



@nicam

Module



```
<!DOCTYPE html>
<html ng-app="store"> ←----- Run this module
  <head> when the document
    <link rel="stylesheet" type="text/css" href="bootstrap.min.css" /> loads.
  </head>
  <body>
    <script type="text/javascript" src="angular.min.js"></script>
    <script type="text/javascript" src="app.js"></script>
  </body>
</html>
```

The code shows the structure of an Angular.js application. It starts with a standard HTML doctype and a root element. The root element has the attribute `ng-app="store"`. Inside the root element, there is a head section containing a link to a bootstrap CSS file. Below the head is a body section containing two script tags: one for the Angular.js library and one for the application-specific code in `app.js`. A handwritten note on the right side of the code block says "Run this module when the document loads." with an arrow pointing to the `ng-app` attribute. The entire code block is enclosed in a dark grey rounded rectangle with a dashed arrow pointing from the bottom right corner towards the bottom of the slide.

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

app.js



SHAPING UP
WITH
ANGULAR.JS



.controllers

Controller

```
<!DOCTYPE html>
<html>
  <body ng-app="myApp">

    <div ng-controller="myFirstController">
      {{myData}}
    </div>

    <script src="angular.js"></script>
    <script src="app.js"></script>
  </body>
</html>
```

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

app.controller('myFirstController', function($scope) {
  $scope.myData = "bla bla";
});
```

{{Data Binding}}



Input

```
<input type="text" ng-model="name"/>
```

Output

```
{ { name } }
```



{"expressions"}

Expressions

Numerical Operations

```
<p>  
  I am {{4 + 6}}  
</p>
```

evaluates to

```
<p>  
  I am 10  
</p>
```

String Operations

```
<p>  
  {{"hello" + " you"}}  
</p>
```

evaluates to

```
<p>  
  hello you  
</p>
```

+ More Operations:
<http://docs.angularjs.org/guide/expression>

SHAPING UP
WITH
ANGULAR.JS



\$scope

Dependency Injection

Controller

```
app.controller('myFirstController', function($scope) {  
  $scope.myData = "bla bla";  
});
```



View

```
<div ng-controller="myFirstController">  
  {{myData}}  
</div>
```

Output

```
<body ng-app="myApp" class="ng-scope">  
  <div ng-controller="FirstController" class="ng-scope ng-binding"> bla bla </div>  
  <script src="angular.js"></script>  
  <script src="app.js"></script>  
  <script src="blink.js"></script>  
</body>
```



<directives>

```
<div ng-controller="FirstController">
  <blink>{{myData}}</blink>
</div>
```

Directive

```
app.directive('blink', function($timeout) {
  return {
    restrict: 'E',
    transclude: true,
    scope: {},
    controller: function($scope, $element) {
      var colors = ['red', 'green', 'blue', 'orange', 'cyan'];
      var idx = 0;

      function showElement() {
        $element.css("color", colors[idx++ % colors.length]);
        $element.css("display", "inline");
        $timeout(hideElement, 150);
      }

      function hideElement() {
        $element.css("display", "none");
        $timeout(showElement, 150);
      }
      showElement();
    },
    template: '<span ng-transclude></span>',
    replace: true
  };
});
```



Directives We Know & Love

ng-app – attach the Application Module to the page

```
<html ng-app="store">
```

ng-controller – attach a Controller function to the page

```
<body ng-controller="StoreController as store">
```

ng-show / ng-hide – display a section based on an Expression

```
<h1 ng-show="name"> Hello, {{name}}! </h1>
```

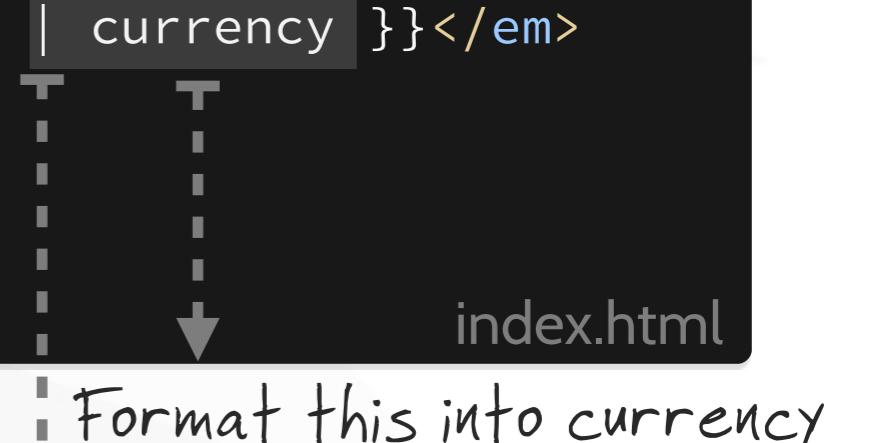
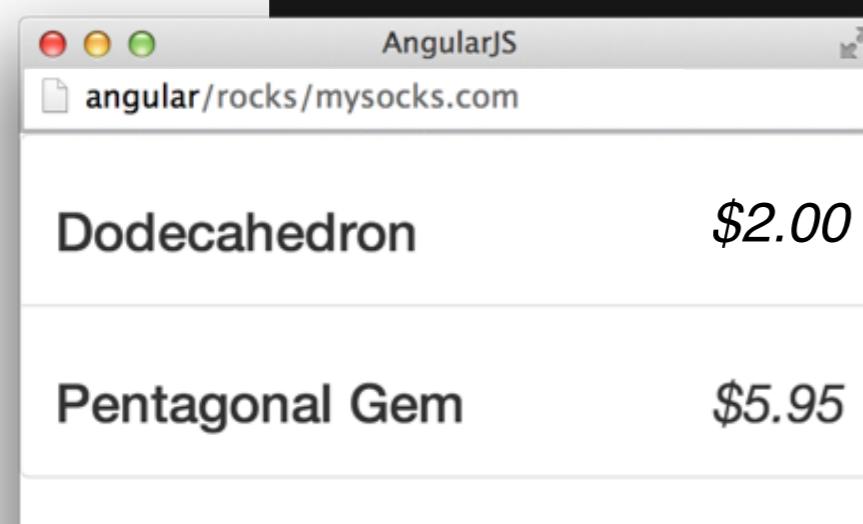
ng-repeat – repeat a section for each item in an Array

```
<li ng-repeat="product in store.products"> {{product.name}} </li>
```

filters | filters | filters | filters

Filters

```
<body ng-controller="StoreController as store">
  <ul class="list-group">
    <li class="list-group-item" ng-repeat="product in store.products">
      <h3>
        {{product.name}}
        <em class="pull-right">{{product.price | currency }}</em>
      </h3>
    </li>
  </ul>
</body>
```



Pipe - "send the output into"
Notice it gives the dollar sign (localized)
Specifies number of decimals

More filters

date

```
{ {'1388123412323' | date:'MM/dd/yyyy @ h:mma'}}
```

12/27/2013 @ 12:50AM

uppercase & lowercase

```
{ 'octagon gem' | uppercase}
```

OCTAGON GEM

limitTo

```
{ 'My Description' | limitTo:8}
```

My Descr

```
<li ng-repeat="product in store.products | limitTo:3">
```

orderBy

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

Will list products by descending price.

Without the - products would list in ascending order.

Custom Filter

Custom Filter

```
app.filter('checkmark', function() {
  return function(input) {
    return input ? '\u2713' : '\u2718';
  };
});
```

Using it in Controllers

```
app.controller('CheckoutController', function ($scope, checkmarkFilter) {
  var filteredValue = checkmarkFilter('blaa');
});
```

```
app.controller('CheckoutController', function ($scope, $filter) {
  var filteredValue = $filter('checkmarkFilter')('blaa');
});
```

Service



Service

Custom Service

```
app.service('someService', function() {
  var container = [];
  this.addItem = function(item) {
    cart.push(item);
  };
});
```

Using it in Controllers

```
app.controller('StoreController', function ($scope, $rootScope, someService) {
```