

Package ‘hmstimer’

August 19, 2024

Title 'hms' Based Timer

Version 0.3.0

Description Tracks elapsed clock time using a `hms::hms()` scalar.

It was originally developed to time Bayesian model runs.

It should not be used to estimate how long extremely fast code takes to execute as the package code adds a small time cost.

License MIT + file LICENSE

URL <https://github.com/poissonconsulting/hmstimer>,

<https://poissonconsulting.github.io/hmstimer/>

BugReports <https://github.com/poissonconsulting/hmstimer/issues>

Depends R (>= 4.0)

Imports hms, lifecycle, rlang

Suggests covr, testthat (>= 3.0.0), withr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.3.2

NeedsCompilation no

Author Joe Thorley [aut, cre] (<<https://orcid.org/0000-0002-7683-4592>>),

Kirill Müller [aut] (<<https://orcid.org/0000-0002-1416-3412>>),

Nadine Hussein [ctb] (<<https://orcid.org/0000-0003-4470-8361>>),

Poisson Consulting [cph, fnd]

Maintainer Joe Thorley <joe@poissonconsulting.ca>

Repository CRAN

Date/Publication 2024-08-19 19:30:07 UTC

Contents

hms_timer	2
local_timer	3
tmr_ceiling	3
tmr_elapsed	4
tmr_floor	5
tmr_format	6
tmr_is_started	7
tmr_is_stopped	7
tmr_is_titled	8
tmr_print	9
tmr_reset	10
tmr_round	10
tmr_start	11
tmr_stop	12
tmr_timer	13
tmr_title	13
tmr_title<-	14
with_timer	15

Index	16
--------------	-----------

hms_timer	<i>hms Timer</i>
-----------	------------------

Description

A hms Timer is a `hms::hms()` scalar which if running has an attribute named `start` that specifies the system time when the timer was started.

Details

The elapsed time is the value of the scalar plus the difference between the current system time and the system time when the timer was started.

Examples

```
str(tmr_timer())
str(tmr_timer(1.5, start = TRUE))

x <- tmr_timer(1, start = TRUE)
print(x)
Sys.sleep(0.1)
print(x)
print(tmr_elapsed(x))
print(x)
```

local_timer	<i>Local Timer</i>
-------------	--------------------

Description

Called for the side effect of providing a message of the time required to execute the rest of the function.

Usage

```
local_timer(..., title = "", srcref = TRUE, .local_envir = rlang::caller_env())
```

Arguments

...	These dots are for future extensions and must be empty.
title	A string of the title.
srcref	A flag specifying whether to print the source reference.
.local_envir	The environment to use for scoping.

See Also

[with_timer\(\)](#)

Examples

```
fun <- function() {  
  local_timer()  
  Sys.sleep(0.1)  
  10  
}  
fun()
```

tmr_ceiling	<i>Ceiling hms Timer</i>
-------------	--------------------------

Description

Rounds a [hms_timer\(\)](#) up to the nearest second.

Usage

```
tmr_ceiling(x)
```

Arguments

x	A hms_timer() .
---	---------------------------------

Value

A [hms_timer\(\)](#).

See Also

Other round: [tmr_floor\(\)](#), [tmr_format\(\)](#), [tmr_round\(\)](#)

Examples

```
tmr_ceiling(tmr_timer(18.9))
tmr_ceiling(tmr_timer(122.1))
```

tmr_elapsed	<i>Elapsed Time hms Timer</i>
-------------	-------------------------------

Description

Returns the elapsed time for a [hms_timer\(\)](#) as a [hms_timer\(\)](#).

Usage

```
tmr_elapsed(x)
```

Arguments

x A [hms_timer\(\)](#).

Details

The elapsed time is the value of the scalar plus the difference between the current system time and the system time when the timer was started.

If the original [hms_timer\(\)](#) was running then the new [hms_timer\(\)](#) is assigned an attribute named start of the current system time.

Value

A [hms_timer\(\)](#) of the elapsed time.

See Also

Other start_stop: [tmr_is_started\(\)](#), [tmr_is_stopped\(\)](#), [tmr_print\(\)](#), [tmr_reset\(\)](#), [tmr_start\(\)](#), [tmr_stop\(\)](#), [tmr_timer\(\)](#)

Examples

```
tmr <- tmr_start(tmr_timer())
print(tmr_elapsed(tmr))
Sys.sleep(0.01)
print(tmr_elapsed(tmr))
tmr <- tmr_stop(tmr)
print(tmr_elapsed(tmr))
Sys.sleep(0.01)
print(tmr_elapsed(tmr))
```

tmr_floor

Floor hms Timer

Description

Rounds a [hms_timer\(\)](#) down to the nearest second.

Usage

```
tmr_floor(x)
```

Arguments

x A [hms_timer\(\)](#).

Value

A [hms_timer\(\)](#).

See Also

Other round: [tmr_ceiling\(\)](#), [tmr_format\(\)](#), [tmr_round\(\)](#)

Examples

```
tmr_floor(tmr_timer(18.9))
tmr_floor(tmr_timer(122.1))
```

tmr_format	<i>Format hms Timer</i>
------------	-------------------------

Description

Converts a [hms_timer\(\)](#) to a string of the clock time after rounding it to the number of digits.

Usage

```
tmr_format(x, digits = 3, ..., print_title = TRUE)
```

Arguments

x	A hms_timer() .
digits	A count of the number of decimal places.
...	These dots are for future extensions and must be empty.
print_title	A flag specifying whether to print the title.

Details

Negative values of digit are not permitted.

Value

A character string.

See Also

Other round: [tmr_ceiling\(\)](#), [tmr_floor\(\)](#), [tmr_round\(\)](#)

Examples

```
tmr_format(tmr_timer(61.66))
tmr_format(tmr_timer(61.66), digits = 0)
```

tmr_is_started	<i>Is hms Timer Started</i>
----------------	-----------------------------

Description

Tests if a `hms_timer()` is started (as indicated by the presence of an attribute named `start`).

Usage

```
tmr_is_started(x)
```

Arguments

x A `hms_timer()`.

Value

A flag (TRUE or FALSE).

See Also

Other `start_stop`: `tmr_elapsed()`, `tmr_is_stopped()`, `tmr_print()`, `tmr_reset()`, `tmr_start()`, `tmr_stop()`, `tmr_timer()`

Examples

```
tmr <- tmr_timer(start = TRUE)
print(tmr_is_started(tmr))
tmr <- tmr_stop(tmr)
print(tmr_is_started(tmr))
```

tmr_is_stopped	<i>Is hms Timer Stopped</i>
----------------	-----------------------------

Description

Tests if a `hms_timer()` is stopped (as indicated by the absence of an attribute named `start`).

Usage

```
tmr_is_stopped(x)
```

Arguments

x A `hms_timer()`.

Value

A flag.

See Also

Other start_stop: [tmr_elapsed\(\)](#), [tmr_is_started\(\)](#), [tmr_print\(\)](#), [tmr_reset\(\)](#), [tmr_start\(\)](#), [tmr_stop\(\)](#), [tmr_timer\(\)](#)

Examples

```
tmr <- tmr_timer(start = TRUE)
print(tmr_is_stopped(tmr))
tmr <- tmr_stop(tmr)
print(tmr_is_stopped(tmr))
```

tmr_is_titled	<i>Is hms Timer Title</i>
---------------	---------------------------

Description

Tests if a [hms_timer\(\)](#) has a title (as indicated by the presence of an attribute named start).

Usage

```
tmr_is_titled(x)
```

Arguments

x A [hms_timer\(\)](#).

Value

A flag (TRUE or FALSE).

Examples

```
tmr_is_titled(tmr_timer())
tmr_is_titled(tmr_timer(title = "my timer"))
```

tmr_print	<i>Print hms Timer</i>
-----------	------------------------

Description

Returns the elapsed time for a `hms_timer()` from the system time when the timer was started and the current system time as an hms time.

Usage

```
tmr_print(x, ..., print_title = TRUE)
```

Arguments

`x` A `hms_timer()`.
`...` These dots are for future extensions and must be empty.
`print_title` A flag specifying whether to print the title.

Details

The elapsed time is the value of the scalar plus the difference between the current system time and the system time when the timer was started.

Value

A character string.

See Also

Other start_stop: `tmr_elapsed()`, `tmr_is_started()`, `tmr_is_stopped()`, `tmr_reset()`, `tmr_start()`, `tmr_stop()`, `tmr_timer()`

Examples

```
x <- tmr_start(tmr_timer())  
tmr_print(x)
```

tmr_reset	<i>Reset hms Timer</i>
-----------	------------------------

Description

Resets a `hms_timer()` by creating a new one.

Usage

```
tmr_reset(x, seconds = 0)
```

Arguments

x	A <code>hms_timer()</code> .
seconds	A non-negative numeric scalar of the initial number of seconds.

Value

A `hms_timer()`.

See Also

Other start_stop: `tmr_elapsed()`, `tmr_is_started()`, `tmr_is_stopped()`, `tmr_print()`, `tmr_start()`, `tmr_stop()`, `tmr_timer()`

Examples

```
tmr <- tmr_timer(10)
print(tmr)
tmr_reset(tmr)
```

tmr_round	<i>Round hms Timer</i>
-----------	------------------------

Description

Rounds a `hms_timer()` after updating it to the elapsed time.

Usage

```
tmr_round(x, digits = 0)
```

Arguments

x	A <code>hms_timer()</code> .
digits	A count of the number of decimal places.

Details

Negative values of digit are permitted.

Value

A `hms_timer()`.

See Also

Other round: `tmr_ceiling()`, `tmr_floor()`, `tmr_format()`

Examples

```
tmr_round(tmr_timer(18.9))
tmr_round(tmr_timer(18.9), 1)
tmr_round(tmr_timer(18.9), -1)
tmr_round(tmr_timer(121), -2) # 121 is rounded to 100 seconds
```

tmr_start	<i>Start hms Timer</i>
-----------	------------------------

Description

Starts a `hms_timer()` by adding an attribute named start of the current system time.

Usage

```
tmr_start(x, ..., title = NULL)
```

Arguments

x	A <code>hms_timer()</code> .
...	These dots are for future extensions and must be empty.
title	A string of the title.

Details

If the `hms_timer()` is already started, the function simply issues a warning and returns the original object.

Value

A started `hms_timer()`.

See Also

Other start_stop: `tmr_elapsed()`, `tmr_is_started()`, `tmr_is_stopped()`, `tmr_print()`, `tmr_reset()`, `tmr_stop()`, `tmr_timer()`

Examples

```
tmr <- tmr_start(tmr_timer())
print(tmr_elapsed(tmr))
Sys.sleep(0.01)
print(tmr_elapsed(tmr))
```

tmr_stop

Stop hms Timer

Description

Stops a `hms_timer()` after updating it to the elapsed time.

Usage

```
tmr_stop(x)
```

Arguments

x A `hms_timer()`.

Details

If the `hms_timer()` is already stopped, the function simply issues a warning and returns the original object.

Value

A stopped `hms_timer()`.

See Also

Other start_stop: `tmr_elapsed()`, `tmr_is_started()`, `tmr_is_stopped()`, `tmr_print()`, `tmr_reset()`, `tmr_start()`, `tmr_timer()`

Examples

```
tmr <- tmr_stop(tmr_timer(start = TRUE))
print(tmr_elapsed(tmr))
Sys.sleep(0.01)
print(tmr_elapsed(tmr))
```

tmr_timer	<i>Create hms Timer</i>
-----------	-------------------------

Description

Creates a [hms_timer\(\)](#).

Usage

```
tmr_timer(seconds = 0, start = FALSE, ..., title = "")
```

Arguments

seconds	A non-negative numeric scalar of the initial number of seconds.
start	A flag specifying whether to start the timer.
...	These dots are for future extensions and must be empty.
title	A string of the title.

Value

A [hms_timer\(\)](#).

See Also

Other start_stop: [tmr_elapsed\(\)](#), [tmr_is_started\(\)](#), [tmr_is_stopped\(\)](#), [tmr_print\(\)](#), [tmr_reset\(\)](#), [tmr_start\(\)](#), [tmr_stop\(\)](#)

Examples

```
tmr_timer()  
tmr_timer(1, start = TRUE, title = "my timer")  
class(tmr_timer(2))  
str(tmr_timer(2, start = TRUE, title = "a timer"))
```

tmr_title	<i>Get Title hms Timer</i>
-----------	----------------------------

Description

Returns a flag (character vector) of the title.

Usage

```
tmr_title(x)
```

Arguments

x A `hms_timer()`.

Value

A flag of the title.

See Also

`tmr_title<-()`

Examples

```
tmr_title(tmr_timer())
tmr_title(tmr_timer(title = ""))
tmr_title(tmr_timer(title = "A Title"))
```

`tmr_title<-` *Set Title hms Timer*

Description

Sets the title of a `hms_timer()`.

Usage

```
tmr_title(x) <- value
```

Arguments

x A `hms_timer()`.
value A string of the title.

Value

A copy of the `hms_timer()` with the new title.

See Also

`tmr_title()`

Examples

```
tmr <- tmr_timer(title = "A title")
tmr_print(tmr)
tmr_title(tmr) <- "A different title"
tmr_print(tmr)
tmr_title(tmr) <- NULL
tmr_print(tmr)
```

`with_timer`*With Timer*

Description

With Timer

Usage

```
with_timer(code, ..., title = FALSE, srcref = FALSE)
```

Arguments

<code>code</code>	A line or block of R code.
<code>...</code>	These dots are for future extensions and must be empty.
<code>title</code>	A flag specifying whether to add a title based on code.
<code>srcref</code>	A flag specifying whether to print the source reference.

Value

The result of executing the code.

See Also

[local_timer\(\)](#)

Examples

```
fun <- function() {
  Sys.sleep(0.1)
  10
}
with_timer(fun())

with_timer({
  for (i in 1:2) {
    Sys.sleep(0.1)
  }
  20
})
```

Index

* **round**

tmr_ceiling, 3
tmr_floor, 5
tmr_format, 6
tmr_round, 10

* **start_stop**

tmr_elapsed, 4
tmr_is_started, 7
tmr_is_stopped, 7
tmr_print, 9
tmr_reset, 10
tmr_start, 11
tmr_stop, 12
tmr_timer, 13

hms::hms(), 2
hms_timer, 2
hms_timer(), 3–14

local_timer, 3
local_timer(), 15

tmr_ceiling, 3, 5, 6, 11
tmr_elapsed, 4, 7–13
tmr_floor, 4, 5, 6, 11
tmr_format, 4, 5, 6, 11
tmr_is_started, 4, 7, 8–13
tmr_is_stopped, 4, 7, 7, 9–13
tmr_is_titled, 8
tmr_print, 4, 7, 8, 9, 10–13
tmr_reset, 4, 7–9, 10, 11–13
tmr_round, 4–6, 10
tmr_start, 4, 7–10, 11, 12, 13
tmr_stop, 4, 7–11, 12, 13
tmr_timer, 4, 7–12, 13
tmr_title, 13
tmr_title(), 14
tmr_title<-, 14

with_timer, 15
with_timer(), 3