

KTHLARGESTUNIQUEIF Function

Contents:

- *Basic Usage*
- *Syntax and Arguments*
 - *col_ref*
 - *k_integer*
 - *test_expression*
- *Examples*
 - *Example - Second-most measurements for a specific city*

Extracts the ranked unique 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 can be Integer, Decimal, or Datetime.

KTHLARGESTUNIQUEIF 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 four rows with column values 10 , 9 , 9 , and 8, the the function returns 9 for $k=2$ and 8 for $k=3$.

Input column can be of Integer, Decimal or 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 unique calculation without conditionals, use the KTHLARGESTUNIQUE function. See *KTHLARGESTUNIQUE 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

```
kthlargestuniqueif(POS_Sales, 2, DayOfWeek == 'Saturday')
```

Output: Returns the secondmost value (rank=2) from the POS_Sales column when the DayOfWeek value is Saturday.

Syntax and Arguments

```
kthlargestuniqueif(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 Integer, Decimal, or Datetime values.

NOTE: If the input is in Datetime type, the output is in unixtime format. You can wrap these outputs in the DATEFORMAT function to output the results in the appropriate Datetime format. See *DATEFORMAT Function*.

Usage Notes:

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

k_integer

Integer representing the unique 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 - Second-most measurements for a specific city

This example illustrates how to use the conditional ranking functions.

Functions:

Item	Description
KTHLARGESTIF Function	Extracts the ranked 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 can be Integer, Decimal, or Datetime.
KTHLARGESTUNIQUEIF Function	Extracts the ranked unique 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 can be Integer, Decimal, or Datetime.

Source:

Here is some example weather data:

date	city	rain_cm	temp_C	wind_mph
1/23/17	Valleyville	0.00	12.8	8.8
1/23/17	Center Town	0.31	9.4	5.3
1/23/17	Magic Mountain	0.00	0.0	7.3
1/24/17	Valleyville	0.25	17.2	3.3
1/24/17	Center Town	0.54	1.1	7.6
1/24/17	Magic Mountain	0.32	5.0	8.8
1/25/17	Valleyville	0.02	3.3	6.8
1/25/17	Center Town	0.83	3.3	5.1
1/25/17	Magic Mountain	0.59	-1.7	6.4
1/26/17	Valleyville	1.08	15.0	4.2
1/26/17	Center Town	0.96	6.1	7.6
1/26/17	Magic Mountain	0.77	-3.9	3.0
1/27/17	Valleyville	1.00	7.2	2.8
1/27/17	Center Town	1.32	20.0	0.2
1/27/17	Magic Mountain	0.77	5.6	5.2
1/28/17	Valleyville	0.12	-6.1	5.1
1/28/17	Center Town	0.14	5.0	4.9
1/28/17	Magic Mountain	1.50	1.1	0.4
1/29/17	Valleyville	0.36	13.3	7.3
1/29/17	Center Town	0.75	6.1	9.0
1/29/17	Magic Mountain	0.60	3.3	6.0

Transformation:

In this case, you want to find out the second-most measures for rain, temperature, and wind in Center Town for the week.

Transformation Name	Pivot columns
Parameter: Values	KTHLARGESTIF(rain_cm,2,city == 'Center Town')

Parameter: Max number of columns to create	1
---	---

You can see in the preview that the value is 1.32. Before adding it to your recipe, you change the step to the following:

Transformation Name	Pivot columns
Parameter: Values	<code>KTHLARGESTIF(temp_C,2,city == 'Center Town')</code>
Parameter: Max number of columns to create	1

The value is 20.

For wind, you modify it to be the following, capturing the third-ranked value:

Transformation Name	Pivot columns
Parameter: Values	<code>KTHLARGESTIF(wind_mph,3,city == 'Center Town')</code>
Parameter: Max number of columns to create	1

In the results, you notice that there are two values for 8.8. So you change the function to use the `KTHLARGESTUNIQUEIF` function instead:

Transformation Name	Pivot columns
Parameter: Values	<code>KTHLARGESTUNIQUEIF(wind_mph,3,city == 'Center Town')</code>
Parameter: Max number of columns to create	1

The result value is 7.6. Note that this value appears twice, so if you change the rank parameter in the above transformation to 4, the results would return a different unique ranked value (7.3).

Results:

You can choose to add any of these steps to generate an aggregated result. As an alternative, you can use a `derive` transform to insert these calculated results into new columns.