Package 'cptcity'

December 2, 2024

Type Package

Title 'cpt-city' Colour Gradients

Version 1.1.1

Description Incorporates colour gradients from the 'cpt-city' web archive available at <http://seaviewsensing.com/pub/cpt-city/>.

Depends R (>= 3.5.0)

Imports grDevices

License GPL-3

URL https://github.com/ibarraespinosa/cptcity

BugReports https://github.com/ibarraespinosa/cptcity/issues/

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests covr, testthat

Date 2024-11-30

NeedsCompilation no

Author Sergio Ibarra-Espinosa [aut, cre] (<https://orcid.org/0000-0002-3162-1905>)

Maintainer Sergio Ibarra-Espinosa <zergioibarra@gmail.com>

Repository CRAN

Date/Publication 2024-12-02 09:40:02 UTC

Contents

cpt 2
cptcity 3
cpt_names 3
find_cpt 4
lucky 5
show_cpt 6

Index 7

`cpt` *Function to return colour palettes functions from 'cpt-city'*

Description

This function return a collor palette based on the name or position of the palette.

Usage

```
cpt(
  pal = "mpl_inferno",
  n = 100,
  colorRampPalette = FALSE,
  rev = FALSE,
  frgb = rep(1, 3)
)
```

Arguments

`pal`	Palette of colors available or the number of the position
`n`	integer; number of colors
`colorRampPalette`	Logical; to be used in sf and mapview.
`rev`	Logical; to internally revert order of rgb color vectors.
`frgb`	Numeric; vector of 3 to change internal rgb composition. The order is red, green, blue

Details

The cpt-city web archive comes from: http://seaviewsensing.com/pub/cpt-city/

Value

A colour palette function.

Examples

```
{
library(cptcity)
image(matrix(1:100), col = cpt(pal = "mpl_inferno"))
find_cpt("temperature")
image(matrix(1:100), col = cpt("idv_temperature"))
image(matrix(1:100), col = cpt("idv_temperature", rev = TRUE))
## Not run:
# Do not run
library(ggplot2)
```

```
ggplot(faithfuld, aes(waiting, eruptions)) +
geom_raster(aes(fill = density))

 ggplot(faithfuld, aes(waiting, eruptions)) +
 geom_raster(aes(fill = density)) +
 scale_fill_gradientn(colours = cpt(n = 100))

## End(Not run)
}
```

`cptcity` *A package to return colour gradients from CPTCITY*

Description

Colour palettes comes from http://soliton.vm.bytemark.co.uk/pub/cpt-city/index.html Rhw function `cpt` has two arguments **n** for the numbers and **pal** for the name or number of the palette:

Details

The palettes are available here: http://soliton.vm.bytemark.co.uk/pub/cpt-city/index.html

Author(s)

Maintainer: Sergio Ibarra-Espinosa <zergioibarra@gmail.com> (ORCID)

See Also

Useful links:

- https://github.com/ibarraespinosa/cptcity
- Report bugs at https://github.com/ibarraespinosa/cptcity/issues/

`cpt_names` *Names of the 7140 color gradients of cptcity R Package*

Description

This dataset os a vector with al the names of the gradients of the archive cptcity (http://seaviewsensing.com/pub/cpt-city/) availale in this package. Please, read the documentation of each color gradient in the web page shown above.

Usage

```
data(cpt_names)
```

Format

A vector with the 7140 names of the color gradients

Source

http://seaviewsensing.com/pub/cpt-city/

`find_cpt` *Function to return colour palettes names*

Description

`find_cpt` returns the name of the colour gradient that satisfy the search. It is a searcher. It is a mini mini mini google.

Usage

```
find_cpt(name)
```

Arguments

`name`	character; Word to be searched among the names of the cpt gradients.

Value

names that satisfy the search.

Note

This functions runs grep.

Examples

```
{
library(cptcity)
find_cpt("temperature")
image(matrix(1:100), col = cpt("idv_temperature"))
## Not run:
library(cptcity)
# Do not run
# data names_cpt lazy loaded, already in environment
library(ggplot2)
ggplot(faithfuld, aes(waiting, eruptions)) +
geom_raster(aes(fill = density))

find_cpt("radar")
ggplot(faithfuld, aes(waiting, eruptions)) +
geom_raster(aes(fill = density)) +
scale_fill_gradientn(colours = cpt(n = 10, "ncl_radar"))
```

```
find_cpt("rain")
ggplot(faithfuld, aes(waiting, eruptions)) +
geom_raster(aes(fill = density)) +
scale_fill_gradientn(colours = cpt(pal = "pj_1_a_rainbow"))

## End(Not run)
}
```

---

lucky *Random colour gradient!*

Description

Based on "I'm Feeling Lucky" from Google. As this package includes 7140 colour gradients, it might be hard to find the 'right'

Usage

```
lucky(
  n = 100,
  colorRampPalette = FALSE,
  rev = FALSE,
  message = TRUE,
  nseed,
  frgb = rep(1, 3)
)
```

Arguments

n	integer; number of colors
colorRampPalette	Logical; to be used in sf and mapview.
rev	Logical; to internally revert order of rgb color vectors.
message	Logical; for printing or not the name of the colour gradient
nseed	integer; for reproducing the same colour gradient. See set.seed
frgb	Numeric; vector of 3 to change internal rgb composition The order is red, green, blue

Details

The cpt-city web archive comes from: http://soliton.vm.bytemark.co.uk/pub/cpt-city/index.html

Value

A RANDOM colour palette function including name of the colour gradient and number.

Examples

```
{
library(cptcity)
image(matrix(1:100), col = lucky())
image(matrix(1:100), col = lucky())
image(matrix(1:100), col = lucky())
image(matrix(1:100), col = lucky())
image(matrix(1:100), col = lucky())
image(matrix(1:100), col = lucky(rev = TRUE))
image(matrix(1:100), col = lucky(nseed = 1))
}
```

---

show_cpt *Show me some colors!*

Description

show_cpt returns a color matrix which prints and show the colors

Usage

```
show_cpt(x, label = TRUE)
```

Arguments

x	character; names of the cpt gradients.
label	Logical, to show labels or not.

Value

names color matrix

Examples

```
{
library(cptcity)
show_cpt(find_cpt("radar"))
}
```

Index

∗ **datasets**
cpt_names, 3

cpt, 2, *3*
cpt_names, 3
cptcity, 3
cptcity-package *(*cptcity*)*, 3

find_cpt, *4*, 4

lucky, 5

set.seed, *5*
show_cpt, *6*, 6