Introduction to Directives

A directive is a custom HTML element that extends the power of HTML. Whenever the Angular compiler finds a directive in the DOM(Document Object Model) internally it is a function that gets executed.

Angular directives are divided into three types.

* Component
* Structural
* Attribute

The component in Angular which you learnt earlier is internally a directive. You will now understand about Structural and Attribute directives.

Structural directives are responsible shaping or reshaping the structure of DOM by adding, removing, or manipulating elements.

The most commonly used structural directives are

* ngIf
* ngFor
* ngSwitch

gIf takes a boolean expression as input. It adds or removes elements from DOM based on the value of the expression.

Syntax:

1. \*ngIf='expression'

If the expression evaluates to true then the element will be added to DOM, else it will be removed.

ngFor is a way to present a list of items. It can be attached to a block of HTML that defines how a single item should be displayed. Angular will use that block as a template for rendering each item in the list.

Syntax:

1. \*ngFor = 'let item of itemlist'

Here item is a temporary variable to iterate the itemlist.

You will check if there are any products to be displayed to the user in QuickKart and create a template to iterate and display the products in the next demo.

ngSwitch is another structural directive in Angular which allows us to add or remove DOM elements. It is similar to switch statement of Javascript.

Syntax:

1. <container\_element [ngSwitch]="switch\_expression">
2. <inner\_element \*ngSwitchCase="match\_expression\_1">...</inner\_element>
3. <inner\_element \*ngSwitchCase="match\_expression\_2">...</inner\_element>
4. <inner\_element \*ngSwitchCase="match\_expression\_3">...</inner\_element>
5. <inner\_element \*ngSwitchDefault>...</element>
6. </container\_element>

Observe the below code where choice is a variable declared in the component.

1. <div [ngSwitch]="choice">
2. <div \*ngSwitchCase="'1'">One</div>
3. <div \*ngSwitchCase="'2'">Two</div>
4. <div \*ngSwitchCase="'3'">Three</div>
5. <div \*ngSwitchCase="'4'">Four</div>
6. <div \*ngSwitchCase="'5'">Five</div>
7. <div \*ngSwitchDefault>This is Default</div>
8. </div>

Here, you are binding the ngSwitch directive to the div. Next, we have ngSwitchCase with matching expressions. Whenever the value of choice matches the expression, the element attached to it will be displayed. The ngSwitchDefault is used to display when all the other cases fail.

Here, if you observe [ngSwitch] the [] represent property binding which you will be learning in detail in the upcoming demos.

Attribute directive changes the appearance/behavior of a component/element

Following are built-in attribute directives

* ngStyle
* ngClass

ngStyle directive is used to modify a component/element’s style. If there are more than one css styles to be applied, you can use ngStyle attribute.

Syntax:

1. [ngStyle] = "expression"

The expression can accept an array, string or an object.

ngStyle directive is used to modify a component/element’s style. If there are more than one css styles to be applied, you can use ngStyle attribute.

Syntax:

1. [ngStyle] = "expression"

The expression can accept an array, string or an object.