

# Package ‘navigatr’

May 9, 2026

**Type** Package

**Title** Navigation Menu for Pipe-Friendly Data Processing

**Version** 0.2.1

**Description** Provides a navigation menu to enable pipe-friendly data processing for hierarchical data structures. By activating the menu items, you can perform operations on each item while maintaining the overall structure in attributes.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Imports** cli, lifecycle, pillar, purrr, rlang, stickyr, tidysselect, vctrs

**RoxygenNote** 7.2.3

**Suggests** covr, dplyr, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**BugReports** <https://github.com/UchidaMizuki/navigatr/issues>

**URL** <https://github.com/UchidaMizuki/navigatr>

**NeedsCompilation** no

**Author** Mizuki Uchida [aut, cre]

**Maintainer** Mizuki Uchida <uchidamizuki@vivaldi.net>

**Repository** CRAN

**Date/Publication** 2023-03-26 17:20:15 UTC

## Contents

activate . . . . .	2
itemise . . . . .	3
new_menu . . . . .	4
new_nav_input . . . . .	4
new_nav_menu . . . . .	5
rekey . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

---

activate	<i>Activate or deactivate a menu item</i>
----------	-------------------------------------------

---

### Description

Activates a menu item with the same syntax as `dplyr::pull()`. Activating a menu item allows you to perform operations on the active item. `activate()` turns a `navigatr_nav_menu` object into an `navigatr_item` object, and `deactivate()` turns it back.

### Usage

```
activate(.data, ..., .add = FALSE)

## S3 method for class 'navigatr_nav_menu'
activate(.data, ..., .add = FALSE)

## S3 method for class 'navigatr_item'
activate(.data, ..., .add = FALSE)

deactivate(x, ..., deep = TRUE)

## S3 method for class 'navigatr_nav_menu'
deactivate(x, ..., deep = TRUE)

## S3 method for class 'navigatr_item'
deactivate(x, ..., deep = TRUE)
```

### Arguments

<code>.data</code>	A <code>navigatr_nav_menu</code> object.
<code>...</code>	In <code>activate()</code> , one or more variables passed to <code>dplyr::pull()</code> . In <code>deactivate()</code> , unused (for extensibility).
<code>.add</code>	Whether to add new variables to the path indices. If <code>FALSE</code> (default value), the menu will be deactivated first by <code>deactivate()</code> .
<code>x</code>	A <code>navigatr_nav_menu</code> object.
<code>deep</code>	If <code>TRUE</code> (default value), deactivate recursively.

### Value

In `activate()`, An `navigatr_item` object. If it inherits from class `navigatr_nav_menu`, the menu will be displayed hierarchically. Otherwise, the active data will be displayed. In `deactivate()`, A `navigatr_nav_menu` object.

**Examples**

```
library(dplyr)

mn1 <- new_nav_menu(key = c("band_members", "band_instruments"),
                    value = list(band_members, band_instruments))

mn1 |>
  activate(band_members) |>
  filter(band == "Beatles")

# Items can also be specified as integers
mn1 |>
  activate(2)

mn1 |>
  activate(-1) |>
  deactivate()

# To activate items in a nested menu, specify multiple variables
mn2 <- new_nav_menu(key = c("key1", "key2"),
                    value = list(mn1, mn1))
mn2 |>
  activate(key1, band_members)
```

---

itemise

*Set items*

---

**Description**

Set items

**Usage**

```
itemise(.data, ...)
```

```
itemize(.data, ...)
```

**Arguments**

.data            A navigatr\_nav\_input object.  
...             Key-value pairs.

**Value**

A navigatr\_nav\_input object.

---

 new\_menu

*Deprecated functions*


---

### Description

Deprecated functions

### Usage

```
new_menu(
  key = character(),
  value = list(),
  attrs = NULL,
  ...,
  class = character()
)
```

### Arguments

key	A unique character vector.
value	A list of values corresponding to the keys.
attrs	A data frame for additional attributes of items (an empty data frame by default). When an item becomes active, the attrs will be added to its attributes.
...	Additional arguments passed to <code>vctrs::new_data_frame()</code> .
class	A character vector of subclasses passed to <code>vctrs::new_data_frame()</code> .

### Value

A `navigatr_nav_menu` object, a subclass of class `data.frame`.

---

 new\_nav\_input

*Build a new input form*


---

### Description

To create a new input form, give `new_nav_input()` a unique key and the corresponding list of values. By default, the values are empty characters. Each line shows the menu items (keys on the left, value summaries on the right). The summaries are `pillar::obj_sum` outputs, so you can change the printing methods. Each menu item can be changed by `itemise()`.

**Usage**

```
new_nav_input(
  key = character(),
  value = list(character()),
  ...,
  class = character()
)
```

**Arguments**

key	A unique character vector.
value	A list of values corresponding to the keys. By default, the values are empty characters.
...	Additional arguments passed to <code>vctrs::new_data_frame()</code> .
class	A character vector of subclasses passed to <code>vctrs::new_data_frame()</code> .

**Value**

A `navigatr_nav_input` object, a subclass of class `data.frame`.

**See Also**

[itemise\(\)](#)

---

new\_nav\_menu

*Build a new menu*

---

**Description**

To build a new menu, give `new_nav_menu()` unique keys and a list of their corresponding values. Each line shows the menu items (keys on the left, value summaries on the right). The summaries are [pillar::obj\\_sum](#) outputs, so you can change the printing methods. Each menu item can be accessed by [activate\(\)](#).

**Usage**

```
new_nav_menu(
  key = character(),
  value = list(data.frame()),
  attrs = NULL,
  ...,
  class = character()
)
```

**Arguments**

key	A unique character vector.
value	A list of values corresponding to the keys.
attrs	A data frame for additional attributes of items (an empty data frame by default). When an item becomes active, the attrs will be added to its attributes.
...	Additional arguments passed to <code>vctrs::new_data_frame()</code> .
class	A character vector of subclasses passed to <code>vctrs::new_data_frame()</code> .

**Value**

A `navigatr_nav_menu` object, a subclass of class `data.frame`.

**See Also**

[activate\(\)](#)

**Examples**

```
library(dplyr)

band <- new_nav_menu(key = c("band_members", "band_instruments"),
                    value = list(band_members, band_instruments))

band

# You can also build a nested menu
bands <- new_nav_menu(key = c("key1", "key2"),
                    value = list(band, band))

bands
```

---

rekey

*Rename key names*


---

**Description**

Rename key names

**Usage**

```
rekey(.data, ...)
```

## S3 method for class 'navigatr\_nav'

```
rekey(.data, ...)
```

## S3 method for class 'navigatr\_item'

```
rekey(.data, ...)
```

```
rekey_with(.data, .fn, .keys = tidyselect::everything(), ...)
```

**Arguments**

<code>.data</code>	For <code>rekey()</code> , A <code>navigatr_nav</code> or <code>navigatr_item</code> object. For <code>rekey_with()</code> , A <code>navigatr_nav</code> object.
<code>...</code>	For <code>navigatr_nav</code> objects, use <code>new_name = old_name</code> . For <code>navigatr_item</code> objects, a scalar character of the new key name.
<code>.fn</code>	A function used to transform the selected <code>.keys</code> .
<code>.keys</code>	Keys to rename; defaults to all keys.

**Value**

A `navigatr_nav` or `navigatr_item` object.

# Index

activate, 2  
activate(), 5, 6  
  
deactivate (activate), 2  
dplyr::pull(), 2  
  
itemise, 3  
itemise(), 4, 5  
itemize (itemise), 3  
  
new\_menu, 4  
new\_nav\_input, 4  
new\_nav\_menu, 5  
  
pillar::obj\_sum, 4, 5  
  
rekey, 6  
rekey\_with (rekey), 6  
  
vctrs::new\_data\_frame(), 4–6