Project edi Controller.js

angular.module('app.controllers')  
 .controller('ProjectEditController', ['$scope', '$location', '$routeParams', '$cookies', '$q', 'Project', 'Client', 'appConfig', '$filter',  
 **function** ($scope, $location, $routeParams, $cookies, $q, Project, Client, appConfig, $filter) {  
  
 Project.get({id: $routeParams.id}, **function**(data) {  
 $scope.project = data;  
 $scope.clientSelected = data.client.data;  
 });  
  
 //retornando os status de app config  
 $scope.status = appConfig.project.status;  
  
 $scope.due\_date =  
 {  
 status:  
 {opened: **false**}  
 };  
 $scope.open = **function** ($event) {  
 $scope.due\_date.status.opened = **true**;  
 };  
  
  
 $scope.save = **function**() {  
 **if**($scope.formProject.$valid) {  
 $scope.project.owner\_id = $cookies.getObject('user').id;  
 Project.update({id: $scope.project.id}, $scope.project, **function**() {  
 $location.path('/projects');  
 });  
 }  
 };  
  
 //QUANDO FOR SELECIONAR NOVAMENTE PEGAR O NOME E JOGAR  
 $scope.formatName = **function** (model) {  
 **if**(model){  
 **return** model.name;  
 }  
 **return** "";  
 };  
  
 //$scope.getClients = function (name) {  
 // //$promise - trava a execução do javascript ate os dados serem retornados  
 // return Client.query({  
 // search:name,  
 // searchFields:'name:like'  
 // }).$promise;  
 //};  
  
 $scope.getClients = **function** (name) {  
 **var** deffered = $q.defer();  
 //$promise - trava a execução do javascript ate os dados serem retornados  
 Client.query({  
 search:name,  
 searchFields:'name:like'  
 }, **function**(data){  
 **var** result = $filter('limitTo')(data.data,5);  
 deffered.resolve(result);  
 }, **function**(error){  
 deffered.reject(error);  
 });  
 **return** deffered.promise;  
 };  
  
  
 $scope.selectClient = **function** (item) {  
 $scope.project.client\_id = item.id;  
  
 };  
  
 }]);

ProjectNew Controller.js

angular.module('app.controllers')  
 .controller('ProjectNewController', [  
 '$scope', '$location', '$cookies', '$routeParams', '$q', 'Project', 'Client', 'appConfig', '$filter',  
 **function** ($scope, $location, $cookies, $routeParams, $q, Project, Client, appConfig, $filter) {  
 $scope.project = **new** Project();  
 $scope.clients = {};  
 //retornando os status de app config  
 $scope.status = appConfig.project.status;  
  
 $scope.due\_date =  
 {  
 status:  
 {opened: **false**}  
 };  
 $scope.open = **function** ($event) {  
 $scope.due\_date.status.opened = **true**;  
 };  
  
 $scope.save = **function**() {  
 **if**($scope.formProject.$valid) {  
 $scope.project.owner\_id = $cookies.getObject('user').id;  
 $scope.project.$save().then(**function** () {  
 $location.path('/projects');  
 });  
 }  
 };  
  
 //QUANDO FOR SELECIONAR NOVAMENTE PEGAR O NOME E JOGAR  
 $scope.formatName = **function** (model) {  
 **if**(model){  
 **return** model.name;  
 }  
 **return** "";  
 };  
  
 //$scope.getClients = function (name) {  
 // //$promise - trava a execução do javascript ate os dados serem retornados  
 // return Client.query({  
 // search:name,  
 // searchFields:'name:like'  
 // }).$promise;  
 // };  
 //$scope.getClients = function (name) {  
 // return ClientProject.query({  
 // id: $routeParams.id,  
 // search: name,  
 // searchFields: 'name:like'  
 // }).$promise;  
 //}  
  
 //$q - protela as promessas com seus resultados  
  
 $scope.getClients = **function** (name) {  
 **var** deffered = $q.defer();  
 //$promise - trava a execução do javascript ate os dados serem retornados  
 Client.query({  
 search:name,  
 searchFields:'name:like'  
 }, **function**(data){  
 **var** result = $filter('limitTo')(data.data,5);  
 deffered.resolve(result);  
 }, **function**(error){  
 deffered.reject(error);  
 });  
 **return** deffered.promise;  
 };  
  
  
 $scope.selectClient = **function** (item) {  
 $scope.project.client\_id = item.id;  
  
 };  
 }  
 ]);

Form Project View.html

<label class="col-md-4 control-label">Client</label>  
<div class="col-md-6">  
 <input type="text" autocomplete="off" class="form-control" name="client\_id"  
 typeahead="o as o.name for o in getClients($viewValue)"  
 typeahead-input-formatter="formatName($model)"  
 typeahead-on-select="selectClient($item)"  
 typeahead-editable="false"  
 ng-model="clientSelected" required/>  
  
 <div ng-messages="formProject.client\_id.$error" class="help-block" ng-show="formProject.client\_id.$touched">  
 <div ng-message="required">Campo obrigatório!!</div>  
 <div ng-message="editable">Selecione um client válido!</div>  
 </div>  
</div>