

KTHLARGESTDATEIF Function

Contents:

- *Basic Usage*
- *Syntax and Arguments*
 - *col_ref*
 - *k_integer*
 - *test_expression*
- *Examples*
 - *Example - KTHLARGESTDATE functions*

Extracts the ranked Datetime value from the values in a column, where $k=1$ returns the maximum value, when a specified condition is met. The value for k must be between 1 and 1000, inclusive. Inputs must be Datetime.

KTHLARGESTDATEIF calculations are filtered by a conditional applied to the group.

For purposes of this calculation, two instances of the same value are treated as the same value of k . So, if your dataset contains three rows with column values 2020-02-15, 2020-02-14, and 2020-02-14, then KTHLARGESTDATEIF returns 2020-02-14 for $k=2$ and 2020-02-14 for $k=3$.

Input column must be of Datetime type. Other values column are ignored. If a row contains a missing or null value, it is not factored into the calculation.

NOTE: When added to a transformation, this function is applied to the current sample. If you change your sample or run the job, the computed values for this function are updated. Transformations that change the number of rows in subsequent recipe steps do not affect the values computed for this step.

To perform a simple k th largest calculation on Datetime values without conditionals, use the KTHLARGESTDATE function. See *KTHLARGESTDATE Function*.

For a version of this function that applies to non-Datetime values, see *KTHLARGESTIF Function*.

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

```
kthlargestdateif(transDate, 2, salesPerson == 'jsmith')
```

Output: Returns the secondmost recent Date (rank=2) from the transDate column when the salesPerson value is jsmith.

Syntax and Arguments

```
kthlargestdateif(col_ref, limit, test_expression) [group:group_col_ref] [limit:limit_count]
```

Argument	Required?	Data Type	Description
col_ref	Y	string	Reference to the column you wish to evaluate.

k_integer	Y	integer	The ranking of the value to extract from the source column
test_expression	Y	string	Expression that is evaluated. Must resolve to true or false

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

For more information on the `group` and `limit` parameter, see *Pivot Transform*.

col_ref

Name of the column whose values you wish to use in the calculation. Inputs must be Datetime values.

Usage Notes:

Required?	Data Type	Example Value
Yes	String that corresponds to the name of the column	transactionDate

k_integer

Integer representing the ranking of the value to extract from the source column.

NOTE: The value for `k` must be an integer between 1 and 1,000 inclusive.

- `k=1` represents the maximum value in the column.
- If `k` is greater than or equal to the number of values in the column, the minimum value is returned.
- Missing and null values are not factored into the ranking of `k`.

test_expression

This parameter contains the expression to evaluate. This expression must resolve to a Boolean (`true` or `false`) value.

Usage Notes:

Required?	Data Type	Example Value
Yes	String expression that evaluates to true or false	(LastName == 'Mouse' && FirstName == 'Mickey')

Examples

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

Example - KTHLARGESTDATE functions

This example illustrates how you can apply conditionals to calculate minimum, maximum, and most common date values:

- `KTHLARGESTDATE` - Extracts the ranked Datetime value from the values in a column, where `k=1` returns the maximum value. See *KTHLARGESTDATE Function*.
- `KTHLARGESTUNIQUEDATE` - Extracts the unique ranked Datetime value from the values in a column, where `k=1` returns the maximum value. See *KTHLARGESTUNIQUEDATE Function*.

- `KTHLARGESTDATEIF` - Extracts the ranked Datetime value from the values in a column that meet a specified condition. See *KTHLARGESTDATEIF Function*.
- `KTHLARGESTUNIQUEDATEIF` - Extracts the ranked unique Datetime value from the values in a column that meet a specified condition. See *KTHLARGESTUNIQUEDATEIF Function*.

Source:

Here is some example transaction data:

Date	Product	Units	UnitCost	OrderValue
3/28/2020	ProductA	4	10.00	40.00
3/8/2020	ProductB	4	20.00	80.00
3/12/2020	ProductC	2	30.00	60.00
3/23/2020	ProductA	1	10.00	10.00
3/20/2020	ProductB	2	20.00	40.00
3/12/2020	ProductC	9	30.00	270.00
3/28/2020	ProductA	5	10.00	50.00
3/23/2020	ProductB	8	20.00	160.00
3/16/2020	ProductC	9	30.00	270.00
3/8/2020	ProductA	5	10.00	50.00
3/10/2020	ProductB	3	20.00	60.00
3/13/2020	ProductC	1	30.00	30.00
3/12/2020	ProductA	7	10.00	70.00
3/10/2020	ProductB	7	20.00	140.00
3/24/2020	ProductC	9	30.00	270.00
3/15/2020	ProductA	8	10.00	80.00
3/10/2020	ProductB	5	20.00	100.00
3/10/2020	ProductC	4	30.00	120.00

Transformation:

The following transformation computes the third highest date in the `Date` column:

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	<code>kthlargestdate(Date, 3)</code>
Parameter: New column name	'kthlargestdate'

This transformation computes the third highest unique value in the `Date` column:

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	<code>kthlargestuniquedate(Date, 3)</code>

Parameter: New column name	'kthlargestuniquedate'
-----------------------------------	------------------------

Following transformation calculates the 3rd highest date value when the OrderValue > 200:

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	kthlargestdateif(Date, 3, OrderValue > 200)
Parameter: New column name	'kthlargestdateif'

Following transformation calculates the 3rd highest unique date value when the OrderValue > 200:

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	kthlargestuniquedateif(Date, 3, OrderValue > 200)
Parameter: New column name	'kthlargestuniquedateif'

Results:

Date	Product	Units	UnitCost	OrderValue	kthlargestdate	kthlargestuniquedate	kthlargestdateif	kthlargestu
3/28/2020	ProductA	4	10.00	40.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/8/2020	ProductB	4	20.00	80.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/12/2020	ProductC	2	30.00	60.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/23/2020	ProductA	1	10.00	10.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/20/2020	ProductB	2	20.00	40.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/12/2020	ProductC	9	30.00	270.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/28/2020	ProductA	5	10.00	50.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/23/2020	ProductB	8	20.00	160.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/16/2020	ProductC	9	30.00	270.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/8/2020	ProductA	5	10.00	50.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/10/2020	ProductB	3	20.00	60.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/13/2020	ProductC	1	30.00	30.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/12/2020	ProductA	7	10.00	70.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/10/2020	ProductB	7	20.00	140.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020

3/24 /2020	ProductC	9	30.00	270.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/15 /2020	ProductA	8	10.00	80.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/10 /2020	ProductB	5	20.00	100.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020
3/10 /2020	ProductC	4	30.00	120.00	03-24-2020	03-23-2020	03-23-2020	03-23-2020