

# Splitrows Transform

## Contents:

- *Basic Usage*
- *Syntax and Parameters*
  - *col*
  - *on*
  - *quote*
  - *quoteEscapeChar*
- *Examples*
  - *Example - splitrows with CSV data*

**NOTE:** Transforms are a part of the underlying language, which is not directly accessible to users. This content is maintained for reference purposes only. For more information on the user-accessible equivalent to transforms, see *Transformation Reference*.

Splits a column of values into separate rows of data based on the specified delimiter. You can split rows only on String literal values. Pattern-based row splitting is not supported.

**NOTE:** The `splitrows` transform must be the first one in your recipe. When a dataset is loaded for the first time in the Transformer page, a `splitrows` transform may added as the first step of the recipe. You cannot add another `splitrows` transform later in your recipe. For more information, see *Initial Parsing Steps*.

## Basic Usage

If you load CSV data into the Transformer page and then review the first recipe step in the Recipe panel, it might look like the following:

```
splitrows col: column1 on: '\r'
```

**Output:** The above splits all of the CSV data, which is stored as a comma-separated values in `column1` initially. The delimiter for the end of the row is a carriage return, which is indicated by the `\r` escaped value.

## Syntax and Parameters

```
splitrows col:column_ref on:'string_literal' [quote:'quoted_string']
```

Token	Required?	Data Type	Description
splitrows	Y	transform	Name of the transform
col	Y	string	Source column name
on	Y	string	Specifies the end of row delimiter for each value in the source column
quote	N	string	Specifies a quoted object that is omitted from pattern matching
quoteEscapeChar	N	string	Specifies the escape character that is used to precede quote marks.

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

## col

Identifies the column to which to apply the transform. You can specify only one column.

### Usage Notes:

Required?	Data Type
Yes	String (column name)

## on

Identifies the pattern to match. Pattern can be a string literal, Trifacta® pattern, or regular expression pattern.

**NOTE:** Value must be a string. For this transform, the parameter defines the string on which to split the current row and add the data after the string to the new row.

### Usage Notes:

Required?	Data Type
Yes	String literal

## quote

Can be used to specify a string as a single quoted object.

**NOTE:** This parameter value must be a single character.

```
splitrows col: MyCol on: '\r\n' quote: '''
```

**Output:** Splits the `MyCol` column into separate rows on the return-newline character string (`\r\n`). Values contained within double quotes (`"`) are treated as strings, even if they contain `\r\n` values.

### Usage Notes:

Required?	Data Type
No	String

## quoteEscapeChar

By default, the platform assumes the following characters are used to escape quote marks in text-based formats that use quotes to identify fields:

- **JSON:** Platform assumes that `\` is used.
- **All other file formats:** Platform assumes that `"` is used.

Optionally, you can specify the character that is used to escape quote marks in each recipe. Typically, this value is specified for processing JSON data or for customizing the transform for your specific data.

```
splitrows col: MyCol on: '\r\n' quote: ''' quoteEscapeChar:'''
```

## Usage Notes:

Required?	Data Type
No	String literal (single character)

## Examples

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

### Example - splitrows with CSV data

#### Unstructured source:

Before you import, your data in CSV format looks like the following:

```
Date,UserId,Message
3/14/16,jjones,"Hi, everyone!

Happy, St. Patrick's Day!"
3/14/16,lsmith,"@jjones, it's on 3/17."
3/14/16,thughes,lol
3/14/16,jjones,"@lsmith, no harm in celebrating twice!"
```

#### Notes:

- The Message value for the first row of data contains carriage returns, which must be captured in the data value and not used to split the row.
- The Message value for `thughes` is a single unquoted value.

#### Transformation:

When the data is first loaded into the Transformer page, the following step is added as the first step to the recipe:

<b>Transformation Name</b>	Split into rows
<b>Parameter: Column</b>	column1
<b>Parameter: Split on</b>	'\r'
<b>Parameter: Quote escape character</b>	'\"'

This transformation splits the unstructured CSV data on the carriage return. However, values that are stored between double quotes are treated as single strings, and no row breaks are applied to this data.

#### Results:

For CSV data, this step, a `split` step, and a `header` step are typically added automatically as the first steps of the recipe. In the Transformer page, this dataset should look like the following:

Date	UserId	Message
3/14/16	jjones	Hi, everyone! <sup>C</sup> <sub>R</sub> <sup>C</sup> <sub>R</sub> Happy, St. Patrick's Day!

3/14/16	lsmith	@jjones, it's on 3/17.
3/14/16	thughes	lol
3/14/16	jjones	@lsmith, no harm in celebrating twice!

The <sup>C</sup><sub>R</sub> marker is used to indicate a carriage return in the data.