

# Compare Values

## Contents:

- *Compare Numeric Values*
- *Compare Boolean Values*
- *Compare Date Values*
- *Compare String Values*

Depending on the data type, you can compare values in separate columns or single columns against fixed values.

## Compare Numeric Values

You can use basic comparison operators to perform comparisons on your data. In this example, the `compareCol` column is generated as the evaluation of `3 < 6`, which is `true`:

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	(3 < 6)
<b>Parameter: New column name</b>	'compareCol'

For more information, see *Comparison Operators*.

## Compare Boolean Values

Boolean values can be `true` or `false`, so comparisons like the following can be applied to a Boolean set of values:

<b>Transformation Name</b>	Edit column with formula
<b>Parameter: Columns</b>	Attendance
<b>Parameter: Formula</b>	IF(isSeated == true,true,Attendance)

In the above case, the value in `Attendance` is set to `true` if the value in the `isSeated` column is `true`. Otherwise, the current value in `Attendance` is used.

## Compare Date Values

You can use the `DATEDIF` function to compare two date values, as in the following, which compares the number of days between `startCol` and `endCol` values:

**NOTE:** Both parameters of the `DATEDIF` function must be column references containing valid date values.

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula

<b>Parameter: Formula</b>	DATEDIF(startCol, endCol, 'day')
<b>Parameter: New column name</b>	'DurationInDays'

See *DATEDIF Function*.

## Compare String Values

See *Compare Strings*.