AngularJS

JavaScript MVC Framework

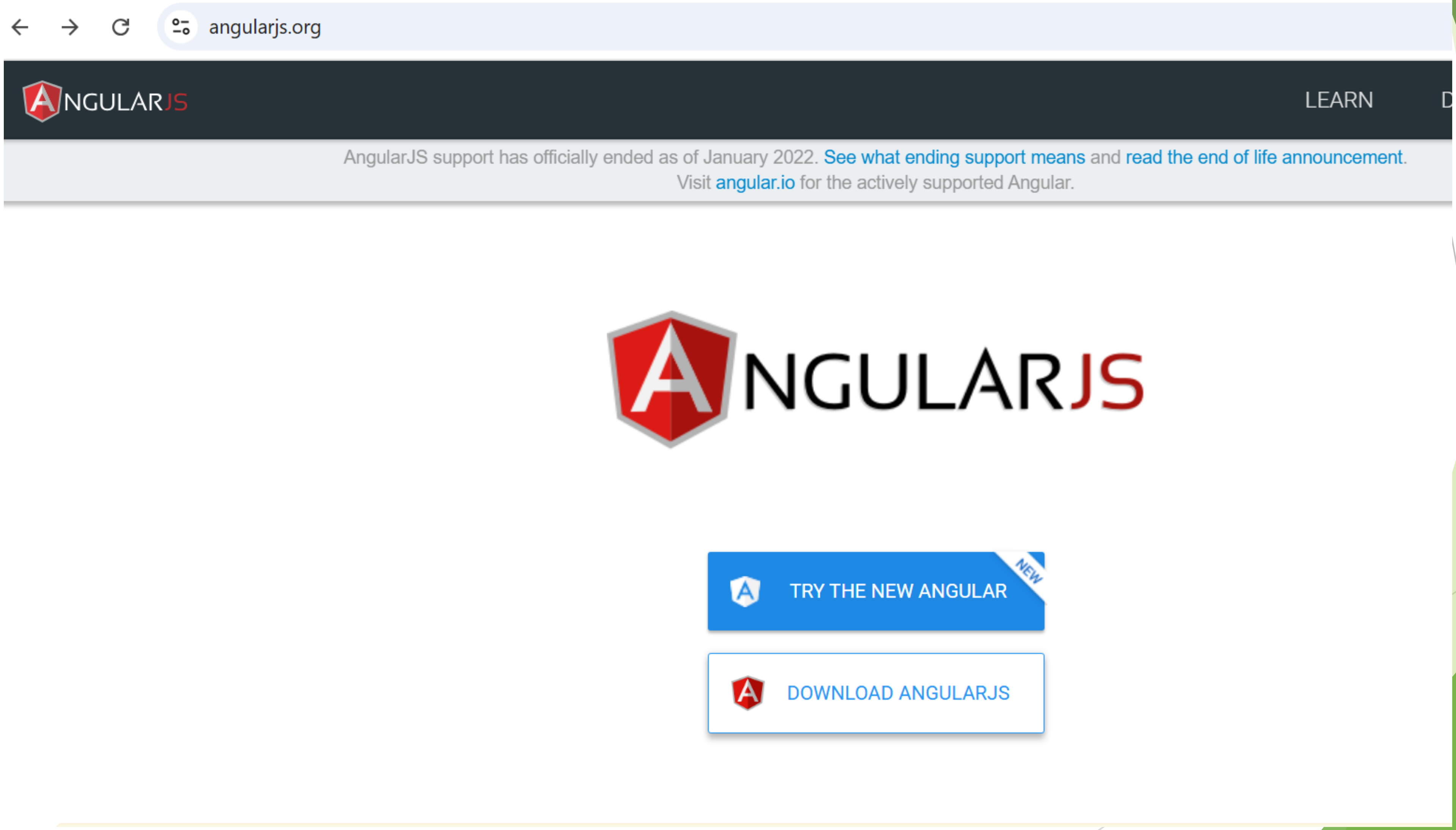
The general features of AngularJS are:

- AngularJS is an efficient framework that can create Rich Internet Applications (RIA).
- AngularJS provides developers an options to write client-side applications using JavaScript in a clean Model View Controller (MVC) way.
- Applications written in AngularJS are cross-browser compliant. AngularJS automatically handles JavaScript code suitable for each browser.
- AngularJS is open source, completely free, and used by thousands of developers around the world. It is licensed under the Apache license version 2.0.

The key features of AngularJS

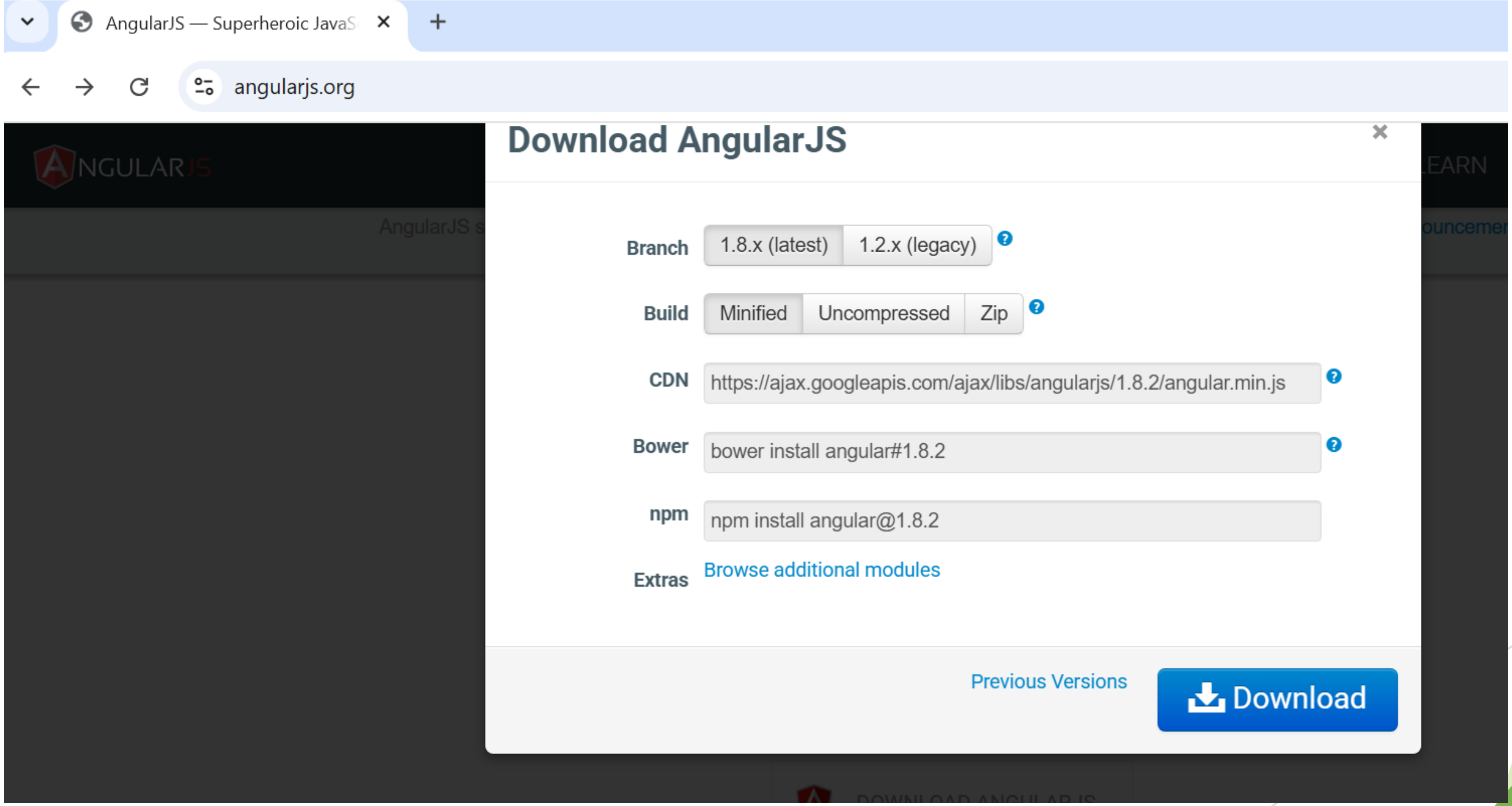
- **Data-binding:** It is the automatic synchronization of data between model and view components.
- MVC Architecture (Model-View-Controller):
- **Dependency Injection:** AngularJS has a built-in dependency injection subsystem that helps the developer to create, understand, and test the applications easily.

Download Angular JS



AngularJS support has officially ended as of January 2022. See what ending support means and read the end of life announcement.
Visit angular.io for the actively supported Angular.

When click “Download angular JS”



Download AngularJS

Branch: 1.8.x (latest) | 1.2.x (legacy)

Build: Minified | Uncompressed | Zip

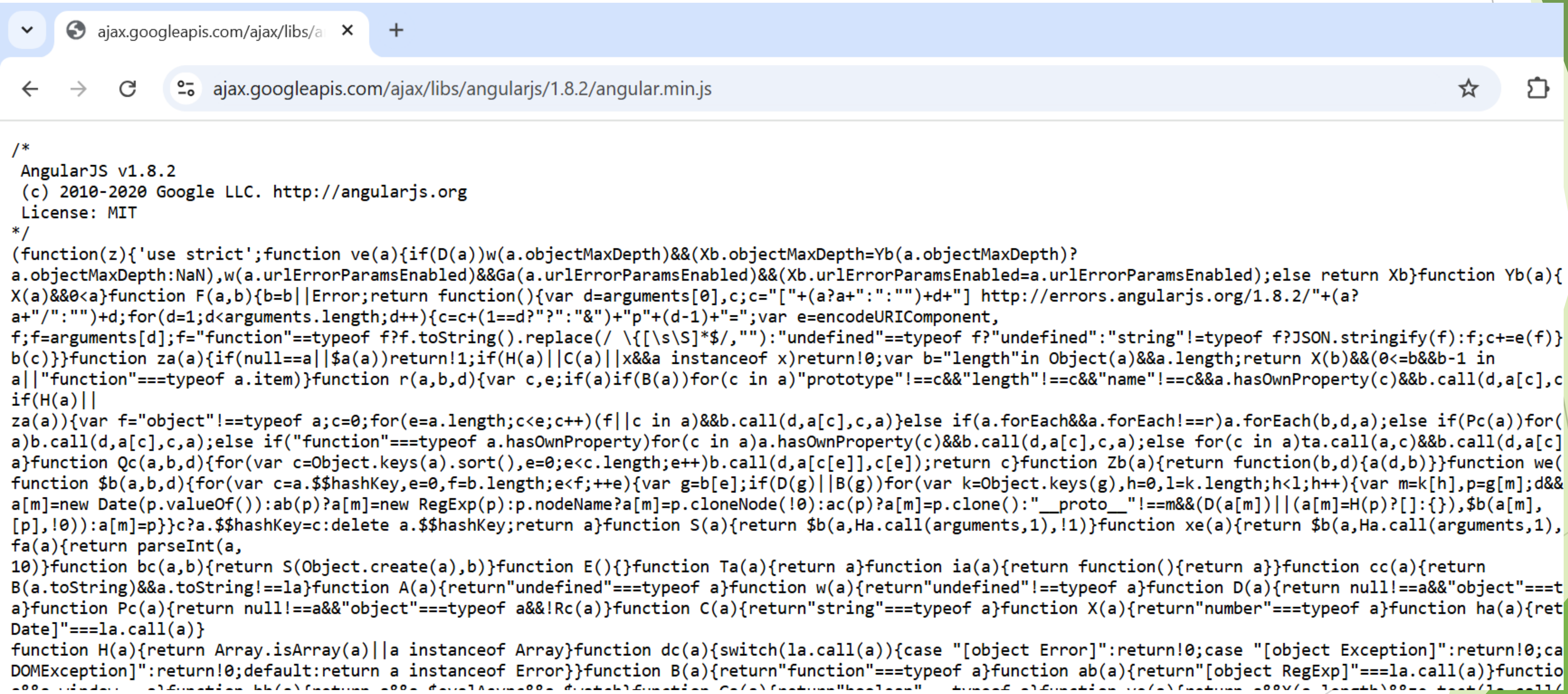
CDN: https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js

Bower: bower install angular#1.8.2

npm: npm install angular@1.8.2

Extras: Browse additional modules

When click “download”


ajax.googleapis.com/ajax/libs/a
ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js

```
/*
 AngularJS v1.8.2
 (c) 2010-2020 Google LLC. http://angularjs.org
 License: MIT
*/
(function(z){'use strict';function ve(a){if(D(a))w(a.objectMaxDepth)&&(Xb.objectMaxDepth=Yb(a.objectMaxDepth)?
a.objectMaxDepth:NaN),w(a.urlErrorParamsEnabled)&&Ga(a.urlErrorParamsEnabled)&&(Xb.urlErrorParamsEnabled=a.urlErrorParamsEnabled);else return Xb}function Yb(a){
X(a)&&0<a}function F(a,b){b=b||Error;return function(){var d=arguments[0],c;c="["+(a?a+":":"")+d+"] http://errors.angularjs.org/1.8.2/"+(a?
a+"/":"")+d;for(d=1;d<arguments.length;d++){c=c+(1==d?"?":"&")+"p"+(d-1)+"=";var e=encodeURIComponent,
f;f=arguments[d];f="function"==typeof f?f.toString().replace(/ \{[\s\S]*$/,""):"undefined"==typeof f?"undefined":"string"!=typeof f?JSON.stringify(f):f;c+=e(f)}
b(c)}}function za(a){if(null==a||$a(a))return!1;if(H(a)||C(a)||x&&a instanceof x)return!0;var b="length"in Object(a)&&a.length;return X(b)&&(0<=b&&b-1 in
a||"function"===typeof a.item)}function r(a,b,d){var c,e;if(a)if(B(a))for(c in a)"prototype"!==c&&"length"!==c&&"name"!==c&&a.hasOwnProperty(c)&&b.call(d,a[c],c
if(H(a)||
za(a)){var f="object"!==typeof a;c=0;for(e=a.length;c<e;c++)(f||c in a)&&b.call(d,a[c],c,a)}else if(a.forEach&&a.forEach!==r)a.forEach(b,d,a);else if(Pc(a))for(
a)b.call(d,a[c],c,a);else if("function"===typeof a.hasOwnProperty)for(c in a)a.hasOwnProperty(c)&&b.call(d,a[c],c,a);else for(c in a)ta.call(a,c)&&b.call(d,a[c]
a}function Qc(a,b,d){for(var c=Object.keys(a).sort(),e=0;e<c.length;e++)b.call(d,a[c[e]],c[e]);return c}function Zb(a){return function(b,d){a(d,b)}}function we(
function $b(a,b,d){for(var c=a.$$hashKey,e=0,f=b.length;e<f;++e){var g=b[e];if(D(g)||B(g))for(var k=Object.keys(g),h=0,l=k.length;h<l;h++){var m=k[h],p=g[m];d&&
a[m]=new Date(p.valueOf()):ab(p)?a[m]=new RegExp(p):p.nodeName?a[m]=p.cloneNode(!0):ac(p)?a[m]=p.clone():"__proto__"!==m&&(D(a[m])||(a[m]=H(p)?[]:{}),$b(a[m],
[p],!0)):a[m]=p}}c?a.$$hashKey=c:delete a.$$hashKey;return a}function S(a){return $b(a,Ha.call(arguments,1),!1)}function xe(a){return $b(a,Ha.call(arguments,1),
fa(a){return parseInt(a,
10)}function bc(a,b){return S(Object.create(a),b)}function E(){}function Ta(a){return a}function ia(a){return function(){return a}}function cc(a){return
B(a.toString)&&a.toString!==la}function A(a){return"undefined"===typeof a}function w(a){return"undefined"!==typeof a}function D(a){return null!==a&&"object"===t
a}function Pc(a){return null!==a&&"object"===typeof a&&!Rc(a)}function C(a){return"string"===typeof a}function X(a){return"number"===typeof a}function ha(a){ret
Date]"===la.call(a)}
function H(a){return Array.isArray(a)||a instanceof Array}function dc(a){switch(la.call(a)){case "[object Error]":return!0;case "[object Exception]":return!0;ca
DOMException]":return!0;default:return a instanceof Error}}function B(a){return"function"===typeof a}function ab(a){return"[object RegExp]"===la.call(a)}functio
```

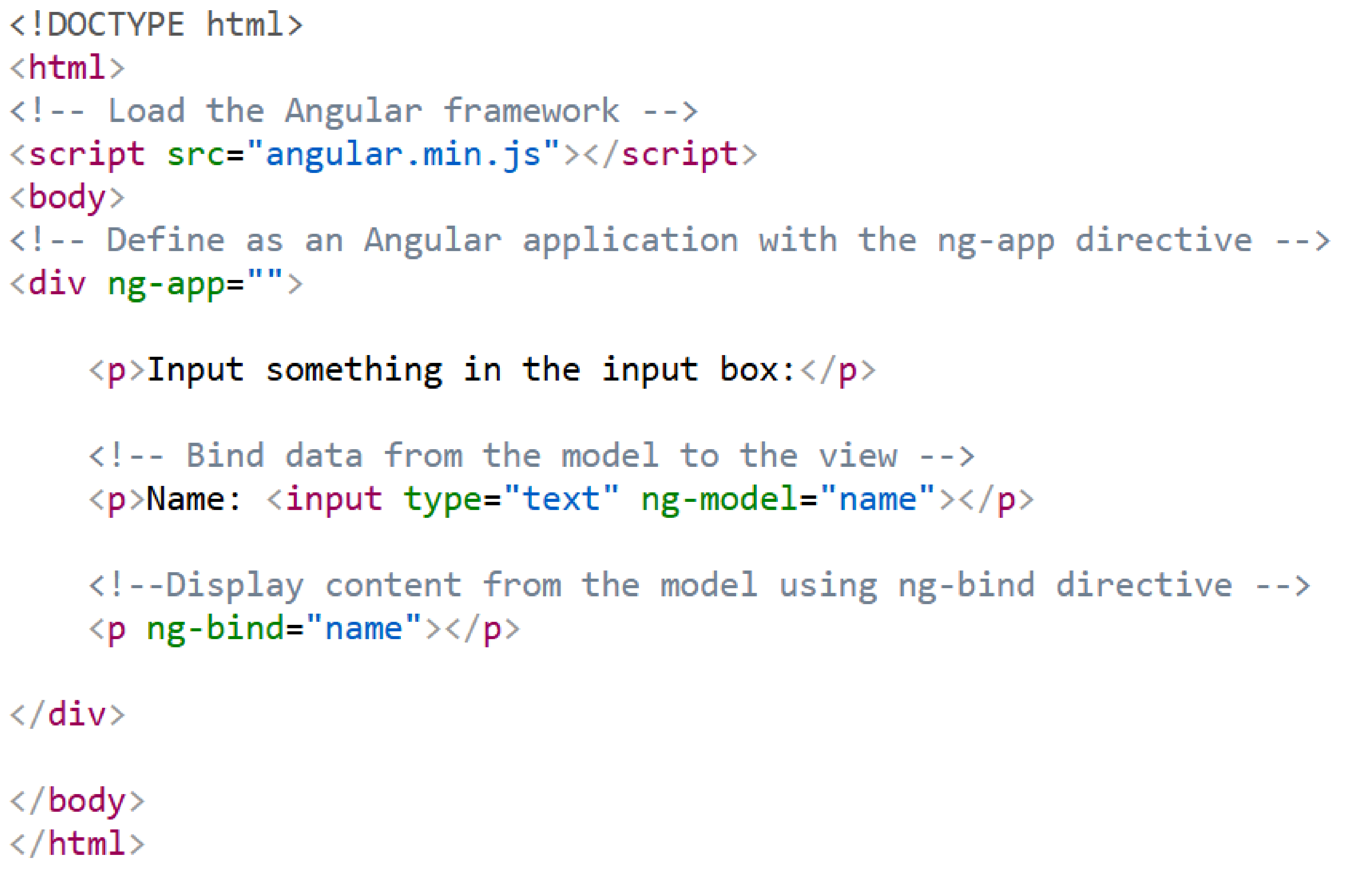
Introduction to AngularJS

- AngularJS is a **JavaScript framework**.

- It can be added to an HTML page with a <script> tag.

- AngularJS extends HTML with **ng-directives**.

- The **ng-app** directive defines an AngularJS application.
- The **ng-model** directive binds the value of HTML controls (input, select, textarea) to application data.
- The **ng-bind** directive binds application data to the HTML view.
- AngularJS starts automatically when the web page has loaded.
- The **ng-init** directive initializes AngularJS application variables.

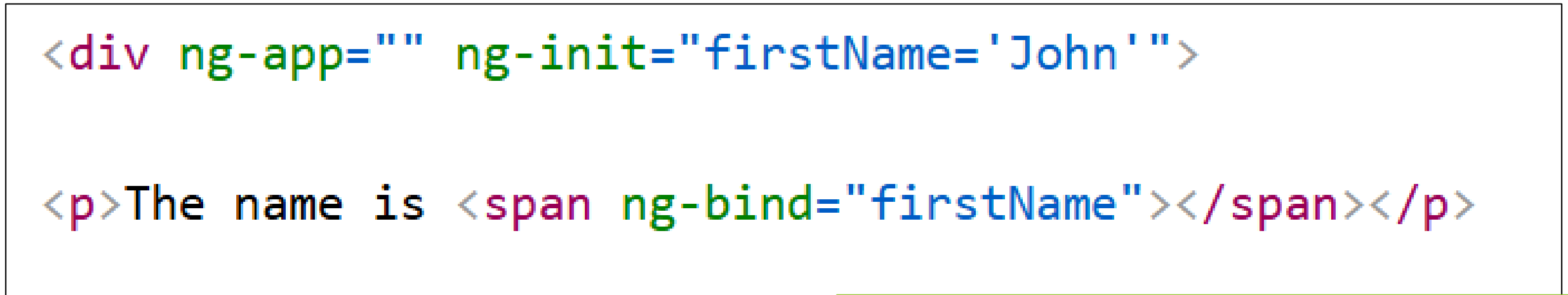


```
<!DOCTYPE html>
<html>
<!-- Load the Angular framework -->
<script src="angular.min.js"></script>
<body>
<!-- Define as an Angular application with the ng-app directive -->
<div ng-app="">

    <p>Input something in the input box:</p>

    <!-- Bind data from the model to the view -->
    <p>Name: <input type="text" ng-model="name"></p>

    <!--Display content from the model using ng-bind directive -->
    <p ng-bind="name"></p>

</div>

</body>
</html>
```

AngularJS Directives

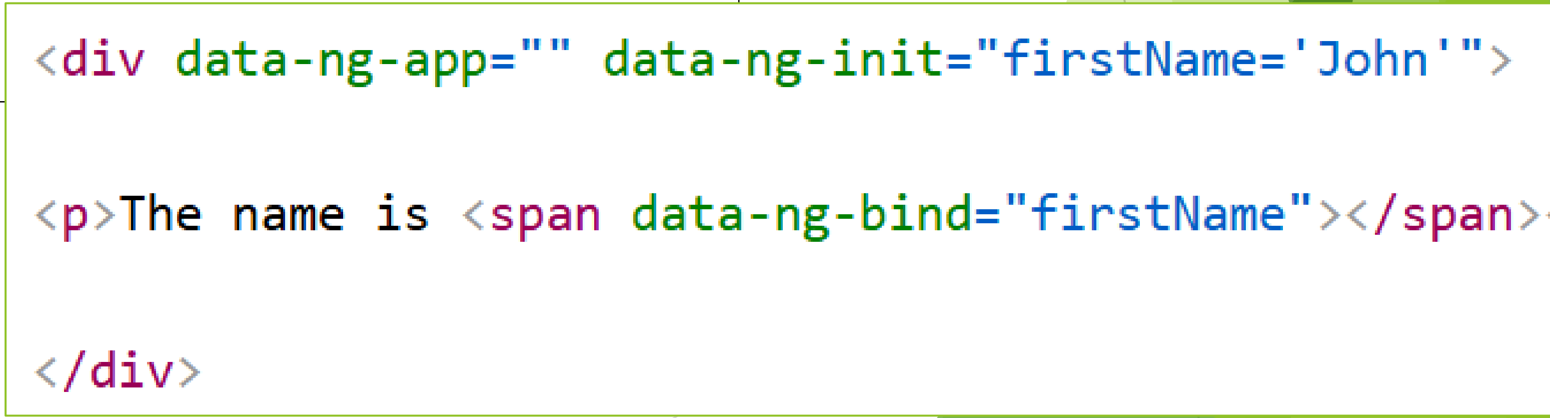
As you have already seen, AngularJS directives are HTML attributes with an **ng** prefix.

- The **ng-init** directive initializes AngularJS application variables.



```
<div ng-app="" ng-init="firstName='John'">

<p>The name is <span ng-bind="firstName"></span></p>

</div>
```

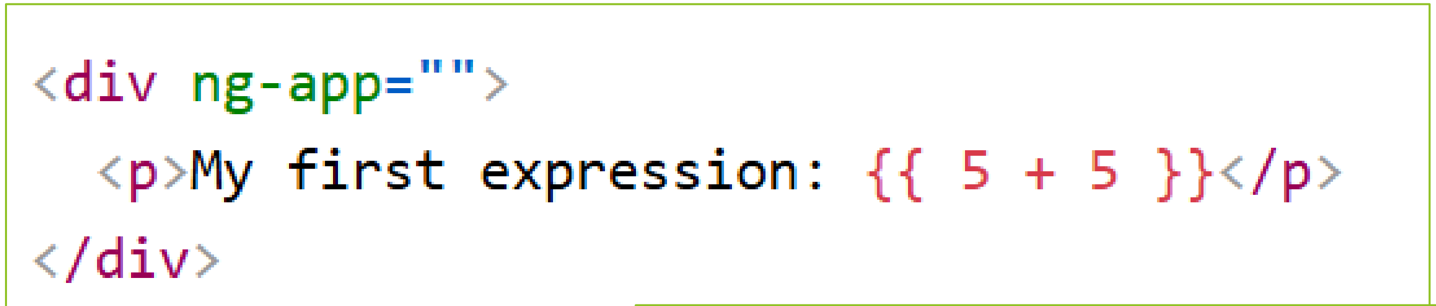
Alternatively with valid HTML:



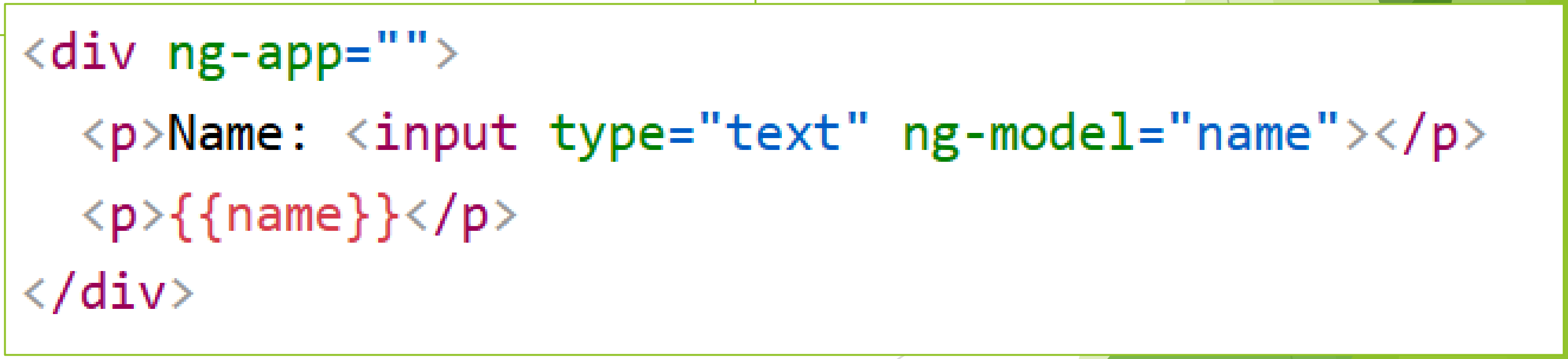
```
<div data-ng-app="" data-ng-init="firstName='John'">

<p>The name is <span data-ng-bind="firstName"></span>

</div>
```

AngularJS Expressions

- AngularJS expressions are written inside double braces: **{{ expression }}**.



```html
<div ng-app="">
  <p>My first expression: {{ 5 + 5 }}</p>
</div>
```



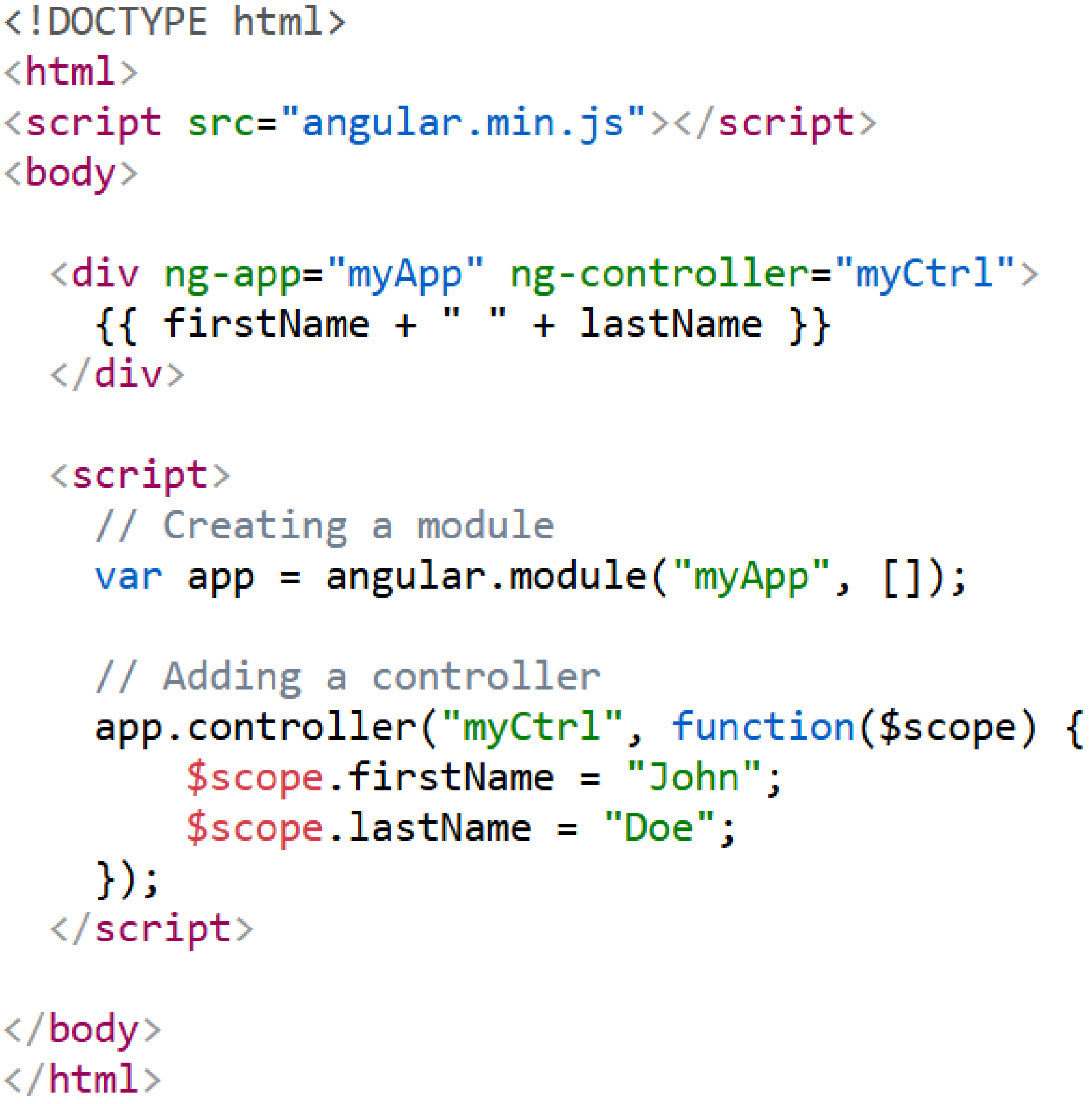
```html
<div ng-app="">
  <p>Name: <input type="text" ng-model="name"></p>
  <p>{{name}}</p>
</div>
```

AngularJS Applications

- AngularJS **modules** define AngularJS applications.
- AngularJS **controllers** control AngularJS applications.
- The **ng-app** directive defines the application, the **ng-controller** directive defines the controller.

AngularJS Modules

- The module is a container for the different parts of an application.
- The module is a container for the application controllers.
- A module is created by using the AngularJS function angular.module



```html
<!DOCTYPE html>
<html>
<script src="angular.min.js"></script>
<body>

  <div ng-app="myApp" ng-controller="myCtrl">
    {{ firstName + " " + lastName }}
  </div>

  <script>
    // Creating a module
    var app = angular.module("myApp", []);

    // Adding a controller
    app.controller("myCtrl", function($scope) {
        $scope.firstName = "John";
        $scope.lastName = "Doe";
    });
  </script>

</body>
</html>
```

AngularJS Controller

- AngularJS controllers are regular **JavaScript Objects**.
- AngularJS applications are controlled by controllers.
- The **ng-controller** directive defines the application controller.
- AngularJS will invoke the controller with a **$scope** object.

Modules and Controllers in Files

- It is common in AngularJS applications to put the module and the controllers in JavaScript files.

```html
<!DOCTYPE html>
<html>
<script src="angular.min.js"></script>
<body>

  <div ng-app="myApp" ng-controller="myCtrl">
    {{ firstName + " " + lastName }}
  </div>

  <script src="myApp.js"></script>
  <script src="myCtrl.js"></script>

</body>
</html>
```

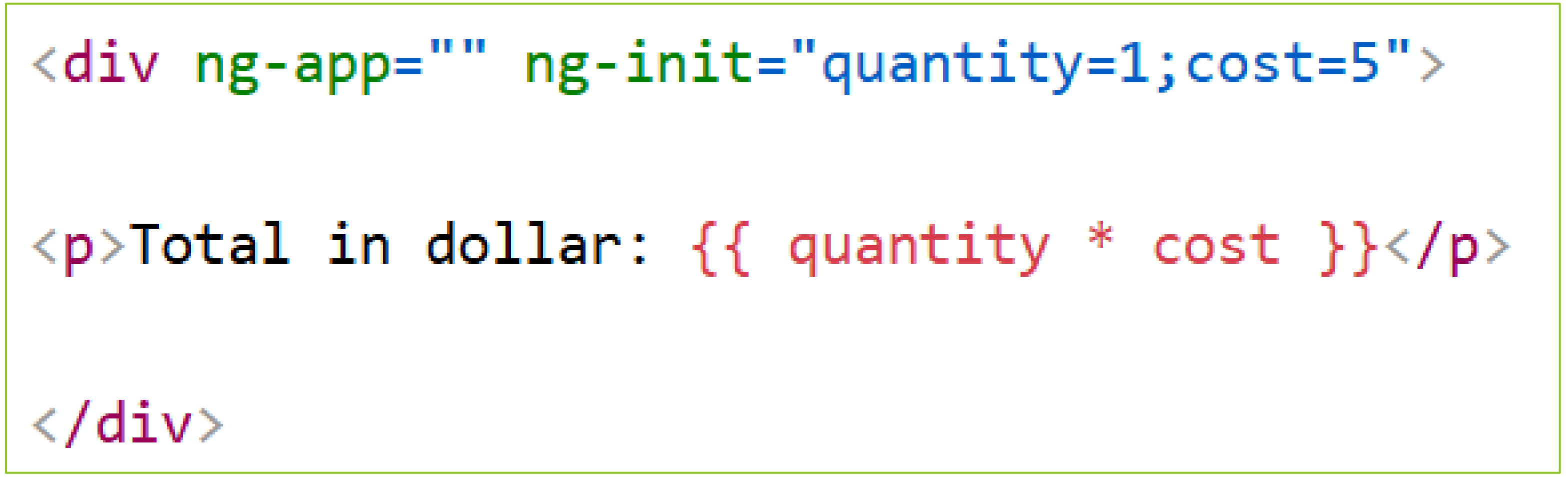
myCtrl.js

```javascript
app.controller("myCtrl", function($scope) {
  $scope.firstName = "John";
  $scope.lastName= "Doe";
});
```

myApp.js

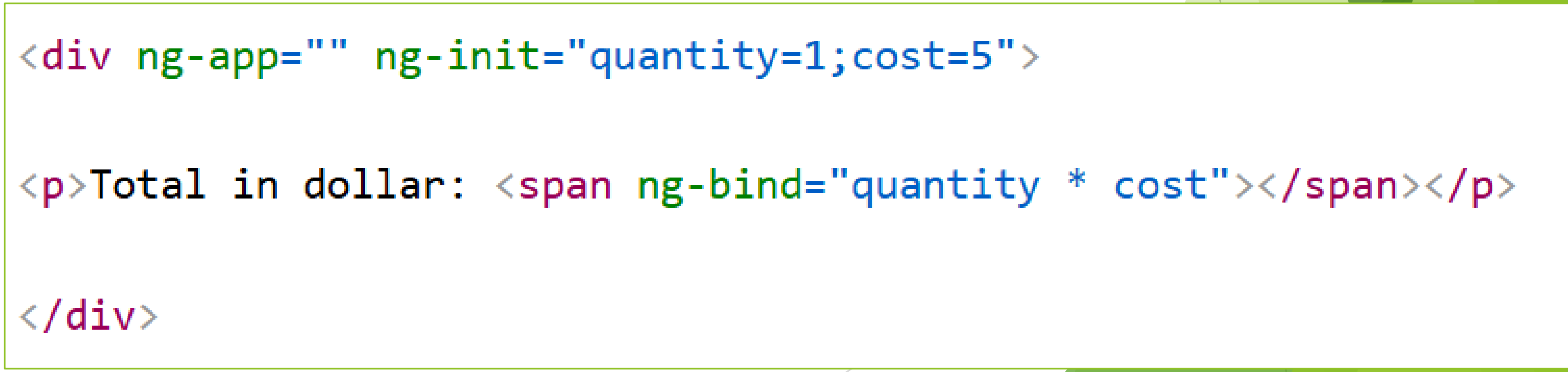
```javascript
var app = angular.module("myApp", []);
```

AngularJS Numbers



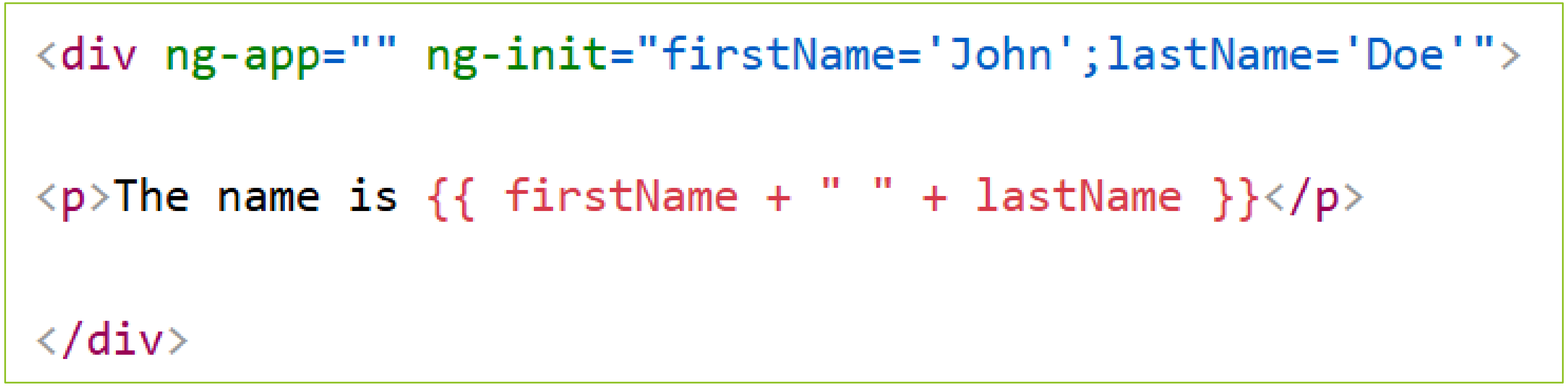
```
<div ng-app="" ng-init="quantity=1;cost=5">

<p>Total in dollar: {{ quantity * cost }}</p>

</div>
```

Or



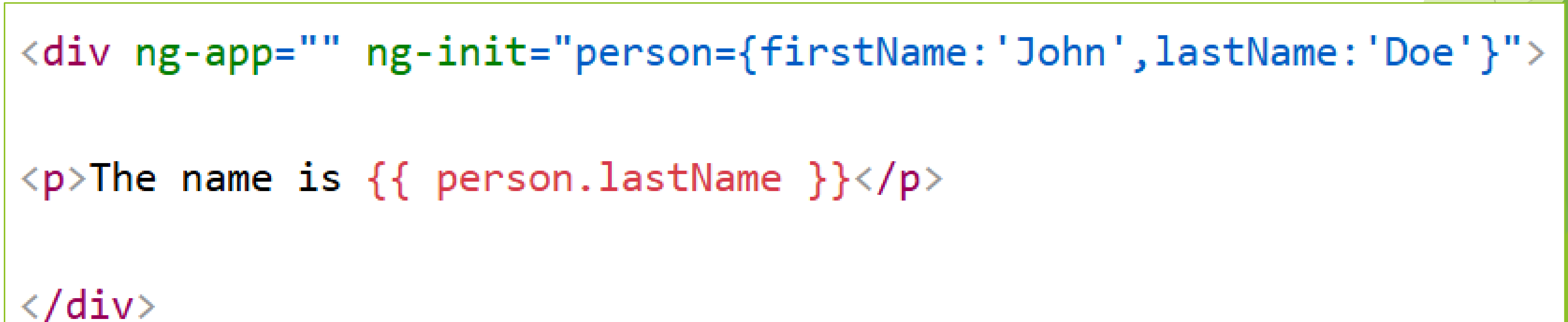
```
<div ng-app="" ng-init="quantity=1;cost=5">

<p>Total in dollar: <span ng-bind="quantity * cost"></span></p>

</div>
```

AngularJS Strings



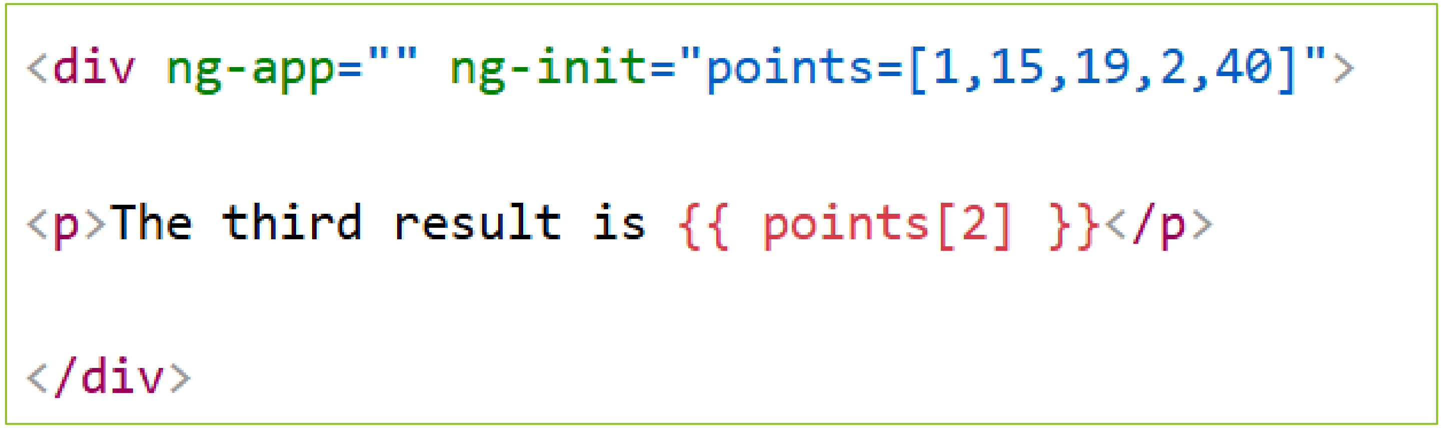
```
<div ng-app="" ng-init="firstName='John';lastName='Doe'">

<p>The name is {{ firstName + " " + lastName }}</p>

</div>
```

AngularJS Objects



```
<div ng-app="" ng-init="person={firstName:'John',lastName:'Doe'}">

<p>The name is {{ person.lastName }}</p>

</div>
```

AngularJS Array



```
<div ng-app="" ng-init="points=[1,15,19,2,40]">

<p>The third result is {{ points[2] }}</p>

</div>
```