

Package ‘gglogger’

July 22, 2025

Type Package

Title Track 'ggplot2' Calls

Version 0.1.6

Description Provides a way to log 'ggplot' component calls, which can be useful for debugging and understanding how 'ggplot' objects are created. The logged calls can be printed, saved, and re-executed to reproduce the original 'ggplot' object.

License MIT + file LICENSE

Imports ggplot2, cli, methods

Suggests rlang, testthat, S7

RoxygenNote 7.3.2

Encoding UTF-8

URL <https://github.com/pwwang/gglogger/>

NeedsCompilation no

Author Panwen Wang [aut, cre] (ORCID: <<https://orcid.org/0000-0002-4614-8970>>)

Maintainer Panwen Wang <pwwang@pwwang.com>

Repository CRAN

Date/Publication 2025-06-19 23:30:02 UTC

Contents

+.gg	2
GGLog-class	2
GGLogs-class	3
ggplot	3
print.GGLog	4
print.GGLogs	4
register	5
Index	6

<code>+.gg</code>	<i>Override + operator for ggplot objects to log calls</i>
-------------------	--

Description

Override + operator for ggplot objects to log calls

Usage

```
## S3 method for class 'gg'
e1 + e2
```

Arguments

- `e1` A ggplot object.
- `e2` A layer to add to the ggplot object.

Value

A ggplot object with logged calls.

GGLog-class	<i>GGLog class</i>
-------------	--------------------

Description

The ‘GGLog’ class is used to store a single ggplot call.

Fields

`code` A string to store the code of the ggplot call.

Methods

```
evaluate(envir = parent.frame()) Evaluate the log.
  @param envir The environment to evaluate the log in.
stringify() Stringify the log.
  @return A string.
```

GGLogs-class

*GGLogs class***Description**

The ‘GGLogs’ class is used to store logs of ggplot calls. Each log is stored as a ‘GGLog’ object. The object can be accessed using the ‘\$logs’ field from a ggplot object.

Fields

logs A list to store the logs.

Methods

add(log) Add a log to the list.

@param log A GGLog object.

evaluate(envir = parent.frame()) Evaluate all logs in the list.

@param envir The environment to evaluate the logs in.

gen_code(setup = "library(ggplot2)") Generate code for all logs in the list.

@param setup A string to setup the environment.

@return A string of code.

stringify() Stringify all logs in the list.

@return A string.

ggplot

*Override ggplot function to log calls***Description**

Override ggplot function to log calls

Usage

```
ggplot(...)
```

Arguments

... Arguments passed to ggplot2::ggplot.

Value

A ggplot object with logged calls.

print.GGLog	<i>Print a GGLog object</i>
-------------	-----------------------------

Description

Print a GGLog object

Usage

```
## S3 method for class 'GGLog'  
print(x, ...)
```

Arguments

x	A GGLog object.
...	Additional arguments.

Value

Display the logs and invisibly return the logs.

print.GGLogs	<i>Print a GGLogs object</i>
--------------	------------------------------

Description

Print a GGLogs object

Usage

```
## S3 method for class 'GGLogs'  
print(x, ...)
```

Arguments

x	A GGLogs object.
...	Additional arguments.

Value

Display the logs and invisibly return the logs.

register	<i>Register a function that returns a ggplot object This is usually implemented by ggplot extensions.</i>
----------	---

Description

Register a function that returns a ggplot object This is usually implemented by ggplot extensions.

Usage

```
register(f, name = NULL)
```

Arguments

f	A function that returns a ggplot object.
name	A string to represent the name of the function.

Value

A function that returns a ggplot object with logged calls.

Index

`+.gg`, [2](#)

`GGLog` (`GGLog-class`), [2](#)

`GGLog-class`, [2](#)

`GGLogs` (`GGLogs-class`), [3](#)

`GGLogs-class`, [3](#)

`ggplot`, [3](#)

`print.GGLog`, [4](#)

`print.GGLogs`, [4](#)

`register`, [5](#)