

# AngularJS 服务(Service)

AngularJS 中你可以创建自己的服务，或使用内建服务。

## 什么是服务？

在 AngularJS 中，服务是一个函数或对象，可在你的 AngularJS 应用中使用。

AngularJS 内建了30 多个服务。

有个 **\$location** 服务，它可以返回当前页面的 URL 地址。

### 实例

```
var app = angular.module('myApp', []);
app.controller('customersCtrl', function($scope, $location) {
    $scope.myUrl = $location.absUrl();
});
```

尝试一下 »

注意 **\$location** 服务是作为一个参数传递到 controller 中。如果要使用它，需要在 controller 中定义。

## 为什么使用服务？

在很多服务中，比如 **\$location** 服务，它可以使用 DOM 中存在的对象，类似 **window.location** 对象，但 **window.location** 对象在 AngularJS 应用中有一定的局限性。

AngularJS 会一直监控应用，处理事件变化，AngularJS 使用 **\$location** 服务比使用 **window.location** 对象更好。

### \$location vs window.location

	window.location	\$location.service
目的	允许对当前浏览器位置进行读写操作	允许对当前浏览器位置进行读写操作
API	暴露一个能被读写的对象	暴露jquery风格的读写器
是否在AngularJS应用生命周期中和应用整合	否	可获取到应用生命周期内的每一个阶段，并且和\$watch整合
是否和HTML5 API的无缝整合	否	是（对低级浏览器优雅降级）
和应用的上下文是否相关	否，window.location.path返回"/docroot/actual/path"	是，\$location.path()返回"/actual/path"

## \$http 服务

**\$http** 是 AngularJS 应用中最常用的服务。服务向服务器发送请求，应用响应服务器传送过来的数据。

## 实例

使用 **\$http** 服务向服务器请求数据:

```
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope, $http) {
    $http.get("welcome.htm").then(function (response) {
        $scope.myWelcome = response.data;
    });
});
```

尝试一下 »

以上是一个非常简单的 **\$http** 服务实例，更多 **\$http** 服务应用请查看 [AngularJS Http 教程](#)。

## \$timeout 服务

AngularJS **\$timeout** 服务对应了 JS **window.setTimeout** 函数。

### 实例

两秒后显示信息:

```
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope, $timeout) {
    $scope.myHeader = "Hello World!";
    $timeout(function () {
        $scope.myHeader = "How are you today?";
    }, 2000);
});
```

尝试一下 »

## \$interval 服务

AngularJS **\$interval** 服务对应了 JS **window.setInterval** 函数。

### 实例

每一秒显示信息:

```
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope, $interval) {
    $scope.theTime = new Date().toLocaleTimeString();
    $interval(function () {
        $scope.theTime = new Date().toLocaleTimeString();
    }, 1000);
});
```

尝试一下 »

## 创建自定义服务

你可以创建自定义服务，链接到你的模块中：

创建名为**hexafy**的服务：

```
app.service('hexafy', function() {
  this.myFunc = function (x) {
    return x.toString(16);
  }
});
```

要使用自定义服务，需要在定义控制器的时候独立添加，设置依赖关系：

### 实例

使用自定义的服务 **hexafy** 将一个数字转换为16进制数：

```
app.controller('myCtrl', function($scope, hexafy) {
  $scope.hex = hexafy.myFunc(255);
});
```

[尝试一下 »](#)

## 过滤器中，使用自定义服务

当你创建了自定义服务，并连接到你的应用上后，你可以在控制器，指令，过滤器或其他服务中使用它。

在过滤器 **myFormat** 中使用服务 **hexafy**：

```
app.filter('myFormat', ['hexafy', function(hexafy) {
  return function(x) {
    return hexafy.myFunc(x);
  };
}]);
```


[尝试一下 »](#)

在对象数组中获取值时你可以使用过滤器：


创建服务 **hexafy**：

```
<ul>
<li ng-repeat="x in counts">{{x | myFormat}}</li>
</ul>
```

[尝试一下 »](#)[← AngularJS Select\(选择框\)](#)[AngularJS 动画 →](#)



7 篇笔记

 写笔记