Package 'listcomp'

May 24, 2020

Title List Comprehensions

Version 0.3.0

Description An implementation of list comprehensions as purely syntactic sugar with a minor runtime overhead. It constructs nested for-loops and executes the byte-compiled loops to collect the results.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Suggests testthat

Imports rlang, compiler, digest

URL https://github.com/dirkschumacher/listcomp

BugReports https://github.com/dirkschumacher/listcomp/issues

NeedsCompilation no

Author Dirk Schumacher [aut, cre, cph]

Maintainer Dirk Schumacher <mail@dirk-schumacher.net>

Repository CRAN

Date/Publication 2020-05-24 07:30:02 UTC

R topics documented:

gen_list *List comprehensions*

Description

Create lists of elements using an expressive syntax. Internally nested for-loops are created and compiled that generate the list.

Usage

```
gen_list(element_expr, ..., .compile = TRUE)
```

Arguments

element_expr	an expression that will be collected
...	either a logical expression that returns a length 1 result. A named list of equal length sequences that are iterated over in parallel or a named parameter with an iterable sequence.
.compile	compile the resulting for loop to bytecode befor eval

Details

For parallel iterations all elements in the list need to be of equal length. This is not checked at runtime at the moment.

Value

A list of all generated values. The element-type is determined by the parameter element_expr.

Examples

```
gen_list(c(x, y), x = 1:10, y = 1:10, x + y == 10, x < y)
z <- 10
gen_list(c(x, y), x = 1:10, y = 1:10, x + y == !!z, x < y)

# it is also possible to iterate in parallel by passing a list of
# sequences
gen_list(c(x, y), list(x = 1:10, y = 1:10), (x + y) %in% c(4, 6))
```

Index