

[AngularJS 教程](#)[AngularJS 简介](#)[AngularJS 表达式](#)[AngularJS 指令](#)[AngularJS 模型](#)[AngularJS Scope\(作用域\)](#)[AngularJS 控制器](#)[AngularJS 过滤器](#)[AngularJS Service](#)[AngularJS Http](#)[AngularJS Select](#)[AngularJS 表格](#)[AngularJS SQL](#)[AngularJS HTML DOM](#)[AngularJS 事件](#)[AngularJS 模块](#)[AngularJS 表单](#)[AngularJS 输入验证](#)[AngularJS API](#)[AngularJS Bootstrap](#)[← AngularJS 包含](#)[AngularJS 依赖注入 →](#)

## AngularJS 动画

AngularJS 提供了动画效果，可以配合 CSS 使用。

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

```
<script src="http://cdn.static.runoob.com/libs/angular.js/1.4.6/angular-animate.min.js"></script>
```

还需在应用中使用模型 ngAnimate :

```
<body ng-app="ngAnimate">
```

## 什么是动画？

动画是通过改变 HTML 元素产生的动态变化效果。

### 实例

勾选复选框隐藏 DIV:

```
<body ng-app="ngAnimate">

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

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

</body>
```

AngularJS 包含

AngularJS 动画

AngularJS 依赖注入

AngularJS 路由

AngularJS 应用

AngularJS 实例

AngularJS 实例

AngularJS 参考手册

AngularJS 参考手册

尝试一下 »



应用中动画不宜太多，但合适的使用动画可以增加页面的丰富性，也可以更易让用户理解。

如果我们应用已经设置了应用名，可以把 ngAnimate 直接添加在模型中：

## 实例

```
<body ng-app="myApp">

<h1>隐藏 DIV: <input type="checkbox" ng-model="myCheck"></h1>

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

<script>
var app = angular.module('myApp', ['ngAnimate']);
</script>
```

尝试一下 »

## 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。

AngularJS 添加/移除 class 的指令：

- ng-show
- ng-hide
- ng-class
- ng-view
- ng-include

- ng-repeat
- ng-if
- ng-switch

ng-show 和 ng-hide 指令用于添加或移除 ng-hide class 的值。

其他指令会在进入 DOM 会添加 ng-enter 类，移除 DOM 会添加 ng-leave 属性。

当 HTML 元素位置改变时，ng-repeat 指令同样可以添加 ng-move 类。

此外，在动画完成后，HTML 元素的类集合将被移除。例如：ng-hide 指令会添加一下类：

- ng-animate
- ng-hide-animate
- ng-hide-add (如果元素将被隐藏)
- ng-hide-remove (如果元素将显示)
- ng-hide-add-active (如果元素将隐藏)
- ng-hide-remove-active (如果元素将显示)

---

## 使用 CSS 动画

我们可以使用 CSS transition(过渡) 或 CSS 动画让 HTML 元素产生动画效果，该部分内容你可以参阅我们的 [CSS 过渡教程](#)，[CSS 动画教程](#)。

---

## CSS 过渡

CSS 过渡可以让我们平滑的将一个 CSS 属性值修改为另外一个：

### 实例

在 DIV 元素设置了 .ng-hide 类时，过渡需要花费 0.5 秒，高度从 100px 变为 0:

```
<style>
div {
  transition: all linear 0.5s;
  background-color: lightblue;
  height: 100px;
```

```
}  
.ng-hide {  
  height: 0;  
}  
</style>
```

[尝试一下 »](#)

## CSS 动画


CSS 动画允许你平滑的修改 CSS 属性值:

### 实例

在 DIV 元素设置了 `.ng-hide` 类时, `myChange` 动画将执行, 它会平滑的将高度从 100px 变为 0:

```
<style>  
@keyframes myChange {  
  from {  
    height: 100px;  
  } to {  
    height: 0;  
  }  
}  
div {  
  height: 100px;  
  background-color: lightblue;  
}  
div.ng-hide {  
  animation: 0.5s myChange;  
}  
</style>
```

[尝试一下 »](#)[← AngularJS 包含](#)[AngularJS 依赖注入 →](#)

 点我分享笔记

笔记需要是本篇文章的内容扩展！

[注册邀请码获取方式](#)



Copyright © 2013-2017 菜鸟教程 [runoob.com](#) All Rights Reserved. 备案号：闽ICP备15012807号-1

反馈/建议