

در [پست‌های قبلی](#) با مفهوم ng-app آشنا شدید. دایرکتیو ng-app برای استفاده از راه انداز خودکار فریم ورک Angular (معروف به auto-bootstrap) استفاده می‌شود. در حالت پیش فرض، به ازای هر سند Html فقط می‌توان یک ماژول در Angular تعریف کرد. در سند مربوطه اولین المانی که دارای دایرکتیو ng-app باشد به عنوان عنصر ریشه در نظر گرفته می‌شود و تمام عناصر تعریف شده در محدوده این دایرکتیو قابل استفاده برای ماژول مورد نظر خواهد بود. سایر عناصر حتی اگر ng-app یکسان داشته باشند نادیده گرفته می‌شوند.

ابتدا یک مثال زیر را به روش auto-bootstrap بررسی می‌کنیم:

```
<div ng-app="myApp">
  <div ng-controller="myController as ctrl">
    <span>ng-app #1</span> {{ctrl.firstName}} {{ctrl.lastName}}
  </div>
</div>

<div ng-app="myApp">
  <div ng-controller="myController as ctrl">
    <span>ng-app #2</span> {{ctrl.firstName}} {{ctrl.lastName}}
  </div>
</div>

@section scripts
{
  <script type="text/javascript" src="~/scripts/Modules/module8.js"></script>
}
```

در کنترلر مورد نظر نیز تعاریف به صورت زیر خواهد بود:

```
var app = angular.module('myApp', []);
app.controller('myController', function ()
{
  this.firstName = "Masoud";
  this.lastName = "Pakdel";
});
```

در مثال بالا دو تگ div وجود دارد که به صورت مشترک با استفاده از دایرکتیو ng-app به یک ماژول اشاره می‌کنند. طبق گفته‌ها بالا در روش auto-bootstrap اولین عنصری که دارای دایرکتیو ng-app باشد به عنوان محدوده ماژول مورد استفاده قرار خواهد گرفت در نتیجه سایر المان‌ها (در اینجا منظور تگ div دوم است) نادیده گرفته خواهند شد. پس خروجی به صورت زیر می‌شود:



اما اگر قصد داشته باشیم که در یک سند html دو نقطه شروع تعریف کنیم در حالی که هر کدام از یک منبع داده استفاده نمایند باید bootstrap برنامه را به صورت دستی تعیین کرد. برای این کار کافیه از دستور angular.bootstrap به صورت زیر استفاده نماییم:

پیاده سازی مثال بالا

```
<div id="myAppContainer1">
  <div ng-controller="myController as ctrl">
    <span>ng-app #1</span> {{ctrl.firstName}} {{ctrl.lastName}}
  </div>
</div>

<div id="myAppContainer2">
  <div ng-controller="myController as ctrl">
    <span>ng-app #2</span> {{ctrl.firstName}} {{ctrl.lastName}}
  </div>
</div>

@section scripts
{
  <script type="text/javascript" src="~/scripts/Modules/module8.js"></script>
}
```

اولین تغییر مورد نظر این است که، دایرکتیو ng-app حذف شد و به جای آن id برای تگ div تعیین کردیم. در فایل کنترلر مورد نظر نیز تغییر زیر را اعمال می‌کنیم:

```
var app = angular.module('myApp', []);
app.controller('myController', function ()
{
  this.firstName = "Masoud";
  this.lastName = "Pakdel";
});
angular.bootstrap(document.getElementById("myAppContainer1"), ["myApp"]);
angular.bootstrap(document.getElementById("myAppContainer2"), ["myApp"]);
```

با استفاده از دستور angular.bootstrap می‌توان بر اساس id تعیین شده تگ مورد نظر در سند را به دست آورد و ماژول مورد نظر را به آن نسبت داد.

خروجی مثال بالا:



برخلاف حالت قبل هر دو نقطه شروع به یک منبع داده اشاره می‌کنند و محدودیت حالت قبل برطرف می‌شود.

نظرات خوانندگان

نویسنده: ابوالفضل رجب پور
تاریخ: ۱۹:۴ ۱۳۹۲/۱۰/۱۷

سلام و تشکر
بنظرم جای نمونه برنامه‌های کوچکی که گام به گام توضیح داده بشن خالیه. خیلی خوب و کاربردی مفهوم جا می‌افته. مثلاً نمونه‌ی معروف todo

نویسنده: مسعود پاکدل
تاریخ: ۲۳:۳ ۱۳۹۲/۱۰/۱۷

مثال بالا نمونه ای از یک برنامه کوچک است. بهتر است هر نکته و سرنخ در طی یک پست ارائه شود به جای ذکر تاریخچه، هستی و چیستی مسائل در طی یک پست.