Package 'foodwebr'

September 30, 2025

Type Package

Title Visualise Function Dependencies

Version 1.0.0

Description Easily create graphs of the inter-relationships between functions in an environment.

License MIT + file LICENSE

URL https://lewinfox.com/foodwebr/

BugReports https://github.com/lewinfox/foodwebr/issues

Imports cli, crayon, codetools, DiagrammeR, glue, rlang, stringr, tidygraph

Suggests testthat

Encoding UTF-8

Language en-GB

RoxygenNote 7.3.3

NeedsCompilation no

Author Lewin Appleton-Fox [aut, cre]

Maintainer Lewin Appleton-Fox <lewin.a.f@gmail.com>

Repository CRAN

Date/Publication 2025-09-30 07:30:07 UTC

Contents

`foodweb` *Create a foodweb*

Description

A `foodweb` object describes the relationship of functions in an environment. It has two components: `funmat` (function matrix) which encodes the caller/callee relationships (i.e. which functions call which) and `graphviz_spec` which is a text representation of the graph and is used for the default plotting behaviour.

Usage

```
foodweb(
  FUN = NULL,
  env = parent.frame(),
  filter = !is.null(FUN),
  as.text = FALSE
)
```

Arguments

`FUN`	A function.
`env`	An environment, `parent.frame()` by default. Ignored if `FUN` is not `NULL`.
`filter`	Boolean. If `TRUE`, only functions that are direct descendants or antecedents of `FUN` will be shown.
`as.text`	Boolean. If `TRUE`, rather than rendering the graph the intermediate graphviz specification is returned.

Details

`foodweb()` looks at the global environment by default. If you want to look at another environment you can either pass a function to the `FUN` argument of `foodweb()` or pass an environment to the `env` argument. If `FUN` is provided then the value of `env` is ignored, and the environment of `FUN` will be used.

Value

If `as.text` is `TRUE`, a character vector. Otherwise, a `foodweb` object as described above.

Examples

```
# Create some functions to look at
f <- function() 1
g <- function() f()
h <- function() {
  f()
  g()
```

```
}
i <- function() {
  f()
  g()
  h()
}
j <- function() j()

x <- foodweb()
x

# You can access the components directly or via getter functions
x$funmat
get_graphviz_spec(x)

# Calculate the foodweb of a function in another package
foodweb(glue::glue)
```

`foodweb_matrix` *Create a function caller/callee matrix*

Description

Returns a matrix of 0s and 1s with a row and column for each function in an environment, such that if the function on the x-axis calls the function on the y-axis, the element is 1, otherwise 0.

Usage

```
foodweb_matrix(env = parent.frame())
```

Arguments

`env`	Environment in which to search for functions.

Value

An n x n matrix where *n* is the number of functions in `env`.

`get_funmat` *Extract the function matrix from a* `foodweb` *object.*

Description

Extract the function matrix from a `foodweb` object.

Usage

```
get_funmat(x)
```

Arguments

x	A foodweb

Value

x$funmat - a numeric matrix.

---

get_graphviz_spec *Extract the GraphViz specification from a* foodweb *object.*

Description

Extract the GraphViz specification from a foodweb object.

Usage

```
get_graphviz_spec(x)
```

Arguments

x	A foodweb

Value

x$graphviz_spec - a character scalar.

---

graphviz_spec_from_matrix *Create a* graphviz *specification from a function matrix*

Description

Given a function matrix created by foodweb_matrix(), convert it into a text specification that can be passed to DiagrammeR::grViz().

Usage

```
graphviz_spec_from_matrix(funmat)
```

Arguments

funmat	A function matrix generated by foodweb_matrix().

Value

A text string.

See Also

graphviz.org/

Examples

```
fm <- matrix(c(0, 1, 1, 1, 0, 1, 0, 1, 0), nrow = 3)
colnames(fm) <- rownames(fm) <- c("foo", "bar", "baz")
graphviz_spec_from_matrix(fm)
```

---

is.foodweb *Is an object a* foodweb*?*

Description

Is an object a foodweb?

Usage

```
is.foodweb(x)
```

Arguments

x	The object to test

Value

Boolean

---

print.foodweb_matrix *Print a* foodweb_matrix

Description

Print a foodweb_matrix

Usage

```
## S3 method for class 'foodweb_matrix'
print(x, ...)
```

Arguments

x	A foodweb_matrix
...	Unused

Value

x, invisibly

Index