

Package: rtern (via r-universe)

August 20, 2024

Title A Ternary Conditional Operator for R

Version 0.1.2

Description A small language extension for succinct conditional assignment using `?` and `:`, emulating the conditional ternary operator syntax using in C, Java, JavaScript and other languages.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

URL <https://github.com/grddavies/rtern>

BugReports <https://github.com/grddavies/rtern/issues>

Suggests covr, testthat (>= 3.0.0), spelling, lifecycle

Config/testthat/edition 3

Language en-US

Imports utils, rlang

Repository <https://grddavies.r-universe.dev>

RemoteUrl <https://github.com/grddavies/rtern>

RemoteRef HEAD

RemoteSha d7443c68d0b9b31fd708f7c839e0fce3aef4d864

Contents

?	2
Index	4

?

*Succinct conditional evaluation and assignment***Description****[Experimental]**

? is an in-line if/else operator

Usage

```
lhs ? rhs
```

Arguments

lhs	A logical expression, vector or matrix.
rhs	A pair of values separated by a colon i.e. value_if_true : value_if_false.

Details

The syntax for ? is as follows:

```
condition ? value_if_true : value_if_false
```

The condition is evaluated TRUE or FALSE as a Boolean expression. On the basis of the evaluation of the Boolean condition, the entire expression returns value_if_true if condition is true, but value_if_false otherwise. In the case where the condition is a vector/matrix of Boolean values, the function returns a vector/matrix where each element is either value_if_true or value_if_false based on the truthiness of the elements of the object on the left-hand side. In these cases the behaviour of ? mimics [ifelse](#).

Who has time for if/else?

Value

One of the values in rhs, depending on the truthiness of lhs.

Examples

```
# Conditional evaluation
4 > 3 ? "it_was_true":"it_was_false"
# > "it_was_true"

FALSE ? "it_was_true":"it_was_false"
# > "it_was_false"

# Vectorised evaluation
c(4, 2) < 3 ? "it_was_true":"it_was_false"
# > "it_was_false" "it_was_true"

# Conditional assignment with `<-`
```

```
x <- 4 > 3 ? "it_was_true":"it_was_false"
x
# > "it_was_true"

# Conditional assignment with `=`
y <- 3 > 4 ? "it_was_true":"it_was_false"
y
# > "it_was_false"

# Chaining `?` statements
z <- FALSE ? "true":(FALSE ? "false,true):(TRUE ? "false,false,true":"all false")
z
# > "false,false,true"
```

Index

?, 2

ifelse, 2