```
(function () {
    'use strict';
    angular.module('angular-loading-bar', ['cfp.loadingBarInterceptor']);
    angular.module('chieffancypants.loadingBar', ['cfp.loadingBarInterceptor']);
    angular.module('cfp.loadingBarInterceptor', [])
      .config(['$httpProvider', function ($httpProvider) {
          var interceptor = ['$q', '$cacheFactory', '$timeout', '$rootScope', '$log',
function ($q, $cacheFactory, $timeout, $rootScope, $log) {
              var reqsTotal = 0;
              var reqsCompleted = 0;
              var startTimeout;
              function isCached(config) {
                  var cache;
                  var defaultCache = $cacheFactory.get('$http');
                  var defaults = $httpProvider.defaults;
                  // Choose the proper cache source. Borrowed from angular: $http service
                  if ((config.cache | defaults.cache) && config.cache !== false &&
                    (config.method === 'GET' || config.method === 'JSONP')) {
                      cache = angular.isObject(config.cache) ? config.cache
                        : angular.isObject(defaults.cache) ? defaults.cache
                        : defaultCache;
                  }
                  var cached = cache !== undefined ?
                    cache.get(config.url) !== undefined : false;
                  if (config.cached !== undefined && cached !== config.cached) {
                      return config.cached;
                  config.cached = cached;
                  return cached;
              }
              return {
                  'request': function (config) {
                      if (!config.ignoreLoadingBar && !isCached(config)) {
                              $("#mydiv").show();
                                                                                Show the Div
                          reqsTotal++;
                                                                               when, Request
                                                                                   Starts
                      return config;
                  },
```

```
'response': function (response) {
                      if (!response || !response.config) {
                           return response;
                                                                               Hide the
                      }
                                                                             Request when
                                                                             Request Ends
                      if (!response.config.ignoreLoadingBar &&
!isCached(response.config)) {
                           reqsCompleted++;
                          if (reqsCompleted >= reqsTotal) {
                               $("#mydiv").hide();
                      }
                      return response;
                  },
                  'responseError': function (rejection) {
                      if (!rejection || !rejection.config)
                           return $q.reject(rejection);
                      }
                      if (!rejection.config.ignoreLoadingBar &&
!isCached(rejection.config)) {
                           reqsCompleted++;
                           if (reqsCompleted >=/reqsTotal) {
                               $("#mydiv").hide();
                           } else {
                               $("#mydiv").hide();
                      return $q.reject(rejection);
                                                                                Push the
                  }
              };
                                                                               Intercepter
          }];
          $httpProvider.interceptors.push(interceptor);
      }]);
})();
var app = angular.module("Myapp", ["angular-loading-bar"]);
                                                                            Inject The
                                                                            Module
```

```
#mydiv { ←
             position: absolute;
             top: 0;
             left: 0;
                                                                       Custom Css for the
             width: 100%;
                                                                            Loader
             height: 100%;
             z-index: 1000;
             background-color: grey;
             opacity: .8;
         }
         .ajax-loader {
             position: absolute;
             left: 50%;
             top: 50%;
             height:50px;
             width:50px;
             margin-left: -32px; /* -1 * image width / 2 */
margin-top: -32px; /* -1 * image height / 2 */
             display: block;
        }
                                                                                        The Div that
                                                                                      containg loading
                                                                                           Image
<div id="mydiv" hidden>
        <img class="ajax-loader" src="Images/loading.gif" />
    </div>
```