



IT EDUCATION
CENTRE

Mean Stack Development



❑ Mongo DB

❑ Express JS

❑ Angular JS

❑ Node JS

Introduction and Environment Set Up.

What is Typescript and ECMAScript?, with Features, AOT, JIT, TSC (Transpilation) Work and Cong, code difference in JS and TS

- ▶ **What is Angular?**
- ▶ **What is the framework?**
- ▶ **The need for MVC, MVVM, MVW and MV* Architecture in Web Application**
- ▶ **What is Components-Based Web Development? And the benefits.**
- ▶ **AngularJs (vs) Angular**
- ▶ **Setup for the local development environment**
- ▶ **Angular files and folder structure with JSON configuration**
- ▶ **Role of Node JS and NPM in Angular**
- ▶ **What is CLI? Angular CLI(Command Line Interface) Commands**
- ▶ **Introduction Of Example Project**

MongoDB

- ▶ Introduction to No SQL DataBAse
- ▶ Advantages over RDBMS
- ▶ MongoDB Data Types
- ▶ Installation
- ▶ MongoDB Data Modeling
- ▶ MongoDB Operators
- ▶ MongoDB Commands – Aggregation, Geospecial, Query & Write Operation, Query Catch Command, Authentication, User Management, Role Management, Replication, Session
- ▶ Collection
- ▶ Document
- ▶ CRUD Operation
- ▶ Schema Modelling
- ▶ MongoDB Shell
- ▶ MongoDB Tools – MongoDB Compass, MongoDB BI Connector

Node.js

- ▶ NodeJS Introduction
- ▶ Installation
- ▶ Node.js – Console, REPL, NPM, CL options, Globals, OS, Timer, Errors, DNS, Net, Crypto, TLS/SSL, Debugger, Process, Child Process, Buffers, Streams, File System, Callback, Events, Web Modules.
- ▶ Node.js to execute script
- ▶ Package Manager
- ▶ HTTP Protocol
- ▶ Building HTTP Server
- ▶ Rendering response
- ▶ Connecting to MongoDB Database
- ▶ Performing CRUD operation

Express.js

Express Framework – Installation, Request, Response, Get, Post, Routing, Cookies, File Upload, Middleware, Scaffolding, Template, Handling Form Data, Error Handling, Creating and Consuming RESTful Services, Using Template, MVC pattern, Building Front end controller, Defining Routes, Creating Action, Project Activity.

Angular JS

➤ Introduction of Angular Js

- What is Angular Js
- Use of Angular JS

➤ MVC Architecture

- Model
- View
- Controller

➤ Conceptual Overview

➤ Angular Js Directives (Inbuilt)

- ng-app
- ng-init
- ng-bind
- ng-model
- ng-controller
- ng-cloak
- G. ng-disabled

➤ Setting up the Environment

➤ First Application

➤ Understanding module, controller, \$scope

➤ Data Binding in Angular Js

➤ Types of Data Binding

- One Way Data Binding
- Two Way Data Binding
- One time Data Binding

- ▶ **Digest Cycle and Dirty Checking Process**
- ▶ **Use of \$watch() and \$apply()**
- ▶ **Bootstrapping Process**
 - Auto Bootstrap Process
 - Manual Bootstrap Process
- ▶ **Controllers**
 - Nested Controllers
 - Multiple Controllers and their Scopes
- ▶ **Programming controllers & \$scope object**
- ▶ **Scope Hierarchies - \$rootScope & \$scope**
- ▶ **Adding behavior to a scope object(Method,Array,Object,Property)**
- ▶ **ng-repeat,nested ng-repeat**
- ▶ **ng-if,ng-show,ng-hide**
- ▶ **Filters**
 - Types of inbuilt filter
 - Custom Filter
- ▶ **Validation**
 - ng-valid
 - ng-invalid
 - ng-dirty
 - ng-pristine
 - ng-touched
 - ng-untouched
- ▶ **Services**
 - Factory
 - Services
 - Provider

➤ **Inbuilt Services**

- \$http
- \$location
- \$log
- \$timeout
- \$interval

➤ **Ajax and Communication From Server via http service**

➤ **Fetch data from JSON file**

➤ **Scope Event Propagation or Communication between Controllers via**

- \$on
- \$Emit
- \$Broadcast

➤ **Custom Directive**

- Template
- templateUrl,
- Restrict Keyword in custom directive
- Link Function

➤ **Scope of Directive**

- Shared
- Inherited
- Isolated

➤ **Routing and view**

➤ **CRUD Operation**