Advanced Validation Conditions

This page contains advanced information

Icon

If you are looking for basic information on validation conditions please visit Basic Validation Conditions.

Sometimes there are inputs which require not only a valid range (IE: between 96 and 105), but a specific number of significant figures (96.6, not 96.65 or 96) or other specific structure (A1234). To set this up, you can use a constraint in the **Validation Condition** section for a field with a Regular Expression.

To design and test your regular expressions, you can use this website: http://www.regexr.com/

Here are some basic examples:

- To restrict a text field to only accept letters (no numbers allowed)
  - regex(.,'^[a-zA-Z]+$')   not allowing spaces
  - regex(., '^[a-zA-Z\s]+$')   allowing spaces
- To restrict a numeric ID field to accept only numbers (no letters allowed)- relevant in CloudCare:
  - regex(.,'^[0-9]+$')

NOTE: This is NOT the correct format if the phone number is going to be used for text messaging /CommCare Messaging. For that, the number would without any dashes: 1234567890

Considerations

- It is important that the input type for this question be text, not number, integer, or decimal; therefore you should use the question type "Numeric ID /Phone Number" as this question type is technically text with a numeric keyboard appearance. Numbers are represented and compared by their numeric value, so "65.00" and "65" are equivalent.

- These validation example often don't validate the actual size of the number at all. You can similarly restrict the values before the decimal sign in your expression using regular expressions, but remember that if it is difficult to express whether a value is valid, users may have a hard time entering the correct value.

Examples

Example 1: Require a specific number of significant figures

To require an input with an arbitrary size, but always with two significant figures, you can use the following:

Valid Inputs: 34.00, 4.32, 0.23

Invalid Inputs: 34, 0.0, 2.3, 34.2222

Enter into Validation Condition = regex(., '^[0-9]*\.[0-9][0-9]$')

the two **[0-9]** elements at the end are the placeholders for the significant figure digits. You can add or remove more to manipulate the number of digits

Example 2: Require up to a specific number of significant figures

If you want to specify that the expression *can* have up to two significant figures the expression is somewhat more complex:

Valid Inputs: 34.00, 4.32, 0.23, 34, 3.2

Invalid Inputs: 34.444, 34.0000, 23. (with nothing after the ".)

Enter into Validation Condition = regex(., '^[0-9]*(\.[0-9][0-9]?)?$')

In this case, you can add additional significant figures by appending more **[0-9]?** elements after the final one

Example 3: Require certain types of significant figures

If you want to specify that expression can end in either .0 or .5, the expression is again somewhat different:

Valid Inputs: 20.5, 100.0, 45.5

Invalid Inputs: 20, 20.2, 97.7

```
Enter into Validation Condition = regex(.,'^[0-9]*\.(0|5)$')
```

Example 4: Require a phone number to be entered in a specific format

If you want your phone number to appear in the format 123-456-7890 you can use the following validation condition:

```
regex(.,'^[0-9][0-9][0-9]-[0-9][0-9][0-9]-[0-9][0-9][0-9][0-9]$')
or
regex(.,'^[0-9]{3}-[0-9]{3}-[0-9]{4}$')
```

Example 5: United Kingdom NHS Number

The United Kingdom NHS number is a 10 digit number (9 real, 1 check digit). It is validated using the Modulus 11 algorithm Details about it are here: http://www.datadictionary.nhs.uk/version2/data_dictionary/data_field_notes/n/nhs_number_de.asp?shownav=0

Here's the expression syntax to validate it (assuming your Question ID is 'NHS_Number'):

```
11 - (substr(#form/NHS_Number, 0, 1) * 10 + substr(#form/NHS_Number, 1, 2) * 9 + substr(#form
/NHS_Number, 2, 3) * 8 + substr(#form/NHS_Number, 3, 4) * 7 + substr(#form/NHS_Number, 4, 5) * 6 +
substr(#form/NHS_Number, 5, 6) * 5 + substr(#form/NHS_Number, 6, 7) * 4 + substr(#form/NHS_Number, 7,
8) * 3 + substr(#form/NHS_Number, 8, 9) * 2) mod 11 = substr(#form/NHS_Number, 9, 10)
```