Chapter 1 - Introducing Course

* About Sahosoft
* Course Introduction
* Course Overview
* Where to find the Course Source Code

Chapter 2 - Introducing Angular

* What is Angular?
* Why Angular?
* What You Should Already Know
* Building Blocks of an Angular Application
* Basic Architecture of an Angular Application
* Angular js vs Angular 2 vs Angular 4 vs Angular 5 vs Angular 6 vs Angular 7 vs Angular 8 vs Angular 9 vs Angular 10 vs Angular 11 vs Angular 12
* Angular Semantic Versioning
* Angular Release Frequency and Schedule
* Angular Preview releases
* Support policy and schedule

Chapter 3 - Development Setup of Angular

* Angular Environment Setup
* IDE for writing your code (Editor)
* What is Node.js
* Why does Angular 2 and above need Node.js
* What is Node Package Manager (NPM)
* What is Angular CLI
* Installing VS Code
* Installing Node.js and NPM
* Installing CLI
* Creating a New Project (First App)
* Building the Application
* Running the Application
* Editing the First App

Chapter 4 – Project Structure

* Project Structure
* package json and package-lock json
* Package Version Numbering Syntax
* Updating Packages
* Uninstalling Packages
* Workflow of angular

Chapter 5 - Components

* What is a Component?
* An Example Component
* The Component Class
* Adding a Component to Its Module
* Creating a Component Using Angular CLI
* The Application Root Component
* Brief of @Component Decorators
  + template & templateUrl
  + Styles and StyleUrls
  + preserveWhitespaces
  + viewProvider
  + changeDetection
  + moduleId
  + Interpolation
  + Encapsulation
  + entryComponents
  + animations
  + Selector
  + Input Property
  + Output Property
  + Providers Property
    - Class Provider - useClass
    - Alias Provider - useExisting
    - Value Provider - useValue
    - Factory Provider - useFactory
  + Exportas Property
  + queries Property
  + host Property
  + jit Property
* Component Lifecycle Hooks
* Constructor Vs ngOnInit
* How to Dynamically load a Component in Angular

Chapter 6 - Directives

* What are Directives
  + Component Directives
  + Structural directives
  + Attribute Directive
* Custom Directives
* Custom Structural Directive
  + TemplateRef
  + ViewContainerRef
  + Component view
  + Host view
  + Embedded view
* Custom Attribute Directive
* ElementRef
* Renderer
* Binding to Directive Properties using Custom property binding

Chapter 7 - Structural Directives

* Structural directives
* NgIf
  + NgIf with HTML Elements
  + NgIf Null Check
  + NgIf with Enum
  + ng-template
  + NgIf with Else
  + Angular NgIf-Then-Else
  + Expanding ngIf into template
  + ngIf - ng-template
  + ngIf-else - ng-template
  + ngIf-then-else - ng-template
  + NgIf with Component Elements
  + NgIf with Async Pipe
  + More than one ng-template for then and else block (NgIf-Then-Else)

* NgFor
  + NgForOf with HTML Elements
  + NgForOf with ng-template
  + index, even and odd with HTML Elements
  + index, even and odd with ng-template
  + first and last with HTML Elements
  + first and last  with ng-template
  + NgForOf with Async Pipe with HTML Elements
  + NgForOf with Async Pipe with ng-template
  + trackBy with HTML Elements
  + trackBy with ng-template
* NgSwitch
  + NgSwitch with html element
  + Expanding ngSwitchCase and ngSwitchDefault into template
  + NgSwitch with ng-template
  + NgSwitch with Enum
  + NgSwitch with NgFor and class
  + NgSwitch and NgFor in same element

Chapter 8 - Attribute Directives

* Attribute Directive
* ngStyle
* Alternative syntax of ngStyle
* ngClass
  + NgClass with String
  + NgClass with Array
  + NgClass with Object
* Alternative syntax of NgClass
* NgSwitch with NgFor and NgClass
* NgIf with NgFor and NgClass

Chapter 9 - Data and Event Binding

* Data binding
* Binding Syntax
* Classification of Data Binding
* One-Way Data Binding
* Different types of one-way data binding
* Difference between attribute and property
* Interpolation
* Attribute Binding
* Style Binding
* Class Binding
* Property Binding
* What is the difference between Property Binding and Interpolation?
* Event Binding
* How can we say that - two-way data binding in Angular is actually a combination of both Property Binding and Event Binding?
* Two-Way Data Binding

Chapter 10 - Pipes and Data Formatting

* What are Pipes?
* Formatting Changes in Angular
* What is ngNonBindable?
* Built-in Pipe
* Lowercase pipe
* Uppercase pipe
* Titlecase pipe
* Slice pipe
* Json Pipe
* Decimal pipe
* Percent Pipe
* Currency Pipe
* Date Pipe
* Percent Pipe
* Parameterized pipes
* Multiple pipes (Chaining pipes)
* Async Pipe
* Custom Pipe
* Chaining of Custom Pipes
* Use Built-in Pipe in Custom Pipe
* Pure and Impure Pipes
* Pure Pipe Example
* Impure Pipe Example

Chapter 11 - Decorators

* What is decorators
* @NgModule
* @Component
* @Injectable
* @Directive
* @Pipe
* @Input
* @Output
  + EventEmitter
  + emit()
* @HostBinding
* @HostListener
* @ContentChild
* @ContentChildren
* @ViewChild
* @ViewChildren

   Chapter 12 - Introduction to Typescript and ES6

* What is JavaScript?
* What is Typescript?
* Transpilation
* Programming Languages for Use with Angular
* Typescript Syntax
* Why use Typescript
* Text Editors For Typescript
* Why Typescript is developed while having JavaScript?
* Typescript Version
* Features of Typescript
* Advantage of Typescript over JavaScript
* Disadvantage of Typescript over JavaScript
* Typescript vs. JavaScript
* Difference between JavaScript and Typescript
* Components of Typescript
* TypeScript - Type Annotations
* TypeScript - Variable
* var Declaration
* let Declaration
* Advantages of using let over var
* Const Declaration
* Typescript Data Type
* Static Data Type
  + Built-in or Primitive Type
    - Number Type
    - String Type
    - Boolean Type
    - Void Type
    - Any Type
    - Null Type
  + User-Defined DataType
    - Array
    - Tuple
    - Interface
    - Class
    - Enums
    - Functions
* Generics Data Type
* Decorators Data Type
* Difference between Null and Undefined
* Typescript - Decision Making
  + Typescript - if
  + Typescript - if else Condition
  + Typescript - else if Condition
  + Typescript - switch
* Typescript - Loops
  + Typescript - for Loops
  + Typescript - while Loop
  + Typescript - do. While loop
* Typescript - Data Modifiers
  + public
  + private
  + protected
* Typescript - Function
  + named functions
  + Anonymous functions
* Function Parameters
  + Optional Parameters
  + Default Parameters
* Arrow Function
* Typescript - Rest Parameters
* The Type System – Classes & Objects
* Class Constructors
* Interfaces
* Working with Modules

Chapter 13 - Services and Dependency Injection

* What is a Service?
* Creating a Basic Service
* What is Dependency Injection?
* What Dependency Injection Looks Like
* Injecting Services
* Using a Service in a Component
* Using onInit to Initialize Component Data
* Injector
* Provider
* @Injectable Decorator
* @Inject() Decorator
* providedIn
* Tree Shakeable Providers
* Include the Service in component level
* Include the Service in module level
* Inject Service into other Service (Nested Services)
* Single VS Multiple Instance of Services

Chapter 14 - Router

* Routing and Navigation
* The Component Router
* Component Router Terminology
* Setting up the Component Router
* A Basic App with Routing
* Router Outlet
* Router Link
* Router Link – Client Side
* Router Link – Server-Side
* Creating a Routing Module
* ForRoot() and ForChild()
* Create Component/Service/pipe/directive/class/guard without spec.ts
* Difference-between-the-href-and-routerLink
* Redirecting Routes
* Wildcard Route
* Child Routes / Nested Routes
* Nested Routes (Nesting Children’s under a child)
* Parameterize Route
  + The Snapshot Way
  + The Observable/Stream Way
  + Passing Data During Navigation
  + Creating Routes with Route Parameters
  + Navigating with Route Parameters
  + Retrieving the Route Parameter
* Route Guards
  + Basic information about route guards
  + Route Guards Parameters
  + ActivatedRouteSnapshot
  + RouterStateSnapshot
* Route Guard Types
  + CanActivate guard
  + CanActivateChild guard
  + CanDeactivate guard
  + Resolve guard
  + CanLoad guard
* Named and Multiple Router-Outlets (Auxiliary Routes or secondary routes)
* Difference between [routerLink] and routerLink
* RouterLinkActive
* Navigate
* NavigateByUrl
* Query Parameters

Chapter 15 - Angular Modules

* Introduction of modules
* The Root Module
* @ngmodule decorator
* Purpose of @ngModule
* Types of ngModule
* @ngModule Metadata Properties
* Feature Modules in Angular
* Eager Loading
* Lazy Loading Modules
* Preloading Modules
* Custom Preloading Modules strategy
* Custom Preloading Modules strategy with delay
* How to Create a Module
* Importing BrowserModule or CommonModule…etc.

Chapter 16 - Session Management

* What is HTML Web Storage?
* What is localStorage and sessionStorage?
* Set Entries
* Updating Entries
* Deleting Entries
* Clearing Everything
* Storing Json Objects
* Checking for Items
* Checking for Support
* Iterating Over Items

Chapter 17 - rxjs

* What is Observable
* What is Subscribe
* Angular In-Memory Web API
* Observable and subscribe with Web API
* Observable with Async pipe and ngFor
* Observable with Async pipe and ngIf
* Observable map
* What is Promise
* Difference between observable and promise
* RxJS Subject and BehaviorSubject and replaysubject

Chapter 18 - HTTP Client

* The Angular HTTP Client
* Using The HTTP Client - Overview
* Importing HttpClientModule
* Service Using HttpClient
* Using the Service in a Component
* HTTP VS HttpClient
* HttpModule VS HttpClientModule
* HttpClient get()
* HttpClient post()
* HttpClient put()
* HttpClient delete()
* HttpParams and HttpHeaders
* Use of headers in HttpClient
* Use of observe in HttpClient
* Use of params in HttpClient
* Use of reportProgress in HttpClient
* Use of responseType in HttpClient
* Use of withCredentials in HttpClient
* Requesting JSON using HttpClient.get using subscribe method
* Requesting JSON using HttpClient.get useing async pipe with Observable
* Requesting Text data using HttpClient.get using subscribe method
* Requesting Text data using HttpClient.get useing async pipe with Observable
* Using HttpParams
* Using HttpParams and HttpHeaders
* Using HttpClient.get with observe Property
* Error Handling
* RxJS retry() to Handle Error
* HttpClient.post Body
* HttpClient.post with Headers
* HttpClient.post with observe Option
* HttpClient.post and Response Type
* HttpClient.post with Error Handling
* HttpClient.put Body
* HttpClient.put with Headers
* HttpClient.put with observe Option
* HttpClient.put and Response Type
* HttpClient.put with Error Handling
* HttpClient.delete with Headers
* HttpClient.delete with observe Option
* HttpClient.delete and Response Type
* HttpClient.delete with Error Handling

Chapter 19 – Forms in Angular

* What is form?
* Forms in Angular
* Template Driven Forms
* Model-driven forms (Reactive forms)
* Which one is better - Template Driven or Reactive?

Chapter 20 - Template Driven Forms

* Template Driven Forms
* Template Driven Forms features
* Importing Forms Module
* A Basic Angular Form
* What is NgForm?
* What is NgModel?
* Template Driven Forms with Validation
* What is ng-touched?
* What is ng-untouched?
* What is ng-pristine?
* What is ng-valid?
* What is ng-invalid?
* Binding Input Fields
* Accessing the Form Object
* Binding the Form Submit Event
* The Submit Function
* Basic HTML5 Validation - "required" Attribute
* HTML5 vs. Angular Validation
* Angular Validators
* Angular Validation State
* Controlling when validation is applied
* Displaying Form Validation State
* Displaying Field Validation State
* Displaying Validation State Using Classes
* Disabling Submit when Form is Invalid
* Submitting the Form
* Checkboxes
* Select (Drop Down) Fields
* Rendering Options for Select (Drop Down)
* Date fields
* Radio Buttons

Chapter 21 - Model Driven Forms

* Model Driven Forms
* Setup for Model Driven Forms
* Form Component Setup
* Setup Main FormGroup
* formControlName
* FormControl Object
* Getting Form Values
* FormBuilder Form Initialization
* Validation
* Built-In Validators
* Controlling when validation is applied
* What is FormControl?
* What is FormGroup?
* What is FormBuilder?
* Reactive forms with Validation
* FormGroup Get Value and reset()
* FormGroup setValue() and patchValue()
* valueChanges() and statusChanges()
* What is FormArray?
* Use of FormArray in component
* What is Nested FormArray?
* Use of Nested FormArray in component
* What is Nested FormGroup?
* Use of Nested FormGroup
* Reactive Form Using Angular Material Design
* Custom Validator
* Using a Custom Validator

Chapter 22 - Angular Material, bootstrap, jQuery, Cookies

* What is Angular Material?
* Angular Material - Features
* Angular Material - install and use
* Angular Material Theming
* Angular Material All Modules
* What is bootstrap?
* Install and Use bootstrap in Angular
* How to Import and Use jQuery in Angular
* What is Cookies?
* Creating Cookies using ngx-cookie-service in Angular
* Cookie methods-Check
* Cookie methods-Get
* Cookie methods-GetAll
* Cookie methods-Set
* Cookie methods-Delete
* Cookie methods-DeleteAll

Chapter 23 - Introduction to Single Page Applications

* What is a Single Page Application (SPA)
* SPA Workflow
* Traditional Web Application Capabilities
* Single Page Application Advantages
* SPA and Traditional Web Sites
* SPA Challenges
* Implementing SPA's Using Angular
* Simple SPA Using Visibility Control
* SPA Using Angular Components
* Implement SPA Using an Angular Component Router

Chapter 24 –Creating and Consuming REST Web Services in Angular

* REST Web Services and Angular
* Understanding REST
* REST Example – Create
* REST Example – Retrieve
* REST Example – Update
* REST Example – Delete
* REST Example – Client Generated ID
* REST Example – JSON
* Knowledge of the REST API
* Common Angular Tasks for REST Communication
* Angular Service Class Using HTTP Client
* RequestOptions
* URL Path Parameters
* Query Parameters
* Common HTTP Request Headers
* Override Default Request Options
* Returning Response Data
* DELETE
* GET
* PUT
* POST
* Security of REST APIs
* Summary

Chapter 25 - Debugging

* Debugging Overview
* Basic Debugging Practices
* Development (Debug) Mode
* Selecting Elements to Inspect
* Inspecting Angular Components
* Using Breakpoints in Angular Code
* Breakpoint in TypeScript Code
* Common Exceptions
* Common Exceptions: 'No such file: package.json'
* Common Exceptions: 'Cant bind to ngModel'
* Common Exceptions: 'router-outlet not a known element'
* Common Exceptions: 'No provider for Router!'

Chapter 26 - Angular Style Guide

* What is the Angular Style Guide?
* Style Categories
* Single Responsibility
* Naming
* Coding Conventions
* App Structure and Angular Modules
* Components
* Directives and Services

Chapter 27- Overview of JSON Web Tokens (JWT)

* JSON Web Tokens
* How JWT Tokens Are Used
* Adding JWT to HTTP Header
* How The Server Makes Use of JWT Tokens?
* Angular HTTP Client Interceptors
* JWT Authentication

Chapter 28 – Third Party Packages Implementation

* CSS and SCSS and SASS
* feather-icons
* How to Install Font-Awesome in Angular?
* Implement Toastr Notification in Angular
* angular-count-to
* NgbTabsetLink
* sweetalert2
* install bootstrap in angular
* NG Bootstrap - Angular Powered Bootstrap widgets
* ngx-ckeditor
* ngx-datatable
* ng2-smart-table In Angular
* ng-chartist
* ng2-charts
* ng2-google-charts
* ng5-slider
* ngx-image-zoom
* Stripe payment gateway in angular

Chapter 29 – Live Project

* e-commerce application
* 01All concepts of Angular 2 - 12
* 02Problem Solving & Coding Skill
* 03TypeScript
* 04Angular Material
* 05BootStrap
* 06rxjs
* 07Web API
* 08Live Project Training