

Package ‘prettycode’

December 12, 2017

Title Pretty Print R Code in the Terminal

Version 1.0.1

Author Gábor Csárdi

Maintainer Gábor Csárdi <csardi.gabor@gmail.com>

Description Replace the standard print method for functions with one that performs syntax highlighting, using ANSI colors, if the terminal supports them.

License MIT + file LICENSE

LazyData true

URL <https://github.com/r-lib/prettycode#readme>

BugReports <https://github.com/r-lib/prettycode/issues>

RoxygenNote 6.0.1

Imports crayon, utils, withr

Suggests covr, mockery, testthat

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2017-12-12 22:33:59 UTC

R topics documented:

default_style	2
highlight	2
prettycode	3
print.function	3

Index	4
--------------	----------

default_style	<i>Default prettycode syntax highlighting style</i>
---------------	---

Description

A style function must return a named list of functions. Possible entries:

- reserved: reserved words
- number: numeric literals
- null: the NULL constant
- operator: operators, including assignment
- call: function calls
- string: character literals
- comment: comments

Usage

```
default_style()
```

Details

Each entry in a list must be a function that takes a character scalar, and returns a character scalar. The default style adds ANSI formatting to the code.

Note that you can also change the code if you like, e.g. to include a unicode arrow character instead of the two-character assignment operator.

Examples

```
highlight(deparse(get), style = default_style())
```

highlight	<i>Syntax highlight R code</i>
-----------	--------------------------------

Description

Syntax highlight R code

Usage

```
highlight(code, style = default_style())
```

Arguments

code	Character vector, each element is one line of code.
style	Style functions, see default_style() .

Value

Character vector, the highlighted code.

Examples

```
highlight(deparse(1s))
cat(highlight(deparse(1s)), sep = "\n")
```

```
prettycode
```

Pretty Print R Code in the Terminal

Description

Replace the standard print method for functions with one that performs syntax highlighting, using ANSI colors, if the terminal supports them.

```
print.function
```

Print a function with syntax highlighting

Description

Print a function with syntax highlighting

Usage

```
## S3 method for class 'function'
print(x, useSource = TRUE, style = default_style(), ...)
```

Arguments

x	Function to print.
useSource	Whether to use the stored source code, if available.
style	The highlight style to use, see default_style() .
...	Not used currently, for compatibility with the <code>print</code> generic.

Value

The function, invisibly.

Index

`default_style`, [2](#)

`default_style()`, [2](#), [3](#)

`highlight`, [2](#)

`prettycode`, [3](#)

`prettycode-package (prettycode)`, [3](#)

`print.function`, [3](#)