

آموزش KnockoutJS - آموزش If binding در KnockoutJS

هدف

این Binding باعث می شود که تنها در صورتی که مقدار عبارتی مشخص برابر با true شود (یا مقدار شبه true ای مانند شيء تهی یا رشته های غیر خالی) بخشی از markup، در سند شما ظاهر شود (و ویژگی های data-bind آن اعمال شود).

If نقش مشابهی با visible binding دارد. تفاوت آن ها در این است که در visible binding همیشه markup در DOM باقی می ماند و ویژگی های data-bind آن همواره اعمال می شود (visible binding برای شروع کردن مشاهده پذیری المان نگهدارنده تنها از CSS استفاده می کند. این در حالی است که if binding به صورت فیزیکی markup را در DOM شما اضافه می کند یا از آن حذف می کند و تنها در صورتی که مقدار عبارت برابر با true باشد، binding ها را تنها بر زیرمجموعه ها اعمال می کند.

مثال 1

این مثال نشان می دهد که if binding می تواند به صورت پویا بخش هایی از markup را همزمان با تغییر مقادیر قابل مشاهده اضافه و یا حذف کند.

Display message

Source code: View

```
<label><input type="checkbox" data-bind="checked: displayMessage" /> Display message</label>
```

```
<div data-bind="if: displayMessage">Here is a message. Astonishing.</div>
```

Source code: View model

```
ko.applyBindings({
  displayMessage: ko.observable(false)
});
```

مثال 2

در این مثال، المان <div> در Mercury خالی و در Earth شلوف است. دلیل این اتفاق این است که مشخصه capital مربوط به Earth، یک مشخصه غیر تهی است، این در حالی است که این مشخصه برای Mercury تهی است.

```

<ul data-bind="foreach: planets">
  <li>
    Planet: <b data-bind="text: name"> </b>
    <div data-bind="if: capital">
      Capital: <b data-bind="text: capital.cityName"> </b>
    </div>
  </li>
</ul>

```

```

<script>
  ko.applyBindings({
    planets: [
      { name: 'Mercury', capital: null },
      { name: 'Earth', capital: { cityName: 'Barnsley' } }
    ]
  });
</script>

```

درک این نکته بسیار مهم است که برای اینکه کد بالا به خوبی کار کند، وجود if binding حیاتی است. بدون این binding، بعد از اینکه کد تلاش کند تا capital.cityName را در متن Mercury جایی که Capital تهی است ارزیابی کند، با خطا مواجه خواهید شد. در جاوا اسکریپت نمی توانید زیر مشخصه های مقادیر تهی یا نامشخص را ارزیابی کنید.

پارامترها

- پارامتر اصلی

پارامتر اصلی عبارتی است که می خواهید آن را ارزیابی کنید. اگر نتیجه ارزیابی برابر با true یا مقداری مشابه با true باشد، در این صورت contained markup در سند حاضر شده و تمامی ویژگی های data-bind در آن اعمال خواهند شد. اما اگر نتیجه ارزیابی عبارت برابر با false شود، در این صورت contained markup بدون اینکه هیچ binding ای بر آن اعمال شود، حذف می شود.

اگر عبارت شما شامل هر گونه مقدار قابل مشاهده ای باشد، در این صورت این عبارت هر زمان که یکی از این مقادیر تغییر کند، مجدداً ارزیابی خواهد شد. در ادامه پس از تغییر نتیجه این عبارت، markup موجود در بلوک if می تواند اضافه یا حذف شود. data-bind هر زمان که مجدداً اضافه شود، ویژگی های آن بر روی نسخه جدیدی از contained markup اعمال خواهند شد.

- پارامترهای اضافی

• ندارد

نکته : استفاده از `if` بدون المان نگهدارنده

گاهی اوقات ممکن است که بدون استفاده از المان نگهدارنده ای که بتواند `if binding` را نگهداری کند، بخواهید حضور یا عدم حضور بخشی از markup را کنترل کنید. برای مثال، ممکن است بخواهید که ظاهر شدن یا ظاهر نشدن المان `<i>` را در کنار دیگر المان های هم نوع خود (siblings) کنترل کنید:

```
<ul>
  <li>This item always appears</li>
  <li>I want to make this item present/absent dynamically</li>
</ul>
```

در این حالت نمی توانید `if` را در `` قرار دهید (زیرا در این صورت ابتدا اثر خود را بر `<i>` هم می گذارد) و نگهدارنده دیگری هم نمی توانید در اطراف `<i>` دوم قرار دهید (چون در HTML امکان اضافه کردن نگهدارنده های اضافی در درون `` وجود ندارد).

برای حل این مشکل می توانید از سینتکس جریان کنترل بدون نگهدارنده استفاده کنید. این سینتکس مبتنی بر تگ های توضیحی می باشد. برای مثال :

```
<ul>
  <li>This item always appears</li>
  <!-- ko if: someExpressionGoesHere -->
    <li>I want to make this item present/absent dynamically</li>
  <!-- /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.