



## 第12讲 AngularJS动画

# 教学目标

---

1. 元素指令基本动画
2. 自定义动画
3. javascript动画

# 1

## 元素基本指令动画

AngularJS 提供了动画效果，可以配合 CSS 使用。

## 什么是AngularJS动画？

AngularJS动画是通过改变 HTML 元素产生的动态变化效果。

AngularJS提供了\$animate动画服务。可以使用动画的指令（如：ng-hide）在整个指令的生命周期中随着各种指示，触发等进行动作，显示动画效果。

\$animate动画服务封装在**ngAnimate**中。

# 1

## 使用动画服务

AngularJS中**ngAnimate**模块支持动画效果，但是**ngAnimate**模块并未包含在AngularJS核心库中，因此使用ngAnimate需要在定义Module时声明对其的引用。

AngularJS 使用动画需要引入 angular-animate.js 库。

```
<script src="angular-animate.js"></script>
```

还需在应用中使用模型 ngAnimate：

```
angular.module('myApp', ['ngAnimate']);
```

# 1

## 使用动画服务

隐藏 DIV: `<input type="checkbox" ng-model="myCheck">`

`<div ng-hide="myCheck"></div>`

```
div {  
    transition: all linear 0.5s;  
    background-color: lightblue;  
    height: 200px; /*width:300px;*/  
}  
  
.ng-hide {  
    height: 0; /*width:0*/  
}
```

Demo:ex01.html

## ②

# 分析——隐藏元素

隐藏元素就是为元素添加 “ng-hide” 样式

```
<div ng-hide="myCheck" class></div>
```

默认状态：  
myCheck=false

```
<div ng-hide="myCheck" class="ng-animate ng-hide-animate ng-hide-add ng-hide-add-active" data-ng-animate="2"></div>
```

```
<div ng-hide="myCheck" class="ng-hide"></div>
```

ng-animate : 动画 ng-hide-animate : 隐藏动画

ng-hide-add (添加class样式)

ng-hide-add-active (添加class样式)



## ②

# 分析——显示元素

显示元素就是为元素移除 “ng-hide” 样式

```
<div ng-hide="myCheck" class="ng-hide"></div>
```

改变状态：  
myCheck=true

```
<div ng-hide="myCheck" class="ng-animate ng-hide-animate ng-hide-remove ng-hide-remove-active" data-ng-animate="2"></div>
```

```
<div ng-hide="myCheck" class></div>
```

ng-animate : 动画 ng-hide-animate : 隐藏动画

ng-hide-remove (移除class样式)

ng-hide-remove-active (移除class样式)

## ngAnimate 做了什么？

- ◆ ngAnimate模型可以添加或移除 class 。
- ◆ ngAnimate模型并不能使 HTML 元素产生动画，但是 ngAnimate会监测事件，类似隐藏显示 HTML 元素，如果事件发生 ngAnimate 就会使用预定义的 class 来设置 HTML 元素的动画。

## 4

# 使用CSS动画

```
div {  
  transition: all linear 0.5s;  
  background-color: lightblue;  
  height: 100px;  
  width: 100%;  
  position: relative;  
  top: 0;  
  left: 0;  
}  
.ng-hide {  
  height: 0;  
  width: 0;  
  background-color: transparent;  
  top: -200px;  
  left: 200px;  
}
```

Demo:ex02.html

2

自定义动画

## 使用 \$animate 动画服务

指令事件	初始类	结束类	触发该CSS类变化的指令
hide	.ng-hide-add	.ng-hide-add-active	ng-hide,ng-show
show	.ng-hide-add-active	.ng-hide-add	ng-hide,ng-show
Add class	.class-add	.class-add-active	ng-class, class='{{{}}}'
Remove class	.class-remove	.class-remove-active	ng-class, class='{{{}}}'

使用 \$animate 动画服务

指令事件	初始类	结束类	触发该CSS类变化的指令
enter	.ng-enter	.ng-enter-active	ng-repeat,ng-include,ng-if,nf-view
leave	.ng-leave	.ng-leave-active	ng-repeat,ng-include,ng-if,nf-view
move	.ng-move	.ng-move-active	ng-repeat

## 2

# 自定义动画样式

```
div {transition: all linear 0.5s;background:
lightblue; height: 100px;width: 100%;}
.ng-hide {height: 0;width: 0;background:
transparent;}
.ng-hide-add{background:#f00}
.ng-hide-add-active{background:#f0f}
.ng-hide-remove{background:#ff0}
.ng-hide-remove-active{background:#0ff}
```

Demo:ex03.html

## ②

# 自定义动画样式

```
.tomato {  
  transition: all linear 0.5s;  
  background-color: coral;  
  padding: 40px;  
  height: 200px;  
}  
.tomato-add{height:100px}  
.tomato-add-active{height:50px}  
.tomato-remove{height:50px}  
.tomato-remove-active{height:100px}
```

Demo:ex04.html



## 2

# 自定义动画样式

```
<div ng-if="myVar" class="fade">
```

```
<style>
```

```
  .fade {color: #ff9933;font-size:20px;  
         transition: 1s linear all;}
```

```
  .fade.ng-enter { opacity: 0; }
```

```
  .fade.ng-enter.ng-enter-active { opacity: 1; }
```

```
  .fade.ng-leave { opacity: 1; }
```

```
  .fade.ng-leave.ng-leave-active { opacity: 0; }
```

```
</style>
```

Demo:ex05.html

## 2

# 自定义动画样式

```
<li ng-repeat="item in items" class="fade">
```

```
<style>
```

```
  .fade {color: #ff9933;font-size:20px;  
        transition: 1s linear all;}
```

```
  .fade.ng-enter { opacity: 0; }
```

```
  .fade.ng-enter.ng-enter-active { opacity: 1; }
```

```
  .fade.ng-leave { opacity: 1; }
```

```
  .fade.ng-leave.ng-leave-active { opacity: 0; }
```

```
</style>
```

Demo:ex06.html

# 3

javascript 动画

**\$animate服务**提供了基本的DOM操作功能，如在DOM中插入、移除和移动元素，以及添加和删除类。这个服务是ngAnimate的核心服务，为CSS和Javascript提供了高档次的动画。

常用的方法：

- 1、 `enter(element,[done]);`
- 2、 `leave(element,[done]);`
- 3、 `move(element,[done]);`

## 1、enter(element, [done]);

在DOM中，将一个元素插入到元素后面或作为第一个子元素插入父元素。一旦完成，done()将会被回调。

**element**：被插入到DOM的元素。

**done**：当元素被插入DOM后执行的回调函数。

## 2、`leave(element,[done]);`

从DOM中移除元素。一旦完成，`done()`将会被调用

`element`：将会被从DOM中移除的元素。

`done`：当元素被从DOM删除后执行的回调函数。

## 3、`move(element,[ done]);`

将提供的元素在DOM中移动位置，在父元素的内部或者兄弟元素之间。一旦完成，该`done()`将会被回调

**element** : 在DOM中被移动的元素。

**done** : 当元素在DOM中被移动后执行的回调函数。

3

## \$animate服务创建动画

```
angular.module( 'myApp' )  
  .animation( '.name-animation' , function( ) {  
    return {  
      event: function(elem, done){.....}  
      leave: function(elem, done){.....}  
      move: function(elem, done){.....}  
    };  
  });
```



在AngularJS中创建JavaScript动画时需要注意：

**1、动画的名字以点 (.) 开头**

**2、所有动画行为接受两个参数：**

- 在当前即将运行动画的DOM元素中的一个对象，要么是在jQuery没有赶在AngularJS加载之前进行加载的一个jQlite对象，要么是一个jQuery对象。
- 一个动画结束时的回调函数。指令对应的行为动作被暂停，直到回调函数被调用。

使用ng-repeat指令动画

```
<style type="text/css" media="screen">
  .fade { transition: 1s linear all; }
  .fade.ng-enter { opacity: 0; }
  .fade.ng-enter.ng-enter-active { opacity: 1; }
  .fade.ng-leave { opacity: 1; }
  .fade.ng-leave.ng-leave-active { opacity: 0; }
</style>
```

```
<li ng-repeat="item in items" class="fade">
```

Demo:listing07.html

## 4

## javascript动画

```
app.animation('.fade', function(){  
  return {  
    enter: function (element, done) {  
      .....  
    },  
    leave: function (element, done) {  
      .....  
    },  
    move: function (element, done) {  
      .....  
    }  
  }  
});
```

```
<li ng-repeat="item in items" class="fade">
```

```
});
```

Demo:listing08.html

# 为指令添加动画

`<button my-dir>淡入/淡出</button>`

```
app.directive("myDir", function($animate, $compile){
  return {
    link: function (scope, element, attr) {
      var isAppend = false;
      var parent = element.parent();
      var box;
      element.on('click', function () {
        isAppend = !isAppend;
        if (isAppend) {
          box = angular.element('<div class="fade"></div>');
          scope.$apply($animate.enter(box, parent, element, function () {
            console.log("Done entering");
          }));
        } else {
          scope.$apply($animate.leave(box, function () {
            console.log("Done leaving");
          }));
        }
      });
    }
  };
});
```

Demo:listing09.html

# 本课小结

---

1. 元素指令基本动画
2. 自定义动画
3. javascript动画

# TNAKS

主讲：王智娟

QQ: 24132228

Email: wangzhijuan@onest.net