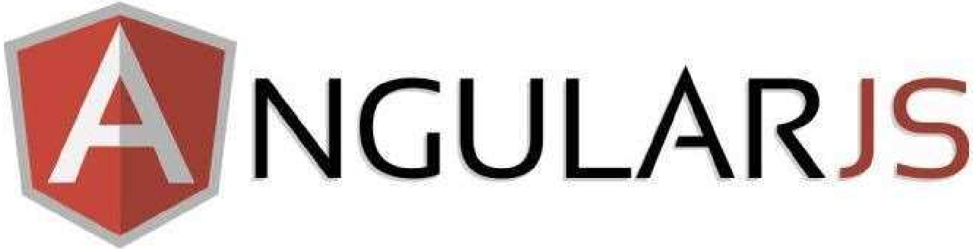
Advanced Topics
in Internet Application Programming

AngularJS SPA JumpStart

Dan Wahlin

DevReach 2013

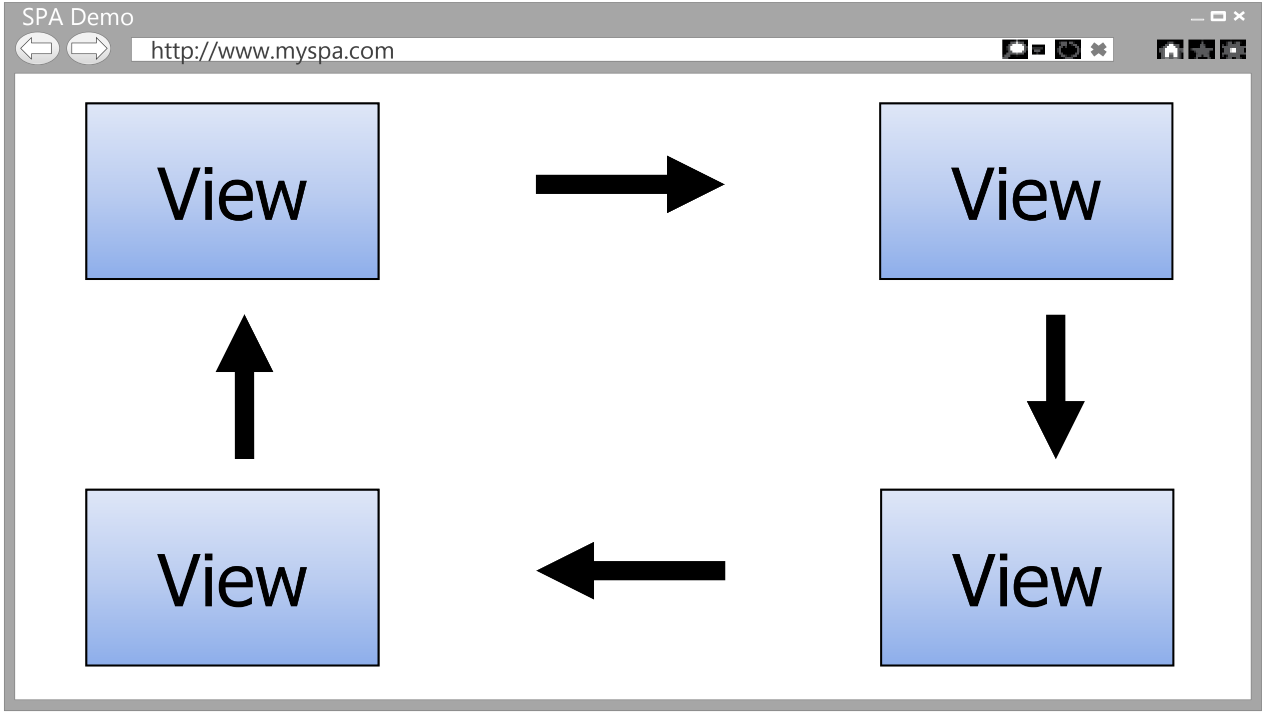


http://angularjs.org

ANGULARJS

Getting Started

Single Page Application (SPA)



The Challenge with SPAs

DOM Manipulation

Routing

Data Binding

History

Caching

Ajax/Promises

Module Loading

Object Modeling

View Loading

Data Binding

MVC

Routing

Testing

jqLite

Templates

History

Factories

AngularJS is a full-featured SPA framework

ViewModel

Controllers

Views

Directives

Controllers

Dependency Injection

Validation

ANGULARJS

Directives, Filters

and Data Binding

What are Directives?

They teach HTML new tricks!

Using Directives and Data Binding

Directive

```
<!DOCTYPE html>
<html ng-app>
<head>
    <title></title>
</head>
<body>
    <div class="container">
        Name: <input type="text" ng-model="name" /> {{ name }}
    </div>

    <script src="Scripts/angular.js"></script>
</body>
</html>
```

Directive

Data Binding Expression

Demo

NG.1.Hello

Iterating with the ng-repeat Directive

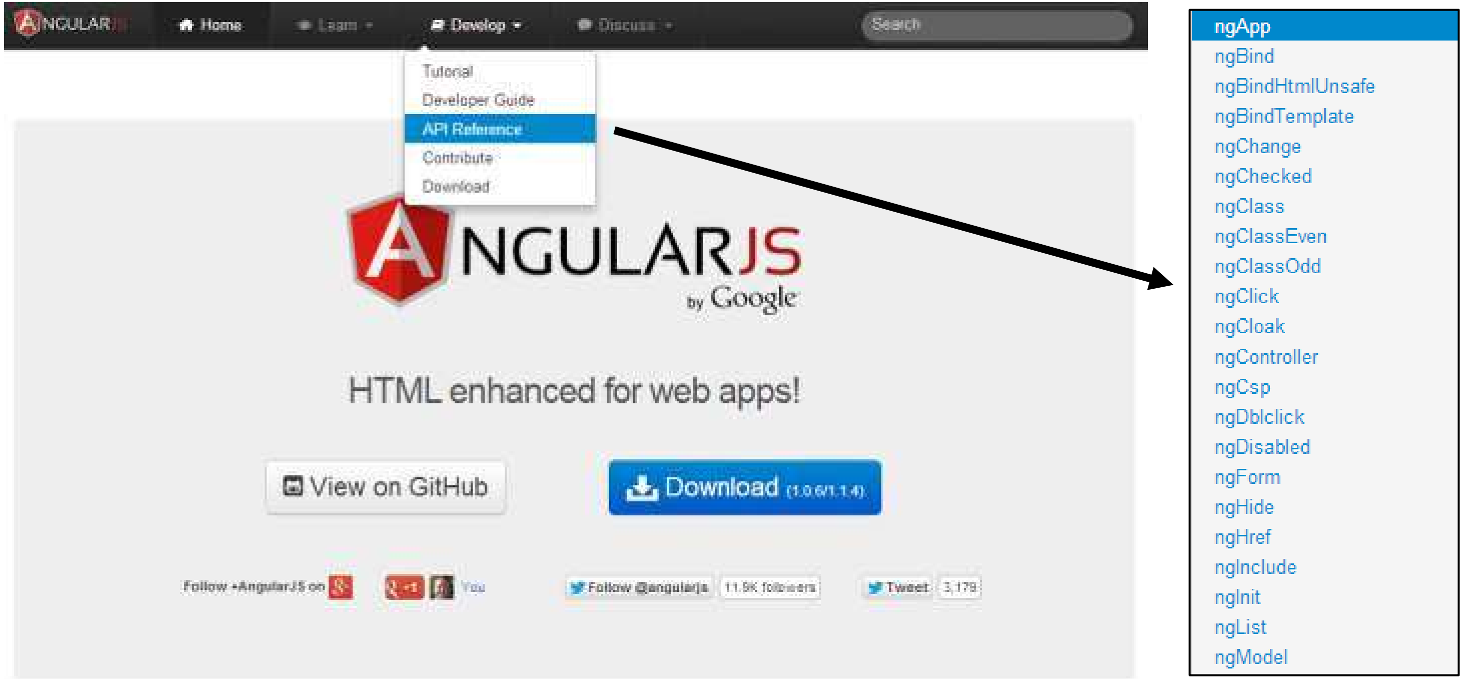
```
<html data-ng-app="">
...

   <div class="container"
      data-ng-init="names=[‘Jojo',‘Miko',‘David',‘Baruch']">

       <h3>Looping with the ng-repeat Directive</h3>
       <ul>
           <li data-ng-repeat="name in names">{{ name }}</li>
       </ul>
   </div>

...
</html>
```

Iterate through names

Demo

NG.2.Repeat

AngularJS Help for Directives



Using Filters

```
<ul>
    <li data-ng-repeat="cust in customers | orderBy:'name'">
        {{ cust.name | uppercase }}
    </li>
</ul>
```

Order customers by name property

Filter customers by model value

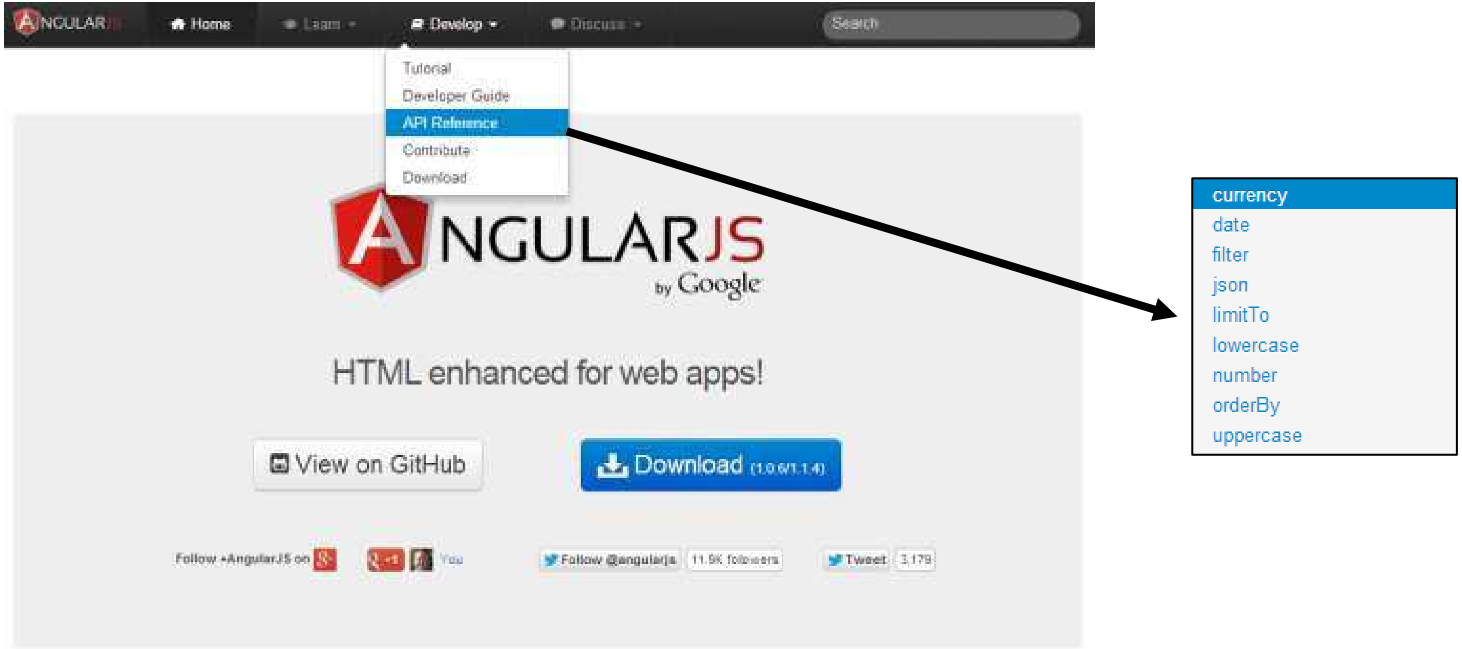
```
<input type="text" data-ng-model="nameText" />
<ul>
    <li data-ng-repeat="cust in customers | filter:nameText | orderBy:'city'">
      {{ cust.name | uppercase}} - {{ cust.city | lowercase}}</li>
</ul>
```

Uppercase 'filter'

Demo

NG.3.Filters

AngularJS Help for Filters



HTML enhanced for web apps!

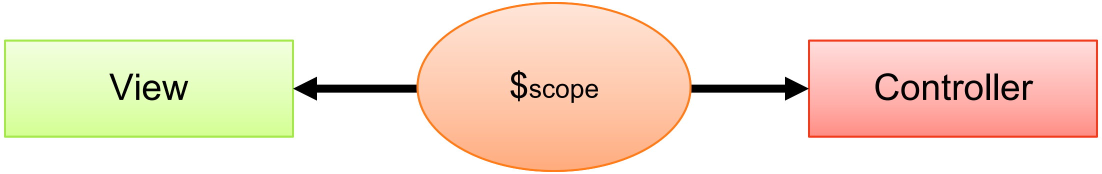
- currency
- date
- filter
- json
- limitTo
- lowercase
- number
- orderBy
- uppercase

ANGULARJS

Views, Controllers

and Scope (ViewModel)

View, Controllers and Scope



View

$scope

Controller

$scope is the "glue" (ViewModel) between a controller and a view

Creating a View and Controller

```
<div class="container" data-ng-controller="SimpleController">
    <h3>Adding a Simple Controller</h3>
    <ul>
        <li data-ng-repeat="cust in customers">
           {{ cust.name }} - {{ cust.city }}
        </li>
    </ul>
</div>

<script>
    function SimpleController($scope) {
        $scope.customers = [
           { name: 'Dave Jones', city: 'Phoenix' },
           { name: 'Jamie Riley', city: 'Atlanta' },
           { name: 'Heedy Wahlin', city: 'Chandler' },
           { name: 'Thomas Winter', city: 'Seattle' }
        ];
    }
</script>
```

Define the controller to use

Access $scope

$scope injected dynamically

Basic controller

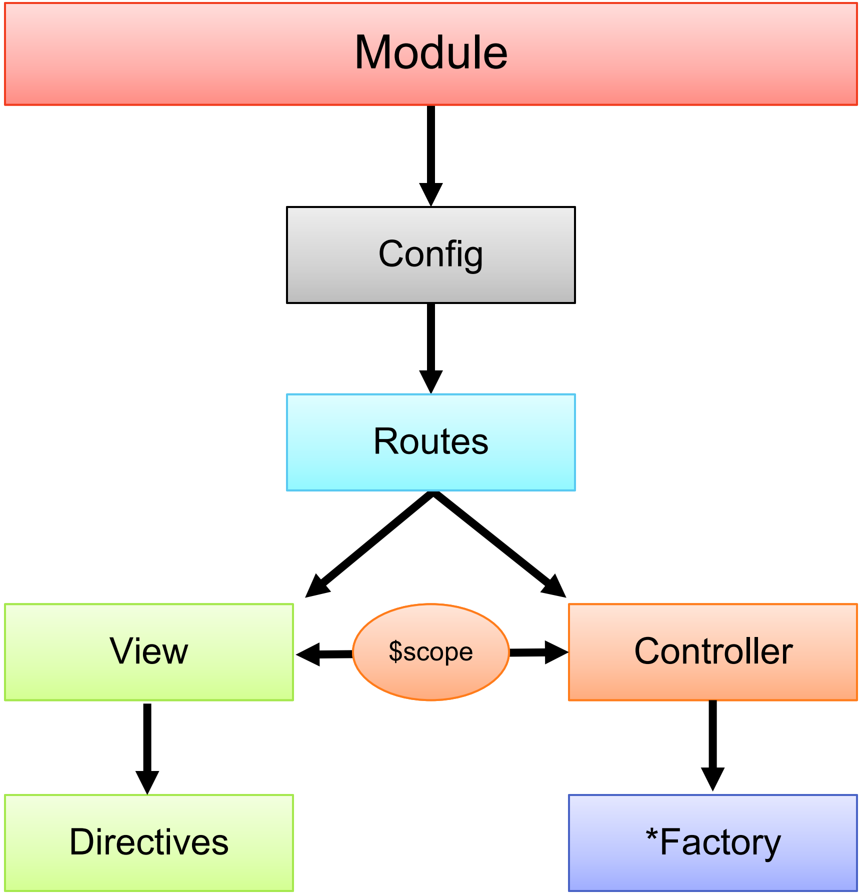
Demo

NG4.Scope

ANGULARJS

Modules, Routes

and Factories


Module

Config

Routes

View

$scope

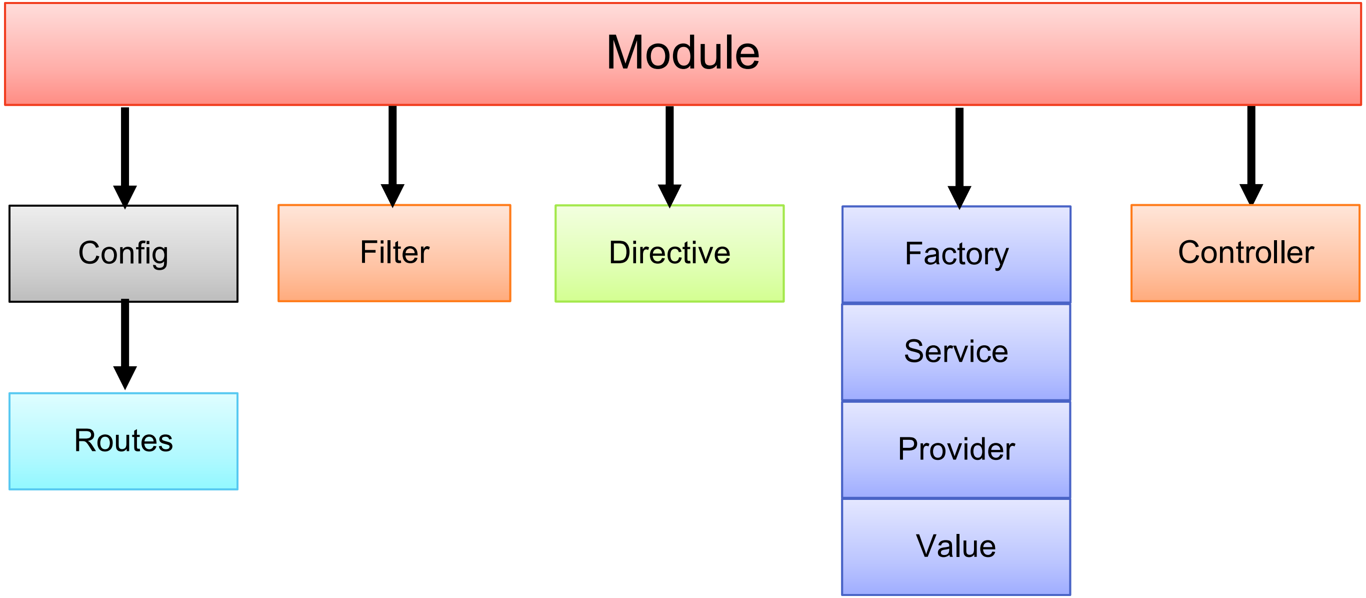
Controller

Directives

*Factory

Modules are Containers

```
<html ng-app="moduleName">
```



Creating a Module

What's the Array for?

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

```
var demoApp = angular.module('demoApp',
  ['helperModule']);
```

Module that demoApp depends on

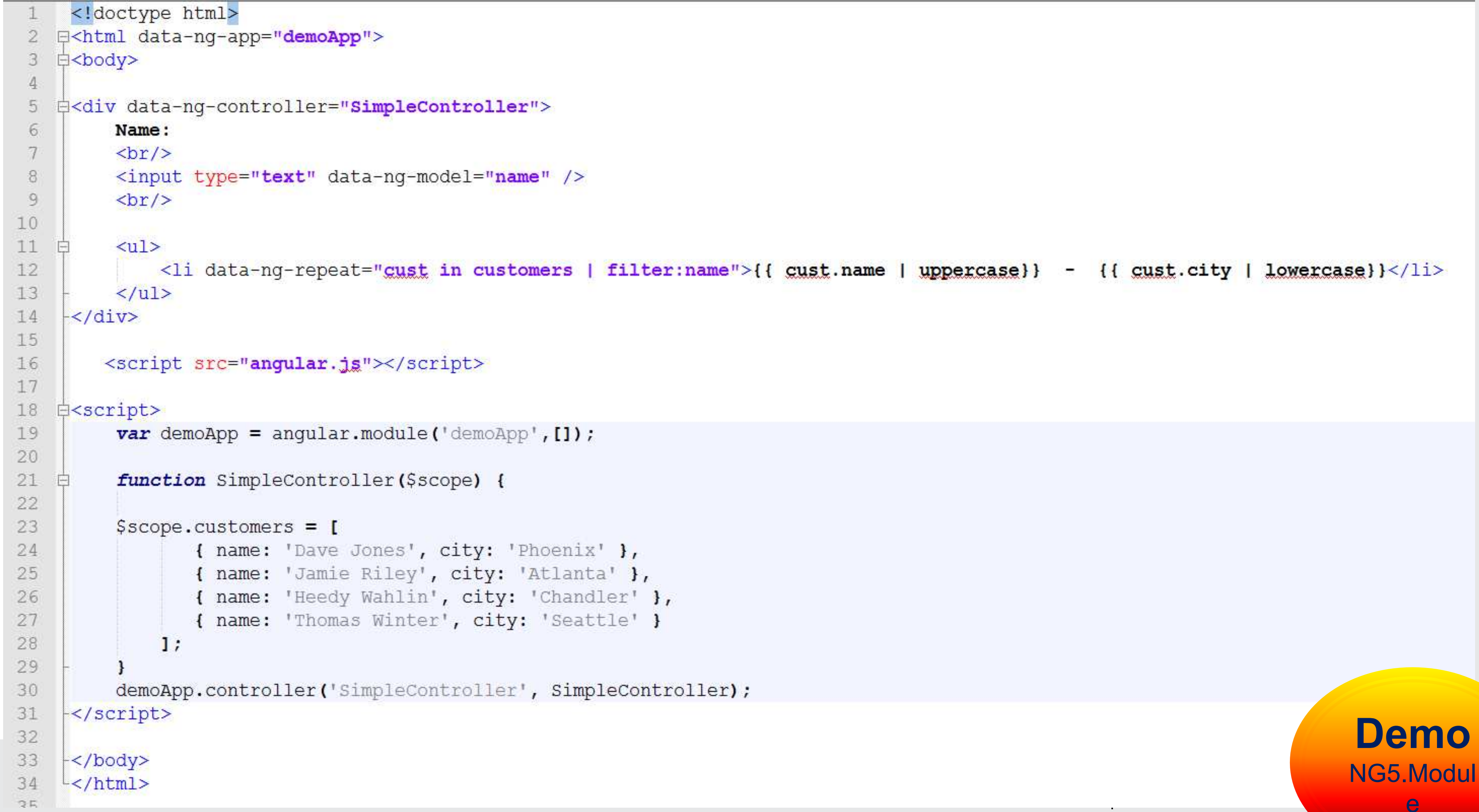
Creating a Controller in a Module

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

Define a Module

Define a Controller

```
demoApp.controller('SimpleController', function ($scope) {
    $scope.customers = [
        { name: 'Dave Jones', city: 'Phoenix' },
        { name: 'Jamie Riley', city: 'Atlanta' },
        { name: 'Heedy Wahlin', city: 'Chandler' },
        { name: 'Thomas Winter', city: 'Seattle' }
    ];

});
```

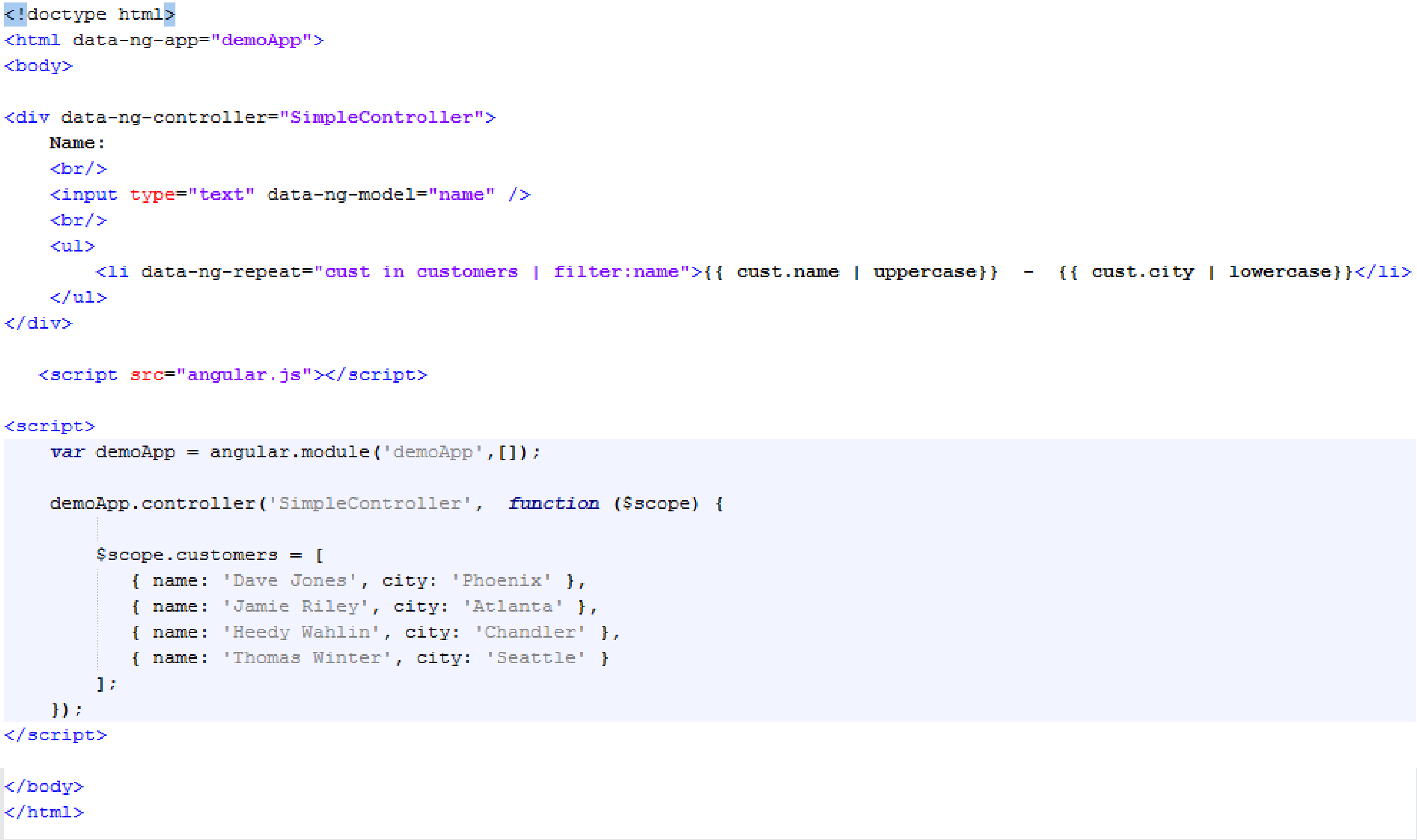


```
<!doctype html>
<html data-ng-app="demoApp">
<body>

<div data-ng-controller="SimpleController">
    Name:
    <br/>
    <input type="text" data-ng-model="name" />
    <br/>

    <ul>
        <li data-ng-repeat="cust in customers | filter:name">{{ cust.name | uppercase}}  -  {{ cust.city | lowercase}}</li>
    </ul>
</div>

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

<script>
    var demoApp = angular.module('demoApp',[]);

    function SimpleController($scope) {

    $scope.customers = [
           { name: 'Dave Jones', city: 'Phoenix' },
           { name: 'Jamie Riley', city: 'Atlanta' },
           { name: 'Heedy Wahlin', city: 'Chandler' },
           { name: 'Thomas Winter', city: 'Seattle' }
        ];
    }
    demoApp.controller('SimpleController', SimpleController);
</script>

</body>
</html>
```

Demo

NG5.Module

Anonymous solution

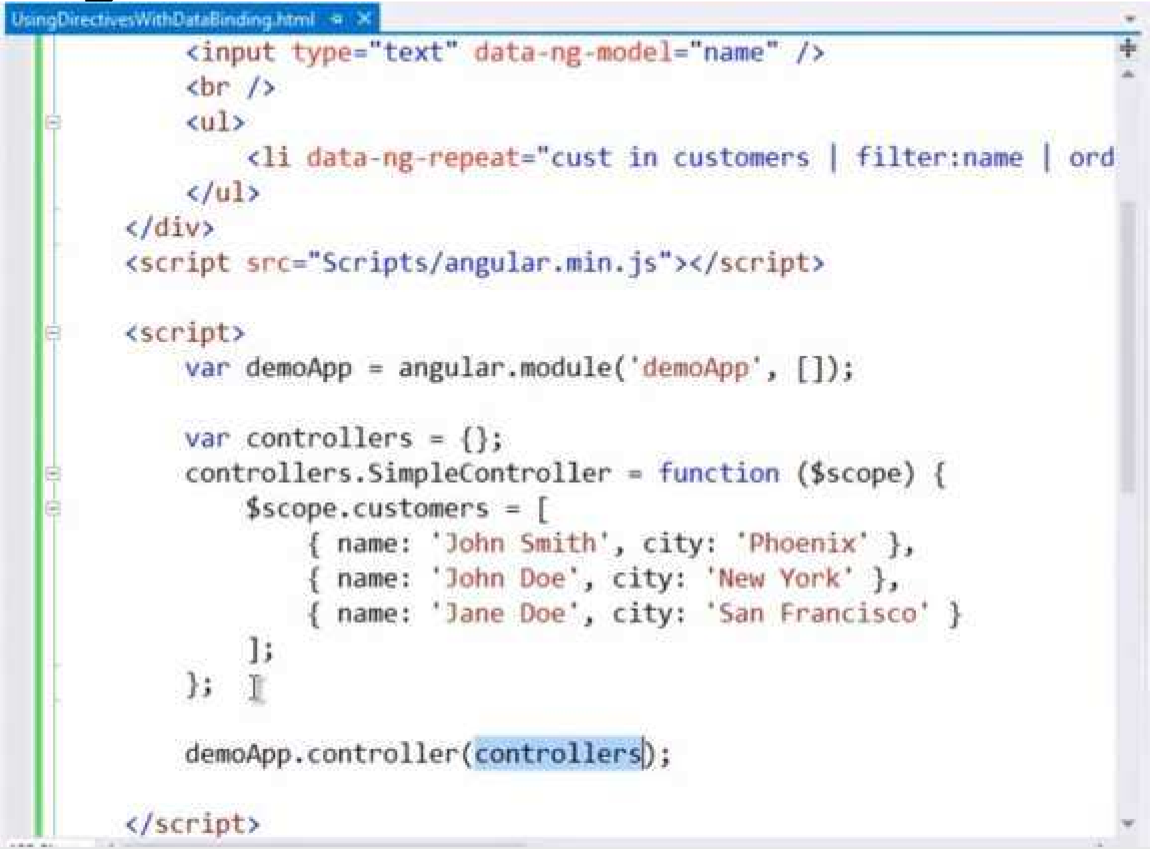


```
<!doctype html>
<html data-ng-app="demoApp">
<body>

<div data-ng-controller="SimpleController">
    Name:
    <br/>
    <input type="text" data-ng-model="name" />
    <br/>
    <ul>
        <li data-ng-repeat="cust in customers | filter:name">{{ cust.name | uppercase}}  -  {{ cust.city | lowercase}}</li>
    </ul>
</div>

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

<script>
    var demoApp = angular.module('demoApp',[]);

    demoApp.controller('SimpleController',  function ($scope) {

        $scope.customers = [
           { name: 'Dave Jones', city: 'Phoenix' },
           { name: 'Jamie Riley', city: 'Atlanta' },
           { name: 'Heedy Wahlin', city: 'Chandler' },
           { name: 'Thomas Winter', city: 'Seattle' }
        ];
    });
</script>

</body>
</html>
```

Demo

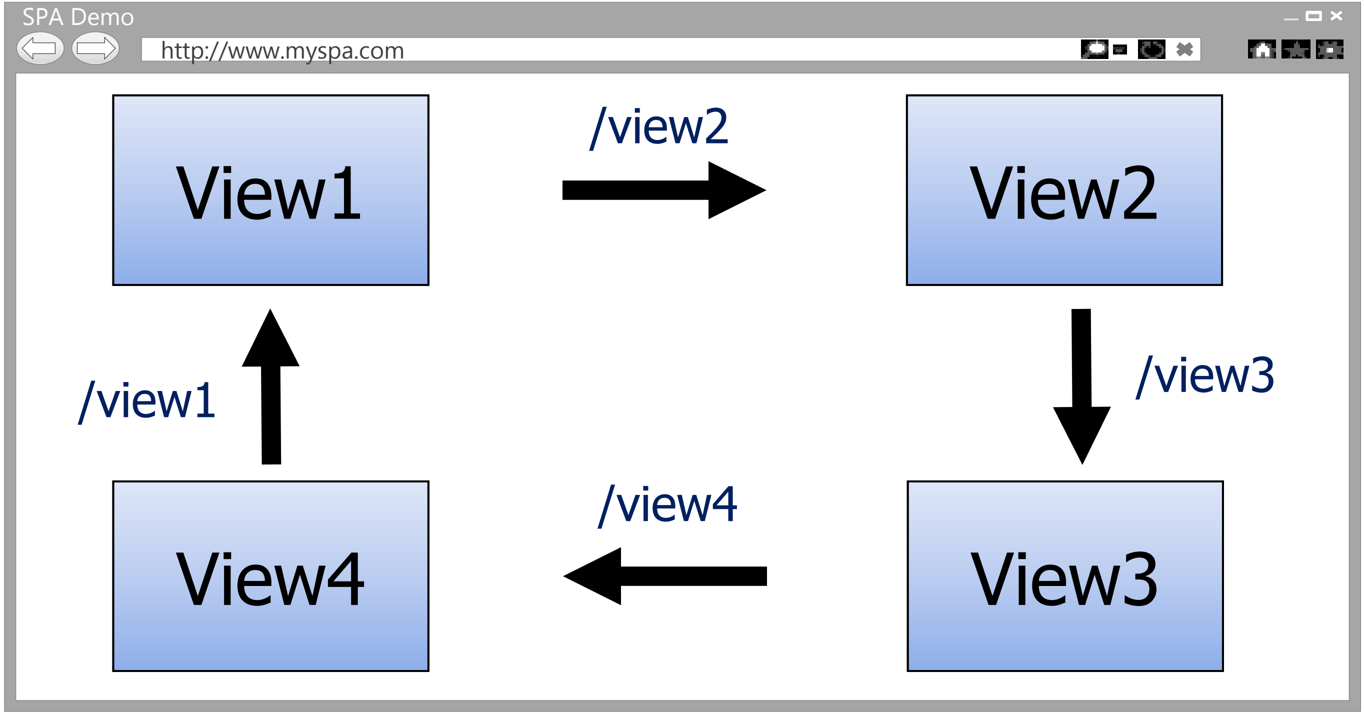
NG6.ModuleAnon

Another way…



```
            <input type="text" data-ng-model="name" />
            <br />
            <ul>
                <li data-ng-repeat="cust in customers | filter:name | ord
            </ul>
        </div>
        <script src="Scripts/angular.min.js"></script>

        <script>
            var demoApp = angular.module('demoApp', []);

            var controllers = {};
            controllers.SimpleController = function ($scope) {
                $scope.customers = [
                    { name: 'John Smith', city: 'Phoenix' },
                    { name: 'John Doe', city: 'New York' },
                    { name: 'Jane Doe', city: 'San Francisco' }
                ];
            };

            demoApp.controller(controllers);

        </script>
```

The Role of Routes (SPA)



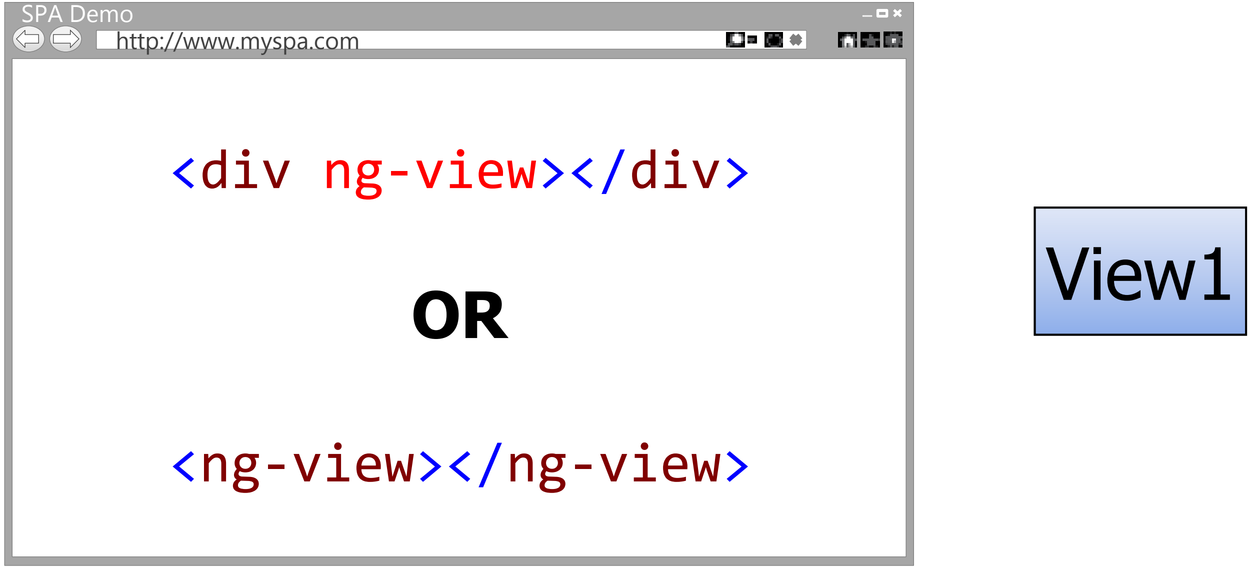
Defining Routes

```
var demoApp = angular.module('demoApp', ['ngRoute']);

demoApp.config(function ($routeProvider) {
    $routeProvider
        .when('/',
            {
                controller: 'SimpleController',
                templateUrl:'View1.html'
            })
        .when('/view2',
            {
                controller: 'SimpleController',
                templateUrl:'View2.html'
            })
        .otherwise({ redirectTo: '/' });
});
```

Define Module Routes

Where do Views Go in a Page?

Dynamically loaded views are injected into the shell page as a module loads:



```
<div ng-view></div>
```

OR

```
<ng-view></ng-view>
```

What if ?

- **We have multiple controllers that want to access the same data ???**

Controller A

Controller B

Controller C

- **If we set the data local to each controller – then we will have 3 different copies + need to synchronize when data is being modified**

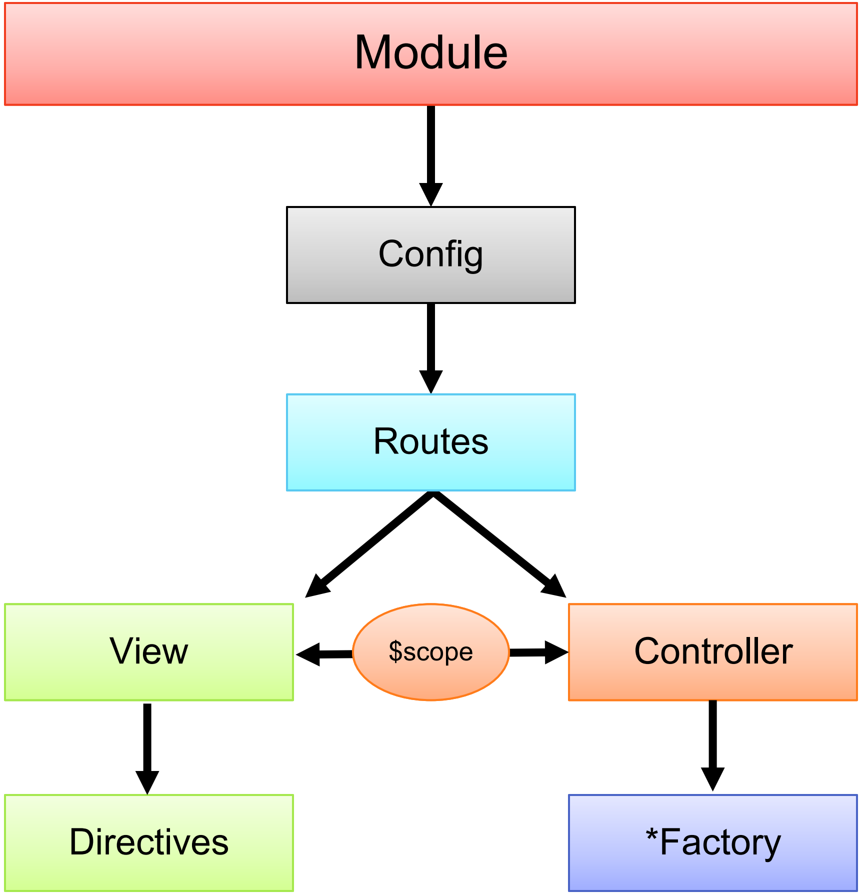
- **The solution: Factory / Service / Provider / Value**

- With the **factory** you actually create an object inside of the factory and return it.
- With the **service** you just have a standard function that uses the this keyword to define function.
- With the **provider** there's a $get you define and it can be used to get the object that returns the data.
- A **value** is just a way to get for instance a config value

The Role of Factories

```
var demoApp = angular.module('demoApp', [])
 .factory('simpleFactory', function () {
    var factory = {};
    var customers = [ ... ];
    factory.getCustomers = function () {
        return customers;
    };
    return factory;
 })
 .controller('SimpleController', function ($scope,
    simpleFactory) {
     $scope.customers = simpleFactory.getCustomers();
});
```

Factory injected into controller at runtime



Module

Config

Routes

View

$scope

Controller

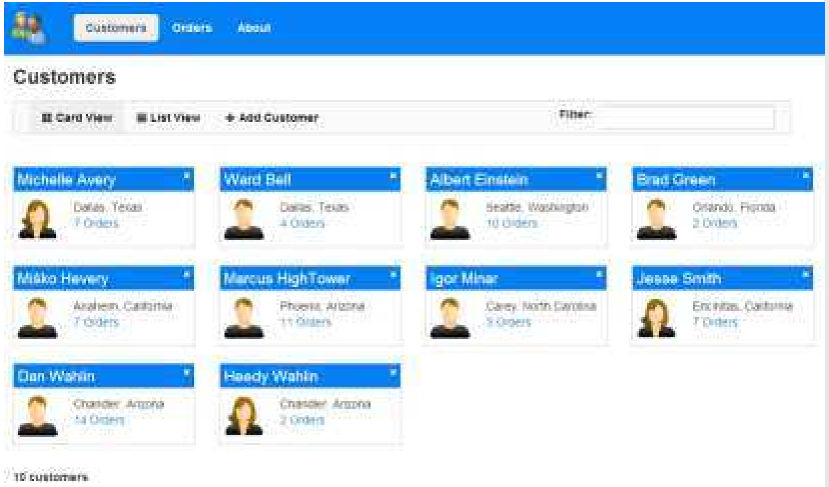
Directives

*Factory

Sample Code

http://tinyurl.com/AngularJSDemos

http://tinyurl.com/CustomerMgr



Customers

Michelle Avery
Dallas, Texas
7 Orders

Ward Bell
Dallas, Texas
4 Orders

Albert Einstein
Seattle, Washington
10 Orders

Brad Green
Orlando, Florida
2 Orders

Miško Hevery
Anaheim, California
7 Orders

Marcus HighTower
Phoenix, Arizona
11 Orders

Igor Minar
Carey, North Carolina
5 Orders

Jesse Smith
Encinitas, California
7 Orders

Dan Wahlin
Chandler, Arizona
14 Orders

Heedy Wahlin
Chandler, Arizona
2 Orders

10 customers