

# Package: anybadger (via r-universe)

September 10, 2024

**Type** Package

**Title** Create Custom Pipeline Badges

**Version** 0.1.1

**Author** Lorenzo Meninato

**Maintainer** Lorenzo Meninato <lorenzomeninato@gmail.com>

**URL** <https://github.com/lmeninato/anybadger>

**BugReports** <https://github.com/lmeninato/anybadger/issues>

**Description** You can use this package to create custom pipeline badges in a standard 'svg' format. This is useful for a company to use internally, where it may not be possible to create badges through external providers. This project was inspired by the 'anybadge' library in python.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** R6, glue, purrr, config, magrittr

**RoxygenNote** 7.0.2

**Suggests** testthat (>= 2.1.0), withr, xml2

**Repository** <https://lmeninato.r-universe.dev>

**RemoteUrl** <https://github.com/lmeninato/anybadger>

**RemoteRef** HEAD

**RemoteSha** aa3abe54538ab5664add5b8ea87c5957d1293787

## Contents

Badge . . . . .	2
create_badge . . . . .	4
<b>Index</b>	<b>5</b>

---

Badge

*The Badge class is used to create badges.*

---

## Description

The Badge class is used to create badges.

The Badge class is used to create badges.

## Details

It is likely easier to use the [create\\_badge](#) to create the badge svg in a single step.

## Methods

### Public methods:

- [Badge\\$new\(\)](#)
- [Badge\\$create\\_svg\(\)](#)
- [Badge\\$clone\(\)](#)

### Method `new()`: Badge constructor

Used to create badge object.

#### *Usage:*

```
Badge$new(  
  label,  
  value,  
  color = "#4c1",  
  num_padding_chars = 0.5,  
  thresholds = NULL,  
  font_name = "DejaVu Sans,Verdana,Geneva,sans-serif",  
  font_size = 11,  
  label_text_color = "#fff",  
  value_text_color = "#fff",  
  svg_template_path = NULL  
)
```

#### *Arguments:*

`label` left hand side of badge, e.g. "pipeline" in pipeline status badge

`value` right hand side of badge, e.g. "passing" in pipeline status badge

`color` to view available colors: `config::get("colors", file = anybadger:::get_sys("config.yml"))`  
alternatively you can also pass in the hex of your desired color. For instance, "#fe7d37" or "orange", either is accepted.

`num_padding_chars` NULL, can be passed in, but calculated automatically based on text length

`thresholds` TODO, thresholds are not implemented yet, coming soon!

`font_name` NULL, valid svg font will work, but sizing might be off

`font_size` NULL, if passing custom svg font

label\_text\_color NULL, set this to change the label text color  
 value\_text\_color NULL, set this to change the value text color  
 svg\_template\_path NULL, to use a different template in svg (not recommended)

*Returns:* NULL

*Examples:*

```
tmp <- tempfile()
b <- Badge$new(label = "Pipeline",
               value = "Passing")
b$create_svg(tmp)
```

**Method** create\_svg(): Fills in the svg template

*Usage:*

```
Badge$create_svg(path = "default_badge.svg")
```

*Arguments:*

path file path to save badge svg to

*Returns:* invisibly returns the svg text

*Examples:*

```
tmp <- tempfile()
b <- Badge$new(label = "Any",
               value = "Badger")
b$create_svg(tmp)
```

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*

```
Badge$clone(deep = FALSE)
```

*Arguments:*

deep Whether to make a deep clone.

## Examples

```
## -----
## Method `Badge$new`
## -----

tmp <- tempfile()
b <- Badge$new(label = "Pipeline",
               value = "Passing")
b$create_svg(tmp)

## -----
## Method `Badge$create_svg`
## -----

tmp <- tempfile()
b <- Badge$new(label = "Any",
               value = "Badger")
b$create_svg(tmp)
```

---

create_badge	<i>Create badge svg</i>
--------------	-------------------------

---

**Description**

Create badge svg

**Usage**

```
create_badge(path, ...)
```

**Arguments**

path	path to save svg to
...	parameters to pass to @seealso [Badge]

**Value**

svg text

**Examples**

```
tmp <- tempfile()
create_badge(tmp, label = "any", value = "badger", color = "fuchsia")
```

# Index

Badge, [2](#)

create\_badge, [2](#), [4](#)