

Validating Form Fields

Why Validate Content?

The simplest way of validation is to mark a form element as mandatory. The validation then checks whether the element has been filled with content as required.

A second validation level is to validate the field contents themselves, for example to prevent incorrect or incomplete entries. For example, an input field for (German) postal codes should consist of exactly five digits or an e-mail address field should contain a valid email address. A validation of entries can be done using regular expressions.

The form element [Text Box](#) therefore contains the attribute **Regular Expression** where you can enter the desired regular expressions. When validating the data entered in the input field, the system then checks whether the entries comply with the required rules.



See [ICU's Regular Expressions package](#) for an overview of regular expression metacharacters and operators.

On this Page:

- [Why Validate Content?](#)
- [Helpful Regular Expressions](#)

Related Pages:

- [Supported Form Elements](#)
 - [Text Box](#)
- [Regular Expressions](#)

Helpful Regular Expressions

We have compiled a list of helpful regular expressions for you:

Regular Expression	Description	Use Case
<code>[0-9]{5}</code>	Only digits and exactly five digits are accepted.	Validate a German zip code.
<code>[0-9/-]*</code>	The field may only contain (any number of) digits, slash, space and minus sign, or it may be blank.	Check the input of phone numbers.
<code>[A-Za-z0-9]+</code>	Only letters and numbers are allowed.	Prevent usage of special characters in the input.
<code>[a-zA-Z0-9 öäüÖÄÜ \$%&]*</code>	Allowed input: Any number of upper and lower case letters, digits, umlauts and the special characters \$, %, & are permitted.	Suitable for checking a password that may only contain the special characters specified in the regular expression.
<code>[a-zA-Z0-9\$%&]{8,16}</code>	Allowed input: Upper and lower case letters (but no umlauts), digits and the special characters \$, %, &. And the input must consist of at least 8, but not more than 16 characters.	Suitable for checking a password that may only contain the special characters specified in the regular expression and must have a certain length.
<code>[a-zA-Z0-9._-]+@[a-zA-Z0-9._-]+\.[a-z]{2,}</code>	Allowed input: <ul style="list-style-type: none">• String (containing letters, digits, dot, underscore or minus) followed by an @-sign and another string (containing letters, digits, dot, underscore or minus) followed by a dot and two or more letters.	Validate the input of an email address.
<code>(http://www\ https://www\ http://\ https://)?[a-z0-9]+(\ [a-z0-9]+)\.[a-z]{2,5}(:[0-9]{1,5})?(V.*)?</code>	Allowed input: <ul style="list-style-type: none">• URL with prefix http://• URL with prefix https://• URL without prefix	Checks if the input contains a valid URL.



Further helpful hints on regular expressions can be found at <https://wiki.selfhtml.org>. A tester for self-created regular expressions can be found at <http://www.regexpal.com>.