| 创建自定义的指令：**directive** | |
| --- | --- |
| 除了 AngularJS 内置的指令外，我们还可以创建自定义指令。  你可以使用 .directive 函数来添加自定义的指令。  要调用自定义指令，HTML 元素上需要添加自定义指令名。  使用驼峰法来命名一个指令， runoobDirective, 但在使用它时需要以 - 分割, runoob-directive: AngularJS 实例 <body ng-app="myApp"> <runoob-directive></runoob-directive> <script> var app = angular.module("myApp", []); app.directive("runoobDirective", function() {     return {         template : "<h1>自定义指令!</h1>"     }; }); </script> </body>  你可以通过以下方式来调用指令：  元素名  属性  类名  注释  以下实例方式也能输出同样结果:  元素名  <runoob-directive></runoob-directive>  属性  <div runoob-directive></div>  类名  <div class="runoob-directive"></div>  注释  <!-- directive: runoob-directive --> 限制使用 你可以限制你的指令只能通过特定的方式来调用。 实例 通过添加 **restrict** 属性,并设置值为 "A", 来设置指令只能通过属性的方式来调用:  var app = angular.module("myApp", []); app.directive("runoobDirective", function() {     return {         restrict : "A",         template : "<h1>自定义指令!</h1>"     }; });  **restrict** 值可以是以下几种:   * E 作为元素名使用 * A 作为属性使用 * C 作为类名使用 * M 作为注释使用   **restrict** 默认值为 EA, 即可以通过元素名和属性名来调用指令。 | |
|  | **本周时间（本周周一的上一天）** |
| **创建自定义的指令** |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |