**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*001\_BasicFilterDemo.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="filterApp">

<head></head>

<body>

<div data-ng-controller="FilterCntrl">

{{'IGATE'|lowercase}}</br>

{{'IGATE'|lowercase}}</br>

{{1234|currency}}</br>

{{1234|currency:'Rs.'}}</br>

{{123.7866|number:2}}</br>

{{{Id:222,Name:'vaishali'}|json}}</br>

{{today|date:'fullDate'}}</br>

{{today|date:'d-m-y'}}</br>

</div>

<!-- <div data-ng-controller="FilterCntrl">

{{formattedDate}}</br>

</div> -->

</body>

<script src="Scripts/angular.js"></script>

<script type="text/javascript">

var filterApp=angular.module("filterApp",[]);

filterApp.controller("FilterCntrl",function($scope){

$scope.today=new Date();

<!-- $scope.formattedDate=$filter('date')(new Date(),'dd-MM-yyyy'); -->

});</script></html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*002\_AngularJSFilterDemo.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html data-ng-app="">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Angular JS Module Demo\_4.html</title>

</head>

<body data-ng-init="customers=[{firstName:'Vaishali',city:'Pune'},

{firstName:'Ameya',city:'Noida'},

{firstName:'Sarita',city:'Delhi'},

{firstName:'Vrushali',city:'Pune'}]"

bgcolor="pink"">

<input type="text" name="txtFirstName" data-ng-model="name"/>

{{name}}<br/>

<ul>

<li data-ng-repeat="cust in customers |orderBy:'firstName'">

{{cust.firstName |uppercase}}-{{cust.city}}

</li>

</ul>

<h2>\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*</h2>

<input type="text" name="txtLastName" data-ng-model="firstNameModel"/>

<ul>

<li data-ng-repeat="cust in customers |filter:firstNameModel|orderBy:'firstName'">

{{cust.firstName |uppercase}}-{{cust.city}}

</li>

</ul>

<h2>\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*</h2>

</body>

<script src="Scripts/angular.js"></script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*003-AngularJs-Filters.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="guestPhoneBookApp">

<head>

<title>AngularJS-Filters</title>

<link href="css/stylesheet.css" type="text/css" rel="stylesheet"/>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<div id="site-wrapper">

<div id="header-wrapper">

<p>Guest Phonebook application</p>

</div>

<div id="body-wrapper">

<table width="100%">

<tr>

<td align="center">

<div ng-controller="GuestController">

<table border="1" cellpadding="10" >

<thead>

<tr>

<th colspan="3">Applying Filters</th>

</tr>

<tr>

<th align="right">Sort by :</th>

<th>

<select ng-model="sortColumn">

<option value="+guestId">Id</option>

<option value="+guestName">Name</option>

</select>

</th>

<th>

<input ng-model="sortOrder" name="order"

type="radio"/>Ascending

<input ng-model="sortOrder" name="order"

type="radio" value="reverse"/>Descending

</th>

</tr>

<tr>

<th>Id</th>

<th>Name</th>

<th>Contact Number</th>

</tr>

</thead>

<tbody>

<tr ng-repeat="guest in guests |orderBy:sortColumn:sortOrder">

<td>{{guest.guestId}}</td>

<td>{{guest.guestName | uppercase}}</td>

<td>{{guest.contactNumber}}</td>

</tr>

</tbody>

</table>

</div>

</td>

</tr>

</table>

</div>

<div id="footer-wrapper">

<p>&copy;Copyrights Reserved 2014 IGATE.</p>

</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript" >

var guestNameSpace = angular.module('guestPhoneBookApp',[]);

guestNameSpace.controller('GuestController',function($scope){

$scope.guests = [

{

"guestId":101,

"guestName":"Anil",

"contactNumber":"+91-8645635092"

},

{

"guestId":102,

"guestName":"Ganesh",

"contactNumber":"+91-9946945634"

},

{

"guestId":103,

"guestName":"Karthik",

"contactNumber":"+91-9986173092"

},

{

"guestId":104,

"guestName":"Shashank",

"contactNumber":"+91-9986464892"

},

{

"guestId":105,

"guestName":"Abishek",

"contactNumber":"+91-9958659892"

},

{

"guestId":106,

"guestName":"Nachiket",

"contactNumber":"+91-8997358092"

},

{

"guestId":107,

"guestName":"Vaishali",

"contactNumber":"+91-9739179037"

}

];

$scope.sortColumn = "+guestId"; // - for descending Order

});

guestNameSpace.controller('GuestAjaxCntrl',function($scope,$http){

var promise = $http.get("data/guests.json");

var onSuccess = function(response){

$scope.guests = response.data;

};

var onError = function(statusText){

$scope.error = "Program Works well with Mozilla firefox & Safari. To make it work with chrome browser, run chrome.exe with following command from run window(Win+R key). chrome.exe -allow-file-access-from-files -disable-web-security";

};

promise.then(onSuccess,onError);

});

</script></html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*004-CustomFilterDemo.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="filterApp">

<head>

<title> Custom Filter Demo</title>

</head>

<body bgcolor="pink">

<h3>Capitalize First Char Of Given String Demo</h3>

{{'vaishali'|capitalizeChar}}

</body>

<script src="Scripts/angular.js"></script>

<script type="text/javascript">

var filterApp=angular.module('filterApp',[]) ;

filterApp.filter('capitalizeChar', function(){

return function(input){

if(input)

return input[0].toUpperCase()+ input.slice(1);

}

}

);

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*001-AnJsFactoryService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating Custom Service Using Factory</h1>

<div ng-app="serviceApp" ng-controller="ServiceCntrl">

<div>{{mycompany.Name}} - {{mycompany.Location}}</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript">

var app = angular.module("serviceApp", []);

app.factory("companyService", function() {

return {

company : {

"Name" : "IGATE",

"Location" : "India"

}

};

});

app.controller('ServiceCntrl', function($scope, companyService) {

$scope.mycompany = companyService.company;

})

</script></html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*002-AnJsServiceService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- If we want to register an instance of a service using a constructor function, we can use service(),

which enables us to register a constructor function for our service object.

-->

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating Custom Service Using Service</h1>

<div ng-controller="ServiceCntrl">

<div>{{mycompany.Name}} - {{mycompany.Location}}</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript">

var app = angular.module("serviceApp", []);

/\*constructor function\*/

var Company = function() {

this.getCompanyInfo = function() {

return {

"Name" : "IGATE Hinjavadi",

"Location" : "Pune "

};

};

}

app.service('companyService', Company); // service() function

app.controller('ServiceCntrl', function($scope, companyService) {

console.log(companyService.getCompanyInfo().Name);

$scope.mycompany = companyService.getCompanyInfo();

})

</script></html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*003-AnJsProviderService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- These factories are all created through the $provide service, which is responsible for instantiating

these providers at run time.

A provider is an object with a $get() method. The $injector calls the $get method to create a new

instance of the service. The $provider exposes several different API methods for creating a service,

each with a different intended use.

At the root of all the methods for creating a service is the provider method. The provider() method

is responsible for registering services in the $providerCache.

Technically, the factory() function is shorthand for creating a service through the provider()

method wherein we assume that the $get() function is the function passed in.

The two method calls are functionally equivalent and will create the same service. -->

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating Custom Service Using Provider</h1>

<div ng-app="serviceApp" ng-controller="ServiceCntrl">

<div>{{mycompany.Name}} - {{mycompany.Location}}</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript">

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*004-AnJs$ProvideService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

var app = angular.module("serviceApp", []);

app.provider("companyService", {

$get : function() {

return {

company : {

"Name" : "IGATE Noida",

"Location" : "Delhi"

}

};

}

});

app.controller('ServiceCntrl', function($scope, companyService) {

$scope.mycompany = companyService.company;

})

</script>

</html**>**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*004-AnJs$ProvideService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating Custom Service Using $provide</h1>

<div ng-controller="ServiceCntrl">

<div>{{company.Name}} - {{company.Location}}</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript">

var app = angular.module("serviceApp",[]);

app.config(function($provide){

$provide.factory('companyService',function(){

return {

company : {"Name":"IGATE London", "Location":"UK"}

};

});

});

app.controller('ServiceCntrl',function($scope,companyService){

$scope.company = companyService.company;

});

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*005-AnJsConstantService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- It’s possible to register an existing value as a service that we can later inject into other parts of our

app as a service. For instance, let’s say we want to set an apiKey for a back-end service. We can store

that constant value using constant(). -->

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating Custom Service Using Factory</h1>

<div ng-app="serviceApp" ng-controller="ServiceCntrl">

<div>{{mycompany}} - {{mycompanyLocation}}</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript">

var app = angular.module("serviceApp", []);

app.constant('Name', 'IGATE ');

app.constant('Location', 'Chennai');

app.config(function(Name) {

//We can resolve companyName here

});

app.config(function(Location) {

//We can resolve Location here

});

app.controller('ServiceCntrl', function($scope, Name,Location) {

$scope.mycompany = Name;

$scope.mycompanyLocation = Location;

})

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*006-AnJsValueService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- If the return value of the $get method in our service is a constant, we don’t need to define a fullblown

service with a more complex method. We can simply use the value() function to register the

service. -->

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating Custom Service Using Factory</h1>

<div ng-app="serviceApp" ng-controller="ServiceCntrl">

<div>{{mycompany.Id}} - {{mycompany.Name}}</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript">

var app = angular.module("serviceApp",[]);

app.value('employee',{Id: 714709, Name:'Karthik'});

app.controller('ServiceCntrl', function($scope, employee) {

$scope.mycompany = employee;

})

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*009\_1-AnJs$QService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- AngularJS web apps are entirely client-side applications.

As we’ve seen, we can write AngularJS

applications without integrating with a back end at all and still have a dynamic, responsive web

app.

Without a back end, we are limited to only showing information that we have at load time. Angular

provides us several methods if we want to integrate our AngularJS app with information from a

remote server. -->

<html ng-app="qServiceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating $q Service</h1>

<div ng-controller="QServiceCntrl">{{myValue}}</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script>

var app = angular.module("qServiceApp", []);

app.controller('QServiceCntrl',function($scope, $q) {

$scope.addOne=function(num){

var q=$q.defer();

if(angular.isNumber(num))

{

q.resolve(num+1);

}

else

{

q.resolve('Nan');

}

return q.promise;

}

$scope.myValue=8;

$scope.promise=$scope.addOne($scope.myValue);

$scope.promise.then(function(v){$scope.myValue=v;},

function(err){$scope.myValue=err;}

);

});

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*009\_2-AnJs$QService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- AngularJS web apps are entirely client-side applications.

As we’ve seen, we can write AngularJS

applications without integrating with a back end at all and still have a dynamic, responsive web

app.

Without a back end, we are limited to only showing information that we have at load time. Angular

provides us several methods if we want to integrate our AngularJS app with information from a

remote server. -->

<html ng-app="qServiceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating $q Service</h1>

<div ng-controller="QServiceCntrl">{{myValue}}</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script>

var app = angular.module("qServiceApp", []);

app.controller('QServiceCntrl',['$scope','$q',function($scope, $q) {

$scope.addOne=function(num){

var q=$q.defer();

if(angular.isNumber(num))

{

q.resolve(num+1);

}

else

{

q.resolve('Nan');

}

return q.promise;

}

$scope.myValue="6";

$scope.promise=$scope.addOne($scope.myValue);

$scope.promise.then(function(v){$scope.myValue=v;},

function(err){$scope.myValue=err;}

);

}]);

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*009\_3-AnJs$QService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<!-- AngularJS web apps are entirely client-side applications.

As we’ve seen, we can write AngularJS

applications without integrating with a back end at all and still have a dynamic, responsive web

app.

Without a back end, we are limited to only showing information that we have at load time. Angular

provides us several methods if we want to integrate our AngularJS app with information from a

remote server. -->

<html ng-app="serviceApp">

<head>

<title>Service - Custom Service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Creating $q Service</h1>

<div ng-app="serviceApp" ng-controller="ServiceCntrl">

<div>

<input type="button" value="$q Service Test" ng-click="test()" />

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script>

var app = angular.module("serviceApp", []);

app.controller('ServiceCntrl', function($scope, $q) {

$scope.test = function() {

var deferred = $q.defer();

var promise = deferred.promise;

promise.then(

function(response) {

alert('Success :' + response.message);

},

function(response) {

alert('Error :' + response.message);

}

);

/\*uncomment to resolve the promise\*/

deferred.resolve({

message : 'Promise will gets fullfilled'

});

<!-- uncomment to reject the promise-->

deferred.reject({message:'Promise gets rejected'});

}

});

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*010-LocaleService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="serviceApp">

<head>

<title>Service - $locale service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Service - $locale service</h1>

<div ng-controller="ServiceController">

<h2>{{ todayDate | date:dateFormat }}</h2>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<!--<script type="text/javascript" src="Scripts/angular-locale\_ta-in.js"></script>

<script type="text/javascript" src="Scripts/angular-locale\_kn-in.js"></script>-->

<script type="text/javascript" src="Scripts/angular-locale\_hi-in.js"></script>

<script type="text/javascript">

var app = angular.module('serviceApp',[])

app.controller("ServiceController",function($scope,$locale){

$scope.todayDate = new Date();

$scope.dateFormat = 'fullDate';

//$scope.dateFormat = $locale.DATETIME\_FORMATS.fullDate;

})

</script>

</html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*011\_1-HttpServiceDemo.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="myApp">

<head>

<title> Dependancy Injection Demo</title>

</head>

<body bgcolor="pink" data-ng-controller="ServiceController">

<h2> Data Accessed from json using http request</h3>

<ul data-ng-repeat="tempGuest in guestList" >

<li>{{tempGuest.guestId}}{{tempGuest.guestName}}</li>

</ul>

</body>

<script src="Scripts/angular.js"></script>

<script type="text/javascript">

var app=angular.module("myApp",[]);

app.factory("loadGuest",function($http,$log){

return{

retrieveGuest:function(successObj){

$http({

method:'GET',

url:'data/guests.json'

})

.success(function(data,status){

successObj(data);

$log.info(data,status);

}

)

.error(function(data,status){

$log.info(data,status);

}

)

}

}

}

);

app.controller("ServiceController",function($scope,loadGuest){

loadGuest.retrieveGuest(function(data){

$scope.guestList=data;

}

);

}

);

</script></html>

**\*\*\*\*\*\*\*\*\*\*\*011\_2\_HttpServiceDemo.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="myApp">

<head>

<title> Dependancy Injection Demo</title>

</head>

<body bgcolor="pink" data-ng-controller="ServiceController">

<h2> Data Accessed from json using http request</h3>

<ul data-ng-repeat="tempGuest in guestList" >

<li>{{tempGuest.Name}}</li>

</ul>

</body>

<script src="Scripts/angular.js"></script>

<script type="text/javascript">

var app=angular.module("myApp",[]);

app.factory("loadGuest",function($http,$log){

return {

retrieveGuest:function(successObj){

var promise= $http({

method:'GET',

//url:'data/guests.json'

url:'https://api.mongolab.com/api/1/databases/igateangulardb/collections/employees?apiKey=Y6KaLZK8I2pEA-zgfk8oGNP6v8IVOLFm'

});

promise .success(function(data,status)

{

successObj(data);

$log.info(data,status);

}

);

promise.error(function(data,status){

$log.info(data,status);

}

) } } }

);

app.controller("ServiceController",function($scope,loadGuest){

loadGuest.retrieveGuest(function(data){

$scope.guestList=data;

}

);

}

);</script></html>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*012-ResourceService.html\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

<html ng-app="serviceApp">

<head>

<title>Service - $resource service</title>

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />

</head>

<body>

<h1>Service - $resource service</h1>

<div ng-controller="AppController">

<button ng-click="showCreate()">Add Employee</button>

<br /><br />

<table border="1" cellpadding="10" ng-show="employees.length">

<thead>

<tr>

<th>

Id

</th>

<th>

Name

</th>

<th>

Contact Number

</th>

<th>

</th>

</tr>

</thead>

<tbody>

<tr data-id="{{employee.\_id}}" ng-repeat="employee in employees">

<td>

{{employee.\_id}}

</td>

<td>

{{employee.Name}}

</td>

<td>

{{employee.ContactNumber}}

</td>

<td>

<button ng-click="editEmployee(employee)">Edit</button>

</td>

</tr>

</tbody>

</table>

<div ng-show="isEditVisible">

<hr />

<h2>Edit Employee</h2>

<form>

<label>Id:</label>

<input type="text" ng-disabled="true" ng-model="editableEmployee.\_id" />

<label>Name:</label>

<input type="text" ng-model="editableEmployee.Name"/>

<label>Contact Number:</label>

<input type="text" ng-model="editableEmployee.ContactNumber"/>

<input type="submit" value="Update" ng-click="updateEmployee(editableEmployee)" />

</form>

</div>

<div ng-show="isCreateVisible">

<hr />

<h2>Add Employee</h2>

<form>

<label>Id:</label>

<input type="text" ng-model="newEmployee.\_id" />

<label>Name:</label>

<input type="text" ng-model="newEmployee.Name" required />

<label>Contact Number:</label>

<input type="text" ng-model="newEmployee.ContactNumber" required />

<input type="submit" value="Save" ng-click="saveEmployee(newEmployee)" />

</form>

</div>

</div>

</body>

<script type="text/javascript" src="Scripts/angular.js"></script>

<script type="text/javascript" src="Scripts/angular-resource.js"></script>

<script type="text/javascript">

var app = angular.module("serviceApp",['ngResource']);

app.factory("EmployeeService", function ($resource) {

//return $resource('data/employees/:Id.json', { Id: '@Id' }, { "update": { method: "PUT" }});

return $resource('https://api.mongolab.com/api/1/databases/igateangulardb/collections/employees?apiKey=Y6KaLZK8I2pEA-zgfk8oGNP6v8IVOLFm', { \_id: '@\_id' }, { "update": { method: "POST" }}, { "delete": { method: "POST" }});

})

app.controller('AppController', function ($scope, EmployeeService) {

$scope.employees = EmployeeService.query();

var createEmployee = function (newEmployee) {

newEmployee.$save();

$scope.employees.push(newEmployee);

};

$scope.showEdit = function () {

$scope.isEditVisible = true;

$scope.isCreateVisible = false;

};

$scope.showCreate = function () {

$scope.isCreateVisible = true;

$scope.isEditVisible = false;

$scope.newEmployee = new EmployeeService();

};

$scope.saveEmployee = function (employee) {

$scope.isCreateVisible = false;

createEmployee(employee);

};

$scope.updateEmployee = function (employee) {

$scope.isEditVisible = false;

employee.$update();

};

$scope.editEmployee = function (employee) {

$scope.isEditVisible = true;

$scope.isCreateVisible = false;

$scope.editableEmployee = employee;

};

$scope.isEditVisible = false;

})

</script>

</html>