

Package: qrencoder (via r-universe)

September 7, 2024

Title Quick Response Code (QR Code) / Matrix Barcode Creator

Version 0.2.1

Maintainer Bob Rudis <bob@rud.is>

Description Quick Response codes (QR codes) are a type of matrix bar code and can be used to authenticate transactions, provide access to multi-factor authentication services and enable general data transfer in an image. QR codes use four standardized encoding modes (numeric, alphanumeric, byte/binary, and kanji) to efficiently store data. Matrix barcode generation is performed efficiently in C via the included 'libqrencoder' library created by Kentaro Fukuchi.

Depends R (>= 3.1.0), raster

License GPL-2

LazyData true

Suggests testthat

LinkingTo Rcpp

Imports Rcpp, base64enc, png

URL <http://github.com/hrbrmstr/qrencoder>

BugReports <https://github.com/hrbrmstr/qrencoder/issues>

RoxygenNote 6.0.1.9000

Repository <https://hrbrmstr.r-universe.dev>

RemoteUrl <https://github.com/hrbrmstr/qrencoder>

RemoteRef HEAD

RemoteSha 95701f7f0b597dc6c547210c9b632c02afc8dcd9

Contents

qrencode	2
qrencoder	2
qrencode_df	3

qrencode_png	3
qrencode_raster	4
qrencode_raw	4
qrencode_svg	5

Index	7
--------------	----------

qrencode	<i>Return a QR encoded string as a matrix</i>
----------	---

Description

Useful if you want to do your own post-processing

Usage

```
qrencode(to_encode)
```

Arguments

to_encode	the data to encode
-----------	--------------------

Examples

```
qrencode("http://rud.is/b")
```

qrencoder	<i>Quick Response Code (QR Code) / Matrix Barcode Creator</i>
-----------	---

Description

Quick Response codes (QR codes) are a type of matrix bar code and can be used to authenticate transactions, provide access to multi-factor authentication services and enable general data transfer in an image. QR codes use four standardized encoding modes (numeric, alphanumeric, byte/binary, and kanji) to efficiently store data. Matrix barcode generation is performed efficiently in C via the included 'libqrencoder' library created by Kentaro Fukuchi.

Author(s)

Bob Rudis (bob@rud.is)

qrcode_df	<i>Return a QR encoded string as an x, y, z data.frame</i>
-----------	--

Description

Useful for `ggplot::geom_raster`

Usage

```
qrcode_df(to_encode)
```

Arguments

to_encode the data to encode

Examples

```
head(qrcode_df("http://rud.is/b"))
```

qrcode_png	<i>Return a QR encoded string as a base 64 encoded inline png</i>
------------	---

Description

Return a QR encoded string as a base 64 encoded inline png

Usage

```
qrcode_png(to_encode)
```

Arguments

to_encode the data to encode

Note

data:image/png;base64, is prepended to the encoded png

Examples

```
cat(qrcode_png("http://rud.is/b"))
```

qrcode_raster *Return a QR encoded string as a raster object*

Description

Return a QR encoded string as a raster object

Usage

```
qrcode_raster(to_encode)
```

Arguments

to_encode the data to encode

Examples

```
library(raster)
old_mar <- par()$mar
par(mar=c(0,0,0,0))
image(qrcode_raster("http://rud.is/b"), asp=1, col=c("white", "black"),
      axes=FALSE, xlab="", ylab="")
par(mar=old_mar)
```

qrcode_raw *Encodes a string as a QR code*

Description

Encodes a string as a QR coder

Usage

```
qrcode_raw(to_encode, version = 0L, level = 0L, hint = 2L,
           caseinsensitive = 1L)
```

Arguments

to_encode character string to encode

version version of the symbol. If 0, the library chooses the minimum version for the given input data.

level error correction level (0 - 3, lowest to highest)

hint tell the library how Japanese Kanji characters should be encoded:

- If "3", the library assumes that the given string contains Shift-JIS characters and encodes them in Kanji-mode.

- If "2" is given, all of non-alphanumerical characters will be encoded as is. If you want to embed UTF-8 string, choose this. Trying to encode UTF-8 with modes will cause an error.
- "0" is "numeric mode",
- "1" is "alphanumeric mode"
- "5" is "ECI mode".

caseinsensitive

case-sensitive(1) or not(0).

See Also

<http://www.qrcode.com/en/about/version.html>

qrcode_svg

Return a QR encoded string as an svg string

Description

Encodes a string as a QR coder

Usage

```
qrcode_svg(to_encode, version = 0L, level = 0L, hint = 2L,
           caseinsensitive = 1L, margin = 0L, size = 3L, dpi = 72L)
```

Arguments

to_encode	character string to encode
version	version of the symbol. If 0, the library chooses the minimum version for the given input data.
level	error correction level (0 - 3, lowest to highest)
hint	tell the library how Japanese Kanji characters should be encoded: <ul style="list-style-type: none"> • If "3", the library assumes that the given string contains Shift-JIS characters and encodes them in Kanji-mode. • If "2" is given, all of non-alphanumerical characters will be encoded as is. If you want to embed UTF-8 string, choose this. Trying to encode UTF-8 with modes will cause an error. • "0" is "numeric mode", • "1" is "alphanumeric mode" • "5" is "ECI mode".
caseinsensitive	case-sensitive(1) or not(0).
margin	width of the margins default is 4
size	module size in dots (pixels); default is 3
dpi	resolution; default = 72

See Also

<http://www.qrcode.com/en/about/version.html>

Index

[qrencode](#), [2](#)
[qrencode_df](#), [3](#)
[qrencode_png](#), [3](#)
[qrencode_raster](#), [4](#)
[qrencode_raw](#), [4](#)
[qrencode_svg](#), [5](#)
[qrencoder](#), [2](#)
[qrencoder-package \(qrencoder\)](#), [2](#)