

Package: vprint (via r-universe)

June 1, 2026

Version 1.2

Date 2025-06-04

Title More Flexible Form of Boolean Verbose

Maintainer Barry Zeeberg <barryz2013@gmail.com>

Author Barry Zeeberg [aut, cre]

Depends R (>= 4.2.0)

Description R functions are not supposed to print text without giving the user the option to turn the printing off or on using a Boolean 'verbose' in a construct like 'if(verbose) print(...)'. But this black/white approach is rather rigid, and an approach with shades of gray might be more appropriate in many circumstances.

License GPL (>= 2)

Encoding UTF-8

VignetteBuilder knitr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

RoxygenNote 7.3.2

Config/testthat/edition 3

NeedsCompilation no

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

Date/Publication 2025-06-04 15:30:02 UTC

RemoteUrl <https://github.com/cran/vprint>

RemoteRef HEAD

RemoteSha 1a1523326ebf2af4a236852bb0bb0e6c0f3df538

Contents

vprint	2
Index	3

vprint

vprint

Description

more flexible form of `if(verbose) print(...)`

Usage

```
vprint(class, verbose = NULL, txt)
```

Arguments

class	integer representing the verbose class
verbose	integer vector representing classes
txt	argument to print

Details

if you are the developer, and you have some debug statements rather than comment them out, you can turn them off and on likewise, a user can request more or less informational comments

suggested standardized class codes for `vprint()`:

-1 = developer debugging only

0 = constitutively turned on

1 = help for new user

2 = follow progress of long computation

3 = primary results

4 = meta info (e.g. dims of a mat before/after trimming)

5 = warnings

6 = errors

Note that the class argument is hardwired into the function code by the developer. For example, for a debugging statement, '-1' is hardwired in. The choice of whether or not to display this message is subsequently governed by the user selecting which values to include in the 'verbose' vector parameter.

Value

returns no values but has side effect of printing some text

Examples

```
vprint(1,1:2,"try me")  
vprint(0,1:2,"try me")  
vprint(3,1:2,"try me")
```

Index

`vprint`, 2