

Package ‘fontquiver’

July 22, 2025

Title Set of Installed Fonts

Version 0.2.1

Description Provides a set of fonts with permissive licences. This is useful when you want to avoid system fonts to make sure your outputs are reproducible.

Depends R (>= 3.0.0)

Imports fontBitstreamVera (>= 0.1.0), fontLiberation (>= 0.1.0)

Suggests testthat, htmltools

License GPL-3 | file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 5.0.1

Collate 'font-getters.R' 'fontset.R' 'fontset-bitstream-vera.R'
'fontset-dejavu.R' 'fontset-liberation.R' 'fontset-symbola.R'
'html-dependency.R' 'utils.R'

NeedsCompilation no

Author Lionel Henry [cre, aut],
RStudio [cph],
George Douros [cph] (Symbola font)

Maintainer Lionel Henry <lionel@rstudio.com>

Repository CRAN

Date/Publication 2017-02-01 00:52:56

Contents

fonts	2
fontset_info	3
fontset_list	3
fontset_register	4
fontset_variants	4
font_families	5

font_variants	5
htmlFontDependency	6
splice_fonts	7
Index	8

fonts	<i>Get elements of a fontset</i>
-------	----------------------------------

Description

fonts() returns the list of all font objects for a given fontset (a collection of fonts provided by fontquiver, such as Liberation or Bitstream Vera). font() returns one specific font. font_symbol() extracts the font marked as symbol font, if it exists in the fontset.

Usage

```
fonts(fontset)

font(fontset, variant, style)

font_symbol(fontset = "Symbola")
```

Arguments

fontset	A string giving the name of a set of fonts (e.g. "Liberation" or "Bitstream Vera"). Use fontset_list() to obtain the list of fontsets registered in your session.
variant	Font variant, as per Fontconfig's nomenclature. Use fontset_variants() to find out which variants are available for a fontset.
style	Font style, as per Fontconfig's nomenclature. Use fontset_styles() to find out which variants are available for a fontset.

Examples

```
font("Bitstream Vera", "Sans", "Roman")

f <- fonts("Liberation")
str(f, 1)

font_symbol("Symbola")
```

fontset_info	<i>Fontset metadata</i>
--------------	-------------------------

Description

This returns a data frame containing metadata about fonts. The columns "variant" and "style" correspond to the properties of the font as reported by fontconfig. The columns "family", "italic" and "bold" provide information in terms of R nomenclature of fonts. The column "base" gives the base filename of the font. See [fontset_list\(\)](#) to obtain a list of available fontsets.

Usage

```
fontset_info(fontset)
```

Arguments

fontset	A string giving the name of a set of fonts (e.g. "Liberation" or "Bitstream Vera"). Use fontset_list() to obtain the list of fontsets registered in your session.
---------	---

See Also

[fontset_list\(\)](#), [font_families\(\)](#), [font_variants\(\)](#)

fontset_list	<i>Get list of all registered fonts</i>
--------------	---

Description

Get list of all registered fonts

Usage

```
fontset_list()
```

fontset_register	<i>Register a font to fontquiver</i>
------------------	--------------------------------------

Description

Register a font to fontquiver

Usage

```
fontset_register(fontset, getter, files)
```

Arguments

fontset	Name of the font set.
getter	Constructor of font objects.
files	Data structure containing font information.

fontset_variants	<i>Fontset variants and styles</i>
------------------	------------------------------------

Description

These functions return the variants and the styles available for a given fontset.

Usage

```
fontset_variants(fontset)

fontset_styles(fontset, variant)
```

Arguments

fontset	A string giving the name of a set of fonts (e.g. "Liberation" or "Bitstream Vera"). Use fontset_list() to obtain the list of fontsets registered in your session.
variant	Font variant, as per Fontconfig's nomenclature. Use fontset_variants() to find out which variants are available for a fontset.

Examples

```
fontset_variants("Liberation")
fontset_styles("Bitstream Vera", "Sans Mono")
```

font_families	<i>Font families and faces</i>
---------------	--------------------------------

Description

Returns families or faces as font files for a given font set. The files are organised by R nomenclature of families ("sans", "serif", "mono", and "symbol") and faces ("plain", "italic", "bold", and "bolditalic"). When a font does not have a combination, NA is reported.

Usage

```
font_families(fontset)

font_faces(fontset, family = c("sans", "serif", "mono", "symbol"))
```

Arguments

- fontset A string giving the name of a set of fonts (e.g. "Liberation" or "Bitstream Vera"). Use [fontset_list\(\)](#) to obtain the list of fontsets registered in your session.
- family One of "sans", "serif", "mono" or "symbol".

Details

Note that the fonts returned by `font_faces()` and `font_families()` are constrained by R's nomenclature of fonts and are thus a subset of those returned by `font_variants()` and `font_styles()`.

See Also

```
font\_variants\(\), fonts\(\)
```

Examples

```
font_families("Bitstream Vera")$mono$bold
font_faces("Bitstream Vera", "mono")
```

font_variants	<i>Font variants</i>
---------------	----------------------

Description

`font_variants()` returns a tree of fonts organised as per `fontconfig`'s nomenclature. The first level contains variants of a font and the second level contains font styles.

Usage

```
font_variants(fontset)

font_styles(fontset, variant)
```

Arguments

fontset	A string giving the name of a set of fonts (e.g. "Liberation" or "Bitstream Vera"). Use <code>fontset_list()</code> to obtain the list of fontsets registered in your session.
variant	Font variant, as per Fontconfig's nomenclature. Use <code>fontset_variants()</code> to find out which variants are available for a fontset.

Details

See also `font_families()` and `font_faces()` for similar collections organised according to R nomenclature of fonts. Note that variants and styles are a super set of R families and faces and may contain more fonts.

See Also

`font_families()`, `fonts()`

Examples

```
font_variants("Bitstream Vera")
font_variants("Bitstream Vera")$Sans$Oblique
font_styles("Bitstream Vera", "Sans")
```

htmlFontDependency	<i>Include font as CSS dependency</i>
--------------------	---------------------------------------

Description

Include font as CSS dependency

Usage

```
htmlFontDependency(...)
```

Arguments

...	Fonts or collections of fonts. See <code>font_families()</code> and <code>font_variants()</code> for creating collections of fonts. You can also supply lists of individual fonts as returned by <code>fonts()</code> .
-----	---

Examples

```
# Create an htmlDependency object:
dep <- htmlFontDependency(font_families("Bitstream Vera"))

# Use the fonts in your dependent css or html files. For example:
# body {
#   font-family: 'Bitstream Vera Sans Mono', courier;
# }
```

splice_fonts

Splice fonts and font collections

Description

splice_fonts() Reduces its arguments to a flat list. It accepts indistinctly font objects, lists of font objects (obtained with [fonts\(\)](#)), or collections of fonts produced by [font_variants\(\)](#) or [font_families\(\)](#). Duplicate fonts are removed from the result.

Usage

```
splice_fonts(...)
```

Arguments

... Fonts or collections of fonts. See [font_families\(\)](#) and [font_variants\(\)](#) for creating collections of fonts. You can also supply lists of individual fonts as returned by [fonts\(\)](#).

Examples

```
splice_fonts(font("Bitstream Vera", "Sans", "Oblique"), font_faces("Bitstream Vera", "mono"))
```

Index

`font (fonts)`, [2](#)
`font_faces`, [6](#)
`font_faces (font_families)`, [5](#)
`font_families`, [3](#), [5](#), [5](#), [6](#), [7](#)
`font_styles`, [5](#)
`font_styles (font_variants)`, [5](#)
`font_symbol (fonts)`, [2](#)
`font_variants`, [3](#), [5](#), [5](#), [6](#), [7](#)
`fonts`, [2](#), [5–7](#)
`fontset_info`, [3](#)
`fontset_list`, [2](#), [3](#), [3](#), [4–6](#)
`fontset_register`, [4](#)
`fontset_styles`, [2](#)
`fontset_styles (fontset_variants)`, [4](#)
`fontset_variants`, [2](#), [4](#), [4](#), [6](#)

`htmlFontDependency`, [6](#)

`splice_fonts`, [7](#)