Angular 18 Online Training

sahosoft

Online Learning Platform

01

By: Chandan Kumar

sahosoft

Online Learning Platform

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/video

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

Where to find the Course Source Code

We will provide source on portal/app

01

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 18 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 7,8, 9,10,11,12,13,14,15,16,17 knowledge is required

Building Blocks of an Angular Application

Following are building blocks of Angular. These are:

- **Modules – No need in Angular 17, 18**
- 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.

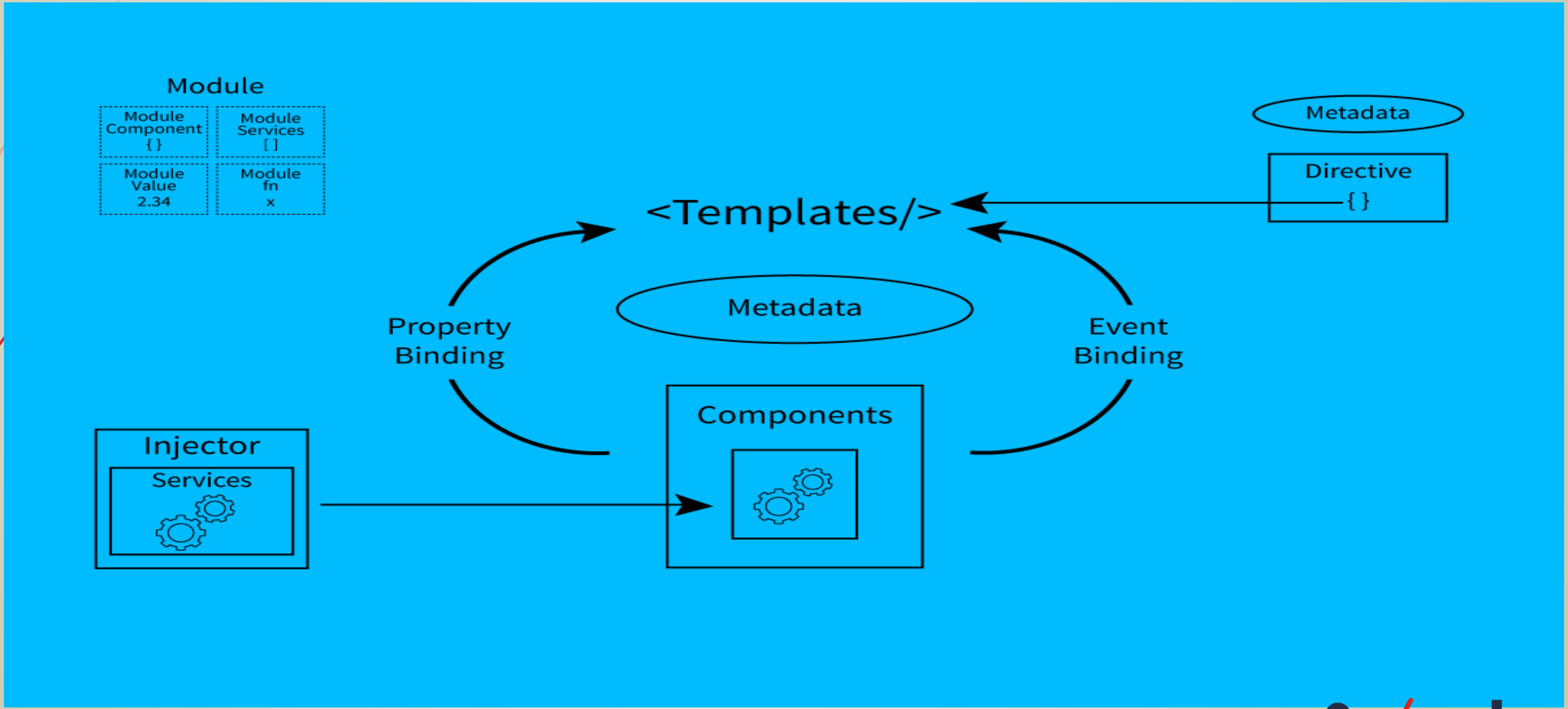
Angular is an open source framework written and maintained by angular team at Google and the Father of Angular is Misko Hevery.

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.

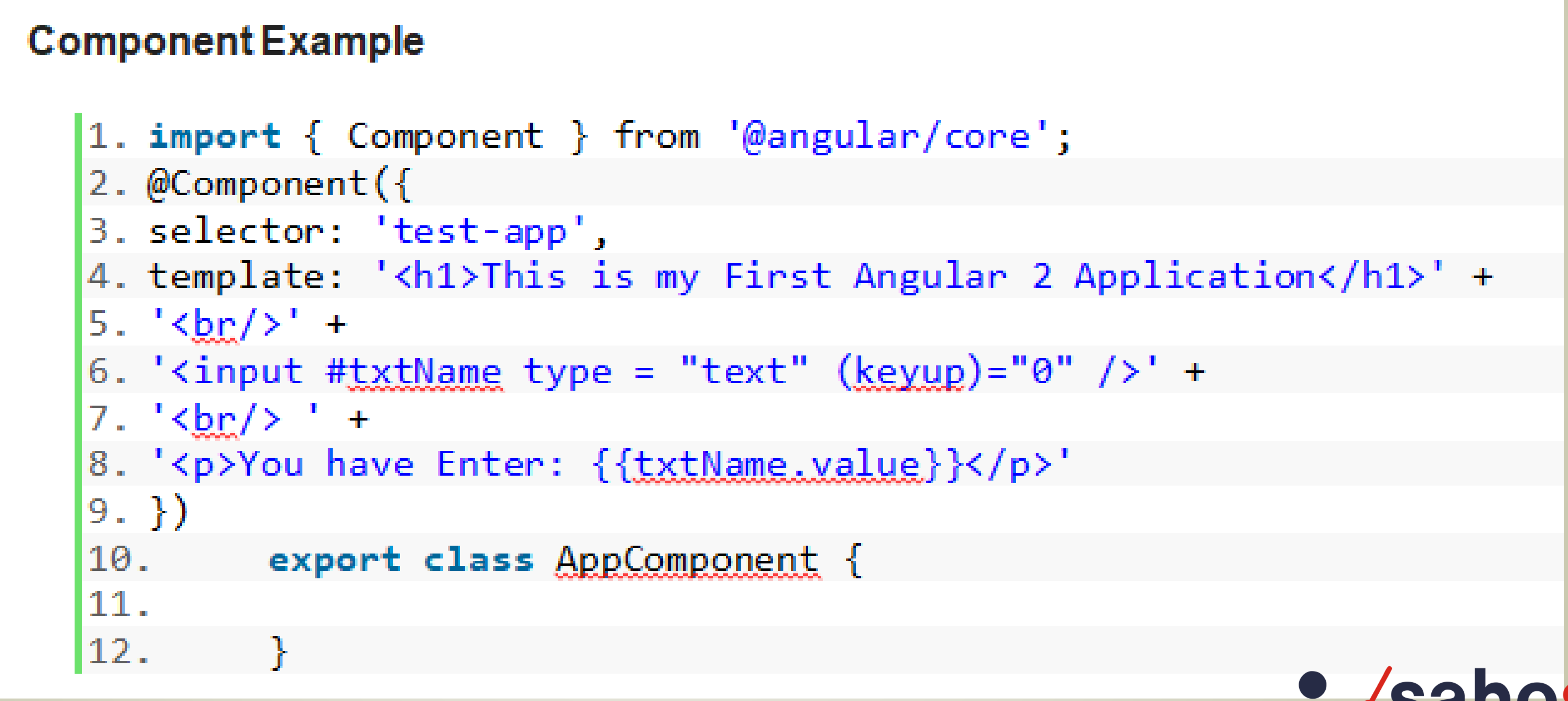
Angular module is a class which is decorated with @NgModule. NgModule takes a single metadata object and its properties describe the module. Following are the important properties of NgModule.

- ***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.



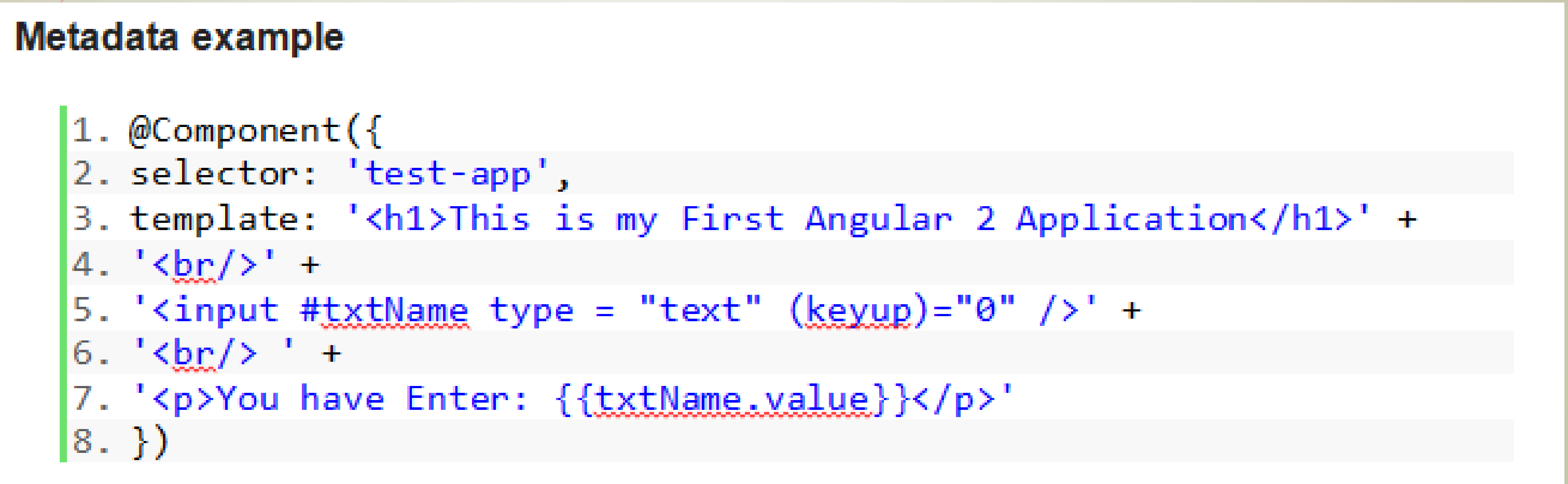
Component Example

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

    }
```

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).

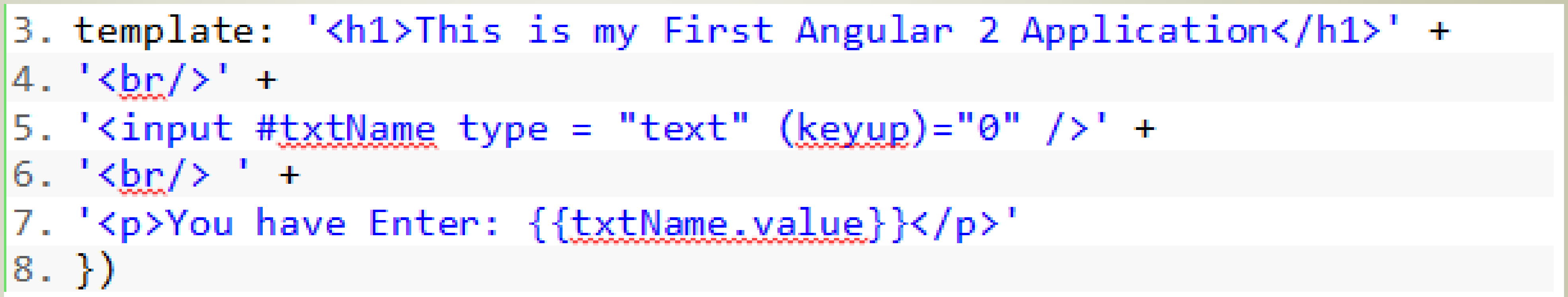


Metadata example

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

Template

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



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

Data Binding

Data binding is a powerful feature of software development technologies. Data bind is the connection bridge between View and the business logic (View Model) of the application. Data binding in AngularJS is the automatic synchronization between the Model and View.

There are four type of binding supported by Angular 2 application,

- ***Interpolation*** - It displays the component value within the HTML tags which is also referred as Expression in Angular 1.x.
- ***Property Binding*** - It passes the value of property from the parent to the property of the child.
- ***Event Binding*** - It fires the event when we click on the components method name.
- ***Two-way Binding*** - It is an important form that combines event and property binding in single notation by using ngModel directive.

.

-

.

Service

Service in Angular is a function or an object that can be used to share the data and the behavior across the application. It is JavaScript function which is used to perform a specific task. It includes the function, values, or any other feature required by the application. Typical examples of services are logging service, data service, message service etc. There is no base class for service.

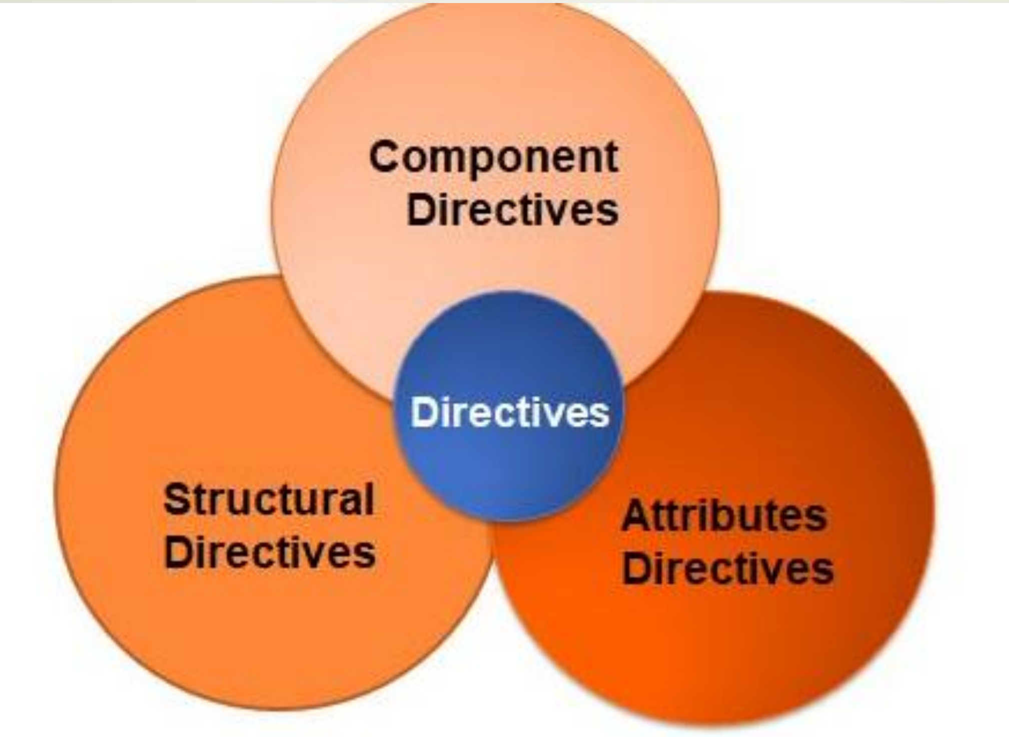
Directive

Basically, directives are used to extend the power of the HTML attributes and to change the appearance or behavior of a DOM element.

Directive in Angular is a javascript class, which is declared as @directive. Angular has 3 types of directives, which are listed below –

-

.



sahosoft
Online Learning Platform

Component Directives

This is the special directive which is always present in an angular app where each of our HTML page associated with this directive.(The directive with a template.)

Component Directive is a class with @Component decorator function. As we know that angular app contains at least one component called AppComponent which is present inside the App-compoennt.ts file.

In this file, we see with the help of selector, @Component which is a decorator function is used to create a component directive.

Even if we create our own component, we will use this decorator function and creating a component directive. It is not possible to render our template without using “selector”property, so with the help of this property, we will create a component directive.

Structural Directives

Structural directive modifies or manipulates the structure of DOM by adding or removing DOM elements. We can say that basically, it works on the structure of a DOM.

Angular provides several structural directives which have a * sign before the directive.

For example, *ngIf and *ngFor, *ngSwitch.

@if @else, @elseif

@for

@switch

Dependency Injection

Dependency Injection is a software design pattern in which objects are passed as dependencies. It helps us remove the hard coded dependencies, and makes dependencies configurable. Using Dependency Injection, we can make components maintainable, reusable, and testable.

Point to remember about Dependency Injection,

- It is stimulated into the Angular framework so that it can be use anywhere in an application.
- The injector is a main mechanism to maintain the service instance and can be created using a provider.
- The provider is the way of creating a service.
- We can register the providers along with injectors

Angular JS 1.x

- AngularJS is also referred to as “Angular.js” or Angular 1.x
- It is a JavaScript-based framework which is open-source and used in front-end web application development.
- It is mainly maintained by Google and by a community of individuals and corporations to address many of the problems faced in developing single-page applications.
- It mainly built to simplify both the development and the testing of applications by providing a framework for client-side model–view–controller (MVC) and model–view–ViewModel (MVVM) architectures, along with components commonly used in rich Internet applications.
- The current stable version is 18
- Angularjs development code is written in JavaScript

Angular 3 was skipped

The reason behind this is that version mismatch between @angular/core, @angular/compiler and @angular/router libraries. The core and router for Angular 2 version are like following:

Name	Version
@angular/core	v2.3.0
@angular/compiler	v2.3.0
@angular/compiler-cli	v2.3.0
@angular/http	v2.3.0
@angular/router	v3.3.0

Angular 3 was skipped

Now the problem is with @angular/router, which is already in a 3.X version. These problems occur because of some active and huge developments on the router section, like route-preload.

Now, the launch of Angular as version 3, with its path in version 4 will create confusion.

To avoid this confusion, they decided to skip version 3 and release it with version 4.0.0 so that every major unit was on the right track.

Angular 4

For most applications, this release is backward compatible with 2.x.x.Angular 2 and was released in March 2018 and there is no major change in Angular 4 from Angular 2. Angular 4 is not the complete rewritten form of Angular 2.

The Angular team has laid emphasis on making angular apps faster and compact.

- **Under the hood changes:** By the new changes, the size of the generated code for your components is reduced by around 60% in most cases.
- Faster Compilation
- Better Bug fixes Alert.
- **TypeScript 2.1 and 2.2 compatibility:** Finally We can use typescript 2.1 or earlier only upto typescript 1.8 was supported.
- ***ngIf/else:** Now the feature to use else clause is also available.
- For email validation in angular 4, there is No need to write a pattern.

Angular 5

Angular 5 was launched in Nov 2018. According to its speed and size, It was way faster and of smaller size than that of Angular 4. Here’s the failure of some of the biggest changes in v5. For the full list, please see the changelog. Following features were introduced in Angular 5.

HttpClient API − HttpClient API was introduced to simplify the HTTP library for Angular applications that rest on the XMLHttpRequest interface exposed by browsers. It is
much secure, faster, efficient than HTTP library and provides additional features like testability, typed request and response objects

Multiple export aliases − To ease the migration process a component can be exported with the help of multiple aliases.

Angular 5

Internationalized Pipes for Number, Date, and Currency −Earlier Angular versions was dependant on the browsers to get the number, date and currency format. This resulted in inconsistency for users but in v5 pipes were updated for better standardization.

Lambda support − lambda expressions with proper names can be used in place of functions.

Build Optimizer - Build Optimizer is also introduced. It contains Angular optimizations applicable to JavaScript code as a TypeScript transform pipeline which optimizes the build size and improves the application speed. Angular CLI uses Build Optimizer automatically.

Improved Compiler − Angular 5 onwards, the compiler supports incremental compilation leading to faster compilation. The compiler uses TypeScript transforms, a new feature of TypeScript 2.3 available onwards.