



SHAPING UP
WITH
ANGULAR.JS



Shaping Up with Angular.JS

Level 2: Filters, Directives, and Cleaner Code

SHAPING UP
WITH
ANGULAR.JS



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>
```



Our Current Code

```
<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}}</em>
      </h3>
    </li>
  </ul>
</body>
```

index.html

AngularJS

angular/rocks/mysocks.com

Dodecahedron	\$2
Pentagonal Gem	\$5.95

There's a better way
to print out prices.



Our First Filter

```
<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>
```

index.html

Dodecahedron	\$2.00
Pentagonal Gem	\$5.95

Format this into currency

Pipe - "send the output into"

Notice it gives the dollar sign (localized)

Specifies number of decimals



Formatting with Filters

Our Recipe `{{ data | filter:options* }}`

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

Will list products by descending price.

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

Without the - products would list in ascending order.



Adding an Image Array to our Product Array

```
var gems = [  
  { name: 'Dodecahedron Gem',  
    price: 2.95,  
    description: '...',  
    images: [ ← ----- Our New Array  
      { ← ----- Image Object  
        full: 'dodecahedron-01-full.jpg',  
        thumb: 'dodecahedron-01-thumb.jpg'  
      },  
      {  
        full: "dodecahedron-02-full.jpg",  
        ...  
      }  
    ]  
  }  
]
```

app.js

To display the first image in a product:

```
{{product.images[0].full}}
```



Using ng-src for Images

Using Angular Expressions inside a **src** attribute causes an error!



```

```

...the browser tries to load the image *before* the Expression evaluates.

```
<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>
```



NG-SOURCE
to the rescue!

index.html





Our Products with Images

AngularJS

angular/rocks/mysocks.com

Flatlander Crafted Gems

– an Angular store –

Pentagonal Gem	\$5.95
	
Dodecahedron	\$2.95
	



Challenges

SHAPING UP
WITH
ANGULAR.JS



More With Directives

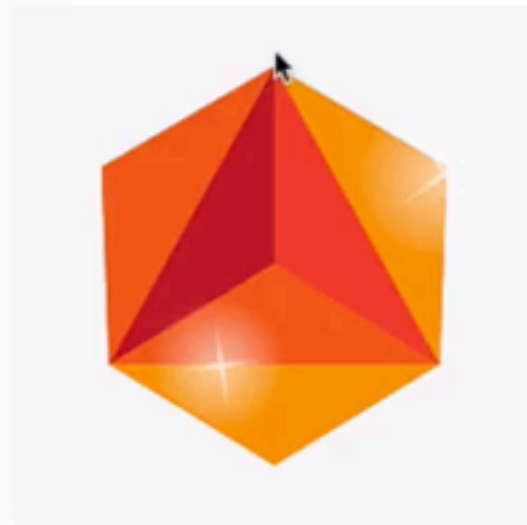


Flatlander Crafted Gems

– an Angular store –

Dodecahedron

\$2.9



Description

Specifications

Reviews

Description

Some gems have hidden qualities beyond their luster, beyond their shine... Dodeca is one of those gems.

How can I make my application more interactive?



A Simple Set of Tabs

```
<section>
  <ul class="nav nav-pills">
    <li> <a href>Description</a> </li>
    <li> <a href>Specifications</a> </li>
    <li> <a href>Reviews</a> </li>
  </ul>
</section>
```

index.html

Description

Specifications

Reviews

Description

Some gems have hidden qualities beyond their luster, beyond their shine... Dodeca is one of those gems.

SHAPING UP
WITH
ANGULAR.JS



Introducing a new Directive!



```
<section>
  <ul class="nav nav-pills">
    <li> <a href ng-click="tab = 1">Description</a> </li>
    <li> <a href ng-click="tab = 2">Specifications</a> </li>
    <li> <a href ng-click="tab = 3">Reviews</a> </li>
  </ul>
  {{tab}}
</section>
```

index.html

Assigning a value to **tab**.

For now just print this value to the screen.



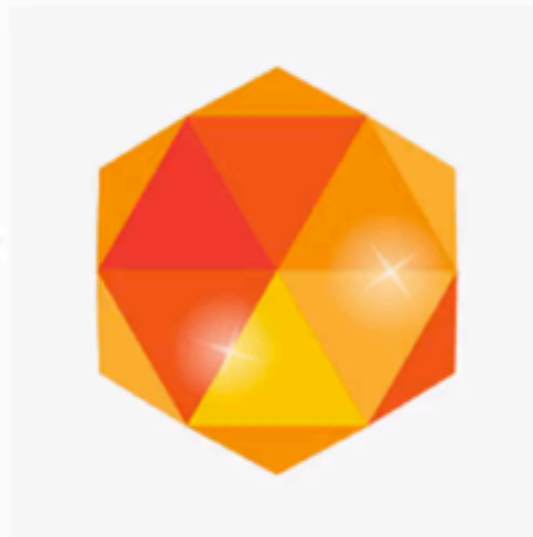
Introducing a new Directive!

Flatlander Crafted Gems

– an Angular store –

Pentagonal Gem

\$5.95



Description

Specifications

Reviews

1

Dodecahedron

\$2.95



Whoa, it's dynamic and stuff...

When `ng-click` changes the value of `tab` ...

... the `{{tab}}` expression automatically gets updated!

Expressions define a 2-way Data Binding ...

this means Expressions are re-evaluated when a property changes.



Let's add the tab content panels

```
↑
┆----- tabs are up here...
. . .
<div class="panel">
  <h4>Description</h4>
  <p>{{product.description}}</p>
</div>
<div class="panel">
  <h4>Specifications</h4>
  <blockquote>None yet</blockquote>
</div>
<div class="panel">
  <h4>Reviews</h4>
  <blockquote>None yet</blockquote>
</div>
```

How do we make the tabs trigger the panel to show?



Let's add the tab content panels

```
<div class="panel" ng-show="tab === 1">
  <h4>Description</h4>
  <p>{{product.description}}</p>
</div>
<div class="panel" ng-show="tab === 2">
  <h4>Specifications</h4>
  <blockquote>None yet</blockquote>
</div>
<div class="panel" ng-show="tab === 3">
  <h4>Reviews</h4>
  <blockquote>None yet</blockquote>
</div>
```


show the panel
if tab is the
right number

Now when a tab is selected it will show the appropriate panel!



Flatlander Crafted Gems

– an Angular store –

Pentagonal Gem	\$5.95
	
Description Specifications Reviews	
Dodecahedron	\$2.95

But how do we set an initial value for a tab?



Setting the Initial Value

`ng-init` allows us to evaluate an expression in the current scope.

```
<section ng-init="tab = 1">
  <ul class="nav nav-pills">
    <li> <a href ng-click="tab = 1">Description</a> </li>
    <li> <a href ng-click="tab = 2">Specifications</a> </li>
    <li> <a href ng-click="tab = 3">Reviews</a> </li>
  </ul>
  . . .
```

index.html



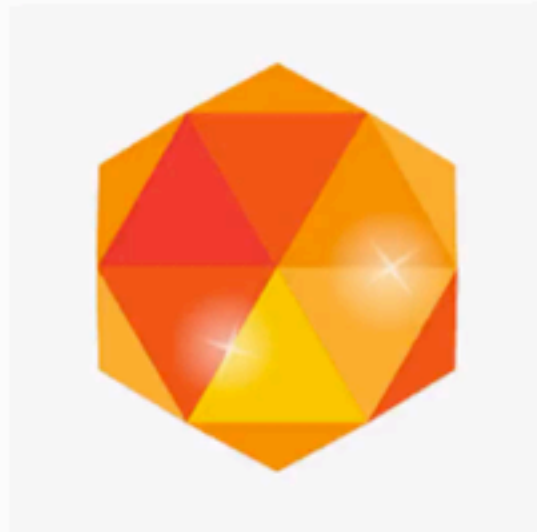
Now with the Initial Tab

Flatlander Crafted Gems

– an Angular store –

Pentagonal Gem

\$5.95



[Description](#)

[Specifications](#)

[Reviews](#)

Description

Origin of the Pentagonal Gem is unknown, hence its low value. It has a very high shine and 12 sides,



Setting the Active Class

We need to set the `class` to `active` if current `tab` is selected ...

```
<section ng-init="tab = 1">
  <ul class="nav nav-pills">
    <li> <a href ng-click="tab = 1">Description</a> </li>
    <li> <a href ng-click="tab = 2">Specifications</a> </li>
    <li> <a href ng-click="tab = 3">Reviews</a></li>
  </ul>
```

index.html



The ng-class directive

```
<section ng-init="tab = 1">
  <ul class="nav nav-pills">
    <li ng-class="{ active:tab === 1 }">
      <a href ng-click="tab = 1">Description</a>
    </li>
    <li ng-class="{ active:tab === 2 }">
      <a href ng-click="tab = 2">Specifications</a>
    </li>
    <li ng-class="{ active:tab === 3 }">
      <a href ng-click="tab = 3">Reviews</a>
    </li>
  </ul>
  . . .
```

Expression to evaluate
If true, set class to "active",
otherwise nothing.

Name of the class to set.

index.html

Flatlander Crafted Gems

– an Angular store –

Pentagonal Gem

\$5.95



Description

Specifications

Reviews

Description

Origin of the Pentagonal Gem is unknown, hence its low value. It has a very high shine and 12 sides,



Feels dirty, doesn't it?

All our application's logic is inside our HTML.

```
<section ng-init="tab = 1">
  <ul class="nav nav-pills">
    <li ng-class="{ active:tab === 1 }">
      <a href ng-click="tab = 1">Description</a>
    </li>
    <li ng-class="{ active:tab === 2 }">
      <a href ng-click="tab = 2">Specifications</a>
    </li>
    <li ng-class="{ active:tab === 3 }">
      <a href ng-click="tab = 3">Reviews</a>
    </li>
  </ul>
  <div class="panel" ng-show="tab === 1">
    <h4>Description </h4>
    <p>{{product.description}}</p>
  </div>
```

How might we pull this logic into a Controller?



Creating our Panel Controller

```
<section ng-init="tab = 1" ng-controller="PanelController as panel">
  <ul class="nav nav-pills">
    <li ng-class="{ active:tab === 1 }">
      <a href ng-click="tab = 1">Description</a>
    </li>
    <li ng-class="{ active:tab === 2 }">
      <a href ng-click="tab = 2">Specifications</a>
    </li>
    <li ng-class="{ active:tab === 3 }">
      <a href ng-click="tab = 3">Reviews</a>
    </li>
  </ul>
  <div class="panel" ng-show="tab === 1">
    <h4>Description </h4>
    <p>{{product.description}}</p>
  </div>
```

...

```
app.controller("PanelController", function(){
});
```



Moving our tab initializer

```
<section ng-controller="PanelController as panel">
  <ul class="nav nav-pills">
    <li ng-class="{ active:tab === 1 }">
      <a href ng-click="tab = 1">Description</a>
    </li>
    <li ng-class="{ active:tab === 2 }">
      <a href ng-click="tab = 2">Specifications</a>
    </li>
    <li ng-class="{ active:tab === 3 }">
      <a href ng-click="tab = 3">Reviews</a>
    </li>
  </ul>
  <div class="panel" ng-show="tab === 1">
    <h4>Description </h4>
    <p>{{product.description}}</p>
  </div>
```

...

```
app.controller("PanelController", function(){
  this.tab = 1;
});
```

app.js



Creating our selectTab function

```
<section ng-controller="PanelController as panel">
  <ul class="nav nav-pills">
    <li ng-class="{ active:tab === 1 }">
      <a href ng-click="panel.selectTab(1)">Description</a>
    </li>
    <li ng-class="{ active:tab === 2 }">
      <a href ng-click="panel.selectTab(2)">Specifications</a>
    </li>
    <li ng-class="{ active:tab === 3 }">
      <a href ng-click="panel.selectTab(3)">Reviews</a>
    </li>
  </ul>
  <div class="panel" ng-show="tab === 1">
    <h4>Description </h4>
    <p>{{product.description}}</p>
  </div>
```

```
app.controller("PanelController", function() {
  this.tab = 1;

  this.selectTab = function(setTab) {
    this.tab = setTab;
  };
});
```



Creating our isSelected function

```
<section ng-controller="PanelController as panel">
  <ul class="nav nav-pills">
    <li ng-class="{ active: panel.isSelected(1) }">
      <a href ng-click="panel.selectTab(1)" >Description</a>
    </li>
    <li ng-class="{ active: panel.isSelected(2) }">
      <a href ng-click="panel.selectTab(2)" >Specifications</a>
    </li>
    <li ng-class="{ active: panel.isSelected(3) }">
      <a href ng-click="panel.selectTab(3)" >Reviews</a>
    </li>
  </ul>
  <div class="panel" ng-show="panel.isSelected(1)">
    <h4>Description </h4>
    <p>{{product.description}}</p>
  </div>
```

...

```
app.controller("PanelController", function() {
  this.tab = 1;

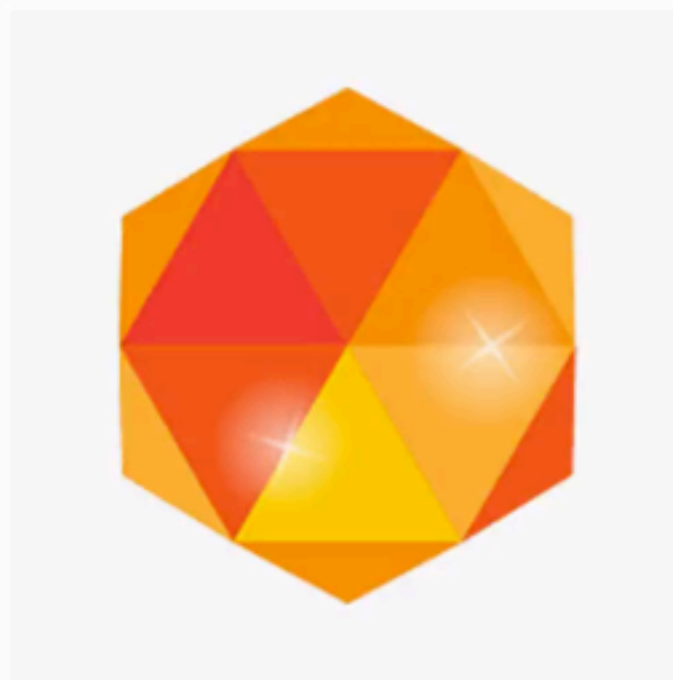
  this.selectTab = function(setTab) {
    this.tab = setTab;
  };
  this.isSelected = function(checkTab) {
    return this.tab === checkTab;
  };
});
```

Flatlander Crafted Gems

– an Angular store –

Pentagonal Gem

\$5.95



Description

Specifications

Reviews

Description

Origin of the Pentagonal Gem is unknown, hence its low value. It has a very high shine and 12 sides,



Challenges

SHAPING UP
WITH
ANGULAR.JS



SHAPING UP
WITH
ANGULAR.JS