SERVICES IN ANGULARJS

- The AngularJS service is a function or object that is available to your AngularJS application.
- Services are JavaScript functions and it performs specific tasks only.
- Inbuilt services are prefixed with $ symbol such as $http, $location, $timeout, $interval.
- Most of the applications create an own service.
- Following are some important 30 built-in services are available in AngularJS.

Why use Services?

- Several services use the DOM objects, but it would have some limitations in your AngularJS application.
- For ex, $location service use the objects that are already in the DOM objects (like window.location). AngularJS controls your application and it handles the changes and events properly.
- AngularJS prefers, Use $location service instead of window.location DOM object.

Create an Own Service in AngularJS:

- To create an own service, connect a service into the module.

Syntax for create an own service in AngularJS:

```
app.service('servicename', function()
{
    ...
});
```

Sample Coding for create an own service in AngularJS:

```
<! DOCTYPE html>
<html>
  <head>
    <title>Wikitechy AngularJS Tutorial</title>
    <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/angular.min.js">
    </script>
  </head>
  <body>
    <div ng-app="myApp" ng-controller="serviceCtrl">
      <h3> Multiplication of two number using Custom Service in
AngularJS</h3>
      <p>Multiply value of 10 and 50 is <b>{{mul}}</b> </p>
    </div>
    <script>
      var app = angular.module('myApp', []);
      app.service('multiply', function() {
        this.myFunc = function (a,b) {
          return a*b;
        }
      });
      app.controller('serviceCtrl', function($scope, multiply) {
        $scope.mul = multiply.myFunc(10,50);
      });
    </script>
  </body>
</html>
```

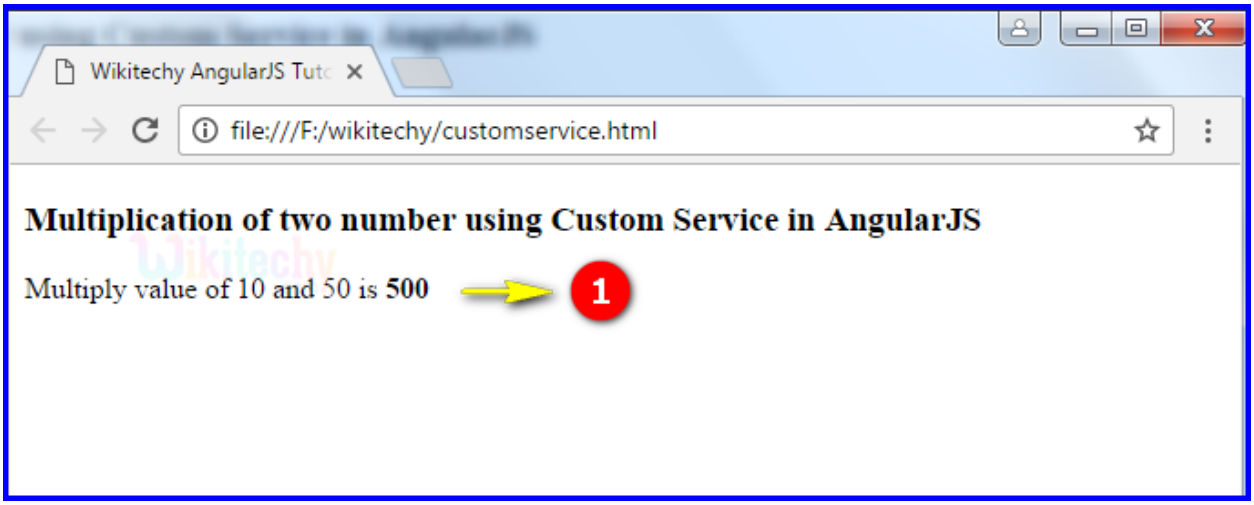
Code Explanation for create an own service in AngularJS:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Wikitechy AngularJS Tutorial</title>
        <script src="https://ajax.googleapis.com/ajax/libs/angularjs/
        1.5.6/angular.min.js"></script>
    </head>
    <body>
        <div ng-app="myApp" ng-controller="serviceCtrl">
            <h3> Multiplication of two number using Custom
                Service in AngularJS</h3>
            <p>Multiply value of 10 and 50 is <b>{{mul}}</b> </p>
        </div>
        <script>
            var app = angular.module('myApp', []);
            app.service('multiply', function() {
                this.myFunc = function (a,b) {
                    return a*b;
                }
            });
            app.controller('serviceCtrl', function($scope, multiply) {
                $scope.mul = multiply.myFunc(10,50);
            });
        </script>
    </body>
</html>
```

1. The ng-app specifies the root element ("myApp") to define AngularJS application.
2. The ng-controller control the data of **"serviceCtrl"** in AngularJS application.
3. Here create an own services (app.service) and the service name is given as "multiply" then the service is connect with the module.

4. The Function (myFunc) is return the multiplication of two values and get the values from the Scope object ($scope.mul).
5. The **"multiply"** service is added with the controller function.
6. $scope.mul is used to declare the multiplication values 10 and 50 and the output will be updated in the <script> tag.

Sample Output for create an own service in AngularJS:



1. The output displays the multiplication of two numbers is 500.

AngularJS Services:

Services	Description
$anchorScroll	scrolls the element related to the specified hash or the current value of $location.hash(), according to the rules are specified in the HTML5
$animate	It is used to design a date for specified format The $animate service exposes a series of DOM utility methods that support for animation hooks. The default behavior is the application of DOM operations. When an animation is detected or enabled $animate will do the heavy lifting to ensure that animation runs with the triggered DOM operation.
$animateCss	It is used to specify the core version of $animateCss. This service will actually performs the animations and **ngAnimate** is default one.
$cacheFactory	It is used to constructs Cache objects and gives access to them.
$compile	This service Compiles an HTML string or DOM into a template and produces a template function, which can be used to link scope and the template together.
$controller	$controller service is responsible for instantiating controllers.
$document	A **jQuery** or **jqLite** element wrapper for the browser's **window.document** object.
$exceptionHandler	Several uncaught exception in angular expressions is delegated to the service. The default implementation is represented by **$log.error** which logs it into the browser console.
$filter	It is used to formats the value of an expression and display to the user.
$http	$http service is used to core Angular service that enables communication with the remote HTTP

	servers via the browser's **XMLHttpRequest** object or **JSONP**
$httpBackend	It is used to specify the **service** that delegates to XMLHttpRequest object or JSONP and it deals with the browser incompatibilities.
$httpParamSerializer	It is used to converts objects to strings by default.
$httpParamSerializerJQLike	Different $http params serializer follows jQuery **param()** method logic. The serializer will sort the params alphabetically.
$interpolate	It is used to compiles a string with markup into an interpolation function. $interpolate services used by
$interval	It is used to specify the Angular's wrapper for **window.setInterval** and the fn function is executed every delay milliseconds
$jsonpCallbacks	This service handles the lifecycle of callback and the JSONP requests. Override this service to customize the callbacks and they are stored in requested url.
$locale	It provides localization rules for various Angular components.
$location	The $location service parses the URL in the browser address bar 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	It is used for logging and default implementation is write the messages into the browser's console
$parse	It is used to Converts Angular expression into a function.
$q	This services helps to run the functions asynchronously, and it can be use their return values (or exceptions)
$rootElement	The root element of Angular application. ngApp was declared or the element passed into angular.bootstrap.

$rootScope	Each application has a single root scope. All other scopes are descendant of the root scope. Normally, scopes provide separation between the model and the view. It is delivered the emission/broadcast and subscription facility.
$sce	This service provides Strict Contextual Excaping services to AngularJS.
$sceDelegate	It is used by the $sce service and it is provide Strict Contextual Escaping(SCE) services to AngularJS.
$templateCache	It is loaded in the template cache for quick retrieval and it is also directly loaded into the cache in a script tag or by consuming the $templateCache service directly.
$templateRequest	The $templateRequest service runs security checks then downloads the provided template using$http it can stores the contents inside of $templateCache. If the HTTP request fails or the response data of the HTTP request is empty, $compile error will be thrown .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.
$timeout	It is used to specify the Angular wrapper for window.setTimeout. The function is wrapped into a try/catch block and delegates any exceptions to **$exceptionHandler** service
$window	It is reference to the brower's window object and the window is globally available in JavaScript, it causes the testablility problems, because it is a global variable. In angular we always refer to the $window service, so it may be overridden, removed or mocked for testing.
$xhrFactory	It is used to create XMLHttpRequest objects.

