
Sistemas Web III

Introdução ao AngularJS

Tiago Cruz de França

tcruz.franca@gmail.com

Introdução

@ Um framework, uma biblioteca escrita em JavaScript

- @ Soma **diretivas** aos atributos HTML e liga dados por meio de **expressões**

@ A primeira versão é de 2012

- @ Projeto iniciado em 2009 por Miško Hevery
 - ▶ Um funcionário da Google
- @ A ideia vingou...
 - ▶ O Google agora dá suporte ao AngularJS

Introdução

@ Adicione a biblioteca a sua página

⌚ É uma biblioteca JavaScript


▶ Basta adicioná-la na sua página com a tag `<script>`

```
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js"></script>
```

▶ Como a inclusão de qualquer JavaScript em páginas HTML

– Observe que no exemplo a biblioteca está nos servidores do google

Pré-Requisitos

 **Antes de continuar você deve conhecer e saber usar:**

 HTML

 CSS

 JavaScript

 **Se você chegou aqui na disciplina de Web-III, você já cumpriu os pré-requisitos**

AngularJS e HTML

```
<!DOCTYPE html>
<html>
<script src=
"http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js"></script>
<body>

<div ng-app="">

<p>Input something in the input box:</p>
<p>Name: <input type="text" ng-model="name"></p>
<p ng-bind="name"></p>

</div>

</body>
</html>
```

AngularJS e HTML

@ AngularJS estende HTML

🌀 Para isso, usa *ng-directives*

- ▶ Diretiva **ng-app** para definir uma aplicação AngularJS
- ▶ Diretiva **ng-model** para ligar o valor do elemento HTML (input, select, textarea) ao dado de aplicação
- ▶ A diretiva **ng-bind** liga o dado de aplicação ao elemento de visualização HTML

Diretivas

@ Começam com ng, como já vimos

@ Para iniciar variáveis, use a diretiva ng-init

► Opcionalmente, é possível usar data-ng para validar o HTML da página

```
<div ng-app="" ng-init="nome='Franca'">
<!-- alternativa -->
<!--<div data-ng-app="" data-ng-init="nome='Tiago'">-->
<p>The name is <span ng-bind="nome"></span></p>
<!--<p>The name is <span data-ng-bind="nome"></span></p>-->
</div>
```

Expressões AngularJS

@ **As expressões são escritas entre duas chaves**

@ Ex: {{expressao}}

▶ O dado será apresentado onde está a expressão

```
<!DOCTYPE html>
<html>
<script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
<body>
<div ng-app="">
    <p>Minha primeira expressão: {{ 5 + 5 }}</p>
</div>
</body>
</html>
```


Bind com Expressões AngularJS

@ Funciona da mesma forma que o ng-bind

```
<!DOCTYPE html>
<html>
<script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
<body>

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

</body>
</html>
```

Aplicações

@ Módulos AngularJS definem aplicações

- @ As aplicações são controladas pelos controllers AngularJS
- @ A aplicação é definida pela diretiva ng-app
- @ A diretiva ng-controller define o controller

Veja o exemplo no próximo slide

Exemplo de Aplicação

```
<div ng-app="myApp" ng-controller="myCtrl">
First Name: <input type="text" ng-model="firstName"><br>
Last Name: <input type="text" ng-model="lastName"><br>
<br>
Full Name: {{firstName + " " + lastName}}
</div>

<script>
var app = angular.module('myApp', []); //definição da aplicação (módulo)
app.controller('myCtrl', function($scope) { //controle da aplicação
    $scope.firstName= "John"; //variáveis começam com estes valores
    $scope.lastName= "Doe";
});
</script>
```

Exemplo de uso de expressão

🔗 Use expressão em qualquer lugar da página

```
<!-- mudando uma propriedade CSS -->  
<div ng-app="" ng-init="myCol='lightblue'">  
  
<input style="background-color:{{myCol}}" ng-model="myCol" value="{{myCol}}">  
  
</div>
```

Exemplo usando números

@ Usando expressões

```
<div ng-app="" ng-init="quantity=1;cost=5">  
  
<p>Total in dollar: {{ quantity * cost }}</p>  
  
</div>
```

@ Mesmo exemplo usando ng-bind

```
<div ng-app="" ng-init="quantity=1;cost=5">  
  
<p>Total in dollar: <span ng-bind="quantity * cost"></span></p>  
  
</div>
```

Exemplo usando Strings

@ Usando expressões

```
<div ng-app="" ng-init="firstName='John';lastName='Doe'">  
  
<p>The name is {{ firstName + " " + lastName }}</p>  
  
</div>
```

@ Mesmo exemplo usando ng-bind

```
<div ng-app="" ng-init="firstName='John';lastName='Doe'">  
  
<p>The name is <span ng-bind="firstName + ' ' + lastName"></span></p>  
  
</div>
```

Objetos AngularJS

@ Os objetos são como os de javascript

```
<div ng-app="" ng-init="person={firstName:'John',lastName:'Doe'}">  
  
<p>The name is {{ person.lastName }}</p>  
  
</div>
```

@ Mesmo exemplo com ng-bind

```
<div ng-app="" ng-init="person={firstName:'John',lastName:'Doe'}">  
  
<p>The name is <span ng-bind="person.lastName"></span></p>  
  
</div>
```

Array

@ São como array javascript

```
<div ng-app="" ng-init="points=[1,15,19,2,40]">  
  
<p>The third result is {{ points[2] }}</p>  
  
</div>
```

@ Mesmo exemplo com ng-bind

```
<div ng-app="" ng-init="points=[1,15,19,2,40]">  
  
<p>The third result is <span ng-bind="points[2]"></span></p>  
  
</div>
```