

در مطلب « [فعال سازی عملیات CRUD در Kendo UI Grid](#) » با نحوه‌ی تعریف مقدماتی اعتبارسنجی فیلدهای تعریف شده، آشنا شدید:

```
fields: {
  "Price": { type: "number", validation: { required: true, min: 1 } }
}
```

در ادامه نگاهی خواهیم داشت به جزئیات تکمیلی امکانات اعتبارسنجی ورودی‌های کاربر در Kendo UI.

HTML 5 و Kendo UI Validation

در HTML 5 امکان تعریف نوع‌های خاص کنترل‌های ورودی کاربر مانند color, email, url, number, range, date, search وجود دارد. برای مثال در اینجا اگر کاربر تاریخ غیرمعتبری را وارد کند، مرورگر پیام اعتبارسنجی متناظری را به او نمایش خواهد داد. همچنین در HTML 5 امکان افزودن ویژگی required نیز به کنترل‌های ورودی پیش‌بینی شده‌است. اما باید در نظر داشت که مرورگرهای قدیمی از این امکانات پشتیبانی نمی‌کنند. در این حالت Kendo UI با تشویق استفاده از روش معرفی شده در HTML 5، با آن یکپارچه شده و همچنین این قابلیت‌های اعتبارسنجی HTML 5 را در مرورگرهای قدیمی نیز میسر می‌کند. Kendo UI Validation جزو [نسخه‌ی سورس باز](#) Kendo UI با مجوز Apache نیز می‌باشد. نمونه‌ای از امکانات اعتبارسنجی توکار HTML 5 را در اینجا مشاهده می‌کنید:

```
<input type="text" name="firstName" required />
<input type="text" name="twitter" pattern="https?://(?:www\.)?twitter\.com/.+i" />
<input type="number" name="age" min="1" max="42" />
<input type="number" name="age" min="1" max="100" step="2" />
<input type="url" name="url" />
<input type="email" name="email" />
```

یکپارچه سازی اعتبارسنجی Kendo UI با اعتبارسنجی HTML 5

در اینجا یک فرم تشکیل شده با ساختار HTML 5 را ملاحظه می‌کنید. هر دو فیلد ورودی، با ویژگی استاندارد required مزین شده‌اند. همچنین توسط ویژگی type، ورودی دوم جهت دریافت آدرس ایمیل معرفی شده‌است. چون فیلد دوم دارای دو اعتبارسنجی تعریف شده است، دارای دو ویژگی data-* برای تعریف پیام‌های اعتبارسنجی متناظر نیز می‌باشد. الگوی تعریف آن‌ها data-[rule]-msg است.

```
<div class="k-rtl">
  <form id="testView">
    <label for="firstName">نام</label>
    <input id="firstName"
      name="firstName"
      type="text"
      class="k-textbox"
      required
      validationmessage="لطفا نامی را وارد کنید">
    <br>
    <label for="emailId">آدرس پست الکترونیک</label>
    <input id="emailId"
      name="emailId"
      type="email"
      dir="ltr"
      required
      class="k-textbox"
      data-required-msg="لطفا ایمیلی را وارد کنید."
      data-email-msg="ایمیل وارد شده معتبر نیست">
    <br>
```

```


</form>
</div>

<script type="text/javascript">
$(function () {
    $("form#testView").kendoValidator();
});
</script>

```

تنها کاری که جهت یکپارچه سازی امکانات اعتبارسنجی Kendo UI با اعتبارسنجی استاندارد HTML 5 باید انجام داد، فراخوانی متد kendoValidator بر روی ناحیه‌ی مشخص شده است.

تعیین محل نمایش پیام‌های اعتبارسنجی

پیام‌های اعتبارسنجی Kendo UI به صورت خودکار در کنار فیلد متناظر با آن نمایش داده می‌شوند. اما اگر نیاز به تعیین مکان دستی آن‌ها وجود داشت (جهت خوانایی بهتر) باید به نحو ذیل عمل کرد:

```

</span>

```

در اینجا span با کلاس k-invalid-msg و ویژگی data-for که به name کنترل ورودی اشاره می‌کند، محل نمایش پیام اعتبارسنجی متناظر با فیلد name خواهد بود.

تعریف سراسری پیام‌های اعتبارسنجی

در مثال فوق، به ازای تک تک فیلدهای ورودی، پیام اعتبارسنجی متناظر با required وارد شد. می‌توان این پیام‌ها را حذف کرد و در قسمت messages متد kendoValidator قرار داد:

```

<script type="text/javascript">
$(function () {
    $("form#testView").kendoValidator({
        messages: {
            // {0} would be replaced with the input element's name
            required: '{0} را تکمیل کنید.',
            email: 'ایمیل وارد شده معتبر نیست.'
        }
    });
});
</script>

```

- به این صورت پیام‌های اعتبارسنجی required و email، به صورت یکسانی به تمام المان‌های دارای این ویژگی‌ها اعمال خواهند شد.

- در این پیام‌ها {0} با مقدار ویژگی name فیلد ورودی متناظر جایگزین می‌شود.

- اگر هم در markup و هم در تعاریف kendoValidator، پیام‌های اعتبارسنجی تعریف شوند، حق تقدم با تعاریف markup خواهد بود.

اعتبارسنجی سفارشی سمت کاربر

علاوه بر امکانات استاندارد HTML 5، امکان تعریف دستورهای اعتبارسنجی سفارشی نیز وجود دارد:

```
<script type="text/javascript">
$(function () {
    $("#form#testView").kendoValidator({
        rules: {
            customRule1: function (input) {
                if (!input.is("[id=firstName]"))
                    return true;

                var re = /^[A-Za-z]+$/;
                return re.test(input.val());
            }
        }, //, customRule1: ....
        messages: {
            // {0} would be replaced with the input element's name
            required: '{0} را تکمیل کنید.',
            email: 'ایمیل وارد شده معتبر نیست.',
            customRule1: 'اعداد مجاز نیستند.'
        }
    });
});
</script>
```

- همانطور که ملاحظه می‌کنید، برای تعریف منطق اعتبارسنجی سفارشی، باید از خاصیت rules ورودی متد kendoValidator شروع کرد. در اینجا نام یک متد callback دلخواهی را وارد کرده و سپس بر اساس منطق اعتبارسنجی مورد نظر، باید true/false را بازگشت داد. برای نمونه در این مثال اگر کاربر در فیلد نام، عدد وارد کند، ورودی او مورد قبول واقع نخواهد شد.
- باید دقت داشت که اگر بررسی input.is صورت نگیرد، منطق تعریف شده به تمام کنترل‌های صفحه اعمال می‌شود.
- پیام متناظر با این دستور سفارشی جدید، در قسمت messages، دقیقاً بر اساس نام callback method تعریف شده در قسمت rules باید تعریف شود.

فراخوانی دستی اعتبارسنجی یک فرم

در حالت پیش فرض، با کلیک بر روی دکمه‌ی ارسال، اعتبارسنجی کلیه عناصر فرم به صورت خودکار انجام می‌شود. اگر بخواهیم در این بین یک پیام سفارشی را نیز نمایش دهیم می‌توان به صورت زیر عمل کرد:

```
<script type="text/javascript">
$(function () {
    $("#form#testView").submit(function (event) {
        event.preventDefault();
        var validator = $("#form#testView").data("kendoValidator");
        if (validator.validate()) {
            alert("validated!");
        } else {
            alert("There is invalid data in the form.");
        }
    });
});
```

```
});
$("form#testView").kendoValidator();
});
</script>
```

در اینجا رخداد submit فرم بازنویسی شده و متد validate آن بر اساس kendoValidator تعریف شده، به صورت دستی فراخوانی می‌شود.

اعتبارسنجی سفارشی در DataSource

در تعریف فیلدهای مدل DataSource، امکان تعریف اعتبارسنجی‌های پیش فرضی مانند min، max، required و امثال آن وجود دارد که نمونه‌ای از آن‌را در بحث [فعال سازی CRUD در Kendo UI Grid](#) مشاهده کردید:

```
fields: {
  "serviceName": {
    type: "string",
    defaultValue: "Inspection",
    editable: true,
    nullable: false,
    validation: { /*...*/ }
  },
  // ...
}
```

برای تعریف اعتبارسنجی سفارشی در اینجا، همانند متد kendoValidator نیاز است یک یا چند callback متد سفارشی را طراحی کرد:

```
schema: {
  model: {
    id: "ProductID",
    fields: {
      ProductID: { editable: false, nullable: true },
      ProductName: {
        validation: {
          required: true,
          custom1: function (input) {
            if (input.is("[name='ProductName']") && input.val()
              != "") {
              input.attr("data-custom1-msg", "نام محصول باید
                با حرف بزرگ انگلیسی شروع شود");
              return /^[A-Z]/.test(input.val());
            }
            return true;
          },
          // custom2: ...
        },
        UnitPrice: { type: "number", validation: { required: true, min:
          1 } },
        Discontinued: { type: "boolean" },
        UnitsInStock: { type: "number", validation: { min: 0, required:
          true } }
      }
    }
  }
}
```

نام این متد که نهایتاً true/false بر می‌گرداند، اختیاری است. نام کنترل جاری همان نام فیلد متناظر است (جهت محدود کردن بازه‌ی اعمال منطق اعتبارسنجی). برای مقدار دهی پیام اعتبارسنجی از متد input.attr و الگوی data-[validationRuleName]-msg استفاده می‌شود. ضمناً به هر تعداد لازم می‌توان در اینجا custom rule تعریف کرد. متد input.val() مقدار کنترل جاری را بر می‌گرداند. برای دسترسی به مقدار سایر کنترل‌ها می‌توان از روش \$("#fieldName").val() استفاده کرد.

نظرات خوانندگان

نویسنده: محمد رعیت پیشه

تاریخ: ۱۷:۱۱ ۱۳۹۳/۰۸/۲۲

ممنون مطلبتون.

آیا اعتبار سنجی در حالت Inline Editing درون گرید هم به همین شکل هست؟

نویسنده: وحید نصیری

تاریخ: ۱۸:۱۱ ۱۳۹۳/۰۸/۲۲

بله. اطلاعات اعتبارسنجی فیلدهای خودش را از data source دریافت می‌کند. مثال [فعال سازی CRUD در Kendo UI Grid](#) را اجرا کنید، این مورد در آن لحاظ شده.