

Package: dotty (via r-universe)

September 4, 2024

Type Package

Title The Unpacking Dot Operator

Version 0.1.0.9000

Description Provides a ``.` object which can be used for unpacking assignments. For example, ``. [rows, columns] <- dim(cars)`` could be used to pull the number of rows and number of columns from `dim(cars)`` into individual variables `rows`` and `columns`` in a single step.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests codetools, testthat (>= 3.0.0)

Config/testthat/edition 3

URL <https://kevinushey.github.io/dotty/>

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

RemoteUrl <https://github.com/kevinushey/dotty>

RemoteRef HEAD

RemoteSha 56c8a4ff9a67f9eafc85ca17b7cc7bd63ecbb9a3

Contents

.....	2
destructure	2
dotify	3
Index	4

The Destructuring Dot Operator

Description

Use ‘dotty’ to perform destructuring assignments. Please see the examples below for usages.

Usage

.

Format

An object of class dotty of length 0.

Examples

```
# extract number of rows, number of columns from mtcars
.[nr, nc] <- dim(mtcars)
c(nr, nc)

# extract first, last element of vector
.[first, .., last] <- c(1, 2, 3, 4, 5)
c(first, last)

# extract a value by name
.[beta = beta] <- list(alpha = 1, beta = 2, gamma = 3)
beta

# unpack nested values
.[x, .[y, .[z]]] <- list(1, list(2, list(3)))
c(x, y, z)

# split version components
.[major, minor, patch] <- getRversion()
c(major, minor, patch)
```

destructure*Destructure an Object*

Description

‘destructure’ is primarily used to help define how an object should be prepared, or transformed, prior to a destructuring assignment. This can be relevant for objects which have unique subsetting semantics – for example, [numeric_version] objects.

Usage

```
destructure(object)
```

Arguments

object An R object.

Details

Packages which would like to define special destructring semantics for certain object classes can implement methods for this class.

dotify	<i>Dotify an R Package</i>
--------	----------------------------

Description

When using ‘dotty’ within an R package, you might see ****NOTE****s during ‘R CMD check’ of the form:

Usage

```
dotify()
```

Details

“‘ N checking R code for possible problems (1.8s) <package>: no visible binding for global variable <variable> Undefined global functions or variables: <variable> “‘

This occurs because the [codetools] package, which is used for static analysis of R code during ‘R CMD check’, does not recognize that e.g. ‘.[apple] <- 42’ would create a variable called ‘apple’ in the current scope. Calling ‘dotty::dotify()’ in your package’s ‘.onLoad()’ will allow ‘dotty’ to patch ‘codetools’ in a way that will allow it to understand ‘dotty’ usages, and so prevent these ‘R CMD check’ notes from being emitted.

Index

* **datasets**

.., 2

.., 2

destructure, 2

dotify, 3

dotty(.), 2