Package ‘decor’

July 1, 2023

Title Retrieve Code Decorations

Version 1.0.2

Description Retrieves code comment decorations for C++
languages of the form '\\ [[xyz]]', which are used for automated
wrapping of C++ functions.

License MIT + file LICENSE

Depends R (>= 3.3.0)

Imports tibble, utils, vctrs (>= 0.5.0)

Suggests covr, testthat

Encoding UTF-8

RoxygenNote 7.2.3

URL `https://github.com/r-lib/decor`

BugReports `https://github.com/r-lib/decor/issues`

NeedsCompilation yes

Author Davis Vaughan [aut, cre] (<`https://orcid.org/0000-0003-4777-038X`>),
Romain François [aut] (<`https://orcid.org/0000-0002-2444-4226`>),
Jim Hester [aut] (<`https://orcid.org/0000-0002-2739-7082`>),
Posit Software, PBC [cph, fnd]

Maintainer Davis Vaughan `<davis@posit.co>`

Repository CRAN

Date/Publication 2023-07-01 18:30:02 UTC

R topics documented:

cpp_decorations *Decorations in a ‘C++‘ file*

Description

Decorations in a ‘C++‘ file

Usage

```
cpp_decorations(pkg = ".", files = cpp_files(pkg = pkg), is_attribute = FALSE)
```

Arguments

pkg	The path to a package’s root directory.
files	Paths to ‘C++‘ files. If given, ‘pkg‘ will not be used.
is_attribute	If ‘TRUE‘ the decorations are C++11 attributes, if ‘FALSE‘ they are comments.

Value

A tibble with the decorations found, containing fields: - file - The filename for the decoration - line - The line the decoration was found - decoration - The name of the decoration - params - Any parameters given with the decoration - context - The text of the decoration line and all lines until the next decoration (or the end of the file).

Examples

```
# Setup
f <- tempfile()
writeLines("[[cpp11::register]] int fun(int x = 1) { return x + 1; }", f)

# Retrieve the decorations in the file
cpp_decorations(files = f, is_attribute = TRUE)

# Cleanup
unlink(f)
```

cpp_files *‘C++‘ files from a package*

Description

‘C++‘ files from a package

Usage

```
cpp_files(pkg = ".")
```

Arguments

pkg	The path to a package's root directory.

Value

A character vector of ‘C++‘ files found in the package, ordered according to the C locale, for stability across different sessions and platforms.

Examples

```
# Setup
pkg <- tempfile()
dir.create(file.path(pkg, "src"), recursive = TRUE)
file.create(file.path(pkg, "src", "code.c"))
file.create(file.path(pkg, "src", "code.cpp"))

# List the files, only the C++ file will be listed
cpp_files(pkg)

# Cleanup
unlink(pkg, recursive = TRUE)
```

---

parse_cpp_function *Parse a C++ function*

Description

Parses a C++ function returning a tibble with the function name and return type and a list column with the arguments of the function.

Usage

```
parse_cpp_function(context, is_attribute = FALSE)
```

Arguments

context	The function context, as obtained by the ‘context‘ column from [cpp_decorations()]
is_attribute	If ‘TRUE‘ the decorations are C++11 attributes, if ‘FALSE‘ they are comments.

Value

A tibble with the following fields: - name - The name of the function - return_type - The return type of the function - args - A list column containing a tibble of the functions arguments - type - The type of the argument - name - The name of the argument - default - The default value of the argument (if any).

Examples

```
# Setup
context <- "int fun(int x) { return x + 1; }"

# Parse the function
parse_cpp_function(context)
```

Index