# AngularJS Part 1

Single Page Applications

What is AngularJS?

MVW?

"Having said, I'd rather see developers build kick-ass apps that are well-designed and follow separation of concerns, than see them waste time arguing about MV* nonsense. And for this reason, I hereby declare AngularJS to be MVW framework - Model-View-Whatever. Where Whatever stands for "whatever works for you"."

-Igor Minar
AngularJS Lead

What is it good for?
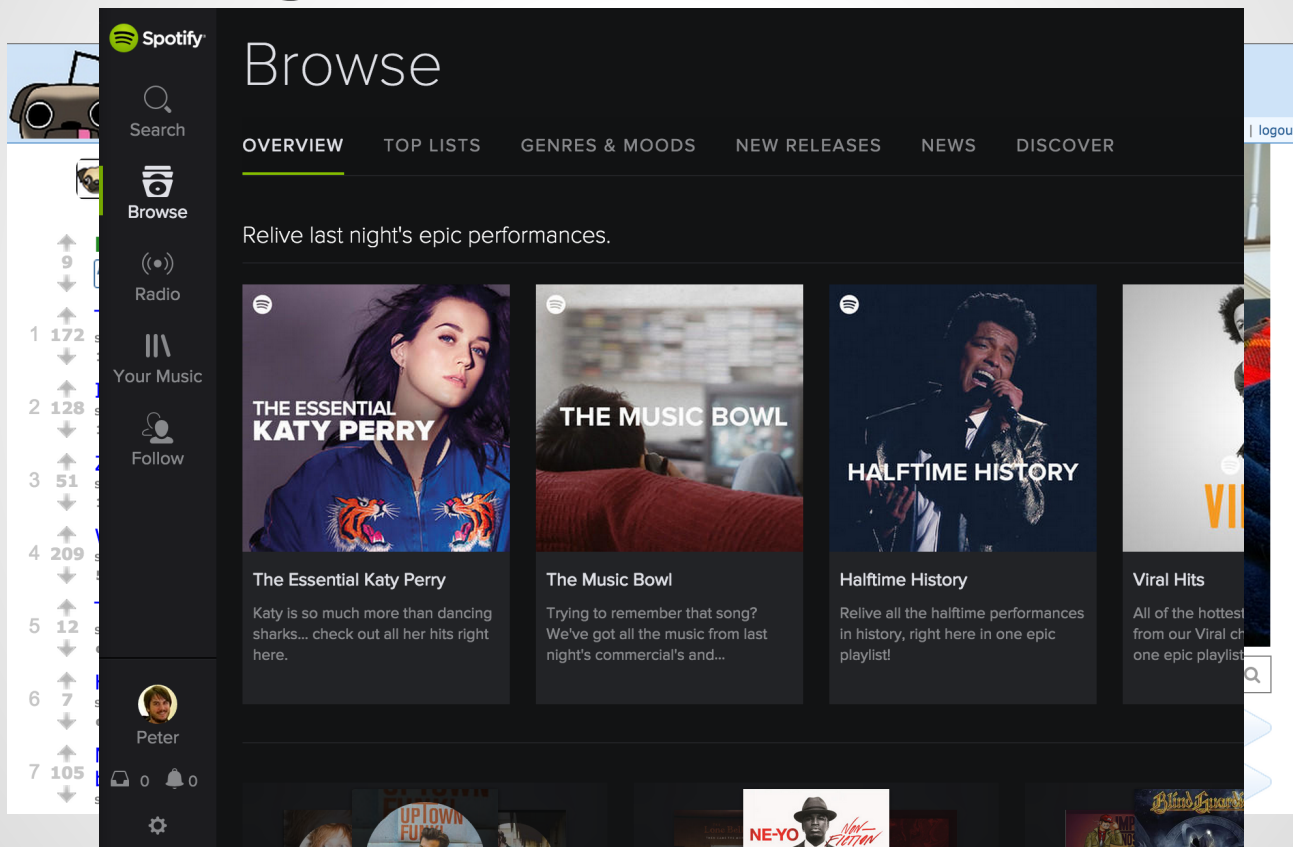
What is a SPA?
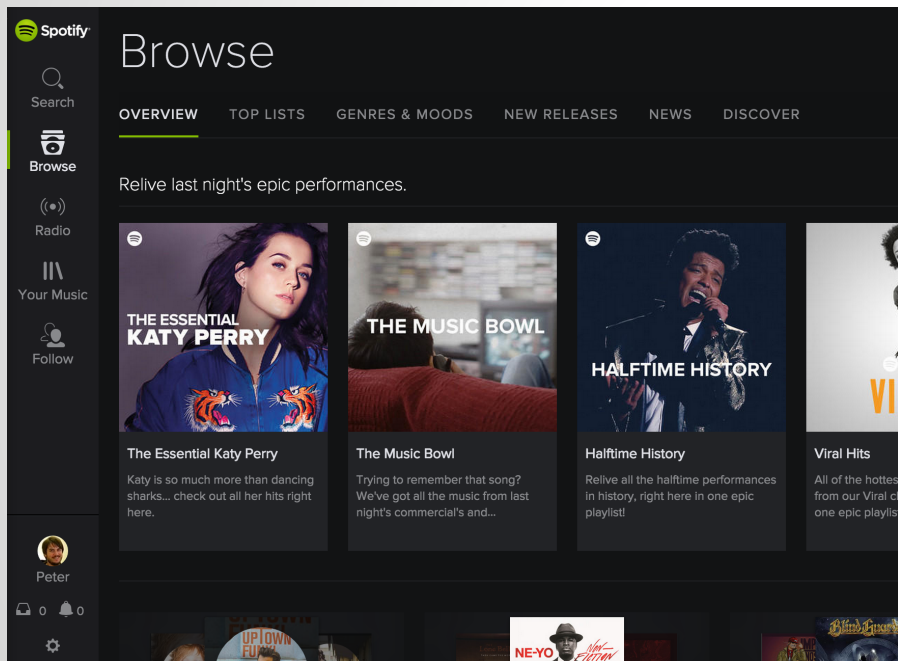
REST

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021"
  },
  "phoneNumber": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "fax",
      "number": "646 555-4567"
    }
  ],
  "gender": {
    "type": "male"
  }
}
```
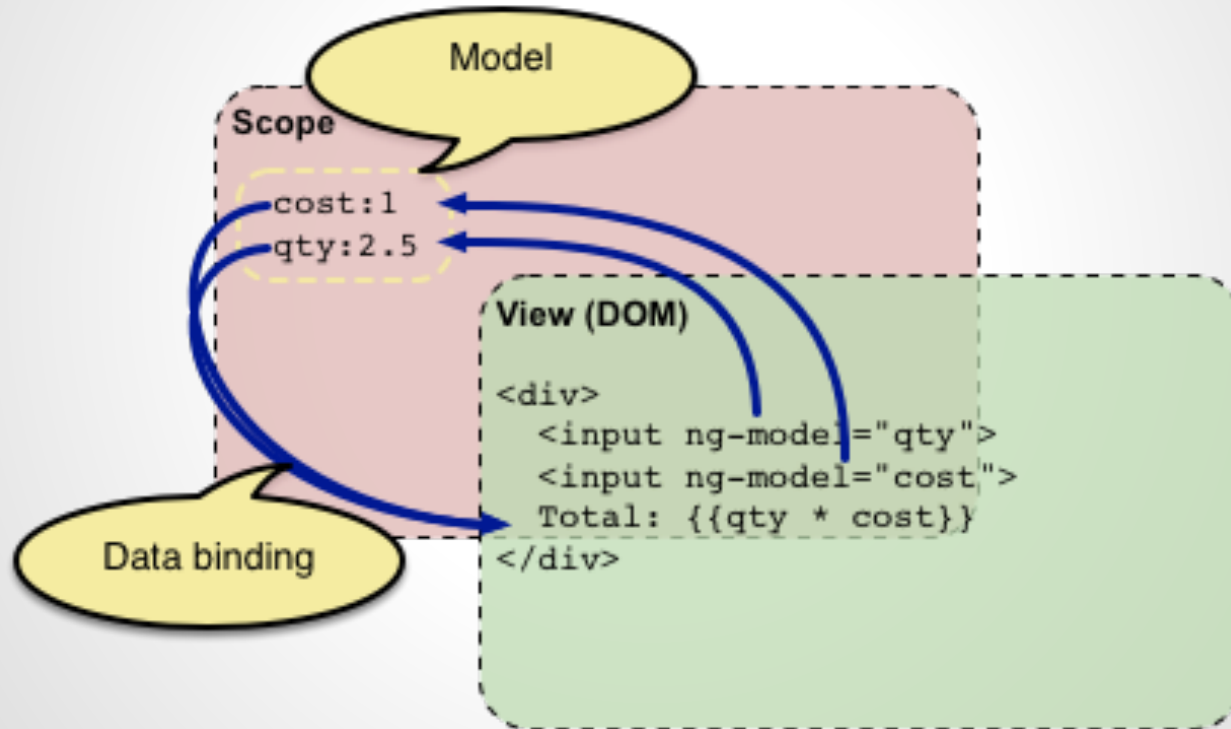
Let's get started

\<p>Hello {{name}}\</p>

\<p>Name: \<input type="text" ng-model="name">\</p>
\<p>Hello {{name}}\</p>

Name: [   ]
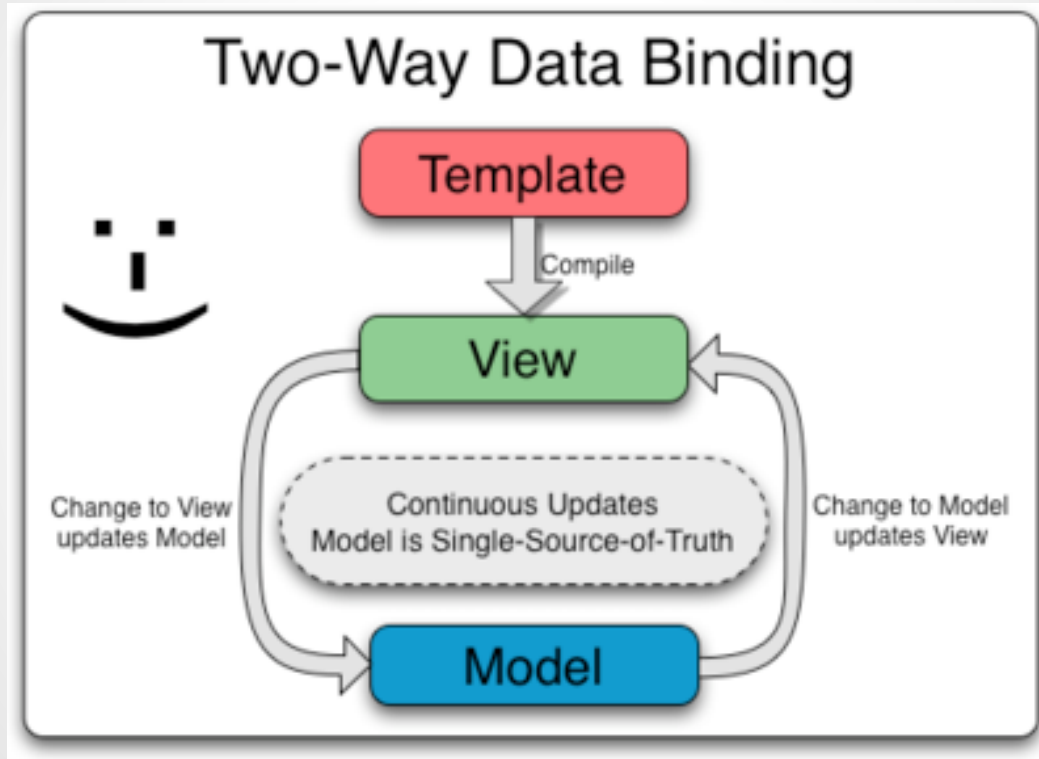
Hello

Let's get started

Let's get started

The Javascript

A Note...

- Examples have been incorporated directly into the repo.
- These examples have been commented and are available for your perusal.

The Javascript

The Javascript

```javascript
1   'use strict';
2   // Firstly, we need to wire this controller into the myApp module
3   angular.module('myApp')
4     // Here we see the name of the controller defined.
5     // Following the name, we see an anonymous function with two arguments.
6     // These arguments are functionality being injected into the controller.
7     // Think of this as being like including scripts in HTML.
8     // Unless they are referenced here, you cannot utilize them.
9     .controller('DemoController', function ($scope, DemoService, Tools) {
10      // The rest of the function defines the behavior & data that the controller will manage.
11      $scope.items = [
12        'Apples',
13        'Oranges',
14        'Bananas',
15        'Pears',
16        'Avocados'
17      ];
18      $scope.sayHello = function(name){
19          alert("Hello " + name);
20      };
21      $scope.sayHelloFromService = DemoService.sayHello;
22    });
```

The Javascript
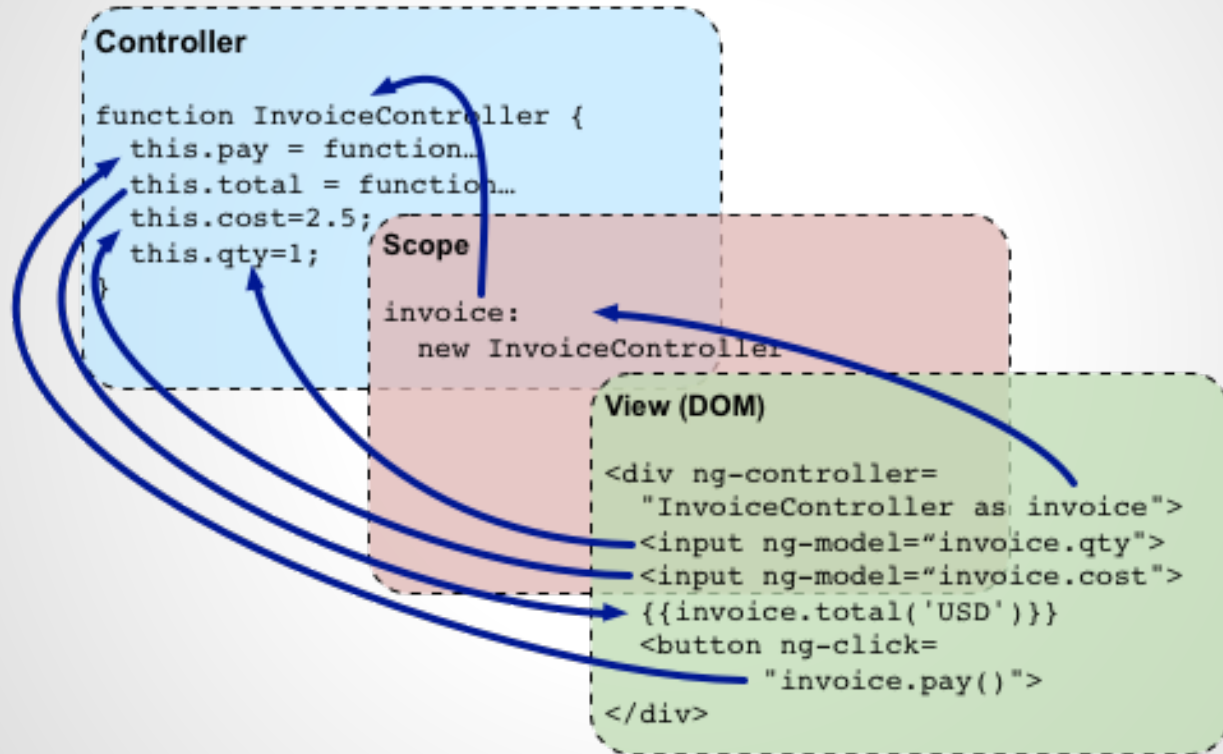
The Javascript - Services

- Does not have direct access to the scope.
- They persist for the lifecycle of the SPA.
- Business logic should live here.

The Javascript - Services
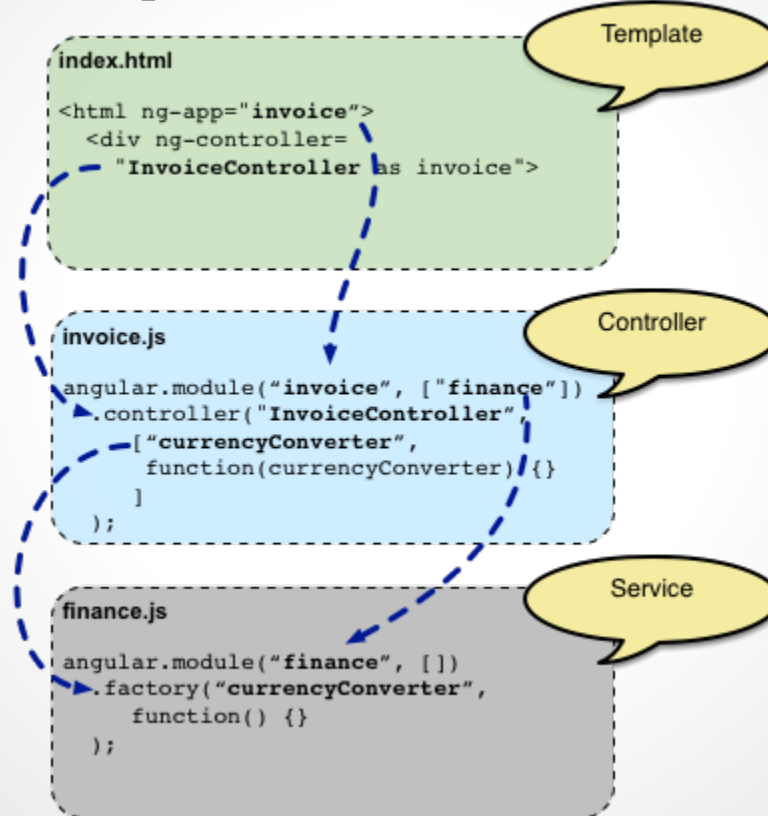
Reviewing Repo Content

```html
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>AngularJS Part 1</title>
    <link rel="stylesheet" type="text/css" href="style/style.css">
  </head>
  <!-- Declare our app on the page -->
  <body ng-app="myApp">

    <!-- Example of how to use ng-include -->
    <div ng-include="'templates/demo.html'"></div>

    <hr>

    <!-- Example of how to instantiate a controller from a template -->
    <div ng-controller="DemoController">
      <p>Name: <input type="text" ng-model="name"></p>
      <p>Hello {{name}}</p>
      <p><button ng-click="sayHello(name)">Say Hello from DemoController</button></p>
      <p><button ng-click="sayHelloFromService(name)">Say Hello from DemoService</button></p>
      <p>Here's a demo list.</p>
      <ul>
        <li ng-repeat="item in items">{{item}}</li>
      </ul>
      <!-- This changes the route, which laods another controller into the ng-view below -->
      <p><a href="#/demo">Try out some of the routes</a>.</p>
    </div>
```

Final Questions?

Best of Luck