# 9.AngularJS XMLHttpRequest

**$http** 是 AngularJS 中的一个核心服务，用于读取远程服务器的数据。

使用格式：

// 简单的 GET 请求，可以改为 POST

$http({

    method: 'GET',

    url: '/someUrl'

}).then(function successCallback(response) {

        // 请求成功执行代码

    }, function errorCallback(response) {

        // 请求失败执行代码

});

### **简写方法**

POST 与 GET 简写方法格式：

$http.get('/someUrl', config).then(successCallback, errorCallback);

$http.post('/someUrl', data, config).then(successCallback, errorCallback);

此外还有以下简写方法：

* $http.get
* $http.head
* $http.post
* $http.put
* $http.delete
* $http.jsonp
* $http.patch

更详细内容可参见：[https://docs.angularjs.org/api/ng/service/$http](https://docs.angularjs.org/api/ng/service/$http" \t "https://www.runoob.com/angularjs/_blank)

## 读取 JSON 文件

以下是存储在web服务器上的 JSON 文件：

## https://www.runoob.com/try/angularjs/data/sites.php

{ "sites": [ { "Name": "菜鸟教程", "Url": "www.runoob.com", "Country": "CN" }, { "Name": "Google", "Url": "www.google.com", "Country": "USA" }, { "Name": "Facebook", "Url": "www.facebook.com", "Country": "USA" }, { "Name": "微博", "Url": "www.weibo.com", "Country": "CN" } ] }

## AngularJS $http

AngularJS $http 是一个用于读取web服务器上数据的服务。

$http.get(url) 是用于读取服务器数据的函数。

### **废弃声明 (v1.5)**

*v1.5 中$http 的 success 和 error 方法已废弃。使用 then 方法替代。*

### **通用方法实例**

## AngularJS1.5 以上版本 - 实例

var app = angular.module('myApp', []); app.controller('siteCtrl', function($scope, $http) { $http({ method: 'GET', url: 'https://www.runoob.com/try/angularjs/data/sites.php' }).then(function successCallback(response) { $scope.names = response.data.sites; }, function errorCallback(response) { // 请求失败执行代码 }); });

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_customers_json3" \t "https://www.runoob.com/angularjs/_blank)**

### **简写方法实例**

## AngularJS1.5 以上版本 - 实例

<div ng-app="myApp" ng-controller="siteCtrl"> <ul> <li ng-repeat="x in names"> {{ x.Name + ', ' + x.Country }} </li> </ul> </div> <script> var app = angular.module('myApp', []); app.controller('siteCtrl', function($scope, $http) { $http.get("https://www.runoob.com/try/angularjs/data/sites.php") .then(function (response) {$scope.names = response.data.sites;}); }); </script>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_customers_json2" \t "https://www.runoob.com/angularjs/_blank)**

## AngularJS1.5 以下版本 - 实例

<div ng-app="myApp" ng-controller="siteCtrl"> <ul> <li ng-repeat="x in names"> {{ x.Name + ', ' + x.Country }} </li> </ul> </div> <script> var app = angular.module('myApp', []); app.controller('siteCtrl', function($scope, $http) { $http.get("https://www.runoob.com/try/angularjs/data/sites.php") .success(function (response) {$scope.names = response.sites;}); }); </script>

**[尝试一下 »](https://www.runoob.com/try/tryit.php?filename=try_ng_customers_json" \t "https://www.runoob.com/angularjs/_blank)**

**应用解析:**

注意：以上代码的 get 请求是本站的服务器，你不能直接拷贝到你本地运行，会存在跨域问题，解决办法就是将 Customers\_JSON.php 的数据拷贝到你自己的服务器上，附：[PHP Ajax 跨域问题最佳解决方案](https://www.runoob.com/w3cnote/php-ajax-cross-border.html" \t "https://www.runoob.com/angularjs/_blank)。

AngularJS 应用通过 **ng-app** 定义。应用在 <div> 中执行。

**ng-controller** 指令设置了 **controller 对象** 名。

函数 **customersController** 是一个标准的 JavaScript **对象构造器**。

控制器对象有一个属性: **$scope.names**。

**$http.get()** 从web服务器上读取静态 **JSON 数据**。

服务器数据文件为：  [https://www.runoob.com/try/angularjs/data/sites.php](https://www.runoob.com/try/angularjs/data/sites.php" \t "https://www.runoob.com/angularjs/_blank)。

当从服务端载入 JSON 数据时，**$scope.names** 变为一个数组。

|  |  |
| --- | --- |
| **IMG_256** | 以上代码也可以用于读取数据库数据。 |