

آموزش Knockout - آموزش With binding در Knockout

هدف

کار این binding ایجاد کردن یک binding text جدید است، به گونه ای که المان های زیرمجموعه در درون متن یک شیء مشخص مقید باشند.

می توانید در کنار دیگر binding های کنترل جریان مانند if و foreach ، از binding های with به صورت لانه ای استفاده کنید.

مثال 1

در اینجا مثال ساده ای از تغییر binding text به یک شیء کودک (child object) را می توانید مشاهده کنید. توجه داشته باشید که در ویژگی های data-bind لازم نیست که حتما latitude یا longitude را در کنار coords به صورت پیشوندی استفاده کنید، زیر binding context به coords تغییر حالت داده است.

```
<h1 data-bind="text: city"> </h1>
<p data-bind="with: coords">
  Latitude: <span data-bind="text: latitude"> </span>,
  Longitude: <span data-bind="text: longitude"> </span>
</p>
```

```
<script type="text/javascript">
  ko.applyBindings({
    city: "London",
    coords: {
      latitude: 51.5001524,
      longitude: -0.1262362
    }
  });
</script>
```

مثال 2

این مثال تعاملی نشان می دهد که :

- With binding به صورت پویا و بر اساس اینکه مقدار مربوط به المان های زیرمجموعه تهی یا نامشخص هست یا خیر، این المان ها را حذف یا اضافه می کند.

- در صورتی که بخواهید از binding context های مادر به داده های یا توابع دسترسی داشته باشید در این صورت می توانید از مشخصه های متنی ویژه مانند \$parent و \$root استفاده کنید.

امتحان کنید :



Source code: View

```
<form data-bind="submit: getTweets">
  Twitter account:
  <input data-bind="value: twitterName" />
  <button type="submit">Get tweets</button>
</form>

<div data-bind="with: resultData">
  <h3>Recent tweets fetched at <span data-bind="text: retrievalDate"> </span></h3>
  <ol data-bind="foreach: topTweets">
    <li data-bind="text: text"></li>
  </ol>

  <button data-bind="click: $parent.clearResults">Clear tweets</button>
</div>
```

Source code: View model

```
function AppViewModel() {
  var self = this;
```

```

self.twitterName = ko.observable('@example');
self.resultData = ko.observable(); // No initial value

self.getTweets = function() {
  var name = self.twitterName(),
      simulatedResults = [
        { text: name + ' What a nice day.' },
        { text: name + ' Building some cool apps.' },
        { text: name + ' Just saw a famous celebrity eating lard. Yum.' }
      ];

  self.resultData({ retrievalDate: new Date(), topTweets: simulatedResults });
}

self.clearResults = function() {
  self.resultData(undefined);
}
}

ko.applyBindings(new AppViewModel());

```

پارامترها

- پارامتر اصلی

پارامتر اصلی، پارامتری است که می‌خواهید از آن به عنوان متن برای مقیدسازی المان‌های زیرمجموعه استفاده کنید.

اگر نتیجه ارزیابی عبارتی که شما وارد می‌کنید، تهی یا نامشخص باشد، در این صورت المان‌های زیرمجموعه به هیچ وجه مقید نمی‌شوند، بلکه در عوض از سند پاک می‌شوند.

اگر عبارتی که وارد می‌کنید شامل هر گونه مقدار قابل مشاهده‌ای باشد، در این صورت هر زمان که یکی از این مقادیر تغییر کنند، عبارت شما مجدداً ارزیابی می‌شود. سپس المان‌های زیرمجموعه کاملاً پاک می‌شوند و نسخه جدیدی از markup به سند شما اضافه می‌شود و در متن نتیجه جدید ارزیابی مقید می‌شود.

- پارامترهای اضافی
- ندارد

نکته : استفاده از with بدون المان نگهدارنده

مانند تمام المان های جریان کنترل دیگر مثل if و foreach ، می توانید از With بدون هیچ المان نگهدارنده ای استفاده کنید. این کار در زمانی مفید است که می خواهید از with استفاده کنید اما وارد کردن یک المان نگهدارنده جدید برای نگهداری with binding (legal) نیست. برای اطلاعات بیشتر به if یا foreach مراجعه کنید.

مثال:

```
<ul>
  <li>Header element</li>
  <!-- ko with: outboundFlight -->
    ...
  <!-- /ko -->
  <!-- ko with: inboundFlight -->
    ...
  <!-- /ko -->
</ul>
```

کامنت های <!-- ko --> و <!-- /ko --> ، به عنوان نشانگرهای آغازین و پایانی عمل می کنند، به این صورت که این کامنت ها المان های مجازی ای را تعریف می کنند که در درون خود markup را نگهداری می کنند. knockoutJS درست مانند وقتی که از المان نگهدارنده واقعی استفاده کنید، این سینتکس مجازی را bind می کند.

وابستگی ها

به غیر از کتابخانه اصلی knockoutJS ، وابستگی دیگری ندارد.