

Validierung

gex url https

Betrifft: **Seite** / **Element**

- [Anwendung im Wizard](#)
 - [Klassifizierung](#)
 - [Syntax \(java expression parser\) verwenden](#)
 - [Formelwizard verwenden](#)
 - [Fehlermeldung erfassen](#)
- [REGEX Validierungen](#)
 - [Verwenden einer Regex](#)
 - [Vom System definierte REGEX Konstanten:](#)
 - [Tutorial](#)
- [Datumsvergleich](#)
- [Checkboxgruppe - maximale Auswahl festlegen](#)

Validierungen dienen zur Überprüfung der gemachten Angaben. Validierungen können unter dem Bereich / Tab "Validierung" der Feldeigenschaften definiert werden.

Anwendung im Wizard

Beschriftung	Hilfetext	Fragetyp	Initialwert	Sichtbarkeit ⓘ	Editierbarkeit	Validierung	Regeln
Eigenschaften	Drucken	Beschreibung					

+ -	Klassifizierung	Nachricht	
<input checked="" type="radio"/>	Fehler	Nein, Bitte überprüfen Sie die Versicherten Nr., Nein, Nein, Nein	

Klassifizierung * Fehler ▼

Syntax (java expression parser) verwenden? ☒

Nur zum Validierungszeitpunkt prüfen? ☐

JEP Syntax

Klassifizierung

Meistens als **Fehler**, damit das "Absenden" der nicht korrekt validierten Daten nicht erfolgen kann.

Syntax (java expression parser) verwenden

Wird diese Option gewählt, muss die Formel manuell erfasst werden. Dies lässt mehr Freiheit zu. Ohne JEP können die Bedingungen zusammengekllickt werden.

Syntax (java expression parser) verwenden? ☒

Nur zum Validierungszeitpunkt prüfen? ☐

JEP Syntax

```
nodeValue(".") != null && !matchesRegex("[0-9]{3}[.]{1}[0-9]{4}[.]{1}[0-9]{4}[.]{1}[0-9]{2}",nodeValue("."))
```

Anzahl verfügbarer Zeichen: 4892

evalFormula

Die Formel muss in der JEP Syntax manuell eingegeben werden.

Formelwizard verwenden

* * kleiner als * 0

Fehlermeldung erfassen

REGEX Validierungen

Im Feld "JEP Syntax" wird die entsprechende Formel eingegeben: `nodeValue(".") != null && !matchesRegEx("REGEX",nodeValue("."))`

- ### Vom System definierte REGEX Konstanten:

Verwendung: `matchesRegEx(<Schlüssel>,nodeValue("."))`

Beispiel: `nodeValue(".") != null && !matchesRegex(REGEX_EMAIL,nodeValue("."))`

Tutorial

JAVA RegularExpressions > <http://www.vogella.com/tutorials/JavaRegularExpressions/article.html>

REGEX 101 (bauen, testen, debuggen): <https://regex101.com/>

Datumsvergleich

Beispiel eines Datumsvergleichs

Berechtigung ab	<input type="text" value="berechtigungAbDatum_01"/>	*	<input type="text" value="TT.MM.JJJJ"/>	
Berechtigung bis	<input type="text" value="berechtigungBisDatum_01"/>	*	<input type="text" value="TT.MM.JJJJ"/>	

Vorgabe ist, dass das BIS-Datum mindestens 30 Tage nach dem AB-Datum ist.

Formel: `(berechtigungBisDatum_01 < DateTools(berechtigungAbDatum_01,"+30D")) && berechtigungBisDatum_01 != NULL`

Beim BIS-Feld > im Wizard > Regeln

Feldeigenschaften (berechtigungBisDatum_01)

Beschreibung	Hilfetext	Frageart	Initialwert	Sichtbarkeit	Editierbarkeit	Validierung	Regeln	Eigenschaften
--------------	-----------	----------	-------------	--------------	----------------	-------------	--------	---------------

Drucken

Beschreibung

Verarbeitung

	Klassifizierung	Nachricht
<input checked="" type="radio"/>	Fehler	Nein, Nein, Das Datum der «Berechtigung bis» muss in der Zukunft liegen und später als das Datum «Ber...

Klassifizierung

* Fehler

Syntax (java expression parser) verwenden?

☒

Nur zum Validierungszeitpunkt prüfen?

☐

Java Implementationsklasse (Condition Interface)

JEP Syntax

(berechtigungBisDatum_01 < DateTools(berechtigungAbDatum_01,"+30D")) && berechtigungBisDatum_01 != NULL

Anzahl verfügbarer Zeichen: 4987

Formel prüfen

☐ Validierungsnachricht

OK

Abbrechen

Weitere Dokumentation "[bootstrap-datetimepicker](#)"

Checkboxgruppe - maximale Auswahl festlegen

- Siehe [separate Seite](#) unter Tipps und Tricks