

Directivas propias

Directivas propias

- Como en los controladores, las directivas se registran en los módulos
- Para registrar una directiva se utiliza la API `module.directive`

```
myModule.directive('greeting', function() {...});
```

Directivas propias

- Una directiva es un nuevo markup al que se le asocia un comportamiento
- Es decir que lleva asociado un template angular

Directivas propias

```
<body ng-app="myApp">  
  <div ng-controller="greetingCtrl">  
    <greeting></greeting>  
  </div>  
</body>
```

Directivas propias

```
var myApp = angular.module('myApp', []);  
myApp.controller('greetingCtrl', function ($scope){  
    $scope.greeting = function(){ $scope.greeting = 'Hola!!!  
Waaasssup?';}  
});
```

Directivas propias

```
myApp.directive('greeting', function() {  
  return {  
    restrict: 'E',  
    template: '<button type="submit" class="btn btn-default"  
ng-click="greetMe()"> Greet Me </button> {{ greeting }}'  
  };  
});
```

Ejercicios

- Utilizando "gna-como-directiva.html" crea una app Angular que esté compuesta de una directiva `<gna></gna>` creada por ti, que agrupe el código html que tenías hasta ahora

Directivas propias: link

- `function link(scope, element, attrs) { ... }` donde:
 - **scope** es un objeto \$scope Angular
 - **element** es el elemento 'jqLite-incrustado'
 - **attrs** es un objeto hash con pares clave-valor de los nombres de los atributos y sus correspondientes valores

Directivas propias: link

- Las directivas que quieren **manipular el DOM** utilizan la función link

Directivas propias: link

```
<body ng-app="myApp">  
  <greeting></greeting>  
</body>
```

Directivas propias: link

```
var myApp = angular.module('myApp', []);
myApp.directive('greeting', function() {
    return {
        restrict: 'E',
        template: '<button type="submit" class="btn btn-default" +
        \'ng-click="greetMe()"> Greet Me </button> {{ greeting }}',
        link: function (scope) {
            scope.greetMe = function(){scope.greeting = 'Hola!!!'
            + 'Waaasssup?';}
        }
    });
});
```

Ejercicios

- Utiliza la función `link`, para poder asociar la lógica de generar números aleatorios a la directiva, y sin hacer más falta el controlador

Función link: attrs

- Desde la directiva podemos acceder a los **atributos del markup** que hemos definido

```
<body ng-app="myApp">  
  <greeting name="Dani"></greeting>  
</body>
```

Función link: attrs

```
var myApp = angular.module('myApp', []);
myApp.directive('greeting', function() {
  return {
    restrict: 'E',
    template: '<button type="submit" class="btn btn-default" ng-click="greetMe()"> Greet Me </button> {{ greeting }}',
    link: function (scope, element, attrs) {
      scope.greetMe = function(){scope.greeting = 'Hola!!!' + attrs.name +
        'Waaasssup?';}
    }
  });
});
```

Ejercicios

- Pásale a tu gna a través de un atributo, el intervalo de tiempo en el que quieres que se genere un nuevo número