BEEVA - CURSO DE ANGULAR 2015
TEORÍA (KATA NIVEL BÁSICO)
Fernando Castro / Juan Ferrer

DIRECTIVAS

DEFINICIÓN

Según la definición que nos proporcionan los propios creadores, las directivas son, a alto nivel, marcadores en el documento DOM (como pueden ser un atributo, un nombre, un comentario o una clase CSS), que nos permiten adjuntar nuevo comportamiento a ese elemento, o incluso transformarlo tanto a él como a sus descendientes.

Con un lenguaje un poco más coloquial, no deja de ser una forma de **extender nuestro HTML**, incluyendo pequeñas cantidades de código o incluso funcionalidades completas.

Las directivas tienen varias funciones en nuestro código, que nos van a permitir:

Reutilización de código: podremos definir partes de código que se repitan en varias zonas de nuestra aplicación, las cuales serán llamadas con una única línea desde las vistas.

Manejo del DOM: sólo podremos realizar cambios sobre el DOM a través de directivas, de tal forma que encapsularemos este tipo de operaciones.

NATIVAS

Existen una gran cantidad de directivas proporcionadas por Angular (nativas) que nos van a permitir construir nuestro código a gran velocidad. Vamos a dar un repaso rápido a las más importantes, agrupándolas por funcionalidad, de tal forma que su aprendizaje sea más efectivo:

DE ESTRUCTURA

ng-app
Con ella se indica el inicio de una aplicación en Angular, define el elemento root de la aplicación ($rootScope), y normalmente se coloca en el elemento principal de la página (en las etiquetas *<body>* o *<html>*).
Sólo un ng-app por documento HTML. Si quisiéramos definir varios en un solo documento, habría que utilizar *angular.bootstrap* (que es la forma de iniciar una aplicación de forma manual), pero no es recomendable (siguiendo el manual de buenas prácticas).
Ejemplo

ng-view
Es un complemento perfecto para el servicio *ng-route*, encargándose de renderizar la plantilla html de la ruta actual dentro de la capa principal. Se actualiza según cambie la ruta, de acuerdo a la configuración mostrada en el servicio *$route*.
Sólo puede definirse un ng-view por aplicación.
Ejemplo

{{ }} / ng-bind
Con estas llaves creamos un bindeo directo desde alguna variable del *$scope*, de tal forma que se actualizará automáticamente cada vez que cambie el valor de la variable.
Ejemplo

Si queremos que al cargar el HTML no se vea la variable sin evaluar, utilizaremos ***ng-bind***.
Ejemplo

ng-cloak
Con esta directiva sólo se muestra el valor de un binding cuando esté evaluado, sirve para prevenir el flash del valor no cargado al utilizar {{ }}.
Ejemplo

CONDICIONALES Y BUCLES

ng-if
Nos permite crear o eliminar elementos del DOM de acuerdo al cumplimiento de una determinada expresión. Si se cumple (true) se inserta el elemento, en caso contrario (false) se elimina.
Ejemplo

ng-else
No existe como directiva nativa, pero se puede utilizar como directiva custom:
https://github.com/zachsnow/ng-elif

ng-show / ng-hide
Es parecida a las anteriores, pero en este caso no crea o elimina el elemento del DOM, simplemente lo oculta mediante una clase CSS predefinida de Angular, *.ng-hide* (seteando el atributo display = none).
Ejemplo

ng-switch
Permite mostrar unos elementos u otros en función de una determinada condición.
Para utilizar esta directiva necesitaremos definir además:

on='expresion': el campo dónde evaluar la condición

ng-switch-when: el valor que debe tener el campo indicado en *on* para mostrar el contenido

ng-switch-default: contenido a mostrar por defecto

Ejemplo

ng-disabled
Para poder habilitar o deshabilitar cualquier elemento.

HTML Code
```
<label>Click me to toggle: <input type="checkbox"
ng-model="checked"></label><br/>
<button ng-model="button" ng-disabled="checked">Button</button>
```

ng-repeat

Lo utilizaremos para instanciar una plantilla por cada ítem asociado a una colección, es decir, nos permite repetir n veces partes definidas de código, pudiendo además aprovechar una serie de propiedades asociadas:

$index: índice de cada elemento repetido
$first: será true si el elemento en cuestión es el primero de la lista
$middle: será true si el elemento se encuentra entre el primero y el último
$last: será true si el elemento es el último de la lista
$event: será true si el índice es par
$odd: será true si el índice es impar

Ejemplo

ESTILOS

ng-class

Será la forma de cargar diferentes estilos de manera dinámica.
Las clases duplicadas no se añaden.
Se pueden cargar estilos de tres formas distintas:

String (con las clases separadas por espacios)
Object (pareja key:value, se aplica el estilo key si se cumple el value – true/false)
Array (cada elemento del array es una clase u objeto)

Ejemplo en la API de Angular

FORMULARIOS

ng-form

Permite anidar formularios unos dentro de otros.
Un formulario es válido cuando todos sus formularios anidados son válidos.
En el documento adjunto **control_errores.odt** podréis observar en profundidad la forma de controlar los fallos más comunes en los formularios.
Para enviar un formulario utilizaremos las directivas *ng-submit* o en *ng-click* (no se deben usar las dos a la vez, para evitar un doble envío).
Ejemplo

ng-ckecked

Con esta directiva podemos cambiar el valor de un input de tipo checkbox, mediante true/false.
No debe utilizarse junto a *ng-model*.
Ejemplo en la API de Angular

ng-options

Se usa para generar dinámicamente una lista de opciones dentro de un input de tipo select, usando para ello un array u objeto asociado.
Ejemplo

ng-value
Es una directiva específica para utilizar con los input de tipo select y radio. Almacena la opción seleccionada en el atributo value de los input mencionados.
Ejemplo en la API de Angular

ng-submit
Lo utilizaremos para el envío de los datos de un formulario.
Ejemplo en la API de Angular

EVENTOS

ng-click
Permite especificar el comportamiento cuando un elemento es pulsado.
Se trata de uno de los elementos más utilizados, enlazando la pulsación con funciones definidas en el controlador, o con código marcado en la propia vista.
Ejemplo

Además del ya mencionado, relacionados con los eventos de ratón podemos utilizar **ng-dblclick**, **ng-mousedown**, **ng-mouseenter**, **ng-mouseleave**, **ng-mousemove**, **ng-mouseover** y **ng-mouseup**.

Si lo que necesitamos es trabajar con teclado, podemos contar con **ng-keydown**, **ng-keypress** y **ng-keyup**.

ng-change
Se encarga de evaluar una expresión en el momento en el que el usuario cambia el modelo.
Es necesario un campo con un *ng-model* asociado para poder trabajar con la directiva.
Ejemplo

OTRAS DIRECTIVAS

ng-src / ng-href
Para cargar diferentes páginas o asociar elementos externos, pudiendo usar binding sin miedo a problemas de carga (se producen problemas al intentar cargar una variable que todavía no ha cogido su valor final). Si no se bindea ningún valor, se pueden utilizar las propiedades originales src y href.

HTML Code
```
<img ng-src="http://www.gravatar.com/avatar/{{hash}}" alt="Description" />

<a id="link-6" ng-href="{{value}}">link</a> (link, change location)
```

ng-init
La utilizaremos para inicializar datos en la aplicación.
No es recomendable su uso en cualquier caso, sólo en algunos muy determinados, como puede ser utilizar propiedades especiales en un *ng-repeat*. Para inicializar valores en un Scope, es mejor hacerlo directamente en el controlador asociado.
Ejemplo en la API de Angular

ng-include
Permite incluir fragmentos HTML, compilarlos y mostrarlos de forma dinámica.
Soporta animaciones y crea su propio Scope asociado.
Ejemplo

BIBLIOGRAFÍA

Se ha recurrido al abundante material disponible en la red para explicar de la mejor manera posible los diferentes conceptos que aparecen en esta documentación:

Página principal de Angular: https://angularjs.org/
API de Angular: https://docs.angularjs.org/api/
Aprendiendo directivas en Angular:
http://frontendlabs.io/2287--aprendiendo-directivas-en-angularjs
Directivas en Angular: http://uno-de-piera.com/directivas-en-angular/

Ejemplos en Angular:
- ng-app: http://jsbin.com/ICOzeFI/2/edit?html,output
- ng-view: http://learnwebtutorials.com/simple-example-of-ng-view-in-angularjs
- {{}}: http://jsbin.com/ODUxeho/1/edit?html,output
- ng-bind: http://jsbin.com/esihUJ/1/edit?html,output
- ng-cloak: http://jsbin.com/AJEboLO/1/edit?html,output
- ng-if: http://jsbin.com/ezEcamo/1/edit?html,output
- ng-show/ng-hide: http://jsbin.com/ihOkagE/1/edit?html,output
- ng-switch: http://jsbin.com/AVihUdi/2/edit?html,output
- ng-class: https://docs.angularjs.org/api/ng/directive/ngClass
- ng-form: http://jsbin.com/UduNeCA/1/edit?html,output
- ng-checked: https://docs.angularjs.org/api/ng/directive/ngChecked
- ng-options: http://jsfiddle.net/qWzTb/
- ng-value: https://docs.angularjs.org/api/ng/directive/ngValue
- ng-submit: https://docs.angularjs.org/api/ng/directive/ngSubmit
- ng-click: http://www.w3schools.com/angular/tryit.asp?filename=try_ng_events_click
- ng-change: http://codepen.io/wurmr/pen/Hgyso
- ng-repeat: http://jsbin.com/akuYUkey/1/edit?html,output
- ng-init: https://docs.angularjs.org/api/ng/directive/ngInit
- ng-include: http://jsfiddle.net/mrajcok/mfha6/