**COMPONENTS**

● Component metadata

● The component selector

● The component template

● Styling a component

● Using other components in a component

● Interpolation and the expression context

● Property binding

● Event binding

● Getting data to the component with input

● Subscribing to component events with output

**DIRECTIVES AND PIPES**

● Structural directives—ng-if

● Structural directives—ng-for

● Attribute directives—built in

● Attribute directives—custom

● Using directive values

● Working with events in directives

● Angular pipes—built in

● Angular pipes—custom

**FORMS**

● Angular forms

● Template-driven forms

● Model-driven forms

● Validation--built in

● Validation—custom

● Error handling

**DEPENDENCY INJECTIONS AND SERVICES**

● How Angular does dependency injection

● Services in angular

● Class constructor injection

● Building a service

● Provider registration at bootstrap

● The inject decorator

● The opaque token

**HTTP**

● The angular 2 HTTP bundle

● Using a mock back end for HTTP calls

● Using HTTP for GET calls

● Using Url Search Params

● Using HTTP for POST, PUT and DELETE calls

**ROUTING**

● The Angular 2 routing bundle

● Route configuration

● Router outlets

● Router links

● Using the router class to navigate

**ANGULAR JS 4.0**

Difference between Angular-1.X and Angular -4.0

Understanding component architecture

Dependency injection design pattern

**NG SET UP**

* Initializing Angular CLI.
* Initializing Web page.
* Set up VS code.
* Adding required templates to VS code.
* Set up GIT LAB

**CREATING ANGULAR APPLICATION**

* Understanding setup files
* Understanding Architecture of Application
* Building Hello World

**CREATING THE GATHERING PROJECT REQUIREMENT**

**BUILDING THE COMPONENT**

* Understanding the decorators
* Understanding the Selectors
* Understanding the Providers
* Referring the Component in the Application
* Adding Component to the module

**BUILDING THE MODULE**

* Understanding NG Module
* Declarations
* Imports
* Exports
* Providers

**REFERRING A MODULE IN ANOTHER MODULE**

**COMPONENT LIFE CYCLE**

* Understanding Component life cycle Hooks
* Understanding various Interfaces
* Consuming classes in component
* Consuming Interfaces in Component
* Consuming Component in a component

**RX JS - REACTIVE PROGRAMING**

* Understanding what is reactive programming
* What are Observables
* What is a Subject
* Map Method

**WORKING WITH RXJS WITH ANGULAR 4.0**

**REST API CALLS**

* HTTP Calls with Observables
* Error handling
* Streaming
* Integrating with third party API"s

**BUILD IN DIRECTIVES**

* \* ngFor
* \* ngIf , Else
* Show and hide elements
* Rendering Images in Angular 4.0
* Rendering Tables in Angular 4.0

**PROVIDERS**

* How to Use Providers
* Communicate with Component
* Working with RXJS in Providers

**CREATING PIPES**

* Understanding Build in pipes
* Creating Custom pipes

**ADVANCES ANGULAR 4.0**

* Working with Routing
* Authenticating and Authorization
* Component to component communication
* Understanding input output methods
* Understanding Subjects
* Understanding Event emitters

**SETTING UP REVERSE PROXY WITH WEBSERVER**

* Page nation
* Calendar
* More

**ANGULAR JS 5.0 FEATURES**

**NODE JS CONTENT**

**INTRODUCTION TO NODE JS**

* **Creating first Node JS app**
* **Understanding NPM**
* **Creating and locating Modules**
* **Specifying dependencies with package. Json**
* **Understanding CPS (Continuation Passing Style)**
* **Understanding the Event Loop**

**NPM  AND CPS**

* **Understanding NPM**
* **Creating and locating Modules**
* **Specifying dependencies with package. Json**
* **Understanding CPS (Continuation Passing Style)**
* **Understanding the Event Loop**

**BUILD APP WITH NODE AND EXPRESS WITH MVC ARCHITECTURE**

* **Creating a simple HTTP server**
* **Understanding and installing Express JS**
* **Creating a basic Express JS app**
* **Understanding Routing in Express JS**

**MONGODB**

* **Connecting to Mongoose**
* **Working with Mongodb**
* **CRUD Operations**
* **Search Operations**
* **Working with Mongodb in Nodejs**
* **Understanding Mongoose**
* **Securing passwords and SSN by encryption**

**EXPRESS JS WITH MIDDLEWARE**

* **Error Handling by using logger**
* **Handling sessions**
* **Ejs templating engine.**
* **Working with public and private content.**

**● PASSPORT AUTHENTICATION**

* + **Local**
  + **Google**
  + **Facebook**
* **Using parameterized routes**
* **Rending view using Express JS**
* **Session Handling ● Working with Multer**
* **Working with Morgan**

**WORKING WITH THIRD PARTY API’s**

* **Working Open Api’s Like Walmart.**
* **Working with Google Api.**

**EVENTS AND STREAMS**

* **Understanding streams**
* **Readable and Writable streams**
* **Piping streams**
* **Understanding Events**
* **Demonstration of using Events**

**SOCKET IO**

* **Understanding Socket.IO**
* **Creating a basic Socket.IO app**
* **Creating a simple chat application using Socket.IO**

**NODE JS BEST PRCTICES**

* **Challenges with Node Js.**
* **Deployment scenarios.**

**CONCLUSION**

**Additional benefits from NBITS**

* Course Material
* Sample resumes and Fine tuning of Resume
* Interview Questions
* Mock Interviews by Real time Consultants
* Certification Questions
* Job Assistance.