# Angular Training

simpli·learn

# Table of **Contents:**

# Program **Overview:**

This Angular Certification Training Course will help you master front-end web development with Angular. Gain in-depth knowledge of Angular concepts such as facilitating the development of single-page web applications, dependency injection, typescript, components, directives, pipes, forms, routing, and testing.

# Program **Features:**

> 50 hours of Blended Learning

> 20 hours of online self-paced learning

> 30 hours of instructor-led training

> Free introductory JavaScript course

> Flexibility to choose classes

> Industry-recognized course completion certificate

# Delivery **Mode:**

**Blended -** Online self-paced learning and live virtual classroom

# Prerequisites**:**

There are no prerequisites for taking this Angular certification training course.

## Target **Audience:**

> UI developers

> Technical project managers

> Full-stack web developers

> Quality assurance specialists

> Architects

> Students who wish to learn front-end UI development

## Key Learning **Outcomes:**

By the end of this online Angular training course, you will be able to:

> Understand the design of single-page applications and how Angular facilitates their development

> Separate the model, view, and controller layers of your application and implement them using Angular

> Master Angular expressions, filters, and scopes

> Build Angular forms

> Write Angular directives

> Unit test and end-to-end test your Angular applications

> Developing interactive and dynamic UI using Bootstrap

## Certification Details and **Criteria:**

> Complete at least 85 percent of the course or attend one complete batch

> Successful completion and evaluation in the project

# Course **Curriculum:**

## Live Virtual Class Curriculum:

## Lesson 01 - Features of TypeScript

> Introduction
> Introduction to TypeScript
> Introduction to Data Types
> Let vs Const
> Introduction to Functions
> Introduction to Parameters
> Introduction to Classes

## Lesson 02 - Features of Angular

> History of Angular
> Understanding Angular
> Set up Angular App
> Angular Building Blocks
> Observables
> Introduction to Reactive Forms
> Introduction to Server Communication

## Lesson 03 - Ngmodule

> Angular Modules
> Routing Module
> Feature Module
> Sharing Module
> Systematically Arranging Components of App

## Lesson 04 - Ng Unit Testing and Observables

> Introduction to Automated Testing
> Types of Automated Testing
> Introduction to Testing Tools
> Introduction to Test Bed
> Handling Component Dependencies
> RxJS - Observable

## Lesson 05 - Bootstrap

> Learning Objectives
> Explain Responsive Web Design (RWD)
> Understand the Bootstrap Grid System
> Learn Bootstrap Components
> Quiz
> Key Takeaways

## Lesson 06 - Binding and Events

> Learning Objectives
> What is Template Model?
> How Angular Binding works and the Type of Bindings
> Understand Angular Built-in Directives
> Basics of Webpack and SystemJS
> Quiz
> Key Takeaways

## Lesson 07 - Dependency Injection and Service

> Learning Objectives
> Understand Dependency Injection (DI)
> Understand DI Application Programming Interface
> Explain a Service
> Describe How to Create a Service
> Explain Injecting a Service
> Quiz
> Key Takeaways

## Lesson 08 - Directives

> Learning Objectives
> Angular Directives
> Types of Angular Directives
> Built-in Angular Directives
> Working with Custom Directives
> Quiz
> Key Takeaways

## Lesson 09 - Pipes

> Learning Objectives
> What is Pipe in Angular
> Understand How Built-in Pipes Work in Angular
> Understand Angular Custom pipes
> Quiz
> Key Takeaways

## Lesson 10 - Forms

> Learning Objectives
> Angular Form Benefits
> Template-Driven Approach
> Model-Driven Approach
> Angular Form Validation
> Quiz
> Key Takeaways

## Lesson 11 - Routing

> Learning Objectives
> Understand How Angular Helps to Achieve SPA Using Routing
> Define the Benefits of @NgModule
> Identify Multiple Ways of Accessing Routes
> Understand the Process of Routing Cycle
> Quiz
> Key Takeaways

## Lesson 12 - HTTP, Promises, and Observables

> Learning Objectives
> Understand Working with RxJS
> Understand Angular Interaction with HTTP GET
> Describe the Process of Sending Data to the Server
> Explain the Difference Between Promises and Observables
> Quiz
> Key Takeaways

## Lesson 13 - Testing

> Learning Objectives
> Understand Tools and Setup
> Understand Testing of Angular Class
> Describe Testing Service
> Understand Testing DOM
> Explain End to End Testing
> Quiz
> Key Takeaways

## Self-paced Curriculum:

## Lesson 01 - Getting Started

> Course Introduction
> What is Angular?
> Angular vs Angular 2 vs Angular 8
> Project Setup and First App
> Editing the First App
> The Course Structure
> How to get the Most out of the Course
> What is TypeScript
> A Basic Project Setup using Bootstrap for Styling

## Lesson 02 - The Basics

- Module Introduction
- How an Angular App gets Loaded and Started
- Components are Important!
- Creating a New Component
- Understanding the Role of AppModule and Component Declaration
- Using Custom Components
- Creating Components with the CLI Nesting Components
- Working with Component Templates
- Working with Component Styles
- Fully Understanding the Component Selector
- Assignment Solution
- What is Databinding
- String Interpolation
- Property Binding
- Property Binding vs String Interpolation
- Event Binding
- Passing and Using Data with Event Binding
- Two-Way-Databinding
- Combining all Forms of Databinding
- Assignment Solution
- Understanding Directives
- Using ngIf to Output Data Conditionally
- Enhancing ngIf with an Else Condition
- Styling Elements Dynamically with ngStyle
- Applying CSS Classes Dynamically with ngClass
- Outputting Lists with ngFor
- Assignment Solution
- Getting the Index when using ngFor

## Lesson 03 - Course Project - The Basics

- Project Introduction
- Planning the App
- Setting up the Application
- Creating the Components
- Using the Components
- Adding a Navigation Bar
- Creating a 'Recipe' Model
- Adding Content to the Recipes Components
- Outputting a List of Recipes with ngFor
- Displaying Recipe Details
- Working on the ShoppingListComponent
- Creating an Ingredient Model
- Creating and Outputting the Shopping List
- Adding a Shopping List Edit Section
- Wrap Up & Next Steps

## Lesson 04 - Debugging

> Understanding Angular Error Messages
> Debugging Code in the Browser Using Sourcemaps
> Using Augury to Dive into Angular Apps

## Lesson 05 - Components  Data binding Deep Dive

> Module Introduction
> Splitting Apps into Components
> Property Event Binding Overview
> Binding to Custom Properties
> Assigning an Alias to Custom Properties
> Binding to Custom Events
> Assigning an Alias to Custom Events
> Custom Property and Event Binding Summary
> Understanding View Encapsulation
> More on View Encapsulation
> Using Local References in Templates
> Getting Access to the Template & DOM with @ViewChild
> Projecting Content into Components with ng-content
> Understanding the Component Lifecycle
> Seeing Lifecycle Hooks in Action
> Lifecycle Hooks and Template Access
> Getting Access to ng-content with @ContentChild
> Wrap Up
> Assignment Solution

## Lesson 06 - Course Project Components Databinding

> Introduction
> Adding Navigation with Event Binding and ngIf
> Passing Recipe Data with Property Binding
> Passing Data with Event and Property Binding (Combined)
> Allowing the User to Add Ingredients to the Shopping List

simpli learn

## Lesson 07 - Directives Deep Dive

> Module Introduction
> ngFor and ngIf Recap
> ngClass and ngStyle Recap
> Creating a Basic Attribute Directive
> Using the Renderer to build a Better Attribute Directive
> Using HostListener to Listen to Host Events
> Using HostBinding to Bind to Host Properties
> Binding to Directive Properties
> What Happens behind the Scenes on Structural Directives
> Building a Structural Directives
> Understanding ngSwitch

## Lesson 08 - Course Project Directives

> Building and Using a Dropdown Directive

## Lesson 09 - Using Services Dependency Injection

> Module Introduction
> Why would you Need Services
> Creating a Logging Service
> Injecting the Logging Service into Components
> Creating a Data Service
> Understanding the Hierarchical Injector
> How many Instances of Service Should It Be
> Injecting Services into Services
> Using Services for Cross-Component Communication
> Assignment Solution

## Lesson 10 - Course Project Services Dependency Injection

> Introduction
> Setting up the Services
> Managing Recipes in a Recipe Service
> Using a Service for Cross-Component Communication
> Adding the Shopping List Service
> Using Services for Pushing Data from A to B
> Adding Ingredients to Recipes
> Passing Ingredients from Recipes to the Shopping List (via a Service)

# Lesson 11 - Changing Pages with Routing

- Module Introduction
- Why do we need a Router
- Setting up and Loading Routes
- Navigating with Router Links
- Understanding Navigation Paths
- Styling Active Router Links
- Navigating Programmatically
- Using Relative Paths in Programmatic Navigation
- Passing Parameters to Routes
- Fetching Route Parameters
- Fetching Route Parameters Reactively
- An Important Note about Route Observables
- Passing Query Parameters and Fragments
- Retrieving Query Parameters and Fragments
- Practicing and some Common Gotchas
- Setting up Child (Nested) Routes
- Using Query Parameters – Practice
- Configuring the Handling of Query Parameters
- Redirecting and Wildcard Routes
- Outsourcing the Route Configuration
- An Introduction to Guards
- Protecting Routes with canActivate
- Protecting Child (Nested) Routes with canActivateChild
- Using a Fake Auth Service
- Controlling Navigation with canDeactivate
- Passing Static Data to a Route
- Resolving Dynamic Data with the resolve Guard
- Understanding Location Strategies
- Wrap Up

## Lesson 12 - Course Project Routing

> Planning the General Structure
> Setting up Routes
> Adding Navigation to the App
> Marking Active Routes
> Fixing Page Reload Issues
> Child Routes - Challenge
> Adding Child Routing Together
> Configuring Route Parameters
> Passing Dynamic Parameters to Links
> Styling Active Recipe Items
> Adding Editing Routes
> Retrieving Route Parameters
> Programmatic Navigation to the Edit Page
> One Note about Route Observables

## Lesson 13 - Understanding Observables

> Module Introduction
> Analyzing Angular Observables
> Getting Closer to the Core of Observables
> Building a Custom Observable
> Errors & Completion
> Observables You!
> Understanding Operators
> Subjects
> Wrap Up

## Lesson 14 - Course Project Observables

> Improving the Reactive Service with Observables (Subjects)

simpli learn

# Lesson 15 - Handling Forms in Angular Apps

> Module Introduction
> Why do we Need Angular s Help
> Template-Driven (TD) vs Reactive Approach
> An Example Form
> TD - Creating the Form and Registering the Controls
> TD - Submitting and Using the Form
> TD - Understanding Form State
> TD - Accessing the Form with @ViewChild
> TD - Adding Validation to check User Input
> TD - Using the Form State
> TD - Outputting Validation Error Messages
> TD - Set Default Values with ngModel Property Binding
> TD - Using ngModel with Two-Way-Binding
> TD - Grouping Form Controls
> TD - Handling Radio Buttons
> TD - Setting and Patching Form Values
> TD - Using Form Data
> TD - Resetting Forms
> Introduction to the Reactive Approach
> Reactive - Setup
> Reactive - Creating a Form in Code
> Reactive - Syncing HTML and Form
> Reactive - Submitting the Form
> Reactive - Adding Validation
> Reactive - Getting Access to Controls
> Reactive - Grouping Controls
> Reactive - Arrays of Form Controls (FormArray)
> Reactive - Creating Custom Validators
> Reactive - Using Error Codes
> Reactive - Creating a Custom Async Validator
> Reactive - Reacting to Status or Value Changes
> Reactive - Setting and Patching Values
> Assignment Solution

## Lesson 16 - Course Project Forms

> Introduction
> TD - Adding the Shopping List Form
> Adding Validation to the Form
> Allowing the Selection of Items in the List
> Loading the Shopping List Items into the Form
> Updating existing Items
> Resetting the Form
> Allowing the the User to Clear (Cancel) the Form
> Allowing the Deletion of Shopping List Items
> Creating the Template for the (Reactive) Recipe Edit Form
> Creating the Form for Editing Recipes
> Syncing HTML with the Form
> Adding Ingredient Controls to a Form Array
> Adding new Ingredient Controls
> Validating User Input
> Submitting the Recipe Edit Form
> Adding a Delete and Clear (Cancel) Functionality
> Redirecting the User (after Deleting a Recipe)
> Adding an Image Preview
> Providing the Recipe Service Correctly
> Deleting Ingredients and Some Finishing Touches

## Lesson 17 - Using Pipes to Transform Output

> Introduction Why Pipes are Useful
> Using Pipes
> Parametrizing Pipes
> Where to learn more about Pipes
> Chaining Multiple Pipes
> Creating a Custom Pipe
> Parametrizing a Custom Pipe
> Example - Creating a Filter Pipe
> Pure and Impure Pipes (or How to fix the Filter Pipe)
> Understanding the 'async' Pipe

simpli·learn

## Lesson 18 - Making HTTP Requests

> A New IDE
> Module Introduction
> How Does Angular Interact with Backends
> The Anatomy of a HTTP Request
> Backend (Firebase) Setup
> Sending a POST Request
> GETting Data
> Using RxJS Operators to Transform Response Data
> Using Types with the HTTPClient
> Outputting Posts
> Showing a Loading Indicator
> Using a Service for HTTP Requests
> Services Components Working Together
> Sending a DELETE Request
> Handling Errors
> Using Subjects for Error Handling
> Using the catchError Operator
> Error Handling & UX
> Setting Headers
> Adding Query Params
> Observing Different Types of Responses
> Changing the Response Body Type
> Introducing Interceptors
> Manipulating Request Objects
> Response Interceptors
> Multiple Interceptors
> Wrap Up

## Lesson 19 - Course Project HTTP

> Module Introduction
> Backend (Firebase) Setup
> Setting Up the DataStorage Service
> Storing Recipes
> Fetching Recipes
> Transforming Response Data
> Resolving Data Before Loading
> Fixing a Bug with the Resolver

simpl·learn

## Lesson 20 - Authentication Route Protection in Angular Apps

- Module Introduction
- How Authentication Works
- Adding the Auth Page
- Switching Between Auth Modes
- Handling Form Input
- Preparing the Backend
- Preparing the Signup Request
- Sending the Signup Request
- Adding a Loading Spinner Error Handling Logic
- Improving Error Handling
- Sending Login Requests
- Login Error Handling
- Creating & Storing the User Data
- Reflecting the Auth State in the UI
- Adding the Token to Outgoing Requests
- Attaching the Token with an Interceptor
- Adding Logout
- Adding Auto-Login
- Adding Auto-Logout
- Adding an Auth Guard
- Wrap Up

## Lesson 21 - Dynamic Components

- Module Introduction
- Adding an Alert Modal Component
- Understanding the Different Approaches
- Using ngIf
- Preparing Programmatic Creation
- Creating a Component Programmatically
- Understanding entryComponents
- Data Binding Event Binding
- Wrap Up

## Lesson 22 - Angular Modules Optimizing Angular Apps

> Module Introduction
> What are Modules
> Analyzing the AppModule
> Getting Started with Feature Modules
> Splitting Modules Correctly
> Adding Routes to Feature Modules
> Component Declarations
> The ShoppingList Feature Module
> Understanding Shared Modules
> Understanding the Core Module
> Adding an Auth Feature Module
> Understanding Lazy Loading
> Implementing Lazy Loading
> More Lazy Loading
> Preloading Lazy-Loaded Code
> Modules Services
> Loading Services Differently
> Ahead-of-Time Compilation
> Wrap Up

## Lesson 23 - Deploying an Angular App

> Module Introduction
> Deployment Preparations and Important Steps
> Using Environment Variables
> Deployment Example - Firebase Hosting

simpl<sub>i</sub>learn

# Lesson 24 - Bonus Working with NgRx in our Project

- Module Introduction
- What is Application State
- What is NgRx
- Getting Started with Reducers
- Adding Logic to the Reducer
- Understanding & Adding Actions
- Setting Up the NgRx Store
- Selecting State
- Dispatching Actions
- Multiple Actions
- Preparing Update & Delete Actions
- Updating Deleting Ingredients
- Expanding the State
- Managing More State via NgRx
- Removing Redundant Component State Management
- First Summary & Clean Up
- One Root State
- Setting Up Auth Reducer & Actions
- Dispatching Auth Actions
- Auth Finished
- And Important Note on Actions
- Exploring NgRx Effects
- Defining the First Effect
- Effects & Error Handling
- Login via NgRx Effects
- Managing UI State in NgRx
- Finishing the Login Effect
- Preparing Other Auth Actions
- Adding Signup
- Adding Signup
- Adding Auto-Login with NgRx
- Adding Auto-Logout
- Finishing the Auth Effects
- Using the Store Devtools
- The Router Store
- Getting Started with NgRx for Recipes
- Fetching Recipe Detail Data
- Fetching Recipes & Using the Resolver
- Fixing the Auth Redirect
- Update, Delete and Add Recipes
- Storing Recipes via Effects
- Cleanup Work
- Wrap Up

simpli·learn

## Lesson 25 - Bonus Angular Universal

> Module Introduction
> Getting Started with Angular Universal
> Working on the App Module
> Adding a Server-Side Build Workflow
> Adding a NodeJS Server
> Pre-Rendering the App on the Server
> Next Steps

## Lesson 26 - Angular Animations

> Introduction
> Setting up the Starting Project
> Animations Triggers and State
> Switching between States
> Transitions
> Advanced Transitions
> Transition Phases
> The void State
> Using Keyframes for Animations
> Grouping Transitions
> Using Animation Callbacks

## Lesson 27 - Adding Offline Capabilities with Service Workers

> Module Introduction
> Adding Service Workers
> Caching Assets for Offline Use
> Caching Dynamic Assets URLs

## Lesson 28 - A Basic Introduction to Unit Testing in Angular Apps

> Introduction
> Why Unit Tests
> Analysing the Testing Setup
> Running Tests
> Adding a Component and some fitting Tests
> Testing Dependencies - Components and Services
> Simulating Async Tasks
> Using 'fakeAsync' and 'tick'
> Isolated vs Non-Isolated Tests

## Lesson 29 - Angular Changes & New Features

> What's New with Angular 8
> A First Look At Angular Elements

## Lesson 30 - Course Roundup

> Course Roundup

# Course End **Project:**

**Build a Product List Web App in Angular**

Implement CRUD functionality in Angular and build an Angular app with the following features:

Product form, product list, view, delete, and update products from the list.

simpl learn

# About **Us:**

Simplilearn is a leader in digital skills training, focused on the emerging technologies that are transforming our world. Our blended learning approach drives learner engagement and is backed by the industry's highest completion rates. Partnering with professionals and companies, we identify their unique needs and provide outcome-centric solutions to help them achieve their professional goals.

For more information, please visit our website:
https://www.simplilearn.com/angular-certification-training

**simplilearn** | simplilearn.com

Founded in 2009, Simplilearn is one of the world's leading providers of online training for Digital Marketing, Cloud Computing, Project Management, Data Science, IT Service Management, Software Development and many other emerging technologies. Based in Bangalore, India, San Francisco, California, and Raleigh, North Carolina, Simplilearn partners with companies and individuals to address their unique needs, providing training and coaching to help working professionals meet their career goals. Simplilearn has enabled over 1 million professionals and companies across 150+ countries train, certify and upskill their employees.

Simplilearn's 400+ training courses are designed and updated by world-class industry experts. Their blended learning approach combines e-learning classes, instructor-led live virtual classrooms, applied learning projects, and 24/7 teaching assistance. More than 40 global training organizations have recognized Simplilearn as an official provider of certification training. The company has been named the 8th most influential education brand in the world by LinkedIn.

**India – United States – Singapore**