

# Angular Directives

2 Comments / 3 minutes of reading / March 9, 2023



[Angular Tutorial](#)

[ngFor Directive](#)



[Child Component in Angular](#)

Angular Directives are the most important features of Angular. In this, tutorial We will look at three types of directives that Angular supports like Component, Structural, and Attribute Directives. We also look at the few of the most commonly used Angular directives.

[Download Source Code](#)

## Table of Contents

[What is Angular Directive](#)

[Component Directive](#)

[Structural Directives](#)

[Commonly used structural directives](#)

[Attribute Directives](#)

[Commonly used Attribute directives](#)

[Building Custom Directives](#)

[Summary](#)

## What is Angular Directive

The Angular directive helps us to manipulate the DOM. You can change the appearance, behavior, or layout of a DOM element using the Directives. They help you to extend HTML

There are three kinds of directives in Angular:

1. Component Directive
2. Structural directives
3. Attribute directives

## Component Directive

Components are special directives in Angular. They are the directive with a template (view) We covered [how to create Components in Angular](#) tutorial.

## Structural Directives

Structural directives can change the DOM layout by adding and removing DOM elements. All structural Directives are preceded by Asterix symbol

### Commonly used structural directives

#### ngFor

The [ngFor](#) is an Angular structural directive, which repeats a portion of the HTML template once per each item from an iterable list (Collection). The [ngFor](#) is similar to [ngRepeat](#) in AngularJS

## Example of ngFor

```
1
2 <tr *ngFor="let customer of customers;">
3   <td>{{customer.customerNo}}</td>
4   <td>{{customer.name}}</td>
5   <td>{{customer.address}}</td>
6   <td>{{customer.city}}</td>
7   <td>{{customer.state}}</td>
8 </tr>
9
10
```

You can read more about the [Angular ngFor Directive](#) tutorial.

## ngSwitch

The [ngSwitch](#) directive lets you add/remove HTML elements depending on a match expression. [ngSwitch](#) directive used along with [ngSwitchCase](#) and [ngSwitchDefault](#)

## The example of ngSwitch

```
1
2 <div [ngSwitch]="Switch_Expression">
3   <div *ngSwitchCase="MatchExpression1"> First Template</div>
4   <div *ngSwitchCase="MatchExpression2">Second template</div>
5   <div *ngSwitchCase="MatchExpression3">Third Template</div>
6   <div *ngSwitchCase="MatchExpression4">Third Template</div>
7   <div *ngSwitchDefault?>Default Template</div>
8 </div>
9
```

You can read more about the [Angular ngSwitch Directive](#) tutorial.

## ngIf

The [ngIf](#) Directives is used to add or remove HTML elements based on an expression. The expression must return a boolean value. If the expression is false then the element is removed, else the element is inserted

### Example of ngIf

```
1  
2 <div *ngIf="condition">  
3   This is shown if condition is true  
4 </div>  
5
```

You can read more about [Angular ngIf Directive](#) tutorial.

## Attribute Directives

An Attribute or style directive can change the appearance or behavior of an element.

### Commonly used Attribute directives

#### ngModel

The ngModel directive is used to achieve the [two-way data binding](#). We have covered ngModel directive in [Data Binding in Angular Tutorial](#)

## ngClass

The [ngClass](#) is used to add or remove the CSS classes from an HTML element. Using the [ngClass](#) one can create dynamic styles in HTML pages

### Example of ngClass

```
1  
2 <div [ngClass]="['first second']">...</div>  
3
```

## ngStyle

[ngStyle](#) is used to change the multiple style properties of our HTML elements. We can also bind these properties to values that can be updated by the user or our components.

### Example of ngStyle

```
1  
2 <div [ngStyle]="{'color': 'blue', 'font-size': '24px', 'font-weight': 'bold'}">  
3   some text  
4 </div>  
5
```

## Building Custom Directives

You can also build custom directives in Angular. The Process is to create a JavaScript class and apply the **@Directive** attribute to that class. You can write the desired behavior in the class.

## Summary

In this tutorial, we introduced you to the **directives in Angular**. In the next few tutorials, we will look at some of the important directives in detail

[Download Source Code](#)

## Read More

1. [Angular Tutorial](#)
2. [Angular Directives](#)
3. [ngFor](#)
4. [ngSwitch](#)
5. [ngIf](#)
6. [ngClass](#)
7. [ngStyle](#)
8. [ngFor Trackby](#)
9. [Custom Directive](#)



[Angular Tutorial](#)

[ngFor Directive](#) →

[Child Component in Angular](#)

## Related Posts