CONVERTTIMEZONE Function

Contents:

Converts Datetime value in specified time zone to corresponding value second specified time zone. Input can be a column of Datetime values, a literal Datetime value, or a function returning Datetime values.

- Inputs with time zone offsets are invalid.
- Specified time zone must be a string literal of one of the support time zone values. For more information, see *Supported Time Zone Values*.

Wrangle vs. SQL: This function is part of Wrangle , a proprietary data transformation language. Wrangle is not SQL. For more information, see *Wrangle Language*.

Basic Usage

Column reference values:

```
converttimezone(myTimestamp,'US/Mountain','US/Pacific')
```

Output: Returns the UTC values of the `myTimestamp` converted from US Mountain time zone to US Pacific time zone.

Syntax and Arguments

```
converttimezone(date, 'enum-timezone1','enum-timezone2')
```

Argument	Required?	Data Type	Description
date	Y	datetime	Name of Datetime column, Datetime literal, or function returning a Datetime value.
enum-timezone-string1, enum-timezone-string2	Y	string	Case-sensitive string literal value corresponding to the source or target time zone.

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

date

Name of a column containing Datetime values, a literal Datetime value, or a function returning Datetime values to convert.

> **Tip:** Use the DATEFORMAT function to wrap values into acceptable formats. See *DATEFORMAT Function*.

If an input value is invalid for Datetime data type, a null value is returned.

- Column references with time zone offsets are invalid.
- Missing values for this function in the source data result in missing values in the output.
- Multiple columns and wildcards are not supported.

Usage Notes:

Required?	Data Type	Example Value
Yes	Datetime (column reference, function, or literal)	`sourceTime`

enum-timezone-string1, enum-timezone-string2

String literal value for the time zone 1) to convert from and 2) to convert to.

> **NOTE:** These values are case-sensitive.

Example values:

```
'America/Puerto_Rico'
'US/Eastern'
'US/Central'
'US/Mountain'
'US/Pacific'
'US/Alaska'
'US/Hawaii'
```

For more information on supported time formatting strings, see *Supported Data Types*.

Examples

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

Example - Time zone conversion

This example shows how you can use the following functions to convert Datetime values to different time zones.

- `CONVERTFROMUTC` - Converts valid Datetime values from UTC time zone to a specified time zone. See *CONVERTFROMUTC Function*.
- `CONVERTTOUTC` - Converts valid Datetime values from a specified time zone to UTC time zone. See *CONVERTTOUTC Function*.
- `CONVERTTIMEZONE` - Converts valid Datetime values from one time zone to another. See *CONVERTTIMEZONE Function*.

Source:

row	datetime
1	2020-03-15
2	2020-03-15 0:00:00
3	2020-03-15 +08:00

4	2020-03-15 1:02:03
5	2020-03-15 4:02:03
6	2020-03-15 8:02:03
7	2020-03-15 12:02:03
8	2020-03-15 16:02:03
9	2020-03-15 20:02:03
10	2020-03-15 23:02:03

Transformation:

When you import the above dates, Trifacta Wrangler Enterprise may not recognize the column as a set of dates. You can use the column menus to format the date values to the following standardized format:

```
yyyy*mm*dd*HH:MM:SS
```

Transformation Name	Change column data type
Parameter: Columns	datetime
Parameter: New type	Date/Time
Parameter: Date/Time type	yyyy*mm*dd*HH:MM:SS

When the type has been changed, row 1 and row 3 have been identified as invalid. You can use the following transformation to remove these rows:

Transformation Name	Filter rows
Parameter: Condition	Custom formula
Parameter: Type of formula	Custom single
Parameter: Condition	ISMISMATCHED(datetime, ['Datetime','yy-mm-dd hh:mm:ss','yyyy*mm*dd*HH:MM:SS'])
Parameter: Action	Delete matching rows

When the Datetime values are consistently formatted, you can use the following transformations to perform conversions. The following tranformation converts the values from UTC to US/Eastern time zone:

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	CONVERTFROMUTC(datetime, 'US\/Eastern')
Parameter: New column name	'datetimeUTC2Eastern'

This transformation now assumes that the date values are in US/Pacific time zone and converts them to UTC:

Transformation Name	`New formula`
Parameter: Formula type	`Single row formula`
Parameter: Formula	`CONVERTTOUTC(datetime, 'US\/Pacific')`
Parameter: New column name	`'datetimePacific2UTC'`

The final transformation converts the date time values between arbitrary time zones. In this case, the values are assumed to be in US/Alaska time zone and are converted to US/Hawaii time zone:

Transformation Name	`New formula`
Parameter: Formula type	`Single row formula`
Parameter: Formula	`CONVERTTIMEZONE(datetime, 'US\/Alaska', 'US\/Hawaii')`
Parameter: New column name	`'datetimeAlaska2Hawaii'`

Results:

row	datetime	datetimeAlaska2Hawaii	datetimePacific2UTC	datetimeUTC2Eastern
2	2020-03-15 00:00:00	2020-03-14 22:00:00	2020-03-15 07:00:00	2020-03-14 20:00:00
4	2020-03-15 01:02:03	2020-03-14 23:02:03	2020-03-15 08:02:03	2020-03-14 21:02:03
5	2020-03-15 04:02:03	2020-03-15 02:02:03	2020-03-15 11:02:03	2020-03-15 00:02:03
6	2020-03-15 08:02:03	2020-03-15 06:02:03	2020-03-15 15:02:03	2020-03-15 04:02:03
7	2020-03-15 12:02:03	2020-03-15 10:02:03	2020-03-15 19:02:03	2020-03-15 08:02:03
8	2020-03-15 16:02:03	2020-03-15 14:02:03	2020-03-15 23:02:03	2020-03-15 12:02:03
9	2020-03-15 20:02:03	2020-03-15 18:02:03	2020-03-16 03:02:03	2020-03-15 16:02:03
10	2020-03-15 23:02:03	2020-03-15 21:02:03	2020-03-16 06:02:03	2020-03-15 19:02:03