

\$provide.decorator



gabe@zoz.com

```
angular.module('app', [])  
  .config(function($provide) {  
    // configuration stuff  
  });
```

```
angular.module('app', [])  
  .config(function($provide) {  
    // configuration stuff  
  });
```

\$provide.service

\$provide.factory

\$provide.value

\$provide.constant

\$provide.provider

```
angular.module('app', [])
```

```
$p.service(...)
```

```
$p.factory(...)
```

```
$p.value(...)
```

```
$p.constant(...)
```

```
$p.provider(...)
```



`$provide.decorator`

“decorator”

angular

```
.module('app', [])  
.factory('SomeService', function() {  
    return 'hello';  
})  
  
.config(function($provide) {  
    $provide.decorator('SomeService',  
        function($delegate) {  
            return $delegate.toUpperCase();  
        });  
});
```



```
angular
  .module('app', [])
  .factory('SomeService', function() {
    return 'hello';
  })
  .config(function($provide) {
    $provide.decorator('SomeService',
      function($delegate) {
        return $delegate.toUpperCase();
      });
  });
```

```
angular
  .module('app', [])
  .factory('SomeService', function() {
    return 'hello';
  })
  .config(function($provide) {
    $provide.decorator('SomeService',
      function($delegate) {
        return $delegate.toUpperCase();
      });
  });
```

```
angular
  .module('app', [])
  .factory('SomeService', function() {
    return 'hello';
  })
  .config(function($provide) {
    $provide.decorator('SomeService',
      function($delegate) {
        return $delegate.toUpperCase();
      });
  });
```

"HELLO"

~~"decorator"~~

component interceptor

"...override or
modify the
behaviour of the
service."

```
$provide.decorator('$q', function($delegate) {  
    var _all = $delegate.all, q = {};  
    angular.copy($delegate, q);  
  
    q.all = function (promises, message) {  
        console.log(message);  
        return _all(promises);  
    };  
  
    $provide.factory('$q+messages', function () {  
        return q;  
    });  
  
    return $delegate;  
});
```

```
$provide.decorator('$q', function($delegate) {  
    var _all = $delegate.all, q = {};  
    angular.copy($delegate, q);  
  
    q.all = function (promises, message) {  
        console.log(message);  
        return _all(promises);  
    };  
  
    $provide.factory('$q+messages', function () {  
        return q;  
    });  
  
    return $delegate;  
});
```



```
$provide.decorator('$q', function($delegate) {  
    var _all = $delegate.all, q = {};  
    angular.copy($delegate, q);  
  
    q.all = function (promises, message) {  
        console.log(message);  
        return _all(promises);  
    };  
  
    $provide.factory('$q+messages', function () {  
        return q;  
    });  
  
    return $delegate;  
});
```

```
$provide.decorator('$state',  
function($delegate) {  
    var state = $delegate,  
        opts = {reload: true, inherit: false, notify: true};  
  
    state.forceReload = function() {  
        return state.go(state.current, state.params, opts);  
    };  
  
    return state;  
});
```

Sources:

[http://www.slideshare.net/
damienklinnert/angular-decorate](http://www.slideshare.net/damienklinnert/angular-decorate)

[http://stackoverflow.com/questions/
21714655/angular-js-angular-ui-
router-reloading-current-state-
refresh-data](http://stackoverflow.com/questions/21714655/angular-js-angular-ui-router-reloading-current-state-refresh-data)

[https://egghead.io/lessons/
angularjs-provide-decorator](https://egghead.io/lessons/angularjs-provide-decorator)

[https://docs.angularjs.org/api/
auto/object/\\$provide](https://docs.angularjs.org/api/auto/object/$provide)

THANKS