

Package: bracer (via r-universe)

September 11, 2024

Title Brace Expansions

Version 1.2.3-0

Description Performs brace expansions on strings. Made popular by Unix shells, brace expansion allows users to concisely generate certain character vectors by taking a single string and (recursively) expanding the comma-separated lists and double-period-separated integer and character sequences enclosed within braces in that string. The double-period-separated numeric integer expansion also supports padding the resulting numbers with zeros.

URL <https://trevorldavis.com/R/bracer/>,
<https://github.com/trevorld/bracer>

BugReports <https://github.com/trevorld/bracer/issues>

Imports stringr

License MIT + file LICENSE

Encoding UTF-8

Suggests knitr, rmarkdown, testthat, V8

VignetteBuilder knitr, rmarkdown

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

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

RemoteUrl <https://github.com/trevorld/bracer>

RemoteRef HEAD

RemoteSha 9c7148d8b616c6f7f47f0693482cc84c69b88ef3

Contents

expand_braces	2
glob	3

Index	4
--------------	----------

expand_braces	<i>Bash-style brace expansion</i>
---------------	-----------------------------------

Description

expand_braces() performs brace expansions on strings. str_expand_braces() is an alternate that returns a list of character vectors. Made popular by Unix shells, brace expansion allows users to concisely generate certain character vectors by taking a single string and (recursively) expanding the comma-separated lists and double-period-separated integer and character sequences enclosed within braces in that string. The double-period-separated numeric integer expansion also supports padding the resulting numbers with zeros.

Usage

```
expand_braces(string, engine = getOption("bracer.engine", NULL))
```

```
str_expand_braces(string, engine = getOption("bracer.engine", NULL))
```

Arguments

string	input character vector
engine	If 'r' use a pure R parser. If 'v8' use the 'braces' Javascript parser via the suggested V8 package. If NULL use 'v8' if 'V8' package detected else use 'r'; in either case send a message() about the choice unless getOption(bracer.engine.inform') is FALSE.

Value

expand_braces() returns a character vector while str_expand_braces() returns a list of character vectors.

Examples

```
expand_braces("Foo{A..F}", engine = "r")
expand_braces("Foo{01..10}", engine = "r")
expand_braces("Foo{A..E..2}{1..5..2}", engine = "r")
expand_braces("Foo{-01..1}", engine = "r")
expand_braces("Foo{{d..d},{bar,biz}}.{py,bash}", engine = "r")
expand_braces(c("Foo{A..F}", "Bar.{py,bash}", "{{Biz}}"), engine = "r")
str_expand_braces(c("Foo{A..F}", "Bar.{py,bash}", "{{Biz}}"), engine = "r")
```

`glob`*Brace and Wildcard expansion on file paths*

Description

`glob()` is a wrapper around `Sys.glob()` that uses `expand_braces()` to support both brace and wildcard expansion on file paths.

Usage

```
glob(paths, ..., engine = getOption("bracer.engine", NULL))
```

Arguments

<code>paths</code>	character vector of patterns for relative or absolute filepaths.
<code>...</code>	Passed to <code>Sys.glob()</code>
<code>engine</code>	If 'r' use a pure R parser. If 'v8' use the 'braces' Javascript parser via the suggested V8 package. If NULL use 'v8' if 'V8' package detected else use 'r'; in either case send a <code>message()</code> about the choice unless <code>getOption("bracer.engine.inform")</code> is FALSE.

Examples

```
dir <- system.file("R", package="bracer")
path <- file.path(dir, "{R,r,S,s}")
glob(path, engine = "r")
```

Index

`expand_braces`, [2](#)
`expand_braces()`, [3](#)

`glob`, [3](#)

`str_expand_braces (expand_braces)`, [2](#)
`sys.glob()`, [3](#)