

Package ‘zipangu’

November 29, 2019

Title Japanese Utility Functions and Data

Version 0.1.0

Description Some data treated by the Japanese R user
require unique operations and processing. These are caused
by address, Kanji, and traditional year representations.
'zipangu' transforms specific to Japan into something more general one.

URL <https://uribo.github.io/zipangu>,
<https://github.com/uribo/zipangu>

BugReports <https://github.com/uribo/zipangu/issues>

Depends R (\geq 3.2)

License MIT + file LICENSE

Imports magrittr (\geq 1.5),
dplyr (\geq 0.8.3),
purrr (\geq 0.3.3),
lifecycle (\geq 0.1.0),
rlang (\geq 0.4.0),
stringr (\geq 1.4.0)

Suggests testthat (\geq 2.1.0),
covr (\geq 3.4.0)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.0.1

R topics documented:

convert_jyear	2
jpnprefs	2
kansuji2arabic	3
separate_address	3

Index	5
-------	---

convert_jyear	<i>Convert Japanese imperial year to Anno Domini</i>
---------------	--

Description

Maturing

Usage

```
convert_jyear(jyear)
```

Arguments

jyear Japanese imperial year (jyear). Kanji or Roman character

Examples

```
convert_jyear("R1")
convert_jyear("Heisei2")
convert_jyear("\u5e73\u6210\u5143\u5e74")
convert_jyear(c("\u662d\u548c10\u5e74", "\u5e73\u621014\u5e74"))
convert_jyear(kansuji2arabic_all("\u5e73\u6210\u4e09\u5e74"))
```

jpnprefs	<i>Prefectural informations in Japan</i>
----------	--

Description

Prefectures dataset.

Usage

```
jpnprefs
```

Format

A tibble with 47 rows 5 variables:

- jis_code: jis code
- prefecture_kanji: prefecture names
- prefecture: prefecture names
- region: region
- major_island:

Examples

```
jpnprefs
```

kansuji2arabic	<i>Convert kansuji character to arabic</i>
----------------	--

Description

Experimental Converts a given Kansuji element such as Ichi (1) and Nana (7) to an Arabic. `kansuji2arabic_all()` converts only Kansuji in the string.

Usage

```
kansuji2arabic(str, convert = TRUE, .under = Inf)
```

```
kansuji2arabic_all(str, ...)
```

Arguments

<code>str</code>	Input vector.
<code>convert</code>	If FALSE, will return as numeric. The default value is TRUE, and numeric values are treated as strings.
<code>.under</code>	Number scale to be converted. The default value is infinity.
<code>...</code>	Other arguments to carry over to <code>kansuji2arabic()</code>

Value

a character or numeric.

Examples

```
kansuji2arabic("\u4e00")
kansuji2arabic(c("\u4e00", "\u767e"))
kansuji2arabic(c("\u4e00", "\u767e"), convert = FALSE)
# Keep Kansuji over 1000.
kansuji2arabic(c("\u4e00", "\u767e", "\u5343"), .under = 1000)
# Convert all character
kansuji2arabic_all("\u3007\u4e00\u4e8c\u4e09\u56db\u4e94\u516d\u4e03\u516b\u4e5d\u5341")
kansuji2arabic_all("\u516b\u4e01\u76ee")
```

separate_address	<i>Separate address elements</i>
------------------	----------------------------------

Description

Experimental Parses and decomposes address string into elements of prefecture, city, and lower address.

Usage

```
separate_address(str)
```

Arguments

`str` Input vector. address string.

Value

A list of elements that make up an address.

Examples

```
separate_address("\u5317\u6d77\u9053\u672d\u5e4c\u5e02\u4e2d\u592e\u533a")
```

Index

*Topic **datasets**

jpnprefs, [2](#)

convert_jyear, [2](#)

jpnprefs, [2](#)

kansuji2arabic, [3](#)

kansuji2arabic_all (kansuji2arabic), [3](#)

separate_address, [3](#)