Service components in ng

|  |  |
| --- | --- |
| **Name** | **Description** |
| [$anchorScroll](https://docs.angularjs.org/api/ng/service/$anchorScroll) | When called, it scrolls to the element related to the specified hash or (if omitted) to the current value of [$location.hash()](https://docs.angularjs.org/api/ng/service/$location#hash), according to the rules specified in the [HTML5 spec](http://www.w3.org/html/wg/drafts/html/master/browsers.html#an-indicated-part-of-the-document). |
| [$animate](https://docs.angularjs.org/api/ng/service/$animate) | The $animate service exposes a series of DOM utility methods that provide support for animation hooks. The default behavior is the application of DOM operations, however, when an animation is detected (and animations are enabled), $animate will do the heavy lifting to ensure that animation runs with the triggered DOM operation. |
| [$animateCss](https://docs.angularjs.org/api/ng/service/$animateCss) | This is the core version of $animateCss. By default, only when the ngAnimate is included, then the $animateCss service will actually perform animations. |
| [$cacheFactory](https://docs.angularjs.org/api/ng/service/$cacheFactory) | Factory that constructs [Cache](https://docs.angularjs.org/api/ng/type/$cacheFactory.Cache) objects and gives access to them. |
| [$templateCache](https://docs.angularjs.org/api/ng/service/$templateCache) | $templateCache is a [Cache object](https://docs.angularjs.org/api/ng/type/$cacheFactory.Cache) created by the [$cacheFactory](https://docs.angularjs.org/api/ng/service/$cacheFactory). |
| [$compile](https://docs.angularjs.org/api/ng/service/$compile) | Compiles an HTML string or DOM into a template and produces a template function, which can then be used to link [scope](https://docs.angularjs.org/api/ng/type/$rootScope.Scope) and the template together. |
| [$controller](https://docs.angularjs.org/api/ng/service/$controller) | $controller service is responsible for instantiating controllers. |
| [$document](https://docs.angularjs.org/api/ng/service/$document) | A [jQuery or jqLite](https://docs.angularjs.org/api/ng/function/angular.element) wrapper for the browser's window.document object. |
| [$exceptionHandler](https://docs.angularjs.org/api/ng/service/$exceptionHandler) | Any uncaught exception in AngularJS expressions is delegated to this service. The default implementation simply delegates to $log.error which logs it into the browser console. |
| [$filter](https://docs.angularjs.org/api/ng/service/$filter) | Filters are used for formatting data displayed to the user. |
| [$httpParamSerializer](https://docs.angularjs.org/api/ng/service/$httpParamSerializer) | Default [$http](https://docs.angularjs.org/api/ng/service/$http) params serializer that converts objects to strings according to the following rules: |
| [$httpParamSerializerJQLike](https://docs.angularjs.org/api/ng/service/$httpParamSerializerJQLike) | Alternative [$http](https://docs.angularjs.org/api/ng/service/$http) params serializer that follows jQuery's [param()](http://api.jquery.com/jquery.param/) method logic. The serializer will also sort the params alphabetically. |
| [$http](https://docs.angularjs.org/api/ng/service/$http) | The $http service is a core AngularJS service that facilitates communication with the remote HTTP servers via the browser's [XMLHttpRequest](https://developer.mozilla.org/en/xmlhttprequest) object or via [JSONP](http://en.wikipedia.org/wiki/JSONP). |
| [$xhrFactory](https://docs.angularjs.org/api/ng/service/$xhrFactory) | Factory function used to create XMLHttpRequest objects. |
| [$httpBackend](https://docs.angularjs.org/api/ng/service/$httpBackend) | HTTP backend used by the [service](https://docs.angularjs.org/api/ng/service/$http) that delegates to XMLHttpRequest object or JSONP and deals with browser incompatibilities. |
| [$interpolate](https://docs.angularjs.org/api/ng/service/$interpolate) | Compiles a string with markup into an interpolation function. This service is used by the HTML [$compile](https://docs.angularjs.org/api/ng/service/$compile) service for data binding. See [$interpolateProvider](https://docs.angularjs.org/api/ng/provider/$interpolateProvider) for configuring the interpolation markup. |
| [$interval](https://docs.angularjs.org/api/ng/service/$interval) | AngularJS's wrapper for window.setInterval. The fn function is executed every delay milliseconds. |
| [$jsonpCallbacks](https://docs.angularjs.org/api/ng/service/$jsonpCallbacks) | This service handles the lifecycle of callbacks to handle JSONP requests. Override this service if you wish to customise where the callbacks are stored and how they vary compared to the requested url. |
| [$locale](https://docs.angularjs.org/api/ng/service/$locale) | $locale service provides localization rules for various AngularJS components. As of right now the only public api is: |
| [$location](https://docs.angularjs.org/api/ng/service/$location) | The $location service parses the URL in the browser address bar (based on the [window.location](https://developer.mozilla.org/en/window.location)) and makes the URL available to your application. Changes to the URL in the address bar are reflected into $location service and changes to $location are reflected into the browser address bar. |
| [$log](https://docs.angularjs.org/api/ng/service/$log) | Simple service for logging. Default implementation safely writes the message into the browser's console (if present). |
| [$parse](https://docs.angularjs.org/api/ng/service/$parse) | Converts AngularJS [expression](https://docs.angularjs.org/guide/expression) into a function. |
| [$q](https://docs.angularjs.org/api/ng/service/$q) | A service that helps you run functions asynchronously, and use their return values (or exceptions) when they are done processing. |
| [$rootElement](https://docs.angularjs.org/api/ng/service/$rootElement) | The root element of AngularJS application. This is either the element where [ngApp](https://docs.angularjs.org/api/ng/directive/ngApp) was declared or the element passed into [angular.bootstrap](https://docs.angularjs.org/api/ng/function/angular.bootstrap). The element represents the root element of application. It is also the location where the application's [$injector](https://docs.angularjs.org/api/auto/service/$injector) service gets published, and can be retrieved using $rootElement.injector(). |
| [$rootScope](https://docs.angularjs.org/api/ng/service/$rootScope) | Every application has a single root [scope](https://docs.angularjs.org/api/ng/type/$rootScope.Scope). All other scopes are descendant scopes of the root scope. Scopes provide separation between the model and the view, via a mechanism for watching the model for changes. They also provide event emission/broadcast and subscription facility. See the [developer guide on scopes](https://docs.angularjs.org/guide/scope). |
| [$sceDelegate](https://docs.angularjs.org/api/ng/service/$sceDelegate) | $sceDelegate is a service that is used by the $sce service to provide [Strict Contextual Escaping (SCE)](https://docs.angularjs.org/api/ng/service/$sce) services to AngularJS. |
| [$sce](https://docs.angularjs.org/api/ng/service/$sce) | $sce is a service that provides Strict Contextual Escaping services to AngularJS. |
| [$templateRequest](https://docs.angularjs.org/api/ng/service/$templateRequest) | The $templateRequest service runs security checks then downloads the provided template using $http and, upon success, stores the contents inside of $templateCache. If the HTTP request fails or the response data of the HTTP request is empty, a $compile error will be thrown (the exception can be thwarted by setting the 2nd parameter of the function to true). Note that the contents of $templateCache are trusted, so the call to $sce.getTrustedUrl(tpl) is omitted when tpl is of type string and $templateCache has the matching entry. |
| [$timeout](https://docs.angularjs.org/api/ng/service/$timeout) | AngularJS's wrapper for window.setTimeout. The fn function is wrapped into a try/catch block and delegates any exceptions to [$exceptionHandler](https://docs.angularjs.org/api/ng/service/$exceptionHandler) service. |
| [$window](https://docs.angularjs.org/api/ng/service/$window) | A reference to the browser's window object. While window is globally available in JavaScript, it causes testability problems, because it is a global variable. In AngularJS we always refer to it through the $window service, so it may be overridden, removed or mocked for testing. |

<https://docs.angularjs.org/api/ng/service>