ANGULAR 7

Brochure

zeolearn™

▶ **24 hours of interactive instructor led training**

▶ **Covers features of Angular 2, 4, 5, 6 and 7**

▶ **Gain basic to advanced Angular skills**

▶ **Learn by doing, experience end to end development cycle**

24 hrs	**120+ hrs**	**2**	**2**
live training	quizzes, assignments	live projects	industry use cases

You Will Learn To

▶ Code using new ES6 and TypeScript language features

▶ Understand and use Angular Forms, Observables, Dependency Injection, and Routing

▶ Unit test all the parts of an Angular application including Modules, Components, Services, and Pipes

▶ Develop reusable elements (even out of angular applications) using angular elements

▶ Explore Angular coding and architecture best practices

▶ Retrieve, update, and delete data using Angular's Http service

▶ Create, build, and deploy an Angular (Angular 6) application using the Angular CLI

▶ Develop dynamic Model-driven forms that are easier to unit test

Pre-Requisites

All attendees must have substantial prior experience developing with JavaScript. If attendees will not have prior JavaScript experience, we would be delighted to precede this class with a one- or two-day intensive JavaScript

hello@zeolearn.com

zeolearn™

Curriculum for Angular 7

Module 1: Introduction

› Introduction to Angular 7

Module 2: Why Angular?

› User Experience similar to a Desktop Application

› Productivity and Tooling

› Performance

› Community

› Full-featured Framework

› Platform for Targeting Native Mobile not just Web Browsers

Module 3: Understanding Angular Versions

› AngularJS (Angular 1.x)

› Angular 2

› Angular 4

› Angular 5

› Angular 6

› Angular 7

Module 4: Typescript & ES6

› Introduction

› What is Typescript

› Why Typescript

› Setup and installation

› IDE support

› Scoping using Let and Const Keywords (ES6)

› Template Literals (ES6)

› Spread Syntax and Rest Parameters (ES6)

› Destructuring (ES6)

Module 5: Power of Types

› Introduction

› Type inference

› Type Annotations

› Number

› Boolean

› String

› Array

› Tuple

› Enum

› Any

› Void

› Null and Undefined

› Never (Typescript 2)

Module 6: Explore Functions

› Introduction

› Using types in functions

› Function as types

› Optional and default parameters

› Arrow functions

Module 7: Classes

› Introduction

› Inheritance

› Access modifiers

› Getters and setters

› Readonly & static

Module 8: Interfaces

› Introduction

› Optional properties and methods

zeolearn™

Module 8: contd...

> Strict structural contract
> Extending interface
> Implementing interface

Module 9: Modules

> Introduction
> Import
> Export
> Default

Module 10: Decorators (Typescript Aspect Oriented Programming)

Module 11: Working with Angular CLI

> Angular CLI
> Anatomy of the project
> Debugging Angular apps
> Working with Augury

Module 12: 8 Main Building blocks of Angular (Birds Eye View Of Angular)

> Modules
> Components
> Templates
> Metadata
> Data binding
> Directives
> Services
> Dependency injection

Module 13: Angular modules

> Why modules
> How to create modules
> Built in modules
> Root Module
> feature module

Module 14: Components

> Introduction
> @Component decorator
> Component configuration object
> Custom components
> Component with templates
> Inline
> External
> Component with Styles
> Inline
> External

Module 15: Templating

> HTML as template
> Data binding
> Interpolation
>   Property Binding
>   Event Binding
> Two way binding
> Template expressions
> Template syntax
> Attribute, class and style bindings
> @Input()
> @Output
> Template reference variables
> Safe navigation operator

Module 16: Directives

> Introduction
> Built-in directives
> Structural directive
> NgIf
> NgFor
> NgSwitch
> Attribute directive
> NgClass
> NgStyle
> NgModel

zeolearn™

Module 17: Pipes

> Introduction
> Built-in pipes
> @Pipe decorator

Module 18: Forms

> Introduction
> @angular/forms library
> Template driven forms
> Form and field validation
> Validation check with ng-pristine,ng-dirty, ng-touched, ng-untouched, ng-valid, ng-invalid
> Show and hide validation error messages
> Form submission with ngSubmit
> Reactive/ Model drive forms
> ReactiveFormsModule
> FormGroup, FormControl classes
> FormBuilder for easy form building
> Validations using Validators
> Setting form model using setValue and patchValue
> Use FormArray to build repeated from controls or form groups

Module 19: Dependency Injection

> Introduction
> Why DI
> @Injectable decorator
> Custom service development
> Registering the service with NgModule using providers key
> Provider Types
> Class
> Factory
> Value

Module 20: Routing And Navigation

> Introduction
> @angular/router library
> Configure routes
> RouterModule.forRoot and RouterModule.forChild
> RouterOutlet, RouterLink, RouterLinkActive
> Nested Routes
> Parametrized routes
> Route guards

Module 21: RxJs Primer

> Introduction
> Why RxJs
> Observable interface
> Streams
> Operators
> Subscription
> Subject
> Schedulers

Module 22: HTTP Deprecated & HttpClient

> Setup installing the module
> Making a request for JSON data
> Typechecking the response
> Error handling
> Sending data to the server
> Making a POST request
> Configuring other parts of the request

Module 22: New Features in Angular 6

> New features in angular cli
> ng update
> ng add
> Angular Elements
> Updates to @angular/material

hello@zeolearn.com

zeolearn™

Module 22: contd...

> CLI workspaces
> Building libraries
> Tree shakable providers
> RxJs 6 Support
> How to upgrade to Angular 6

Module 23: New Features in Angular 7

> TypeScript 3.1 support
> RxJs 6.3 support
> Node 10 support
> New CLI prompts
> Application performance improvements
> New features to Angular CDK & Material
> Virtual scrolling
> Drag and Drop
> Content projection support for angular elements
> New partner launches
> Angular console
> @angular/fire

Module 24.: How to upgrade to Angular 7

Module 25: Testing

> Introduction
> Testing in Typescript
> Testing Component
> Testing Service/Provider
> Testing Pipe

Module 26: Deployment

> Manually
> Using the Angular CLI with Ahead-Of-Time (AOT) Compilation and Tree-Shaking (removing unused library code)

WHY ZEOLEARN

Immersive Hands-on training with a combination of theoretical learning, hands-on exercises, simulations and intensive Q&A sessions.

Lifetime Mentoring Support from our Industry's Expert anytime

Apply theory to practice by building live projects through the duration of training.

Lifetime access to the Recorded Videos

Get Mentored by Industry Practitioners

Lifetime Enrolment for Free in any of the upcoming workshops, if wished to brush up and practice again

Benefits

Benefits of on-line web training

☺ Lifetime Mentoring Support from our Industry's Expert anytime

☺ Lifetime access to the Recorded Videos

☺ Lifetime Enrollment For Free in any of the upcoming workshops, if wished to brush up and practice again.

Benefits of on-site web training

☺ Mentors evaluate / review the Live Project

☺ Deep dive into specific concepts

☺ Assistance provided in implementing the technology in existing / new projects

All our training sessions (on-site and on-line) include course notes along with post course support. We are able to work with your questions and queries on a one-to-one basis as necessary. By delivering our courses live, via the web or on-site, we can create the training environment that best fits your needs.

Featured in

Inc.   **CIO**   **Forbes**   **DIGITAL JOURNAL**   **Entrepreneur**   **Chicago Tribune**   THE HUFFINGTON POST

---

zeolearn™

Zeolearn LLC

12227 Audrianna Drive, Frisco, Texas, USA – 75033

USA +1-315-599-0046   AUS +61-261452089
UK +44-20-79938831   India +91-8065660555