Angular 4 Syllabus

Overview:

Angular is a framework for building client applications in HTML and either JavaScript or a language like Typescript that compiles to JavaScript. The framework consists of several libraries, some of them core and some optional. You write Angular applications by composing HTML templates with Angularized markup, writing component classes to manage those templates, adding application logic in services, and boxing components and services in modules. Then you launch the app by bootstrapping the root module. Angular takes over, presenting your application content in a browser and responding to user interactions according to the instructions you have provided.

Course Objectives:

- Develop modern, complex, responsive and scalable web applications with Angular 4
- Use their gained, deep understanding of the Angular 4 fundamentals to quickly establish themselves as frontend developers
- Fully understand the architecture behind an Angular 4 application and how to use it
- Create single-page applications with one of the most modern JavaScript frameworks out there

Pre-requisite / Target Audience:

- JavaScript OOP basics (constructor pattern, inheritance, polymorphism, this object)
- ES6 - class, extend, arrow function, import
- Typescript is identical to ES6, but has few more important features such as Interface, Type system & Decorators

Module 1: Introduction to Angular

In this module, you will learn what is angular, what are the major differences between Angular 1.X, Angular 2 and Angular 4. We will also learn how to setup in the local environment and a small introduction about typescript.

- What is Angular?
- Features of AngularJS and Angular
- AngularJS **(vs)** Angular
- Steps to Setup for local development environment
- Executing First Angular program using NodeJS and NPM
- Executing First Angular program using Visual Studio.

Module 2: Angular Architecture

In this module, you will learn how angular works and flow of the angular application in depth. Concepts like services, directives, modules, databinding

- Introduction
- Basic Building Blocks of Angular Applications
- Angular Modules and @NgModule decorator
- Angular Libraries
- Component, Templates and Metadata
- Data Binding
- Directives
- Services and Dependency Injection

Module 3: Displaying Data using Templates

In this module, you will learn about directives, directives are the classes that can change the behavior or appearance of the components by using CSS Classes, CSS Styles and events here you will find how to use directives in our application

- About Directives
- Components Properties and Interpolation
- Built-In Directive
    - ngIF

- ngFor
- ngSwitch

- Creating a class for data (Model object)
- Template Expressions
- Working with Arrays
- * vs <template>
- External HTML Template File

Module 4: Data Binding

In this module, you will learn how many kind of databinding and we will discuss each of them in this chapter

- Binding properties and Interpolation
- One-way Binding / Property Binding
- Event Binding
- Two-way Binding
- Two-way binding with NgModel
- Attribute Binding
- Style and Class Binding

Module 5: Styles Binding In Components

In this module, Angular applications are styled with regular CSS. That means we can apply everything we know about CSS stylesheets, selectors, rules, and media queries to our Angular applications directly, this module will help you to how to apply css to our page

- Style and Class Bindings
- Built-In Directives - NgStyle & NgClass
- Using Component Styles
- Special selectors
- Loading Styles into Components

Module 6: Advanced Component Features

In this module we will learn one of the major concepts in angular4 i.e., components and we can learn how to create dynamic component using ngComponentOutlet

- Revising what are Components?
- Components Life Cycle Hooks.
- Dynamic components using ngComponentOutlet.

Module 7: Template Driven Forms

In this module, you will learn in most of the frontend applications we use HTML forms for grouping HTML elements and we provide validations using HTML5 attributes like required, minlength, maxlength, pattern etc

- Introduction
- Create the component that controls the form
- Create a template with the initial form layout
- Bind data properties to each form input control with the ngModel two-way data binding syntax
- Add the name attribute to each form input control
- Add custom CSS to provide visual feedback
- Show and hide validation error messages
- Handle form submission with **ngSubmit**
- Disable the form's submit button until the form is valid
- Resetting the form.

Module 8: Reactive Forms

Model driven forms are more powerful and easy to do functionalities, which are complex when using template driven forms

- Reactive Forms Introduction
- More Form Controls
- Form Control Properties
- setValue and patchValue
- Validating Form Elements

- Submitting and Resetting forms
- Observing and Reacting to Form Changes
- Using FormBuilder

Module 9: Pipes

In this module, you will learn about pipe, a pipe takes in data as input and transforms it to a desired output

- Built-in Pipes
- Using parameters and chaining Pipes
- Custom Pipes
- Parameterized Custom Pipe
- Pipes and Change Detection
- Pure and Impure pipes

Module 10: Custom Directives

In this module, you will learn about attribute and validators, Using custom attribute directive, we can change the color, back-ground, font-size etc., of the HTML host element by using ElementRef

- Custom Attribute Directive
- Using HostListener
- Using HostBinding
- Custom Validator Directive

Module 11: Dependency Injection

In this module, we will learn Dependency Injection (DI) is the software design pattern in which a class receives its dependencies from external sources rather than creating them itself. In addition, we will learn about very interesting topic Dependency Injection in angular.

- Understanding Dependency Injection
- Understanding DI in Angular Framework
- ReflectiveInjector
- Exploring Provider
- Types of Tokens
- Types of Dependencies
- Configuring DI using Providers
- Implementing DI in Angular
- Optional Dependencies

Module 12: Services In Angular

In this module, you will learn about services,In angular services are reusable classes which can be injected in components when it's needed. Using a separate service keeps components lean and focused on supporting the view, and makes it easy to unit test components with a mock service

- Building and Injecting Custom Services
- Service using another Service
- Built-In $http Service
- Promises and Observables
- Get Request

Module 13: Angular Routing

In this module we will learn the introduction for routing in angular and how to navigate between views, how to do parameterized routing.

- Introduction
- Configuring and Navigating
- Parameterized routes
- Nested (or) Child Routes

Module 14: Angular Modules

In this module, you will learn module, An NgModule class is adorned with @NgModule decorator function this will tell the angular application how to compile and run the module code

- AppModule as Root Module

- Feature modules
- Lazy Loading a Module
- Shared Module

Module 15: Performing CRUD Operations as in Realtime Angular Application.
In this module, we will create an application with end-to-end start from server to client, getting response and requests using HTTP service.

- Create and ASP.NET MVC Web API Application
- Add Angular Seed, download files and Configure Project
- Setup Project for Entity Framework Code First Model
- Add Web API Controller to perform CRUD Operations using EF.
- Create Angular Feature and Routing Modules
- Configure Angular to use HTTP and Json Services.
- Program Angular Modules to performing CRUD Operations.

At the end of the course, participants will be able to

- Build native mobile apps for Android, iOS and using Angular4
- Understand the fundamentals of Angular Forms and its architecture
- Present data in beautiful, interactive lists
- Build forms and setting pages
- Implement Single page application (SPA)