

# Package: displease (via r-universe)

December 16, 2024

**Type** Package

**Title** Numeric and Color Sequences with Non-Linear Interpolation

**Version** 1.0.0

**Maintainer** Mike Cheng <mikefc@coolbutuseless.com>

**Description** When visualising changes between two values over time, a strict linear interpolation can look jarring and unnatural. By applying a non-linear easing to the transition, the motion between values can appear smoother and more natural. This package includes functions for applying such non-linear easings to colors and numeric values, and is useful where smooth animated movement and transitions are desired.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**URL** <https://github.com/coolbutuseless/displease>

**BugReports** <https://github.com/coolbutuseless/displease/issues>

**Suggests** farver

**NeedsCompilation** no

**Author** Mike Cheng [aut, cre, cph]

**Repository** CRAN

**Date/Publication** 2024-12-16 17:40:02 UTC

## Contents

seq_color . . . . .	2
seq_ease . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

seq\_color

*Interpolate (non-linearly) between two colors***Description**

Interpolate (non-linearly) between two colors

**Usage**

```
seq_color(
  col1,
  col2,
  n = 100,
  type = "cubic",
  direction = "in-out",
  colorspace = "hcl"
)
```

**Arguments**

col1, col2	the two colors to interpolate between
n	Number of steps for the transition (including the endpoints)
type	Type of motion easing. Default: 'cubic'. Valid values are 'sine', 'quad', 'cubic', 'quart', 'quint', 'exp', 'circle', 'back', 'elastic', 'linear'.
direction	When should the easing apply? Default: "in-out". Valid values are 'in', 'out', 'in-out'. Default: 'in-out'
colorspace	Color space in which to do the interpolation. Default: 'hcl' Can be any colorspace understood by the farver package i.e. "cmy", "cmyk", "hsl", "hsb", "hsv", "lab" (CIE L*ab), "hunterlab" (Hunter Lab), "oklab", "lch" (CIE Lch(ab) / polarLAB), "luv", "rgb" (sRGB), "xyz", "yxy" (CIE xyY), "hcl" (CIE Lch(uv) / polarLuv), or "oklch" (Polar form of oklab). Note: Not all color spaces make sense for interpolation.

**Value**

character vector containing a color sequence

**Examples**

```
n <- 20
cols <- seq_color('red', 'blue', n = n, direction = 'in-out', colorspace = 'hcl')
cols
grid::grid.rect(x = seq(0, 0.95, length.out = n), width = 0.1,
  gp = grid::gpar(fill = cols, col = NA))
```

---

seq_ease	<i>Create a sequence interpolating between two values with the specified non-linear easing.</i>
----------	---

---

**Description**

Create a sequence interpolating between two values with the specified non-linear easing.

**Usage**

```
seq_ease(x1 = 0, x2 = 1, n = 100, type = "cubic", direction = "in-out")
```

**Arguments**

x1, x2	The start and end values of the sequence. Default: 0, 1
n	Number of steps for the transition (including the endpoints)
type	Type of motion easing. Default: 'cubic'. Valid values are 'sine', 'quad', 'cubic', 'quart', 'quint', 'exp', 'circle', 'back', 'elastic', 'linear'.
direction	When should the easing apply? Default: "in-out". Valid values are 'in', 'out', 'in-out'. Default: 'in-out'

**Value**

Numeric vector of length n

**Examples**

```
x <- seq_ease(x1 = 0, x2 = 1, n = 20, type = 'cubic', direction = 'in-out')
x
plot(x)
```

# Index

seq\_color, 2  
seq\_ease, 3