

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	
⊙	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 zusammengeklickt werden.

Syntax (java expression parser) verwenden?

Nur zum Validierungszeitpunkt prüfen?

JEP Syntax

Anzahl verfügbarer Zeichen: 4892

evalFormula

Die Formel muss in der JEP Syntax manuell eingegeben werden.

Formelwizard verwenden

Neue Bedingung hinzufügen

* * kleiner als

- AngabenPerson_email
- AngabenPerson_firstName
- AngabenPerson_lastName

Beim Wizard werden die Bedingung zusammengeklickt. Gleiche Logik wie bei [Sichtbarkeit](#).

Fehlermeldung erfassen

Validierungsnachricht

de

Als Formel interpretieren

Bitte überprüfen Sie die Versicherten-Nr.

REGEX Validierungen

Verwenden einer Regex

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

- Beispiel E-Mail-Adresse: `nodeValue(".") != NULL && !matchesRegEx("^[A-Za-z0-9.!#$%&*+\\-/?^_`{}~]+@[a-zA-Z0-9][(-]?[a-zA-Z0-9]*([-]?[a-zA-Z0-9])?[a-zA-Z0-9]{2,})[.][a-zA-Z]{2,}$")`
- Beispiel UID (Firmennummer): `nodeValue(".") != null && !matchesRegEx("(CHE-)[0-9]{3}[.][0-9]{3}[.][0-9]{3}",nodeValue("."))`
- Beispiel AHV13: `nodeValue(".") != null && !matchesRegEx("[0-9]{3}[.]{1}[0-9]{4}[.]{1}[0-9]{4}[.]{1}[0-9]{2}",nodeValue("."))`
- Beispiel IBAN: `nodeValue(".") != null && !matchesRegEx("[a-zA-Z]{2}[0-9]{2}[a-zA-Z0-9]{4}[0-9]{7}([a-zA-Z0-9]?){0,16}",nodeValue("."))`
- Beispiel E-Mail-Adressen, welche eine bestimmte Domain enthalten (Beispiel Kt. BE):
 - `nodeValue(".") != null && !matchesRegEx("[a-zA-Z0-9.!#$%&*+\\-/?^_`{}~]+@[a-zA-Z][a-zA-Z][a-zA-Z]\\.(be|be|police\\.be|justice\\.be|fk\\.be|bfh|phbern|[a-zA-Z]{2,20}\\.(unibe|unibe|wb|bbz-biel|tfbern|gymkirchenfeld|gymneufeld|koeniz-lerbermatt|gymhofwil|gymbiel-seeland|gfbienne|gymburgdorf|bz|gyminterlaken|gymthun|upd|ceff|pzmag|bzemme)\\.ch$")`
 - (String + @ + (kantons-url) .ch am Ende)
- Beispiel Telefonnummer: `nodeValue(".") != null && !matchesRegEx("^(+)[0]((\\d[/-]){9,15})$",nodeValue("."))`
- Beispiel URL beginnend mit http:// oder https://: `nodeValue(".") != NULL && !matchesRegEx("^https?:\\W\\W.*",nodeValue("."))`
- Beispiel ZEMIS-Nr. (KTGR): `nodeValue(".") != null && !matchesRegEx("[0-9*]{6,8}[.]{1}[0-9]{1}",nodeValue("."))`
(Verhindert die Eingabe von alphanumerischen Zeichen «*» und gibt eine mind. und max. Länge des ersten Teils bis zum Punkt vor «6,8»)

Vom System definierte REGEX Konstanten:

REGEX_EMAIL	<code>[a-zA-Z0-9]+[._a-zA-Z0-9]!#\$%&apos;*+\\-/?^_`{}~]*[a-zA-Z]*@[a-zA-Z0-9]{2,8}.[a-zA-Z]{2,6}</code>
REGEX_SWISS_PHONE	<code>^(+)[0]((\\d[/-]){9,15})\$</code>

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

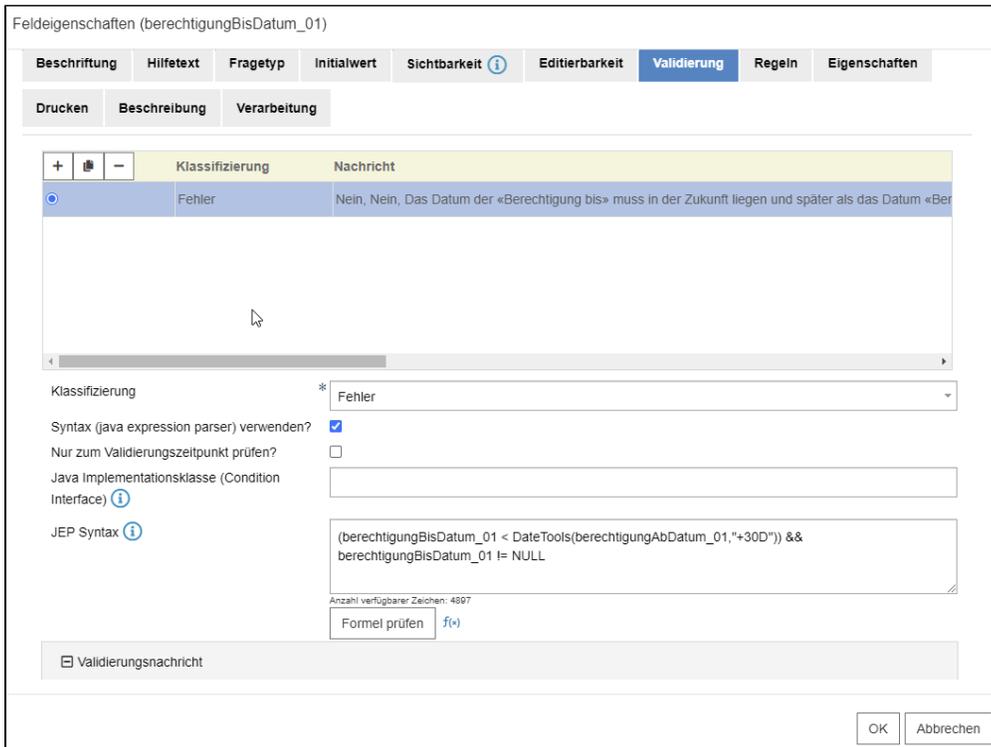
Beispiel eines Datumsvergleichs



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



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

Checkboxgruppe - maximale Auswahl festlegen

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