

Package ‘palr’

August 29, 2016

Type Package

Title Colour Palettes for Data

LazyData yes

Version 0.0.6

Description Colour palettes for data, based on some well known public data sets.

Depends R (>= 2.10)

Suggests knitr, raster, rmarkdown, testthat, covr

VignetteBuilder knitr

License GPL-3

RoxygenNote 5.0.1

URL <https://github.com/AustralianAntarcticDivision/palr>

BugReports <https://github.com/AustralianAntarcticDivision/palr/issues>

NeedsCompilation no

Author Michael D. Sumner [aut, cre]

Maintainer Michael D. Sumner <mdsumner@gmail.com>

Repository CRAN

Date/Publication 2016-07-27 03:09:36

R topics documented:

bathyDeepPal	2
chlPal	2
col2hex	3
icePal	4
mk_timePal	5
oisst	6
palr	6
sstPal	6

Index	8
--------------	----------

 bathyDeepPal

Bathymetry

Description

Deep bathymetry colours.

Usage

```
bathyDeepPal(x, palette = FALSE, alpha = 1)
```

Arguments

x	a vector of data values or a single num (n)
palette	logical, if TRUE return a list with matching colours and values
alpha	value in 0,1 to specify opacity

Details

Colour ramp suitable for deep waters (-5500) to sea level. The palette functions operate in 3 modes: 1) n colours - Pal(6) - returns 6 colours from the palette 2) data - Pal(c(10, 50, 100)) - return colours for 3 ice concentrations 3) palette - Pal(palette = TRUE) - return the full palette and breaks Derived from maps created in Matlab by Emmanuel Laurenceau.

Value

colours, palette, or function, see Details

Examples

```
plot(1:15, pch = 19, cex = 4, col = bathyDeepPal(15))
```

 chlPal

Ocean colour colours for chlorophyll-a.

Description

Ocean colour palette for chlorophyll-a.

Usage

```
chlPal(x, palette = FALSE, alpha = 1)
```

Arguments

x	a vector of data values or a single number
palette	logical, if TRUE return a list with matching colours and values
alpha	value in 0,1 to specify opacity

Details

Flexible control of the chlorophyll-a palette. If x is a single number, the function returns that many colours evenly spaced from the palette. If x is a vector of multiple values the palette is queried for colours matching those values, and these are returned. If x is missing and palette is FALSE then a function is returned that will generate n evenly spaced colours from the palette, as per [colorRampPalette](#).

Value

colours, palette, or function, see Details

References

Derived from http://oceancolor.gsfc.nasa.gov/DOCS/palette_chl_etc.txt.

Examples

```
## Not run:
chl <- raadtools::readchla(xylim = c(100, 110, -50, -40))
## just get a small number of evenly space colours
plot(chl, col = chlPal(10))
## store the full palette and work with values and colours
pal <- chlPal()
## the standard full palette
plot(chl, breaks = pal$breaks, col = pal$cols)
## a custom set of values with matching colours
plot(chl, col = chlPal(pal$breaks[seq(1, length(pal$breaks), length = 10)]))
## any number of colours stored as a function
myfun <- chlPal()
plot(chl, col = myfun(18))
## just n colours
plot(chl, col = chlPal(18))

## End(Not run)
```

col2hex

Colour to hex conversion.

Description

Create colours from colour names in one easy step.

Usage

```
col2hex(x, alpha = 1)
```

Arguments

x vector of colour names or hex strings
alpha optional transparency value in [0,1], can be per colour in x

Value

character string of hex colours

Examples

```
col2hex(c("aliceblue", "firebrick"), alpha = c(1, .5))  
col2hex(c("#FFFFFF", "#123456FF"), alpha = 0.1)
```

icePal	<i>Sea ice colours</i>
--------	------------------------

Description

Colours for sea ice.

Usage

```
icePal(x, palette = FALSE, alpha = 1)
```

Arguments

x a vector of data values or a single num (n)
palette logical, if TRUE return a list with matching colours and values
alpha value in 0,1 to specify opacity

Details

The palette functions operate in 3 modes: 1) n colours - Pal(6) - returns 6 colours from the palette 2) data - Pal(c(10, 50, 100)) - return colours for 3 ice concentrations 3) palette - Pal(palette = TRUE) - return the full palette and breaks

Value

colours, palette, or function, see Details

References

Derived from <http://www.iup.uni-bremen.de/seaiice/amsr/>.

Examples

```
## Not run:
library(raster)
r <- raster(system.file("extdata", "nt_20140320_f17_v01_s.bin", package = "graticule") )
icp <- icePal(palette = TRUE)
## The AMSR colours
plot(r, col = icp$col, zlim = range(icp$breaks),
main = sprintf("NSIDC ice \\% %s", format(getZ(r))))

## End(Not run)
```

mk_timePal

*Time-indexed colour:***Description**

Create a time-indexed colour map, useful for maintaining an absolute scale across time series as a function of date-time.

Usage

```
mk_timePal(x, col)
```

Arguments

x	date-times
col	colours, can be a function or an actual set of colours

Value

function of date-time

Examples

```
dts <- seq(as.Date("1749-01-01"), by = "1 month", length.out = length(sunspots))
d <- data.frame(date = dts, sunspots = as.vector(t(sunspots)))
tpal <- mk_timePal(d$date, col = sstPal(50))
par(mfrow = c(2, 1))
plot(sunspots ~ date, col = tpal(date), data = d)
## colours maintained by absolute date
plot(sunspots ~ date, col = tpal(date), data = d[1500:1800, ], cex = 2)
## we can now insert new points and maintain this colour ramp
d2 <- data.frame(date = seq(min(d$date), max(d$date), by = "5 days"))
d2$sunspots <- approxfun(d$date, d$sunspots)(d2$date)
points(sunspots ~ date, col = tpal(date), data = d2, pch = 19, cex = 0.5)
```

oisst *Sea surface temperature (SST).*

Description

SST example raster data set, at 0.25 degree resolution for global coverage in "longitude180/latitude".

Details

Created using script in data-raw/ using 'raadtools' package.

References

Reynolds, et al.(2007) Daily High-resolution Blended Analyses. Available at <ftp://eclipse.ncdc.noaa.gov/pub/OI-daily/daily-sst.pdf>. Climatology is based on 1971-2000 OI.v2 SST, Satellite data: Navy NOAA19 METOP AVHRR, Ice data: #' NCEP ice Source: NOAA/National Climatic Data Center.

Examples

```
library(raster)
oisst
class(oisst)
writeLines(unlist(metadata(oisst)))
```

palr *palr: colours for data*

Description

palr: colours for data

sstPal *SST colours*

Description

SST colours

Usage

```
sstPal(x, palette = FALSE, alpha = 1)
```

Arguments

<code>x</code>	a vector of data values or a single number
<code>palette</code>	logical, if TRUE return a list with matching colours and values
<code>alpha</code>	value in 0,1 to specify opacity

Value

colours, palette, or function, see Details

References

Derived from http://oceancolor.gsfc.nasa.gov/DOCS/palette_sst.txt.

Examples

```
library(raster)
data(oisst)
sstcols <- sstPal(palette = TRUE)
plot(oisst, col = sstcols$col, zlim = range(sstcols$breaks))
```

Index

[bathyDeepPal](#), [2](#)

[chlPal](#), [2](#)

[col2hex](#), [3](#)

[colorRampPalette](#), [3](#)

[icePal](#), [4](#)

[mk_timePal](#), [5](#)

[oisst](#), [6](#)

[palr](#), [6](#)

[palr-package \(palr\)](#), [6](#)

[sstPal](#), [6](#)