

# Package ‘icecream’

October 3, 2021

**Type** Package

**Title** Print Debugging Made Sweeter

**Version** 0.2.0

**Description** Provides user-friendly and configurable print debugging via a single function, `ic()`. Wrap an expression in `ic()` to print the expression, its value and (where available) its source location. Debugging output can be toggled globally without modifying code.

**License** MIT + file LICENSE

**URL** <https://www.lewinfox.com/icecream/>,  
<https://github.com/lewinfox/icecream>

**BugReports** <https://github.com/lewinfox/icecream/issues>

**Imports** cli, glue, pillar (>= 1.6.1), purrr (>= 0.3.4), rlang

**Suggests** checkmate (>= 2.0.0), testthat (>= 3.0.0), withr

**Config/testthat/edition** 3

**Encoding** UTF-8

**Language** en-GB

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Lewin Appleton-Fox [aut, cre],  
Dominik Rafacz [aut] (<<https://orcid.org/0000-0003-0925-1909>>),  
Ben Stiles [ctb]

**Maintainer** Lewin Appleton-Fox <[lewin.a.f@gmail.com](mailto:lewin.a.f@gmail.com)>

**Repository** CRAN

**Date/Publication** 2021-10-03 10:30:02 UTC

## R topics documented:

enable-disable . . . . .	2
ic . . . . .	2

ic-single-use . . . . .	3
ic_autopeek . . . . .	4
ic_autopeek_header . . . . .	5

<b>Index</b>	<b>6</b>
--------------	----------

---

enable-disable	<i>Enable or disable ic()</i>
----------------	-------------------------------

---

### Description

These functions enable or disable the `ic()` function. While disabled `ic()` will do nothing except evaluate and return its input.

### Usage

`ic_enable()`

`ic_disable()`

### Details

These are just convenience wrappers for `options(icecream.enabled = TRUE/FALSE)` used to align the API with the [Python version](#).

### Value

Returns the old value of the option, invisibly.

### Functions

- `ic_enable`: Enable `ic()`.
- `ic_disable`: Disable `ic()`.

---

ic	<i>User-friendly debug statements</i>
----	---------------------------------------

---

### Description

User-friendly debug statements

### Usage

`ic(x)`

### Arguments

x                    An expression, or nothing

**Value**

If `x` is an expression, returns the result of evaluating `x`. If `x` is missing nothing is returned.

**Examples**

```
f <- function(x) x < 0
ic(f(1))
ic(f(-1))
```

---

`ic-single-use`*Temporarily enable or disable ic()*

---

**Description**

These functions let you evaluate an expression with either `ic()` enabled or disabled without affecting if `ic()` is enabled globally.

**Usage**

```
with_ic_enable(expr)
with_ic_disable(expr)
```

**Arguments**

`expr`            An expression containing the `ic()` function.

**Value**

Returns the result of evaluating the expression.

**Functions**

- `with_ic_enable`: evaluates the expression with `ic()` enabled.
- `with_ic_disable`: evaluates the expression with `ic()` disabled.

**Examples**

```
ic_enable()

fun <- function(x) {
  ic(x * 100)
}

fun(2)
```

```

with_ic_disable(fun(2))

fun(4)

ic_disable()

fun(1)

with_ic_enable(fun(1))

```

---

ic\_peek

*Get descriptive one-line summary of an object*


---

### Description

This function is created as a modification of `utils::str()` function. It is supposed to create more compacted yet informative summary about an object. It's default value of "icecream.peeking.function"

### Usage

```

ic_peek(object, ...)

## S3 method for class 'list'
ic_peek(object, max_summary_length = cli::console_width(), ...)

## S3 method for class 'data.frame'
ic_peek(object, max_summary_length = cli::console_width(), ...)

```

### Arguments

<code>object</code>	The object to be summarized.
<code>...</code>	Other arguments passed to methods.
<code>max_summary_length</code>	Integer. Maximum length of string summarizing the object. By default this is set to the current terminal width.

### Details

This is a generic function. Default method simply calls `utils::str` function.

### Value

The function is mainly used for its side effects – outputting to the terminal. However, it also returns an invisible string of the printed summary.

### Methods (by class)

- `list`: Method for list
- `data.frame`: Method for data.frame

**See Also**

[utils::str\(\) ic\\_peek\(\)](#)

---

*ic\_peek\_header*      *Get a header of the object peeked at*

---

**Description**

Get a header of the object peeked at

**Usage**

```
ic_peek_header(object, ...)
```

**Arguments**

<code>object</code>	The object peeked at.
<code>...</code>	Other arguments passed to methods.

**Details**

This function is used by `ic_peek` to get a header of the summary of a object. It should return object's top-level class name and its dimension.

# Index

`enable-disable`, 2

`ic`, 2

`ic-single-use`, 3

`ic_peek`, 4

`ic_peek_header`, 5

`ic_disable` (`enable-disable`), 2

`ic_enable` (`enable-disable`), 2

`ic_peek()`, 5

`utils::str()`, 4, 5

`with_ic_disable` (`ic-single-use`), 3

`with_ic_enable` (`ic-single-use`), 3