Package: huito (via r-universe)

September 5, 2024

Type Package

Version 0.2.5

Title Reproducible and Flexible Label Design

Description An open-source R package to deploys reproducible and flexible labels using layers. The 'huito' package is part of the 'inkaverse' project for developing different procedures and tools used in plant science and experimental designs. Learn more about the 'inkaverse' project at <https://inkaverse.com/>.

Date 2024-09-04

URL https://huito.inkaverse.com/, https://github.com/flavjack/huito

BugReports https://github.com/flavjack/huito/issues/

Depends magick, cowplot, ggplot2, dplyr, R (>= 2.10)

Imports tidyr, tibble, purrr, sysfonts, showtext, qrcode, pdftools

Suggests knitr, rmarkdown, bookdown, gsheet, inti

VignetteBuilder knitr

License GPL-3 | file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

LazyData true

Repository https://flavjack.r-universe.dev

RemoteUrl https://github.com/flavjack/huito

RemoteRef HEAD

RemoteSha f848b8f1520946135caf7fb78eb7a3e591a9ec05

Contents

`barcode_qr` *Barcode generator*

Description

Generate bar codes using QR codes

Usage

```
barcode_qr(text, color = "black", alpha = 1, ecl = "H")
```

Arguments

`text`	text to convert to QR bar code
`color`	Bar code color
`alpha`	Intensity of the bar code color
`ecl`	Error correction level (percentage). "L" (7), "M" (15), "Q" (25) and "H" (30). Defaults to "H"

Value

plot

Examples

```
library(huito)

barcode_qr("LIMA-2021-11-03_15_3_4")
```

fieldbook *Fieldbook experimental design*

Description

The dataset were obtained using inti package. The data set is a randomize complete block design (RCBD) with three replications.

Usage

```
fieldbook
```

Format

A data frame with 24 rows and 5 variables:

barcode barcode for each experimental unit

plots Plot number

block Blocks (3): number of replication in the design

condition Factor with two levels: irrigated and drought

genotypes Factor with four levels: choclito, salcedo, pandela, puno

huito_fonts *Fonts import*

Description

Import fonts from Google fonts

Usage

```
huito_fonts(fonts = NA)
```

Arguments

fonts	fonts names

Details

For more fonts visit: https://fonts.google.com/

Value

fonts

image_import *Image import*

Description

Import images and include R magick options

Usage

```
image_import(image, opts = NA)
```

Arguments

image	path or url
opts	R magick functions by layers

Value

image

include_barcode *Barcode layer*

Description

Insert barcode in label

Usage

```
include_barcode(
  label,
  value,
  size,
  position = NA,
  type = "static",
  color = "auto",
  units = "cm"
)
```

Arguments

label	label output
value	column or path
size	image size
position	position coordinate
type	type of entry: dynamic or static
color	image color
units	units for the label options

Value

data frame

---

include_image *Image layer*

Description

Insert image in label

Usage

```
include_image(
  label,
  value,
  size,
  position = NA,
  type = "static",
  units = "cm",
  opts = NA
)
```

Arguments

label	label output
value	column or path
size	image size
position	position coordinate
type	type of entry: dynamic or static
units	units for the label options
opts	R magick funtions

Value

data frame

include_shape *Shape layer*

Description

Insert shape in label

Usage

```
include_shape(
  label,
  value = "hexagon",
  size = 5.08,
  position = NA,
  border_color = "black",
  border_width = 1,
  background = NA,
  units = "cm",
  panel_color = NA,
  panel_size = NA
)
```

Arguments

label	label output (table)
value	type of shape (string: "hexagon")
size	shape size (numeric: 5.08)
position	position coordinate (numeric: NA)
border_color	image color (string: "black")
border_width	shape line width (numeric: 1)
background	background color (string: "red")
units	units for shape (string: "cm")
panel_color	panel color (string: NA)
panel_size	panel size (numeric: NA)

Value

data frame

Examples

```
library(huito)

label <- label_layout(data = NA
                , size = c(10, 2.5)
                , background = "yellow"
                ) %>%
          include_shape(
                value = "hexagon"
                , position = c(1.2, 1.25)
                , background = "red"
                , border_width = 1
                , size = 2.4
                #, panel_size = 2.4*1.157175
                )

label %>% label_print("sample")

ts <- label$opts
```

`include_text` *Text layer*

Description

Insert text in label

Usage

```
include_text(
  label,
  value,
  position = NA,
  size = 11,
  font = NA,
  type = "static",
  color = NA,
  angle = 0,
  opts = NA
)
```

Arguments

`label`	label output
`value`	column or string
`position`	position coordinate

size	text size
font	font type
type	type of entry: dynamic or static
color	image color
angle	angle of the text
opts	list arguments from draw_label()

Value

data frame

`label_layout` *Label layout*

Description

Generate labels options

Usage

```
label_layout(
  data = NA,
  size,
  border_width = NA,
  border_color = "black",
  background = NA,
  units = "cm"
)
```

Arguments

data	data frame to build the labels
size	label size (numeric: c(10, 2.5))
border_width	border width (numeric: 0.5)
border_color	border color (string: "transparent")
background	background color (string: "transparent")
units	units for the label options (string: "cm")

Value

data frame

Examples

```
label <- label_layout(size = c(10, 2.5)
                    , border_color = "red"
                    , border_width = 1
                    ) %>%
                   label_print()
```

`label_print` *Label print*

Description

Generate labels based in a data frame

Usage

```
label_print(
  label,
  mode = "sample",
  filename = "labels",
  margin = 0.04,
  paper = c(21, 29.7),
  units = "cm",
  viewer = FALSE,
  smpres = 200,
  nlabels = NA
)
```

Arguments

`label`	Data frame to build the labels or n repeated labels (table/numeric)
`mode`	Label generation (string: "sample/preview", "complete")
`filename`	Labels file name (string: "labels")
`margin`	Labels margins. margin(numeric vector: t = 0, r = 0, b = 0, l = 0)
`paper`	Paper size. Default A4 (numeric vector: 21.0 x 29.7)
`units`	Units for the label options (string: "cm")
`viewer`	Visualization of the label (logial: FALSE)
`smpres`	Sample resolution if viewer = TRUE (numeric: 200)
`nlabels`	Number of labels to generate (numeric: NA)

Value

pdf

Examples

```
library(huito)

fb <- fieldbook

label <- fb %>%
label_layout(size = c(10, 2.5)
             , border_color = "blue"
             ) %>%
  include_image(
    value = "https://flavjack.github.io/inti/img/inkaverse.png"
    , size = c(2.4, 2.4)
    , position = c(1.2, 1.25)
    ) %>%
  include_barcode(
    value = "barcode"
    , size = c(2.5, 2.5)
    , position = c(8.2, 1.25)
    ) %>%
  include_text(value = "plots"
               , position = c(9.7, 1.25)
               , angle = 90
               , size = 15
               , color = "red"
               ) %>%
  include_text(value = "Inkaverse"
               , position = c(4.6, 2)
               , size = 30
               , color = "brown"
               ) %>%
  include_text(value = "condition"
               , position = c(4.6, 1.2)
               , size = 13
               , color = "orange"
               ) %>%
  include_text(value = "genotypes"
               , position = c(4.6, 0.5)
               , size = 13
               , color = "#009966"
               ) %>%
               label_print(mode = "sample")
```

---

shape_hexagon *Shape hexagon*

Description

Hexagon geom shape for ggplot2

Usage

```
shape_hexagon(
  size = 5.08,
  border_width = NA,
  background = NA,
  border_color = "black",
  units = "cm",
  panel_color = "green",
  panel_size = NA
)
```

Arguments

size	hexagon size (numeric: 5.08)
border_width	line width (numeric: 1)
background	background color (string: "transparent")
border_color	border color (string: "black")
units	units for shape (string: "cm")
panel_color	panel color (string: "green")
panel_size	panel size (numeric: NA)

Value

geom

Examples

```
library(huito)

shape_hexagon(border_width = 1
              , background = "red"
              #, panel_size = 5.08
              )
```

Index