

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

## AngularJS در Modules

مدرس : مهندس افشین رفوآ

دوره آموزش SPA

## AngularJS در Modules

در **AngularJS**، ماژول یک برنامه را تعریف می کند.

ماژول در حقیقت نقش یک ظرف برای نگهداری بخش های مختلف یک برنامه را ایفا می کند.

همچنین می توان گفت که ماژول، یک ظرف برای کنترلگرهای برنامه است.

کنترلگرها همیشه متعلق به یک ماژول هستند.

### یک ماژول به همراه یک کنترلگر

برنامه ی زیر ("**myApp**") دارای یک کنترلگر به نام ("**myCtrl**") است:

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <script src="Angular.js"></script>
</head>
<body>
  <div ng-app="myApp" ng-controller="myCtrl">
    {{ firstName + " " + lastName }}
  </div>
  <script>
    var app = angular.module("myApp", []);
    app.controller("myCtrl", function ($scope) {
      $scope.firstName = "John";
      $scope.lastName = "Doe";
    });
  </script>
</body>
</html>
```

## قرار دادن ماژول ها و کنترلگرها در فایل های جدا

در برنامه های **AngularJS**، متداول است که ماژول ها و کنترلگرها را در فایل های مجزای جاوا اسکریپت قرار می دهند.

در مثال زیر، فایل **"myApp.js"** دربردارنده ی یک ماژول برنامه است درحالی که فایل **"myCtrl.js"** حامل کنترلگر می باشد:

از پارامتر **[]** در تعریف ماژول می توان به منظور تعریف ماژول های وابسته بهره گرفت.

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <script src="Angular.js"></script>
</head>
<body>
  <div ng-app="myApp" ng-controller="myCtrl">
    {{ firstName + " " + lastName }}
  </div>
  <script src="myApp.js"></script>
  <script src="myCtrl.js"></script>
</body>
</html>
```

## توابع می توانند فضاهای نامی سراسری را تخریب کنند

در جاوا اسکریپت باید از بکار بردن توابع سراسری تا حد امکان خودداری کرد، زیرا آنها می توانند به راحتی توسط دیگر اسکریپت ها بازنویسی یا نابود شوند.

ماژول های **AngularJS**، با قرار دادن توابع به صورت محلی درون ماژول مربوطه، این مشکل را تا حدی کاهش دهد.

## زمان بارگذاری کتابخانه

اگرچه رایج است که در برنامه های **HTML**، اسکریپت ها در انتهای عنصر **<body>** قرار داده شوند، پیشنهاد می کنیم که شما فایل آدرس دهنده به کتابخانه ی **AngularJS** را در تگ **<head>** و یا در ابتدای المان **<body>** قرار دهید.

دلیل آن این است که فراخوانی **angular.module** تنها می تواند پس از بارگذاری کتابخانه، ترجمه (کامپایل) شود.

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <script src="Angular.js"></script>
</head>
<body>
  <div ng-app="myApp" ng-controller="myCtrl">
    {{ firstName + " " + lastName }}
  </div>
  <script>
    var app = angular.module("myApp", []);
    app.controller("myCtrl", function ($scope) {
      $scope.firstName = "John";
      $scope.lastName = "Doe";
    });
  </script>
</body>
</html>
```