

Directives

ng-controller
ng-app
ng-show
ng-hide
ng-repeat *ng-repeat="product in store.products"*
ng-src
ng-click
ng-init
ng-class
ng-model
ng-include

Modules

```
var app = angular.module('store', [ ]);
```

Expressions

Controllers

```
app.controller('StoreController', function(){  
});
```

Data

JSON

Array

Mostrar datos usando el controlador y los datos JSON

Filters

pipelines |

currency
date:'MM/dd/yyyy @ h:mm'
uppercase
lowercase
limitTo:8

```
ng-repeat="product in store.products | limitTo:3"
ng-repeat="product in store.products | orderBy:'-price'">
```

Crear un controlador con propiedad con valor inicial

Con una función que se llame a través de ng-click y que además asigne valor a una propiedad del controlador

Con otra función que se llame para mostrar información (p. ej ng-class o p.ej ng-show) en función del valor de una propiedad del controlador, que devuelva booleano.

Previsualización de lo escrito en un formulario

ng-model asocia elementos del formulario a propiedades dentro de expresiones.

Añadir datos a json, al array con push

Bootstrap

```
class="nav nav-pills"
```

```
class="panel"
```

```
active      ng-class="{ active:tab === 1 }"
```

Validations

1º quitamos validaciones por defecto de HTML en el form → **novalidate**

2º Marcar los campos requeridos con → **required**

Validacion de formulario (true/false)

```
{{formulario.$valid}}
```

ej.

```
<form name="reviewForm" ng-controller="ReviewController as reviewCtrl"
```

```
ng-submit="reviewForm.$valid && reviewCtrl.addReview(product)" novalidate>
```

??? Pendiente mirar cómo cambiar estilo en función de la validación correcta o errónea del campo del formulario

Creación de Directivas

```
<product-title></product-title>
```

```
app.directive('productTitle', function(){  
  return {  
    restrict: 'E',  
    templateUrl: 'product-title.html'  
  };  
});
```

mezclar controlador con directiva personalizada

controllerAs: