Atitit 数据校验法

目录

[1. 用内置表单数据校验 简单快捷 1](#_Toc16132)

[1.1. input 元素的校验约束 — starting simple 1](#_Toc10253)

[1.1.1. required 属性 1](#_Toc3609)

[1.1.2. 使用正则表达式校验 1](#_Toc2225)

[1.1.3. 限制输入的长度 1](#_Toc8738)

[2. 使用 JavaScript校验表单 （可自定义） 2](#_Toc19412)

[3. 校验库 2](#_Toc30621)

[4. 远程校验 2](#_Toc6360)

# 用内置表单数据校验 简单快捷

[HTML5](https://developer.mozilla.org/en-US/docs/HTML/HTML5" \o "/en-US/docs/HTML/HTML5) 一个特别有用的新功能就是，可以在不写一行脚本代码的情况下，即对用户的输入进行数据校验，这都是通过表单元素的[校验属性](https://developer.mozilla.org/zh-CN/docs/Web/Guide/HTML/HTML5/Constraint_validation)实现的，这些属性可以让你定义一些规则，用于限定用户的输入，比如某个输入框是否必须输入，或者某个输入框的字符串的最小最大长度限制，或者某个输入框必须输入一个数字、邮箱地址等；还有数据必须匹配的模式。如果表

## input 元素的校验约束 — starting simple

### required 属性

最简单的HTML5校验功能是 [required](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \l "attr-required)属性 — 如果要使输入成为必需的，则可以使用此属性标记元素。 当设置此属性时，如果

### 使用正则表达式校验

另一个常用的校验功能是 [pattern](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \l "attr-pattern) 属性, 以 [Regular Expression](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Guide/Regular_Expressions" \o "/en-US/docs/JavaScript/Guide/Regular_Expressions) 作为 value 值. 正则表达式 (regex) 是一个可以用来匹配文本字符串中字符的组合的模式，所以它们是理想的表单校验器，也可以支持 JavaScript 中许多其它的用途。

### 限制输入的长度

所有文本框 ([<input>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \o "HTML <input> 元素用于为基于Web的表单创建交互式控件，以便接受来自用户的数据; 可以使用各种类型的输入数据和控件小部件，具体取决于设备和user agent。) 或 [<textarea>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/textarea" \o "HTML <textarea> 元素表示一个多行纯文本编辑控件。)) 都可以使用[minlength](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \l "attr-minlength) 和 [maxlength](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \l "attr-maxlength) 属性来限制长度. 如果输入的字段长度小于 [minlength](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \l "attr-minlength) 的值或大于 [maxlength](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \l "attr-maxlength) 值则无效. 浏览器通常不会让用户在文本字段中键入比预期更长的值，不过更精细的设置总归是更好的

# 使用 JavaScript校验表单 （可自定义）

如果你想控制原生错误信息的界面外观，或者你想处理不支持HTML内置表单校验的浏览器，则必须使用 Javascript。

# 校验库

* 独立的库（原生 Javascript 实现）：
  + [Validate.js](http://rickharrison.github.com/validate.js/" \o "http://rickharrison.github.com/validate.js/)
* jQuery 插件:
  + [Validation](http://bassistance.de/jquery-plugins/jquery-plugin-validation/" \o "http://bassistance.de/jquery-plugins/jquery-plugin-validation/)
  + [Valid8](http://unwrongest.com/projects/valid8/" \o "http://unwrongest.com/projects/valid8/)

# 远程校验

在某些情况下，执行一些远程校验可能很有用。 当用户输入的数据与存储在应用程序服务器端的附加数据绑定时，这种校验是必要的。 一个应用实例就是注册表单，在这里你需要一个用户名。 为了避免重复，执行一个 AJAX 请求来检查用户名的可用性，要比让先用户发送数据，然后因为表单重复了返回错误信息要好得多。

执行这样的校验需要采取一些预防措施：

* 它要求公开 API 和一些数据；您需要确保它不是敏感数据。
* 网络滞后需要执行异步校验。这需要一些用户界面的工作，以确保如果校验没有适当的执行，用户不会被阻止。

表单数据校验 - 学习 Web 开发 \_ MDN.html