avarteq

SCOPE EXTENSION

`.controller()`   Defines a new controller
`.directive()`   Defines a new directive
`.factory()`   Defines a new factory
`.filter()`   Defines a new filter
`.service()`   Defines a new service
`.provider()`   Defines a new provider
`.value()`   Defines a new value
`.const()`   Defines a new const value

GLOBAL FUNCTIONS

`angular.bind(self, fn, args)`
   Bind function to (other) context
`angular.bootstrap(element, [modules])`
   Bootstrap the Angular application manually
`angular.copy(source, [destination])`
   Create a deep copy of source
`angular.element(element)`
   Wrap a raw DOM element or HTML string in a jQuery element
`angular.equals(obj1, obj2)`
   Check if two objects are equal
`angular.fromJson(json)`
   Deserialize a JSON string
`angular.isArray(value)`
   Check if reference is an Array
`angular.isDate(value)`
   Check if reference is a Date
`angular.isDefined(value)`
   Check if reference is defined
`angular.isNumber(value)`
   Check if reference is a Number
`angular.toJson(obj, [pretty])`
   Serialize input into a JSON string
`angular.version`
   Provide information about AngularJS version

DIRECTIVE OPTIONS

`scope:`
   `falsy`   Use parent scope (default)
   `true`   Create child scope
   `{...}`   Create isolated scope
     `@`   Bind DOM attribute
     `=`   Required bind from parent scope
     `=?`   Optional parent binding
     `=*`   Shallow parent binding
`restrict:`   (default: A, E)
   `A`   Use attribute name (`<div directive>`)
   `E`   Use element name (`<directive>`)
   `C`   Use class name (`<div class="directive">`)
`controller`   Controller class
`replace`   Replace HTML element
`transclude`   Extract the contents of the element where the directive appears and make it available to the directive.
`link`   Register DOM listeners
`template`   Template HTML as a string
`templateUrl`   Load template from specified URL
`templateNamespace`
   String representing the document type used by the markup in the template. (HTML, SVG, Math).

BUILT-IN DIRECTIVES

`ng-app="moduleName"`
   Add rootScope
`ng-controller="controllerName"`
   Add scope
`ng-hide|show="bool"`
   Hide or show an element
`ng-model="obj"`
   Bind view to a model
`ng-repeat="obj"`
   Repeat an array
`ng-view`
   Print a template
`ng-bind`
   Bind variable to a HTML tag as content
`ng-cloak`
   Prevent template from being displayed until the application has loaded

ng-FILTERS

`currency`   Format number as a currency
`date`   Format as a date with a requested format
`filter`   Select a subset from an array
`json`   Convert JavaScript objects to a JSON string
`limitTo`   Get a specified number of elements
`lowercase`   Convert a string to lowercase
`uppercase`   Convert a string to uppercase
`number`   Format output as a number
`orderBy`   Order output by expression

SERVICES

`$anchorScroll`

`$animate.on(event, container, callback)`
DOM animation

`$animateCss`
Trigger CSS-based animation

`$cacheFactory`
Construct Cache objects

`$controller`
Instantiate a controller

`$document`
Angular wrapper for `window.document`

`$exceptionHandler`
Handle UncaughtException

`$filter`
Format data

`$http`
Make HTTP requests

`$httpBackend`
Delegates to XMLHttpRequest object or JSONP

`$httpParamSerializer`
Converts objects to strings

`$httpParamSerializerJQLike`
Alternative `$http` param serializer

`$interpolate`
Compiles a string with markup into an interpolation function

`$interval`
Execute a function every interval

`$locale`
Provides localization rules

`$location`
Provides interface to URL manipulation

`$log`
Service for logging

`$parse`
Converts Angular expression into a function

`$q`
Run functions asynchronously

`$rootElement`
Root element for Angular application

`$rootScope`
Global variable store

`$sce`
Provides strict contextual escaping

`$sceDelegate`
Used by `$sce`

`$templateCache`
Caches template files

`$templateRequest`
Runs security checks and downloads template

`$timeout`
Angular wrapper for `window.setTimeout`

`$window`
Reference to browser window object

`$xhrFactory`
Create XMLHttpRequest objects

PROVIDERS

`$anchorScrollProvider`
Disable automated scrolling

`$animateProvider`
Synchronously performs DOM animation

`$compileProvider`
Synchronously performs DOM animation

`$controllerProvider`
Register a new controller

`$filterProvider`
Register a new filter

`$httpProvider`
Change the default behavior of the `$http` service

`$interpolateProvider`
Configure the interpolation markup

`$locationProvider`
Configure how your application is handling the deep linking paths

`$logProvider`
Configure how the application logs messages

`$parseProvider`
Used for configuring the default behavior of the `$parse` service

`$rootScopeProvider`
Configuration for the `$rootScope` service

`$sceDelegateProvider`
Configuration for the `$rootScope` service

`$sceProvider`
Configuration for the `$sce` service