



Angular 9 : Online Class

Introduction

By: Sahosoft Solutions

Presented by : Chandan Kumar



About Sahosoft

Sahosoft is the best resource for learning Web Technologies Quickly & Easily. Sahosoft website is all about creativity and innovative work in the field of Technology.

We provide **Online Classes, Online Live Project Training, Corporate Training, Membership Plan, web development course videos and articles**. Sahosoft Online Classes are amazing and easy to learn from basic to advanced level.

Sahosoft provides tutorials of different programming languages and Computer subjects. The main purpose of this Course is to provide quality learning content for students and professionals. we understand your attachment with the content, so committed for delivering you the best possible material.

Sahosoft also provide free videos from my YouTube channel and source code and you are free to use it and make changes.

Here is my YouTube channel link:

<https://www.youtube.com/channel/UCcsUx7ZOL1Sa3oylC29VseA/videos>



Course Introduction

In this course, you will learn how simple it is to use Angular to create maintainable and testable single page applications. You will learn how to: bootstrap your Angular application; use services and create custom services; turn your application into a SPA using routing; and create your own custom elements and handle events using directives.

In this course, you will learn Angular and build responsive, enterprise-strength applications that run smoothly on desktop and mobile devices. Angular provides a robust framework that facilitates the development of richly interactive applications running on multiple platforms. You will also learn how to building components, creating directives, modularizing applications, and building template-driven forms.

You will also learn how to address the challenges you encounter in developing single-page applications with the help of this Angular online class. It will not only make your work easier but be of great help in the advancement of your web development career. Prior to taking this course, you need to have experience in web development as well as in coding with JavaScript.



Course Introduction

By the end of attending this online class, you'll be able to:

- Build real client apps with Angular on your own
- Troubleshoot common compile-time and run-time errors
- Write clean and maintainable code like a professional
- Apply best practices when building Angular apps

Sahosoft



Where to find the Course Source Code

We will provide source on Google Drive..

Sahosoft



What is Angular

The Angular is the newest form of the AngularJS, developed by Google, which is an open-source front-end development platform used for building mobile and desktop web applications. Angular is rewritten by the same team that built AngularJS.

It is a JavaScript framework for building web applications and apps in JavaScript, HTML, and Typescript, which is a superset of JavaScript. The Angular now comes with every latest feature you need to build a complex and sophisticated web or mobile application. It contains features like component, Directives, Forms, Pipes, HTTP Services, Dependency Injection and many more.

Angular is one of the most popular frameworks for building client apps with HTML, CSS and Typescript.



Why Angular

Angular is the next big deal. Being the successor of the massive successful AngularJS framework it's bound to frame the future of frontend development in a similar way. The powerful features and capabilities of Angular permit you to build complex, customizable, modern, responsive and user-friendly web applications. It also enables you to create software quicker and with less effort.

As your application grows, structuring your code in a clean and maintainable and more importantly, testable way, becomes more complex. But your life becomes far easier using a framework like Angular.

Angular 8 is the latest version of the Angular framework and simply an update to Angular 2. Angular is faster than AngularJS and offers a much more flexible and modular development approach. After studying this course you become proficient and able to take full advantage of all those features and start developing incredible applications in a reasonable time. Due to the drastic change between Angular 1 and Angular 7 you don't need to have knowledge about AngularJS to be able to benefit from this course and build your futures projects with Angular.



What You Should Already Know

Before you start studying Angular, you must have basic knowledge of

- HTML, Document Object Model (DOM), CSS, but isn't a must-have
- JavaScript
- Typescript
- It also requires the basic concept of OOPs
- NO Angular 1 or Angular 2 or Angular 4 or Angular 5 or Angular 6 or Angular 7 knowledge is required



Building Blocks of an Angular Application

Following are building blocks of Angular. These are:

- Modules
- Components
- Templates
- Metadata
- Data binding
- Directives
- Services
- Dependency Injection

we will discuss one by one in subsequence chapter in this course.



Basic Architecture of an Angular Application

Angular is a platform for developing web and mobile applications. Angular 2 is not just an update of Angular 1.x but Angular 2.0 and higher is re-written and has many breaking changes. It is completely written in Typescript (to meet ES 6 specifications). There will be a huge learning curve for the developers of Angular 2 and higher. And also, architecture of the Angular 2 and higher is different than Angular 1.x.

Angular is a most popular web development framework for developing mobile apps as well as desktop applications.

Angular framework is also utilized in the cross platform mobile development called IONIC and so it is not limited to web apps only.

Angular is an open source framework written and maintained by angular team at Google and the Father of Angular is Misko Hevery.

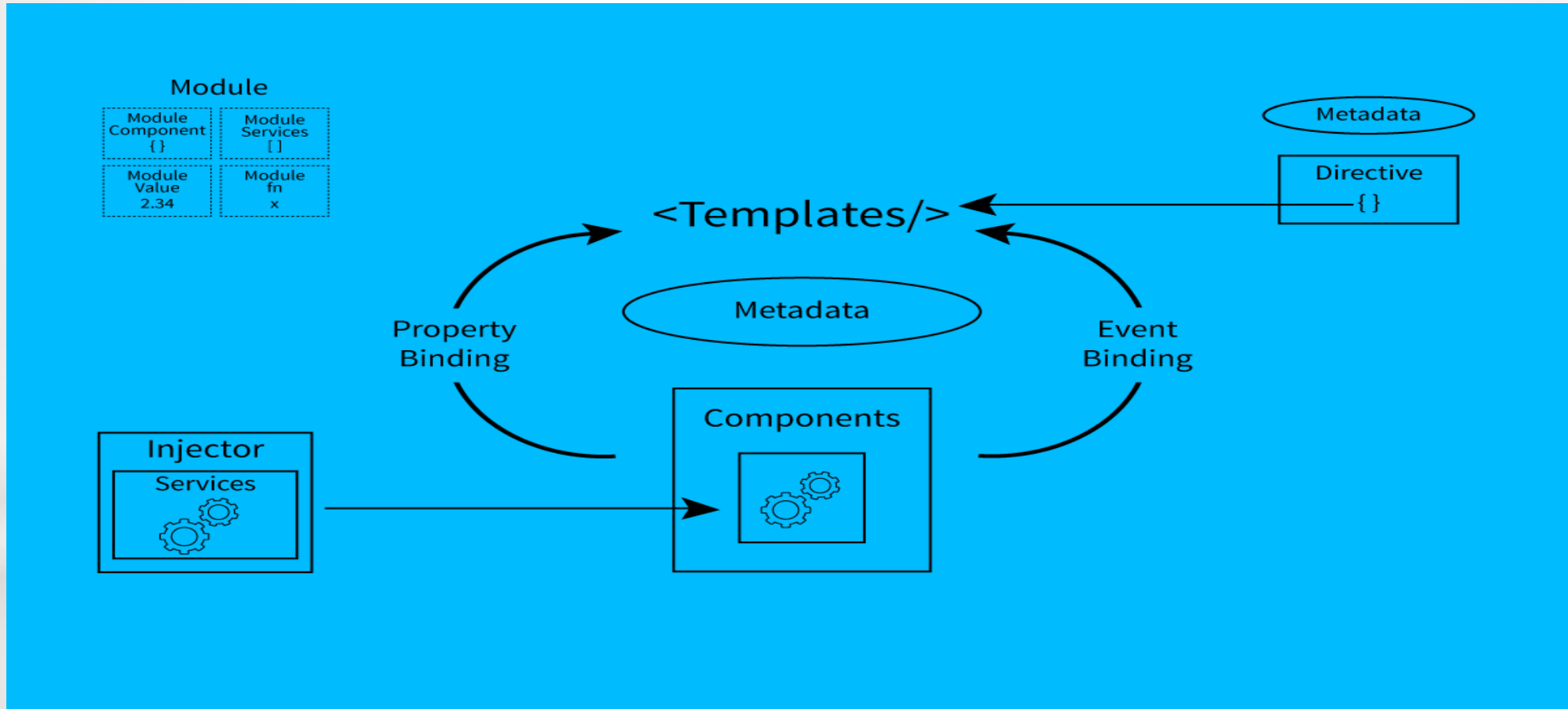
Now, we will discuss the architecture of the Angular 2 and higher.

.



Basic Architecture of an Angular Application

The following diagram shows the architecture of Angular 2 and higher.



Module



Module is the block of code which is designed to perform a single task. We can export the module in form of class. Angular 2 applications have one or more modules. Every Angular application must have at least one module. If Angular application contains only one module, it is referring as root module. Every Angular application has one root module and many more featured modules.

Angular module is a class which is decorated with @NgModule. NgModule takes a single metadata object and its properties describe the module. Following are the important properties of NgModule.

- **exports** - It is the subset of declarations which would be used in the component template of other module.
- *imports* - imports other modules
- **providers** - It is a creator of services. They can be accessible in all the parts of the application.
- **bootstrap** - The root module has to set the bootstrap property. It is used to host all other views.
- **declarations** - It declare the view class that belong to current module. There are three type of view classes supported by Angular components, directives, and pipes..

Component



The component is class with the template that deals with the View of application and it's containing the core logic for the page. We can compare it with the Controller in Angular 1.x. We need to write the application logic inside the class which is used by the View. The component class interacts with the View through Methods and Properties of API.

Component Example

```
1. import { Component } from '@angular/core';
2. @Component({
3.   selector: 'test-app',
4.   template: '<h1>This is my First Angular 2 Application</h1>' +
5.     '<br/>' +
6.     '<input #txtName type = "text" (keyup)="0" />' +
7.     '<br/>' +
8.     '<p>You have Enter: {{txtName.value}}</p>'
9. })
10.   export class AppComponent {
11.
12.   }
```

Metadata



Metadata is the way of defining the processing of a class. In TypeScript, we can define metadata by using decorator. For example, if we define any component in Angular application, we need to tell Angular that this is the component, by using metadata of the class (using @Component decorator).

Metadata example

```
1. @Component({
2.   selector: 'test-app',
3.   template: '<h1>This is my First Angular 2 Application</h1>' +
4.     '<br/>' +
5.     '<input #txtName type = "text" (keyup)="0" />' +
6.     '<br/>' +
7.     '<p>You have Enter: {{txtName.value}}</p>'
8. })
```



Template

The template is the component View that tells Angular how to display the component. It looks like normal HTML.

```
3. template: '<h1>This is my First Angular 2 Application</h1>' +  
4. '<br/>' +  
5. '<input #txtName type = "text" (keyup)="0" />' +  
6. '<br/>' +  
7. '<p>You have Enter: {{txtName.value}}</p>'  
8. })
```