

REMOVESYMBOLS Function

Removes all characters from a string that are not letters, numbers, accented Latin characters, or whitespace.

NOTE: Non-Latin letters are also removed.

Tip: This function also removes common punctuation, such as the following:

```
. , ! & ?
```

To preserve these characters, you might replace them with an alphanumeric text string. For example, the question mark might be replaced by:

```
zzQUESTIONMARKzz
```

After the function has been applied, you can replace these strings with the original 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 example:

```
removesymbols(MyName)
```

Output: Returns the value in the MyName column value with all non-alphanumeric characters removed.

String literal example:

```
removesymbols('Héllö, Wörlds!?!?')
```

Output: Returns the string: Héllö Wörlds.

Wildcard example:

```
removesymbols($col)
```

Output: Strips all non-alphanumeric or space characters from all columns in the dataset.

Syntax and Arguments

```
removesymbols(column_string)
```

| Argument | Required? | Data Type | Description |
|---------------|-----------|-----------|--|
| column_string | Y | string | Name of the column or string literal to be applied to the function |

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

column_string

Name of the column or string constant to be trimmed of symbols.

- Missing string or column values generate missing string results.
- String constants must be quoted ('Hello, World').
- Multiple columns and wildcards are not supported.

Usage Notes:

| Required? | Data Type | Example Value |
|-----------|------------------------------------|---------------|
| Yes | String literal or column reference | myColumn |

Examples

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

Example - String cleanup functions together

The following example demonstrates functions that can be used to clean up strings. These functions include the following:

- TRIM - remove leading and trailing whitespace. See *TRIM Function*.
- REMOVEWHITESPACE - remove leading and trailing whitespace and all whitespace in between. See *REMOVEWHITESPACE Function*.
- REMOVESYMBOLS - remove all characters that are not alpha-numeric or whitespace. See *REMOVESYMBOLS Function*.

Source:

In the following (*space*) and (*tab*) indicate space keys and tabs, respectively. Carriage return and newline characters are also supported by whitespace functions.

| Strings | source |
|----------|-------------------------------------|
| String01 | this source(space)(space) |
| String02 | (tab)(tab)this source |
| String03 | (tab)(tab)this source(space)(space) |
| String04 | this source's? |
| String05 | Why, you @\$%^&*()! |
| String06 | this sörce |
| String07 | (space)this sörce |
| String08 | à mañana |

Transformation:

The following transformation steps generate new columns using each of the string cleanup functions:

| | |
|-----------------------------------|--------------------|
| Transformation Name | New formula |
| Parameter: Formula type | Single row formula |
| Parameter: Formula | TRIM(source) |
| Parameter: New column name | 'trim_source' |

| | |
|-----------------------------------|---------------------------|
| Transformation Name | New formula |
| Parameter: Formula type | Single row formula |
| Parameter: Formula | REMOVEWHITESPACE(source) |
| Parameter: New column name | 'removewhitespace_source' |

| | |
|-----------------------------------|------------------------|
| Transformation Name | New formula |
| Parameter: Formula type | Single row formula |
| Parameter: Formula | REMOVESYMBOLS(source) |
| Parameter: New column name | 'removesymbols_source' |

Results:

| Strings | source | removesymbols_source | removewhitespace_source | trim_source |
|----------|-------------------------------------|-------------------------------------|-------------------------|---------------------|
| String01 | this source(space)(space) | this source(space)(space) | thissource | this source |
| String02 | (tab)(tab)this source | (tab)(tab)this source | thissource | this source |
| String03 | (tab)(tab)this source(space)(space) | (tab)(tab)this source(space)(space) | thissource | this source |
| String04 | this source's? | this sources | thissource's? | this source's? |
| String05 | "Why, you @\$%^&*()!" | Why you | Why,you@\$%^&*()! | Why, you @\$%^&*()! |
| String06 | this sörce | this sörce | thissörce | this sörce |
| String07 | (space)this sörce | (space)this sörce | thissörce | this sörce |
| String08 | à mañana | à mañana | àmañana | à mañana |