

STARTSWITH Function

Contents:

- *Basic Usage*
- *Syntax and Arguments*
 - *column_any*
 - *pattern*
- *Examples*
 - *Example - STARTSWITH and ENDSWITH Functions*

Returns `true` if the leftmost set of characters of a column of values matches a pattern. The source value can be any data type, and the pattern can be a Trifacta pattern, regular expression, or a string.

- The `STARTSWITH` function is ideal for matching based on patterns for any data type. If you need to match strings using a fixed number of characters, you should use the `LEFT` function instead. See *LEFT Function*.
- See *ENDSWITH Function*.

Basic Usage

String literal example:

```
derive type:single value:STARTSWITH(FullName, 'Mr.')
```

Output: Writes `true` into a new column if the first three letters of the `FullName` column value are "Mr.".

Trifacta pattern example:

```
derive type:single value:STARTSWITH(CustId, `{alpha-numeric}{6}`) as:'validateCustId'
```

Output: Generates the `validateCustId` column containing `true` if the `CustId` column begins with a six-digit alpha-numeric sequence. Otherwise, the `validateCustId` column is set to `false`.

Regular expression pattern example:

```
set col:Status value:IF(STARTSWITH(phone, /^(\\+0?1\\s)?\\(?\\d{3}\\)?[\\s.-]\\d{3}[\\s.-]\\d{4}$ /), 'phone - ok', 'phone - error')
```

Output: Sets the value in the `Status` column to `phone - ok` if the value of the `phone` column begins with a value that matches a 10-digit U.S. phone number. Otherwise, the `Status` column is set to `phone - error`.

Syntax and Arguments

```
derive type:single value:STARTSWITH(column_any,pattern)
```

Argument	Required?	Data Type	Description
<code>column_any</code>	Y	any	Name of the column to be applied to the function
<code>pattern</code>	Y	string	Pattern or literal expressed as a string describing the pattern to which to match.

For more information on syntax standards, see *Language Documentation Syntax Notes*.

column_any

Name of the column to be searched.

- Multiple columns and wildcards are not supported.

Usage Notes:

Required?	Data Type	Example Value
Yes	Column reference	myColumn

pattern

Trifacta pattern, regular expression, or string literal to locate in the values in the specified column.

Usage Notes:

Required?	Data Type	Example Value
Yes	String	{zip}

Examples

Tip: For additional examples, see *Common Tasks*.

Example - STARTSWITH and ENDSWITH Functions

The following example demonstrates functions that can be used to evaluate the beginning and end of values of any type using patterns. These functions include the following:

- **STARTSWITH** - check start of values in a specified column against a specific pattern or literal. See *STARTSWITH Function*.
- **ENDSWITH** - check end of values in a specified column against a specific pattern or literal. See *ENDSWITH Function*.

Source:

The following inventory report indicates available quantities of product by product name. You need to verify that the product names are valid according to the following rules:

- A product name must begin with a three-digit numeric brand identifier, followed by a dash.
- A product name must end with a dash, followed by a six-digit numeric SKU.

Source data looks like the following, with the Validation column having no values in it.

InvDate	ProductName	Qty	Validation
04/21/2017	412-Widgets-012345	23	
04/21/2017	04-Fidgets-120341	66	
04/21/2017	204-Midgets-4421	31	
04/21/2017	593-Gidgets-402012	24	

Transform:

In this case, you must evaluate the `ProductName` column for two conditions. These conditional functions are the following:

```
IF(STARTSWITH(ProductName, '#{3}-`), 'Ok', 'Bad ProductName-Brand')
```

```
IF(ENDSWITH(ProductName, `-#{6}`), 'Ok', 'Bad ProductName-SKU')
```

One approach is to use the `derive` transform to create two new test columns and then use a `set` transform based on the evaluation of these two columns. However, using the following, you can compress the evaluation into a single step without creating the intermediate columns:

```
set col: Status value: IF(STARTSWITH(ProductName, '#{3}-`), IF(ENDSWITH(ProductName, `-#{6}`), 'Ok', 'Bad ProductName-SKU'), 'Bad ProductName-Brand')
```

Results:

InvDate	ProductName	Qty	Validation
04/21/2017	412-Widgets-012345	23	Ok
04/21/2017	04-Fidgets-120341	66	Bad ProductName-Brand
04/21/2017	204-Midgets-4421	31	Bad ProductName-SKU
04/21/2017	593-Gidgets-402012	24	Ok