### parsecsv

Parses the comma-separated values (CSV) string or tab-separated values (TSV) string.

#### Syntax

parsecsv [field=TARGET\_FIELD] [overlay=BOOL] [strict=BOOL] [tab=BOOL] [FIELD, ...]

Optional Parameter

**field=TARGET\_FIELD**

Field to be parsed in the input data stream (default: line).

**overlay=BOOL**

Option to control data-overwrite (default: f).

1. t: Overwrites the input data with the parsed data.
2. f: Outputs the parsed data only.

**strict=BOOL**

Option to conform RFC 4180 (default: f).

1. t: Parses the CSV file by strictly applying the RFC 4180: <https://tools.ietf.org/html/rfc4180>. It is the same as when you open the CSV file with Excel. This option cannot be used when tab=t.
2. f: Flexibly parses the CSV file.

**tab=BOOL**

Option to use a tab character as a separator (default: f).

1. t: Uses a tab character as a separator. This is useful for processing tab-separated values (TSV) files.
2. f: Uses a comma (,) as a separator.

Target Object

Target Objects

**FIELD, ...**

Field names for the parsed fields, separated by a comma (,). This option must be placed last in the expression. If names are not provided, default names are used: column0, column1, ..., columnN in order.

#### Usage

Parse a comma-separated text

json "{line: '\"foo\",\"bar\"'}" | parsecsv

Parse a comma-separated text to give names name1 and name2 to fields in order from the front.

json "{line: '\"foo\",\"bar\"'}" | parsecsv name1,name2