Controllers

K. Scott Allen
odetocode.com
@OdeToCode

Controller Basics

- **Controller directive in HTML (ng-controller)**

- **Controller will be a function that Angular invokes**

- **Controller takes a $scope parameter**

- **Attach model to $scope**

```html
<div ng-app>
  <div ng-controller="MainController">


  </div>
</div>
```

```javascript
var MainController = function($scope){

  $scope.message = "Hello!";

};
```

Controller Capabilities

- **Multiple controllers**

- **Complex objects**

- **Nest controllers**

```javascript
var MainController = function($scope) {

  $scope.message = "Hello, Angular!";

};
```

```html
<body ng-controller="MainController">
  <h1>{{message}}</h1>
</body>
```

```html
<body ng-app>
  <h1>Hello Plunker!</h1>

  <div ng-controller="DateController">
    Current time is: {{ date }}
  </div>

  <div ng-controller="QuoteController">
    <div>Stock:{{stock.symbol}}</div>
    <div>Price: {{ stock.price }}</div>

    <div ng-controller="ChartController">
      <img src="{{chart.Source}}" alt="{{chart.description}}">
    </div>
  </div>
</body>
```

$http Service

- **Encapsulates HTTP communication**
  - GET, POST, PUT, DELETE

- **Can "ask" for $http inside a controller**
  - Just add as another parameter to controller function

- **Always returns a promise**
  - A promise to deliver a value in the future

```
var PersonController = function($scope, $http){

}
```
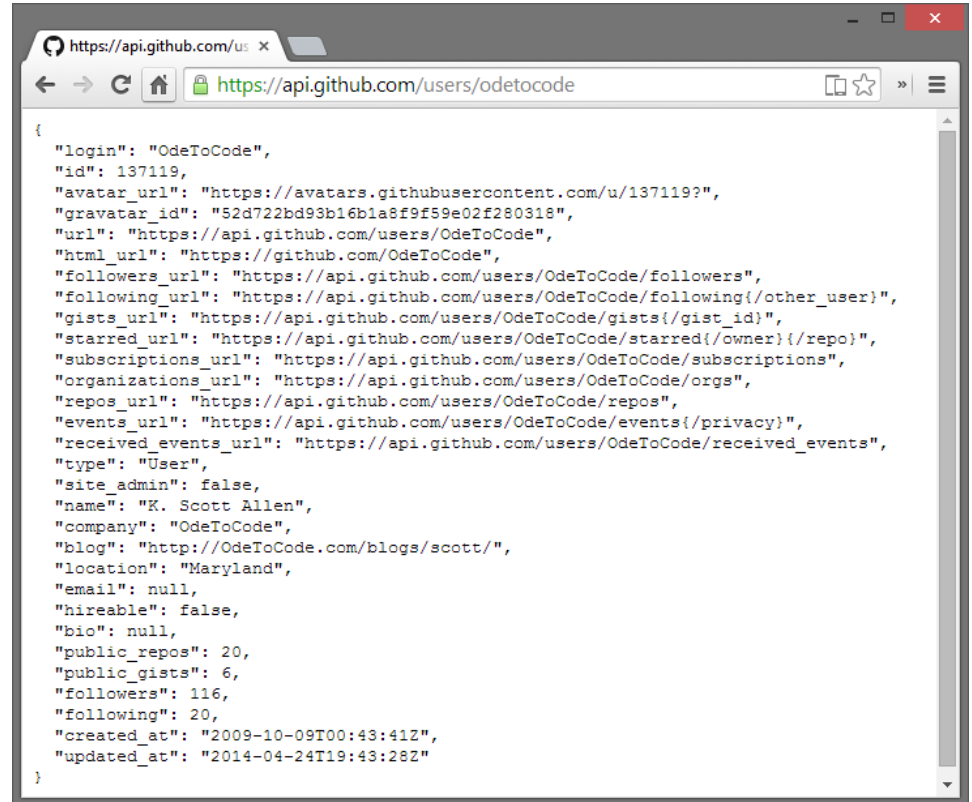
```
var PersonController = function($scope,

    $scope.user = $http.get
};
```

```
var PersonContr

    var promise = };

    promise.then(function(response){
        $scope.user = response.data;
    });

};
```

```
var PersonController = function($scope, $http) {

    $http.get("/users/1783")
        .then(function(response){
            $scope.data = response.data;
        });
```

GitHub API

- **Available from JavaScript in a browser**

- **Returns JSON (easy to convert into objects)**

- **No authentication or client key required**

Modules

- **Controllers usually live in modules**
  - Avoids the global namespace

- **Working with modules**
  - Create a module with a name
  - Register your controllers in the module
  - Tell Angular to use your module with ng-app

```javascript
var app = angular.module("githubViewer", []);
```

```javascript
var MainController = function($scope, $http){
  // ...
};

app.controller("MainController", MainController);
```

```html
<body ng-app="githubViewer">

</body>
```

Summary

- **Controllers manipulate scope**

- **Controllers can live in a module**

- **Controllers can use services like $http**

```javascript
(function() {

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

  var MainController = function($scope, $http) {

    var onUserComplete = function(response) {
      $scope.user = response.data;
    };

    var onError = function(reason) {
      $scope.error = "Could not fetch the user";
    };

    $http.get("https://api.github.com/users/robconery")
      .then(onUserComplete, onError);


    $scope.message = "Hello, Angular!";


  };

  app.controller("MainController", ["$scope", "$http", MainController

}());
```