

در [قسمت قبل](#) با نحوه پیاده سازی مسیریابی در AngularJS آشنا شدیم و در این پست میخواهیم نحوه تعریف و ارسال پارامترها به سیستم مسیریاب را فرا بگیریم.

فرض کنید که میخواهیم در لیست سفارشات قسمتی داشته باشیم برای مشاهدهی جزئیات هر سفارش. پس در صفحه نمایش جزئیات کالا نیاز به کد محصول برای واکنشی آن داریم. در Angular زمانی که داریم مسیرها را تعریف میکنیم این امکان را هم داریم که پارامترهایی را هم برای هر مسیر مشخص کنیم. برای این کار فایل app.js مثال قبل را باز کنید و مسیر ذیل را به آن اضافه کنید :

```
when('/showOrderDetails/:orderId', {
  templateUrl: 'templates/show_order.html',
  controller: 'ShowOrderController'
});
```

در بالا ما پارامتری به نام orderId وارد کرده ایم که میتوانیم توسط \$routeParams در کنترلر به آن دست پیدا کنیم :

```
myFirstRoute .controller('ShowOrderController', function($scope, $routeParams) {
  $scope.order_id = $routeParams.orderId;
});
```

فراموش نکنید که باید پارامتر \$routeParams را به کنترلر خود تزریق کنید.

محتوای فایل index.html را نیز به صورت زیر تغییر دهید :

```
<body ng-app="myFirstRoute" style="
  <div>
  <div>
  <div>
  <table dir="rtl">
  <thead>
  <tr>
  <th>#</th><th>کد</th><th>نام محصول</th><th></th>
  </tr>
  </thead>
  <tbody>
  <tr>
  <td>1</td><td>1234</td><td>15" Samsung Laptop</td>
  <td><a href="#showOrderDetails/1234">جزئیات محصول</a></td>
  </tr>
  <tr>
  <td>2</td><td>5412</td><td>2TB Seagate Hard drive</td>
  <td><a href="#showOrderDetails/5412">جزئیات محصول</a></td>
  </tr>
  <tr>
  <td>3</td><td>9874</td><td>D-link router</td>
  <td><a href="#showOrderDetails/9874">جزئیات محصول</a></td>
  </tr>
  </tbody>
  </table>

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

  <script src="js/bootstrap.js"></script>
  <script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.0.7/angular.min.js"></script>
  <script src="app.js"></script>

</body>
```

نکته‌ی مهم در کد بالا قرار دادن کد کالا بعد از مسیر است، مانند : `showOrderDetails/5412` # و محتویات فایل `templates/show_order.html` :

```
<h2>سفارش شماره<{{order_id}}</h2>
<b>#<{{order_id}}</b>.
محل قرار گیری جزئیات سفارش شماره
```

برنامه را اجرا کنید تا نتیجه را ببینید.

بارگزاری View های محلی توسط تگ <script> :

در بعضی موارد لزومی ندارد که اطلاعات View را از یک فایل دیگر بخوانید و یا حتی اینقدر View شما کوچک است که تمایل دارید آن را به همراه فایل اصلی `index.html` حمل کنید به جای اینکه آن را در یک فایل جدا نگهداری کنید. دایرکتیوی به نام `ng-template` وجود دارد که این امکان را به ما میدهد تا بتوانیم `template` View های کوچکی را در داخل فایل اصلی قرار دهیم. با استفاده از تگ `<script>` به شکل زیر میشود این کار را انجام داد :

```
<script type="text/ng-template" id="add_order.html">
  <h2>ثبت سفارش</h2>
  <{{message}}>
</script>
```

برای درک بهتر مثالی را تهیه میکنیم .

فایل `app.js` مثال قبل را باز کنید و مسیرهای زیر را نیز به آن اضافه کنید :

```
when('/AddNewOrder', {
  templateUrl: 'add_order.html',
  controller: 'AddOrderController'
}).
when('/ShowOrders', {
  templateUrl: 'show_orders.html',
  controller: 'ShowOrdersController'
});
```

سپس دو کنترلر زیر را نیز به آن اضافه کنید :

```
myFirstRoute.controller('AddOrderController', function($scope) {
  $scope.message = 'صفحه نمایش ثبت سفارش جدید';
});

myFirstRoute.controller('ShowOrdersController', function($scope) {
  $scope.message = 'صفحه نمایش لیست سفارشات';
});
```

فایلی به نام `index2.html` برای صفحه اصلی برنامه با محتوای زیر تعریف کنید :

```
<body ng-app="myFirstRoute" style="
  <div>
    <div>
      <div>
        <ul>
          <li><a href="#AddNewOrder">ثبت سفارش جدید</a></li>
          <li><a href="#ShowOrders">نمایش سفارشات</a></li>
        </ul>
      </div>
      <div>
        <div ng-view></div>
      </div>
    </div>
  </div>

  <script type="text/ng-template" id="add_order.html">
    <h2>ثبت سفارش</h2>
    <{{message}}>
```

```

</script>

<script type="text/ng-template" id="show_orders.html">
    <h2> نمایش سفارشات </h2>
    {{message}}
</script>

<script src="js/bootstrap.js"></script>
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.0.7/angular.min.js"></script>
<script src="app.js"></script>

</body>

```

همانطور که مشاهده میکنید در کد بالا از 2 تگ اسکریپت برای قرار دادن محتوای View استفاده کرده ایم که خاصیت type آن برابر با text/ng-template و خاصیت id آن نام View template است و دیگر فایل مجزایی برای Viewها ایجاد نکردیم. Angular به صورت خودکار محتوای داخل تگهای Script را به محض فراخوانی آدرسهای موجود در ویژگی id هر تگ به وسیلهی سیستم مسیر یابی، در داخل دایرکتیو ng-view قرار میدهد. پروژه را اجرا کنید تا نتیجه را مشاهده کنید.

افزودن دادههای سفارشی به سیستم مسیریابی :

بیشتر اوقات ممکن است نیاز داشته باشید تا دادههای خاصی را در مسیرهای معینی ارسال کنید. برای مثال ممکن است شما بخواهید از یک کنترلر در مسیرهای مختلف استفاده کنید و برای هر مسیر یک دادهی خاص را نیز ارسال میکنید. به مثال زیر توجه کنید :

```

when('/AddNewOrder', {
    templateUrl: 'templates/add_order.html',
    controller: 'CommonController',
    foodata: 'addorder'
}).
when('/ShowOrders', {
    templateUrl: 'templates/show_orders.html',
    controller: 'CommonController',
    foodata: 'showorders'
});

sampleApp.controller('CommonController', function($scope, $route) {
    //access the foodata property using $route.current
    var foo = $route.current.foodata;

    alert(foo);
});

```

در هر دو مسیر از کنترلر CommonController استفاده کرده ایم با این تفاوت که در مسیر اول یعنی /AddNewOrder یک خاصیت با نام foodata با مقدار addorder تعریف شده است و در مسیر دوم با مقدار showorder. ما میتوانیم با تزریق \$route به کنترلرمان، توسط دستور :

```
$route.current.foodata
```

مقدار موجود در آن را بخوانیم.

نظرات خوانندگان

نویسنده:

علی رضایی

تاریخ:

۲۰:۱۲ ۱۳۹۳/۰۱/۰۵

با سلام و تشکر فراوان جهت اشتراک دانسته هایتان؛

یک سوال:

در صورتی که بخواهیم بخش پایانی مقاله یعنی « افزودن داده‌های سفارشی به سیستم مسیریابی » را پیاده کنیم، همه چیز درست کار میکند، اما اگر از controllerAs استفاده کنیم دیگر route.current\$ در دستر نیست و undefiend میشود. شما به این مشکل برخوردید؟

ممنون

نویسنده:

ناصر طاهری

تاریخ:

۰:۳ ۱۳۹۳/۰۱/۰۶

مشکلی رخ نداد.

1 - ماژول مسیریابی (ngRoute) رو باید تزریق کنید به ماژول اصلی :

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

2- \$route رو به کنترلر هم تزریق کنید :

```
myFirstRoute.controller('ShowPage1Controller', function ($route) {
  this.message = 'Content of page-one.html';
  //access the foodata property using $route.current
  var foo = $route.current.foodata;
  alert(foo);
});
```

3- این هم یک نمونه از تنظیمات مسیریابی :

```
myFirstRoute.config(['$routeProvider',
  function($routeProvider) {
    $routeProvider.
      when('/pageOne', {
        templateUrl: 'templates/page_one.html',
        controller: 'ShowPage1Controller',
        controllerAs: 'tCtrlOne',
        foodata: 'valueOne'
      }).
      when('/pageTwo', {
        templateUrl: 'templates/page_two.html',
        controller: 'ShowPage2Controller',
        controllerAs: 'tCtrlTwo',
        foodata: 'valueTwo'
      }).
      otherwise({
        redirectTo: '/pageOne'
      });
  }]);
```

البته اینجا فقط برای تصحیح اشتباه است. وگر نه به ControllerAs ارتباطی ندارد. به احتمال زیاد مشکل شما عدم تزریق \$route به کنترلر بوده.