



EBook Gratis

APRENDIZAJE angularjs-directive

Free unaffiliated eBook created from
Stack Overflow contributors.

#angularjs-
directive

Tabla de contenido

Acerca de	1
Capítulo 1: Empezando con la directiva angularjs	2
Observaciones.....	2
Examples.....	2
Instalación o configuración.....	2
Construyendo un componente reutilizable.....	2
Tu primera directiva.....	3
Mensaje emergente de éxito / error que utiliza una función de enlace simple.....	4
Capítulo 2: Directivas de uso común	6
Examples.....	6
ngConfirmClick: Confirmar antes de evaluar la expresión.....	6
Creditos	7

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [angularjs-directive](#)

It is an unofficial and free angularjs-directive ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official angularjs-directive.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con la directiva angularjs

Observaciones

Las directivas AngularJS son elementos personalizados en HTML (como un atributo, nombre de elemento, comentario o clase CSS) que le dicen a AngularJS que adjunte un comportamiento específico a ese elemento DOM, o incluso que transforme el elemento DOM y sus elementos secundarios. En resumen, cuando creamos una directiva, AngularJS tratará ese elemento de manera diferente.

Examples

Instalación o configuración

Directivas viene con la propia biblioteca AngularJS. Se puede crear una directiva de ejemplo como:

```
angular.module('simpleDirective', [])
.directive('helloData', function() {
  return {
    template: 'Hello, {{data}}'
  };
});
```

Y puede ser utilizado como:

JS:

```
angular.module('app', ['simpleDirective'])
.controller('Controller', ['$scope', function($scope) {
  $scope.data = 'World';
}])
```

HTML

```
<div ng-controller="Controller">
  <div hello-data></div>
</div>
```

Se compilará como:

Hola Mundo

Construyendo un componente reutilizable.

Las directivas se pueden utilizar para construir componentes reutilizables. Aquí hay un ejemplo de un componente de "carpeta de usuario":

userBox.js

```
angular.module('simpleDirective', []).directive('userBox', function() {
  return {
    scope: {
      username: '=username',
      reputation: '=reputation'
    },
    templateUrl: '/path/to/app/directives/user-box.html'
  };
});
```

Controller.js

```
var myApp = angular.module('myApp', ['simpleDirective']);

myApp.controller('Controller', function($scope) {
  $scope.user = "John Doe";
  $scope.rep = 1250;
});
```

myPage.js

```
<html lang="en" ng-app="myApp">
  <head>
    <script src="/path/to/app/angular.min.js"></script>
    <script src="/path/to/app/controllers/Controller.js"></script>
    <script src="/path/to/app/directives/userBox.js"></script>
  </head>

  <body>

    <div ng-controller="Controller">
      <user-box username="user" reputation="rep"></user-box>
    </div>

  </body>
</html>
```

user-box.html

```
<div>{{username}}</div>
<div>{{reputation}} reputation</div>
```

Tu primera directiva

Nuestra primera directiva de elementos no hará mucho: solo calculará $2+2$ y se llamará en html así:

```
<my-calculator></my-calculator>
```

Observe que el nombre de la directiva es `myCalculator` (en CamelCase), pero en html se usa como `my-calculator` (en caso de error).

Como queremos que nuestra directiva se use como elemento html, usaremos `restrict: 'E'` .

Cada directiva tiene la plantilla que será compilada e insertada. Nuestra directiva es muy simple, por lo que insertaremos nuestro html como cadena en un parámetro de `template` .

```
// directives/my-calculator.js

angular.module('exampleApp', [])
.directive('myCalculator', function() {
  return {
    restrict: 'E',
    template: '<span> My directive can calculate 2+2: {{2+2}} </span>'
  };
});
```

HTML

```
<!DOCTYPE html>
<html ng-app="exampleApp">

  <head>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/angular.min.js"></script>
    <script src="my-calculator.js"></script>
  </head>

  <body>
    Here is my first directive:
    <my-calculator></my-calculator>
  </body>

</html>
```

El resultado se verá así:

Aquí está mi primera directiva: Mi directiva puede calcular 2 + 2: 4

Si quieres jugar con el ejemplo en vivo, ve a [plunkr](#) .

Mensaje emergente de éxito / error que utiliza una función de enlace simple

La función de enlace es la mejor manera en directivas personalizadas para manipular DOM. Toma tres atributos como entrada (alcance, elemento, atributo) en secuencia

Ámbito: su ámbito de aplicación local objeto de directiva.

elemento: elemento html en el que se utiliza la directiva.

atributo: da acceso a todos los atributos utilizados en el elemento referido.

```

// on success call or similarly error, warning, info in controller
$scope.message={
    text: "Saved Successfully",
    type: "SUCCESS"
};

<user-info msg="message"> </user-info> //in html

var mainApp = angular.module("mainApp", []);
mainApp.directive('userInfo', function() {
    var directive = {};
    directive.restrict = 'E';

    directive.scope = {
        message : "=msg"
    },

    directive.link = function(scope, element, attributes) {
        if(scope.message.type==='SUCCESS')
            scope.message.text = 'SUCCESS: '+scope.message.text+' !';
        else if(scope.message.type==='ERROR')
            scope.message.text = 'ERROR: '+scope.message.text+' !';
        else if(scope.message.type==='WARNING')
            scope.message.text = 'WARNING: '+scope.message.text+' !';
        else if(scope.message.type==='INFO')
            scope.message.text = 'INFO: '+scope.message.text+' !'

        element.on('click', function(event) { //on click of div pop-up will smoothly
close
            $(this).fadeOut();
            });
        },
        directive.template = '<div ng-class={{message.type}}>'+ // one can
create different bg-color as per type of message and width/height
            '<div class="message-text">{{message.text}}</div>'+ //message
text will be printed
            '</div>';

        return directive;
    });
};

```

Lea Empezando con la directiva angularjs en línea: <https://riptutorial.com/es/angularjs-directive/topic/1855/empezando-con-la-directiva-angularjs>

Capítulo 2: Directivas de uso común

Examples

ngConfirmClick: Confirmar antes de evaluar la expresión.

Descripción:

Evaluar la expresión después de la confirmación del usuario.

Argumentos:

- `ng-confirm-click` :(*expresión*) Expresión para evaluar cuando se confirma.
- `ng-confirm-message` :(*plantilla*) Mensaje que se mostrará en el cuadro de diálogo de confirmación.

Código:

```
Directives.directive("ngConfirmClick", ["$parse", "$interpolate", function ($parse, $interpolate)
{
  return {
    restrict:"A",
    priority:-1,
    compile:function(ele, attr){
      var fn = $parse(attr.ngConfirmClick, null, true);
      return function ngEventHandler(scope, ele) {
        ele.on('click', function (event) {
          var callback = function () {
            fn(scope, {$event: "confirm"});
          };
          var message = $interpolate(attr.ngConfirmMessage)(scope) || 'Are you
sure?';

          if(confirm(message)) {
            if (scope.$root.$$phase) {
              scope.$evalAsync(callback);
            } else {
              scope.$apply(callback);
            }
          }
        });
      };
    }
  };
});
```

[Ejemplo de trabajo](#)

Lea Directivas de uso común en línea: <https://riptutorial.com/es/angularjs-directive/topic/5099/directivas-de-uso-comun>

Creditos

S. No	Capítulos	Contributors
1	Empezando con la directiva angularjs	Asim K T , Charlie H , Community , Daniel , ganqqwerty , Tjaart van der Walt , Vishal Singh
2	Directivas de uso común	MMhunter