

Package ‘slap’

April 24, 2024

Title Simplified Error Handling

Version 2024.4.1

Description Alternative to using `withCallingHandlers()` in the simple case of catch and rethrow. The ``%!%`` operator evaluates the expression on its left hand side, and if an error occurs, the right hand side is used to construct a new error that embeds the original error.

License MIT + file LICENSE

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.1

Imports cli, rlang

URL <https://github.com/tadascience/slap>, <https://slap.tada.science>

BugReports <https://github.com/tadascience/slap/issues>

NeedsCompilation no

Author Romain François [aut, cre],
tada.science [cph, fnd]

Maintainer Romain François <romain@tada.science>

Repository CRAN

Date/Publication 2024-04-24 14:00:02 UTC

R topics documented:

slap	2
Index	4

slap

Slap Operator

Description

Slap Operator

Usage

```
expr %!% message
```

```
expr %!!% message
```

Arguments

expr An expression or quosure to evaluate carefully

message A message meant to be formatted by `cli::cli_bullets()` or a function.

Value

If `expr` succeeds, its result is returned.

When `expr` generates an error, the `%!%` and `%!!%` operators catch it and embed it in a new error thrown by `cli::cli_abort()`.

If `message` evaluates to a character vector, it is used as the `message` argument of `cli::cli_abort()`.

If `message` evaluates to a function, the function is called with one argument: the caught error from evaluating `expr`.

When the current environment has an `error_call` object, it is used as the `call` argument of `cli::cli_abort()`.

Examples

```
# g() throws an error
g <- function() {
  stop("ouch")
}

# h() catches that error and embed it in a new error
# with "bam" as its message, the g() error as the parent error,
# and the caller environment as call=
h <- function(error_call = rlang::caller_env()) {
  g() %!% "bam"
}

# f() will be used as the error call
f <- function() {
  h()
}
```

```
# Error in `f()`:  
# ! bam  
# Caused by error in `g()`:  
# ! ouch  
tryCatch(f(), error = function(err) {  
  print(err, backtrace = FALSE)  
})
```

Index

%!%(slap), 2

%%(slap), 2

cli::cli_abort(), 2

cli::cli_bullets(), 2

slap, 2