# 通过终端调试 AngularJS 应用

来源：<http://www.oschina.net/translate/angularjs-console>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 当我们构建AngularJS应用时，通过浏览器（如Chrome，Firefox和IE）的JavaScript控制台访问应用中隐藏的数据和服务总会有些困难。下面是一些简单的技巧可以帮助我们通过Javascript控制台来查看或者控制正在运行的Angular应用，使得应用可以比较容易进行测试，修改，甚至实时的修改我们的Angular应用： 1： 访问作用域 通过一行简单的JS程序访问页面中任何作用域（甚至是隔离的作用域！）：  [?](http://www.oschina.net/translate/angularjs-console)   |  |  | | --- | --- | | 1  2 | > angular.element(targetNode).scope()  -> ChildScope {$id: "005", this: ChildScope, $$listeners: Object, $$listenerCount: Object, $parent: Scope…} |   对于隔离作用域：  [?](http://www.oschina.net/translate/angularjs-console)   |  |  | | --- | --- | | 1  2 | > angular.element(targetNode).isolateScope()  -> Scope {$id: "009", $$childTail: ChildScope, $$childHead: ChildScope, $$prevSibling: ChildScope, $$nextSibling: Scope…} |   这里用`targetNode`作为HTML节点的引用。你可以非常轻松的通过`document.querySelector()`来创建一个`targetNode` | |
|
| 2：查看作用域树 有些时候，我们需要查看页面中作用域层次来有效的调试我们的应用。[AngularJS Batarang](https://chrome.google.com/webstore/detail/angularjs-batarang/ighdmehidhipcmcojjgiloacoafjmpfk?hl=en)正是我们需要的一款Chrome浏览器的扩展，可以展示当前作用域层次，并具有其他非常有用的特性。  http://static.oschina.net/uploads/space/2014/0701/172716_Bb35_12.png 3： 抓取任何服务 无论ngApp在哪里定义，我们都可以使用注入器功能来抓取任何的服务的引用（如果使用angular的bootstrap方法，则可以手动抓取$rootElement）：  [?](http://www.oschina.net/translate/angularjs-console)   |  |  | | --- | --- | | 1  2 | > angular.element('html').injector().get('MyService')  -> Object {undo: function, redo: function, \_pushAction: function, newDocument: function, init: function…} |   然后我们就可以对该服务进行调用，就像我们可以将服务注入一样。 |

|  |  |  |
| --- | --- | --- |
| 4: 访问控制器使用指令 一些指令定义了一个拥有某些额外（通常是分享）功能的控制器。为了从控制台访问一个给定指令的控制器实例，只需使用 controller() 方法：  [?](http://www.oschina.net/translate/angularjs-console)   |  |  | | --- | --- | | 1  2 | > angular.element('my-pages').controller()  -> Constructor {} |   最后一种做法更高级并且不常用。 5: Chrome 控制台特性 Chrome浏览器的控制台有一堆[不错的捷径](https://developer.chrome.com/devtools/docs/commandline-api) 来调试浏览器应用。这是一些Angular开发中最好的做法：   * $0-$4: 访问最近在查看窗口中进行选取的 5 个DOM元素。选择抓取的范围非常方便。 * $(selector)和$$(selector): 分别是querySelector() 和 querySelectorAll的一个快速的替代   感谢 [@zgohr](http://twitter.com/zgohr) 提供这种方法! |