Package ‘templates’

March 15, 2024

Title A System for Working with Templates
Version 0.4.0
Description Provides tools to work with template code and text in R. It aims to provide a simple substitution mechanism for R-expressions inside these templates. Templates can be written in other languages like 'SQL', can simply be represented by characters in R, or can themselves be R-expressions or functions.
Depends R (>= 3.4.0)
Imports stringr, magrittr
License MIT + file LICENSE
Encoding UTF-8
Suggests testthat, knitr, rmarkdown, covr
RoxygenNote 6.1.1
VignetteBuilder knitr
NeedsCompilation no
Author Sebastian Warnholz [aut, cre]
Maintainer Sebastian Warnholz <wahani@gmail.com>
Repository CRAN
Date/Publication 2024-03-15 11:50:02 UTC

R topics documented:

  tmpl ................................................................. 2
tmplUpdate ........................................................... 3

Index 4

1
tmpl

Template constructors

Description
tmpl is the constructor function for template objects.

Usage
tmpl(.t, ...)

## S3 method for class 'character'
tmpl(.t, ..., .envir = parent.frame())

## S3 method for class 'formula'
tmpl(.t, ...)

## S3 method for class 'tmpl'
tmpl(.t, ...)

## S3 method for class 'function'
tmpl(.t, ...)

Arguments
.t something that can be interpreted as template. See defined methods for options.
...
.envir (environment) the environment in which template snippets are evaluated. For formulas and functions their environment is used.

Details
Objects of class tmpl are stored as a character of length one. They can contain 'snippets' to be evaluated. These snippets are identified by an opening {{ and closing }}. The environment in which they are evaluated is stored in the object. They can be further augmented by supplying arguments in ... .

See Also
tmplUpdate, tmplEval

Examples
tmpl("Hi {{ toupper(a) }}!", a = "there")
tmpl(~ {y <- {{ a }}}, a ~ x + 1)
tmpl(function(x) {{ a }} + x, a = 1)
### tmplUpdate

**Update and evaluate templates**

**Description**

Functions operating on `tmpl` objects. They can be updated and / or evaluated as an expression.

**Usage**

```r
tmplUpdate(.t, ...)  
## S3 method for class 'tmpl'  
tmplUpdate(.t, ...)  
## S3 method for class 'function'  
tmplUpdate(.t, ...)  

tmplEval(.t, ..., .envir = new.env(parent = parent.frame()))  

tmplAsFun(.t, ...)  
```

**Arguments**

- `.t`  
  (tmpl) and object of class `tmpl`
- `...`  
  (name = value | name ~ value) name-value expressions used to update the snippets in `x`
- `.envir`  
  (environment) the environment in which the template is evaluated

**Details**

`tmplUpdate` will evaluate all snippets in a template. Objects are searched for in the list of arguments supplied as `...` and the environment of the template. The results are substituted within the snippets.

`tmplEval` will evaluate the template in place or in the specified environment after substituting the elements in `...`

**Examples**

```r  
tmpl("This is {{ a }} very similar to {{ b }}", a = "actually", b = "sprintf")  
tmpl("But I consider it to be ({{ sprintf('%i', a) }}) orthogonal", a = 1.0)  
tmpl("and ({{ sprintf('%i', b) }}) with a different scope:", b = 2.0)  
tmpl("SELECT {{ var }} FROM {{ table }} WHERE {{ condition }};",  
  var = "someVar", table = "someTable", condition = "primaryKey = 1")  
template <- tmpl("cat({{ toupper(x) }})\n")  
tmplUpdate(template, x ~ "hi")  
tmplEval(template, x ~ "hi")
```
Index

tmpl, 2, 3
tmplAsFun (tmplUpdate), 3
tmplEval, 2
tmplEval (tmplUpdate), 3
tmplUpdate, 2, 3