

Package ‘infuser’

October 4, 2017

Type Package

Title A Very Basic Templating Engine

Version 0.2.8

Date 2017-10-03

Author Bart Smeets

Maintainer Bart Smeets <bartsmeeets86@gmail.com>

Description This light-weight templating package provides flexible and user-friendly value substitution in string templates or template files (e.g. SQL files) at runtime.

License MIT + file LICENSE

LazyData TRUE

Suggests testthat, knitr, rmarkdown, dbplyr, magrittr

URL <https://github.com/bart6114/infuser/>

BugReports <https://github.com/bart6114/infuser/issues>

RoxygenNote 6.0.1

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2017-10-04 11:57:42 UTC

R topics documented:

infuse	2
print.infuse	3
print_requested_params	3
read_template	4
trim	4
variables_requested	4

Index	6
--------------	----------

 infuse

Infuse a template with values.

Description

For more info and usage examples see the README on the [infuser github page](#). To help prevent **SQL injection attacks** (or other injection attacks), use a transformation function to escape special characters and provide it through the `transform_function` argument. `build_sql` is a great default escaping function for SQL templating. For templating in other languages you will need to build/specify your own escaping function.

Usage

```
infuse(file_or_string, ...,
       variable_identifier = getOption("variable_identifier"),
       default_char = "|", collapse_char = ",",
       transform_function = function(value) return(value),
       verbose = getOption("verbose"), simple_character = FALSE,
       strict = FALSE)
```

Arguments

<code>file_or_string</code>	the template file or a character string containing the template
<code>...</code>	different keys with related values, used to fill in the template (if first passed item is a list/environment the contents of this will be processed instead)
<code>variable_identifier</code>	the opening and closing character that denounce a variable in the template, defaults to <code>c("{", "}")</code> and can be set persistently using e.g. <code>options(variable_identifier=c("{", "})")</code>
<code>default_char</code>	the character use to specify a default after
<code>collapse_char</code>	the character used to collapse a supplied vector
<code>transform_function</code>	a function through which all specified values are passed, can be used to make inputs safe(r). <code>dplyr::build_sql</code> is a good default for SQL templating.
<code>verbose</code>	verbosity level
<code>simple_character</code>	if TRUE returns only a character vector, else adds the <code>infuser</code> class to the returned object.
<code>strict</code>	if TRUE stops processing when a requested parameter is not supplied, else will simply leave the parameter as-is

<code>print.infuse</code>	<i>prints/shows the result of the <code>infuse</code> function using the <code>cat</code> function</i>
---------------------------	--

Description

prints/shows the result of the `infuse` function using the `cat` function

Usage

```
## S3 method for class 'infuse'  
print(x, ...)
```

Arguments

<code>x</code>	output of the <code>infuse</code> function
<code>...</code>	further arguments passed to or from other methods.

<code>print_requested_params</code>	<i>Prints requested parameters</i>
-------------------------------------	------------------------------------

Description

Prints requested parameters

Usage

```
print_requested_params(params_requested)
```

Arguments

<code>params_requested</code>	the list of requested parameters
-------------------------------	----------------------------------

read_template	<i>Reads the template from a file or string</i>
---------------	---

Description

Reads the template from a file or string

Usage

```
read_template(file_or_string)
```

Arguments

file_or_string the file or character string to read

trim	<i>Returns string w/o leading or trailing whitespace</i>
------	--

Description

Returns string w/o leading or trailing whitespace

Usage

```
trim(x)
```

Arguments

x to string to strip

variables_requested	<i>Shows which variables are requested by the template</i>
---------------------	--

Description

Shows which variables are requested by the template

Usage

```
variables_requested(file_or_string, variable_identifier = c("{", "}"),
  default_char = "|", verbose = FALSE)
```

Arguments

`file_or_string` the template file or a string containing the template
`variable_identifier`
the opening and closing character that denounce a variable in the template
`default_char` the character use to specify a default after
`verbose` verbosity level

Index

`build_sql`, 2

`infuse`, 2

`print.infuse`, 3

`print_requested_params`, 3

`read_template`, 4

`trim`, 4

`variables_requested`, 4