

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

Directive ها در AngularJS

مدرس : مهندس افشین رفوآ

دوره آموزش SPA

Directive ها در AngularJS

Angular یک چارچوب کاری یا کتابخانه ی جاوا اسکریپت هست که به کمک خصیصه های نوینی به نام **Directive** ها (دستور ها) ، به برنامه نویسی یا طراح وب این اجازه را می دهد که **HTML** را بسط داده و قابلیت های آن را گسترش دهد.

Angular Directives

Directive ها با استفاده از پیشوند **ng** ، خصیصه های **HTML** را بسط می دهند.
دستور **ng-app** یک برنامه ی **AngularJS** را مقدار دهی اولیه (آغاز و تعریف) می کند.
دستور **ng-init** داده های برنامه (**app data**) را مقدار دهی اولیه می کند.
دستور **ng-model** مقدار کنترل های دریافت ورودی **HTML** از قبیل **input** و **select** و **textarea** را به داده های برنامه مقید **bind** می کنند.

```
<!DOCTYPE html>
<html>
<head>
<title></title>
<script src="Angular.js"></script>
</head>
<body>
<div ng-app="" ng-init="firstName='John'">
<p>Input something in the input box:</p>
<p>Name: <input type="text" ng-model="firstName"></p>
```

```
<p>You wrote: {{ firstName }}</p>
</div>
</body>
</html>
```

دستور **ng-app** همچنین به **AngularJS** اطلاع می دهد که المان **<div>** صاحب برنامه ی **AngularJS** می باشد.

اتصال داده (Data Binding)

عبارت **{{ firstName }}** که در مثال بالا مشاهده کردید، یک عبارت اتصال داده **AngularJS** است.

اتصال داده یا **data binding** عبارت های **AngularJS** را با داده های **AngularJS** هماهنگ و منطبق (**synchronize**) می سازد.

به طور مثال، **{{ firstName }}** با **ng-model="firstName"** برابر می باشد.

در مثال زیر مشاهده می کنید که دو فیلد دریافت متن (**text field**) با دستور **ng-model** مساوی و برابر شده اند:

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <script src="Angular.js"></script>
</head>
<body>
  <div data-ng-app="" data-ng-init="quantity=1;price=5">
    <h2>Cost Calculator</h2>
    Quantity: <input type="number" ng-model="quantity">
    Price: <input type="number" ng-model="price">
    <p><b>Total in dollar:</b> {{quantity * price}}</p>
  </div>
</body>
</html>
```

همان طور که در مبحث پیشین مطرح شد، استفاده از دستور **ng-init** برای مقداردهی زیاد توصیه نمی شود. در مباحث آینده با روش بهتری برای مقدار دهی اولیه ی آشنا خواهیم شد.

تکرار کردن المان های HTML

دستور **ng-repeat** به منظور تکرار کردن المان **HTML** بکار رفته و عملکردی مشابه حلقه را دارد:

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <script src="Angular.js"></script>
</head>
<body>
  <div data-ng-app="" data-ng-init="names=['Jani','Hege','Kai']">
    <p>Looping with ng-repeat:</p>
    <ul>
      <li data-ng-repeat="x in names">
        {{ x }}
      </li>
    </ul>
  </div>
</body>
</html>
```

در زیر از دستور **ng-repeat** بر روی یک آرایه از اشیا استفاده شده است:

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <script src="Angular.js"></script>
</head>
<body>
  <div ng-app="" ng-init="names=[
{name:'Jani',country:'Norway'},
{name:'Hege',country:'Sweden'},
{name:'Kai',country:'Denmark'}]">
    <p>Looping with objects:</p>
    <ul>
      <li ng-repeat="x in names">
        {{ x.name + ', ' + x.country }}
      </li>
    </ul>
  </div>
</body>
</html>
```

AngularJS برای آن دسته از برنامه های کاربردی که نیاز به پیاده کردن عملیات **Update**، **Create**

Delete، **Read** (ایجاد، خواندن، آپدیت، حذف) بر روی پایگاه داده دارند بسیار مناسب می باشد.

کافی است فرض کنید که این اشیا سطرهایی از یک پایگاه داده می باشند.

دستور ng-app

این **directive** در واقع عنصر آغازین (**root element**) یک برنامه ی کاربردی را تعریف می کند.

ng-app directive زمانی که صفحه بارگذاری می شود، به صورت خودکار راه اندازی (**auto-bootstrap**) و مقدار دهی اولیه می شود.

در فواصل آینده یاد خواهید گرفت چگونه **ng-app** می تواند به منظور اتصال کد ماژول ها، دارای یک مقدار (مانند "**ng-app="myModule"**) باشد.

دستور in-init

این دستور مقادیر اولیه ی برنامه را مشخص می کند.

استفاده از **in-init** برای مقداردهی اولیه چندان متداول نیست، اغلب از یک کنترلر یا ماژول برای **initialization** استفاده می کنیم.

در مباحث بعدی درک بهتری نسبت به کنترلر ها و ماژول ها بدست خواهید آورد.

دستور ng-model

دستور **ng-model** مقادیر کنترل های دریافت کننده ورودی **HTML** همچون **input**، **select** و **textarea** را به داده های برنامه ی کاربردی متصل (**bind**) می کند.

دستور **ng-model** همچنین قادر به انجام عملیات زیر می باشد:

اعتبارسنجی داده های برنامه ی کاربردی از جمله عدد، ایمیل و اطلاعات لازم را بر عهده می گیرد.

وضعیت داده های برنامه را مشخص می کند: نامعتبر، **dirty** (دستخوش تغییر قرار گرفته)، خطا و دستکاری شده (**touched**) و غیره ...

کلاس های **css** را برای المان های **HTML** فراهم می نماید.

المان های **HTML** را به فرم های **HTML** مقید (**bind**) می کند.

دستور ng-repeat

دستور **ng-repeat** به ازای هر آیتم موجود در یک مجموعه یا آرایه، عناصر **HTML** را **clone** (تکثیر) می کند.

Tahildadeh