

Package: cstime (via r-universe)

July 1, 2026

Title Date and Time Functions for Public Health Purposes

Version 2025.10.13

Description Provides easy and consistent time conversion for public health purposes. The time conversion functions provided here are between date, ISO week, ISO yearweek, ISO year, calendar month/year, season, season week.

License MIT + file LICENSE

URL <https://niphr.github.io/cstime/>, <https://github.com/niphr/cstime>

BugReports <https://github.com/niphr/cstime/issues>

Encoding UTF-8

LazyData true

LazyDataCompression xz

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Depends R (>= 3.5.0)

Imports data.table, magrittr

Suggests rmarkdown, ggplot2, knitr, testthat (>= 3.0.0), readxl, glue, lubridate

Config/testthat/edition 3

VignetteBuilder knitr

Config/Needs/website niphr/cstemplate

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

Date/Publication 2026-07-01 09:02:24 UTC

RemoteUrl <https://github.com/niphr/cstime>

RemoteRef HEAD

RemoteSha a85394db70799ccb89c0447026c53c9db946933d

Contents

date_to_calmonth_c	3
date_to_calmonth_n	3
date_to_calyear_c	4
date_to_calyear_n	5
date_to_calyearmonth_c	6
date_to_isoquarter_c	7
date_to_isoquarter_n	8
date_to_isoweek_c	9
date_to_isoweek_n	10
date_to_isoyear_c	11
date_to_isoyear_n	12
date_to_isoyearquarter_c	13
date_to_isoyearweek_c	14
date_to_season_c	15
date_to_seasonweek_n	15
dates_by_isoyearweek	16
isoweek_to_seasonweek_n	17
isoyear_to_last_date	18
isoyear_to_last_isoweek_n	19
isoyear_to_last_isoyearweek_c	20
isoyearweek_to_isoquarter_c	21
isoyearweek_to_isoquarter_n	21
isoyearweek_to_isoweek_c	22
isoyearweek_to_isoweek_n	23
isoyearweek_to_isoyear_c	24
isoyearweek_to_isoyear_n	25
isoyearweek_to_isoyearquarter_c	25
isoyearweek_to_last_date	26
isoyearweek_to_season_c	27
isoyearweek_to_seasonweek_n	28
keep_sundays	28
keep_sundays_and_latest_date	29
keep_sundays_and_latest_date_internal	30
nor_workdays_by_date	30
nor_workdays_by_isoyearweek	31
now_c	32
season_to_last_date	32
seasonweek_to_isoweek_c	33
seasonweek_to_isoweek_n	34

Index

35

date_to_calmonth_c *Date to calendar month (character)*

Description

Converts a date to its calendar month number, returned as a zero-padded character string.

Usage

```
date_to_calmonth_c(x = lubridate::today())

## Default S3 method:
date_to_calmonth_c(x)

## S3 method for class 'character'
date_to_calmonth_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_calmonth_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The month is returned as two digits, "01" for January through "12" for December.

Value

Calendar month as a character vector ("01" to "12").

Examples

```
date_to_calmonth_c(as.Date("2021-08-11"))
date_to_calmonth_c("2021-01-01")
```

date_to_calmonth_n *Date to calendar month (numeric)*

Description

Converts a date to its calendar month number, returned as a number.

Usage

```
date_to_calmonth_n(x = lubridate::today())

## Default S3 method:
date_to_calmonth_n(x)

## S3 method for class 'character'
date_to_calmonth_n(x = lubridate::today())

## S3 method for class 'Date'
date_to_calmonth_n(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Value

Calendar month as an integer vector (1 for January to 12 for December).

Examples

```
date_to_calmonth_n(as.Date("2021-08-11"))
date_to_calmonth_n("2021-01-01")
```

date_to_calyear_c *Date to calendar year (character)*

Description

Converts a date to its calendar (Gregorian) year, returned as a character string.

Usage

```
date_to_calyear_c(x = lubridate::today())

## Default S3 method:
date_to_calyear_c(x)

## S3 method for class 'character'
date_to_calyear_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_calyear_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

Unlike the ISO year (see [date_to_isoyear_c\(\)](#)), the calendar year is taken directly from the date, so 2021-01-01 is calendar year "2021".

Value

Calendar year as a character vector (e.g. "2021").

Examples

```
date_to_calyear_c(as.Date("2021-08-11"))
date_to_calyear_c("2021-01-01")
```

date_to_calyear_n	<i>Date to calendar year (numeric)</i>
-------------------	--

Description

Converts a date to its calendar (Gregorian) year, returned as a number.

Usage

```
date_to_calyear_n(x = lubridate::today())

## Default S3 method:
date_to_calyear_n(x)

## S3 method for class 'character'
date_to_calyear_n(x = lubridate::today())

## S3 method for class 'Date'
date_to_calyear_n(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

Unlike the ISO year (see [date_to_isoyear_n\(\)](#)), the calendar year is taken directly from the date, so 2021-01-01 is calendar year 2021.

Value

Calendar year as an integer vector (e.g. 2021).

Examples

```
date_to_calyear_n(as.Date("2021-08-11"))
date_to_calyear_n("2021-01-01")
```

date_to_calyearmonth_c

Date to calendar yearmonth (character)

Description

Converts a date to a combined calendar year and month string of the form "yyyy-Mmm".

Usage

```
date_to_calyearmonth_c(x = lubridate::today())
```

```
## Default S3 method:
date_to_calyearmonth_c(x)
```

```
## S3 method for class 'character'
date_to_calyearmonth_c(x = lubridate::today())
```

```
## S3 method for class 'Date'
date_to_calyearmonth_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The output combines the calendar year and the zero-padded calendar month, separated by "-M", for example "2021-M08".

Value

Calendar yearmonth as a character vector (e.g. "2021-M08").

Examples

```
date_to_calyearmonth_c(as.Date("2021-08-11"))
date_to_calyearmonth_c("2021-01-01")
```

date_to_isoquarter_c *Date to ISO quarter (character)*

Description

Converts a date to an ISO-week-based quarter (1 to 4), returned as a character string.

Usage

```
date_to_isoquarter_c(x = lubridate::today())

## Default S3 method:
date_to_isoquarter_c(x)

## S3 method for class 'character'
date_to_isoquarter_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoquarter_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The quarter is derived from the ISO week. See [date_to_isoquarter_n\(\)](#) for the week-to-quarter boundaries.

Value

ISO quarter as a character vector (e.g. "3").

Examples

```
date_to_isoquarter_c(as.Date("2021-08-11"))
date_to_isoquarter_c("2021-01-01")
```

date_to_isoquarter_n *Date to ISO quarter (numeric)*

Description

Converts a date to an ISO-week-based quarter (1 to 4), returned as a number.

Usage

```
date_to_isoquarter_n(x = lubridate::today())

## Default S3 method:
date_to_isoquarter_n(x)

## S3 method for class 'character'
date_to_isoquarter_n(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoquarter_n(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The quarter is derived from the ISO week rather than the calendar month: weeks 1 to 13 are quarter 1, weeks 14 to 26 are quarter 2, weeks 27 to 39 are quarter 3, and weeks 40 onwards (including week 53 in long ISO years) are quarter 4.

Value

ISO quarter as an integer vector (1 to 4).

Examples

```
date_to_isoquarter_n(as.Date("2021-08-11"))
date_to_isoquarter_n("2021-01-01")
```

date_to_isoweek_c	<i>Date to ISO week (character)</i>
-------------------	-------------------------------------

Description

Converts a date to its ISO 8601 week number, returned as a zero-padded character string.

Usage

```
date_to_isoweek_c(x = lubridate::today())

## Default S3 method:
date_to_isoweek_c(x)

## S3 method for class 'character'
date_to_isoweek_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoweek_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

ISO weeks run Monday to Sunday and are numbered 01 to 52 or 53. Week 01 is the week containing the first Thursday of the ISO year. The week is returned as two digits, e.g. "01" or "53".

Value

ISO week as a character vector (e.g. "32").

Examples

```
date_to_isoweek_c(as.Date("2021-08-11"))
date_to_isoweek_c("2021-01-01")
```

date_to_isoweek_n *Date to ISO week (numeric)*

Description

Converts a date to its ISO 8601 week number, returned as a number.

Usage

```
date_to_isoweek_n(x = lubridate::today())

## Default S3 method:
date_to_isoweek_n(x)

## S3 method for class 'character'
date_to_isoweek_n(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoweek_n(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

ISO weeks run Monday to Sunday and are numbered 1 to 52 or 53. Week 1 is the week containing the first Thursday of the ISO year.

Value

ISO week as an integer vector (1 to 53).

Examples

```
date_to_isoweek_n(as.Date("2021-08-11"))
date_to_isoweek_n("2021-01-01")
```

date_to_isoyear_c *Date to ISO year (character)*

Description

Converts a date to its ISO 8601 week-based year, returned as a character string.

Usage

```
date_to_isoyear_c(x = lubridate::today())

## Default S3 method:
date_to_isoyear_c(x)

## S3 method for class 'character'
date_to_isoyear_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoyear_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The ISO 8601 week-based year is not always the same as the calendar year. ISO weeks run Monday to Sunday, and week 1 is the week containing the year's first Thursday. As a result the first days of January can belong to the last ISO week of the previous year, and the last days of December can belong to ISO week 1 of the following year. For example 2021-01-01 is a Friday that falls in ISO week 53 of ISO year 2020.

Value

ISO year as a character vector (e.g. "2021").

Examples

```
date_to_isoyear_c(as.Date("2021-08-11"))
date_to_isoyear_c("2021-01-01")
```

date_to_isoyear_n *Date to ISO year (numeric)*

Description

Converts a date to its ISO 8601 week-based year, returned as a number.

Usage

```
date_to_isoyear_n(x = lubridate::today())

## Default S3 method:
date_to_isoyear_n(x)

## S3 method for class 'character'
date_to_isoyear_n(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoyear_n(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The ISO 8601 week-based year can differ from the calendar year near the start and end of January and December. See [date_to_isoyear_c\(\)](#) for the rules used to assign weeks and years.

Value

ISO year as an integer vector (e.g. 2021).

Examples

```
date_to_isoyear_n(as.Date("2021-08-11"))
date_to_isoyear_n("2021-01-01")
```

date_to_isoyearquarter_c
Date to ISO yearquarter (character)

Description

Converts a date to a combined ISO year and quarter string of the form "yyyy-Qn".

Usage

```
date_to_isoyearquarter_c(x = lubridate::today())

## Default S3 method:
date_to_isoyearquarter_c(x)

## S3 method for class 'character'
date_to_isoyearquarter_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoyearquarter_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The output combines the ISO year (see [date_to_isoyear_c\(\)](#)) and the ISO-week-based quarter (see [date_to_isoquarter_c\(\)](#)), for example "2021-Q3".

Value

ISO yearquarter as a character vector (e.g. "2021-Q3").

Examples

```
date_to_isoyearquarter_c(as.Date("2021-08-11"))
date_to_isoyearquarter_c("2021-01-01")
```

date_to_isoyearweek_c *Date to ISO yearweek (character)*

Description

Converts a date to a combined ISO 8601 year and week string of the form "yyyy-ww".

Usage

```
date_to_isoyearweek_c(x = lubridate::today())

## Default S3 method:
date_to_isoyearweek_c(x)

## S3 method for class 'character'
date_to_isoyearweek_c(x = lubridate::today())

## S3 method for class 'Date'
date_to_isoyearweek_c(x = lubridate::today())
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The output combines the ISO year (see [date_to_isoyear_c\(\)](#)) and the zero-padded ISO week (see [date_to_isoweek_c\(\)](#)), separated by a hyphen, for example "2021-32". Because the ISO year can differ from the calendar year, 2021-01-01 maps to "2020-53".

Value

ISO yearweek as a character vector (e.g. "2021-32").

Examples

```
date_to_isoyearweek_c(as.Date("2021-08-11"))
date_to_isoyearweek_c("2021-01-01")
```

date_to_season_c *Date to season (character)*

Description

Maps a date to the surveillance season it belongs to, written as "yyyy/yyyy".

Usage

```
date_to_season_c(x)
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The date is first converted to an ISO yearweek with [date_to_isoyearweek_c\(\)](#) and then to a season with [isoyearweek_to_season_c\(\)](#). Seasons start at ISO week 35, so dates in early January belong to the season that began the previous calendar year.

Value

Season as a character vector (e.g. "2020/2021").

Examples

```
date_to_season_c(c("2021-01-01", "2021-12-01"))
date_to_season_c(as.Date("2021-09-01"))
```

date_to_seasonweek_n *Date to season week (numeric)*

Description

Maps a date to its position within the surveillance season, where season week 1 corresponds to ISO week 35.

Usage

```
date_to_seasonweek_n(x)
```

Arguments

x A Date object, or a character string in the format 'yyyy-mm-dd'.

Details

The date is first converted to an ISO yearweek with `date_to_isoyearweek_c()` and then to a season week with `isoyearweek_to_seasonweek_n()`. As with `isoweek_to_seasonweek_n()`, ISO week 53 maps to the half-step season week 18.5.

Value

Season week as a numeric vector.

Examples

```
date_to_seasonweek_n(c("2021-01-01", "2021-12-01"))
date_to_seasonweek_n(as.Date("2021-09-01"))
```

dates_by_isoyearweek *Dates of each weekday within ISO yearweeks*

Description

A lookup table giving, for every ISO yearweek between 1900 and 2100, the date of each day of that week (Monday to Sunday) plus convenience list-columns that group those dates.

Usage

```
dates_by_isoyearweek
```

Format

A `data.table::data.table` with one row per ISO yearweek and the following columns:

isoyear Integer. ISO year.

isoyearweek Character. ISO yearweek, "yyyy-ww".

mon Date of Monday.

tue Date of Tuesday.

wed Date of Wednesday.

thu Date of Thursday.

fri Date of Friday.

sat Date of Saturday.

sun Date of Sunday.

weekdays List column. The dates Monday to Friday.

weekend List column. The dates Saturday and Sunday.

days List column. The dates Monday to Sunday.

Source

Generated from a daily date sequence using ISO 8601 week rules. See data-raw/1_gen-data.R in the package source.

Examples

```
library(data.table)
# Look up the Monday and Sunday of two ISO yearweeks
dates_by_isoyearweek[isoyearweek %in% c("2021-01", "2021-02"), .(isoyearweek, mon, sun)]

# Constructing a vector of dates without removing the Date class
do.call("c", dates_by_isoyearweek[isoyearweek %in% c("2021-01", "2021-02")]$weekdays)
```

```
isoweek_to_seasonweek_n
```

ISO week to season week (numeric)

Description

Maps an ISO week number to its position within the surveillance season, where season week 1 corresponds to ISO week 35.

Usage

```
isoweek_to_seasonweek_n(x)

## Default S3 method:
isoweek_to_seasonweek_n(x)

## S3 method for class 'character'
isoweek_to_seasonweek_n(x)

## S3 method for class 'numeric'
isoweek_to_seasonweek_n(x)
```

Arguments

x ISO week as a number between 1 and 53.

Details

Surveillance seasons start at ISO week 35, so ISO week 35 is season week 1, ISO week 36 is season week 2, and so on, wrapping around the new year. ISO week 53 (which only occurs in long ISO years) maps to the half-step season week 18.5 so that the surrounding weeks keep consistent numbering.

Value

Season week as a numeric vector (ISO week 53 returns 18.5).

Examples

```
isoweek_to_seasonweek_n(35)
isoweek_to_seasonweek_n(c(31, 53))
```

```
isoyear_to_last_date  ISO year to last date (Sunday)
```

Description

Returns the date of the Sunday that ends the last ISO week of a given ISO year.

Usage

```
isoyear_to_last_date(x)

## Default S3 method:
isoyear_to_last_date(x)

## S3 method for class 'character'
isoyear_to_last_date(x)

## S3 method for class 'numeric'
isoyear_to_last_date(x)
```

Arguments

x ISO year as a number or character string, e.g. 2020 or "2020".

Details

ISO weeks end on Sunday, so the returned date is the Sunday of the final ISO week. Because ISO years and calendar years are not aligned, this date can fall in early January of the following calendar year (for example the last date of ISO year 2020 is 2021-01-03). The year is accepted as either a number or a character string.

Value

A `base::Date` vector giving the last Sunday of each ISO year.

Examples

```
isoyear_to_last_date(c(2019, 2020, 2021))
isoyear_to_last_date("2020")
```

```
isoyear_to_last_isoweek_n
      ISO year to last ISO week (numeric)
```

Description

Returns the number of the last ISO week in a given ISO year, that is, the count of ISO weeks in that year.

Usage

```
isoyear_to_last_isoweek_n(x)

## Default S3 method:
isoyear_to_last_isoweek_n(x)

## S3 method for class 'character'
isoyear_to_last_isoweek_n(x)

## S3 method for class 'numeric'
isoyear_to_last_isoweek_n(x)
```

Arguments

x ISO year as a number or character string, e.g. 2020 or "2020".

Details

This is 52 for most years and 53 for long ISO years such as 2020. The year is accepted as either a number or a character string.

Value

Last ISO week of the year as an integer vector (52 or 53).

Examples

```
isoyear_to_last_isoweek_n(c(2019, 2020, 2021))
isoyear_to_last_isoweek_n("2020")
```

isoyear_to_last_isoyearweek_c
ISO year to last ISO yearweek (character)

Description

Returns the last ISO yearweek of a given ISO year as a "yyyy-ww" string.

Usage

```
isoyear_to_last_isoyearweek_c(x)

## Default S3 method:
isoyear_to_last_isoyearweek_c(x)

## S3 method for class 'character'
isoyear_to_last_isoyearweek_c(x)

## S3 method for class 'numeric'
isoyear_to_last_isoyearweek_c(x)
```

Arguments

x ISO year as a number or character string, e.g. 2020 or "2020".

Details

Most ISO years have 52 weeks, so the result is usually "yyyy-52". ISO years that contain 53 weeks (such as 2020) instead return "yyyy-53". The year is accepted as either a number or a character string.

Value

Last ISO yearweek of the year as a character vector (e.g. "2020-53").

Examples

```
isoyear_to_last_isoyearweek_c(c(2019, 2020, 2021))
isoyear_to_last_isoyearweek_c("2020")
```

```
isoyearweek_to_isoquarter_c
      ISO yearweek to ISO quarter (character)
```

Description

Maps an ISO yearweek to its ISO-week-based quarter (1 to 4), returned as a character string.

Usage

```
isoyearweek_to_isoquarter_c(x)

## Default S3 method:
isoyearweek_to_isoquarter_c(x)

## S3 method for class 'character'
isoyearweek_to_isoquarter_c(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The quarter is derived from the ISO week part of the input. See [isoyearweek_to_isoquarter_n\(\)](#) for the week-to-quarter boundaries.

Value

ISO quarter as a character vector (e.g. "2").

Examples

```
isoyearweek_to_isoquarter_c("2020-19")
```

```
isoyearweek_to_isoquarter_n
      ISO yearweek to ISO quarter (numeric)
```

Description

Maps an ISO yearweek to its ISO-week-based quarter (1 to 4), returned as a number.

Usage

```
isoyearweek_to_isoquarter_n(x)

## Default S3 method:
isoyearweek_to_isoquarter_n(x)

## S3 method for class 'character'
isoyearweek_to_isoquarter_n(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The quarter is derived from the ISO week part of the input. Weeks 1 to 13 are quarter 1, weeks 14 to 26 are quarter 2, weeks 27 to 39 are quarter 3, and weeks 40 onwards (including week 53) are quarter 4.

Value

ISO quarter as an integer vector (1 to 4).

Examples

```
isoyearweek_to_isoquarter_n("2020-19")
```

```
isoyearweek_to_isoweek_c
                          ISO yearweek to ISO week (character)
```

Description

Extracts the ISO week from an ISO yearweek string and returns it as a character string.

Usage

```
isoyearweek_to_isoweek_c(x)

## Default S3 method:
isoyearweek_to_isoweek_c(x)

## S3 method for class 'character'
isoyearweek_to_isoweek_c(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The input is split on the hyphen into year and week, and the week part is returned. The year part is ignored.

Value

ISO week as a character vector (e.g. "19").

Examples

```
isoyearweek_to_isoweek_c("2020-19")
```

```
isoyearweek_to_isoweek_n
      ISO yearweek to ISO week (numeric)
```

Description

Extracts the ISO week from an ISO yearweek string and returns it as a number.

Usage

```
isoyearweek_to_isoweek_n(x)

## Default S3 method:
isoyearweek_to_isoweek_n(x)

## S3 method for class 'character'
isoyearweek_to_isoweek_n(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The input is split on the hyphen into year and week, and the week part is returned. The year part is ignored.

Value

ISO week as an integer vector (1 to 53).

Examples

```
isoyearweek_to_isoweek_n("2020-19")
```

```
isoyearweek_to_isoyear_c
```

ISO yearweek to ISO year (character)

Description

Extracts the ISO year from an ISO yearweek string and returns it as a character string.

Usage

```
isoyearweek_to_isoyear_c(x)
```

```
## Default S3 method:
```

```
isoyearweek_to_isoyear_c(x)
```

```
## S3 method for class 'character'
```

```
isoyearweek_to_isoyear_c(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The input is split on the hyphen into year and week, and the year part is returned. The week part is ignored.

Value

ISO year as a character vector (e.g. "2020").

Examples

```
isoyearweek_to_isoyear_c("2020-10")
```

```
isoyearweek_to_isoyear_n
      ISO yearweek to ISO year (numeric)
```

Description

Extracts the ISO year from an ISO yearweek string and returns it as a number.

Usage

```
isoyearweek_to_isoyear_n(x)

## Default S3 method:
isoyearweek_to_isoyear_n(x)

## S3 method for class 'character'
isoyearweek_to_isoyear_n(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The input is split on the hyphen into year and week, and the year part is returned. The week part is ignored.

Value

ISO year as an integer vector (e.g. 2020).

Examples

```
isoyearweek_to_isoyear_n("2020-10")
```

```
isoyearweek_to_isoyearquarter_c
      ISO yearweek to ISO yearquarter (character)
```

Description

Maps an ISO yearweek to a combined ISO year and quarter string of the form "yyyy-Qn".

Usage

```
isoyearweek_to_isoyearquarter_c(x)

## Default S3 method:
isoyearweek_to_isoyearquarter_c(x)

## S3 method for class 'character'
isoyearweek_to_isoyearquarter_c(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

The output keeps the year part of the input and appends the ISO-week-based quarter (see [isoyearweek_to_isoquarter_c\(\)](#)) for example "2020-Q2".

Value

ISO yearquarter as a character vector (e.g. "2020-Q2").

Examples

```
isoyearweek_to_isoyearquarter_c("2020-19")
```

```
isoyearweek_to_last_date
                          ISO yearweek to last date (Sunday)
```

Description

Returns the date of the Sunday that ends a given ISO yearweek.

Usage

```
isoyearweek_to_last_date(x)

## Default S3 method:
isoyearweek_to_last_date(x)

## S3 method for class 'character'
isoyearweek_to_last_date(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2020-19" for the 19th ISO week of 2020.

Details

ISO weeks run Monday to Sunday, so the returned date is the Sunday of the supplied yearweek.

Value

A `base::Date` vector giving the Sunday of each ISO yearweek.

Examples

```
isoyearweek_to_last_date(c("2019-19", "2020-01"))
```

```
isoyearweek_to_season_c
```

```
ISO yearweek to season (character)
```

Description

Maps an ISO yearweek to the surveillance season it belongs to, written as "yyyy/yyyy".

Usage

```
isoyearweek_to_season_c(x)
```

```
## Default S3 method:
```

```
isoyearweek_to_season_c(x)
```

```
## S3 method for class 'character'
```

```
isoyearweek_to_season_c(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2021-01".

Details

Seasons start at ISO week 35 (season week 1). ISO weeks 35 and later belong to the season beginning in that calendar year, while earlier weeks belong to the season that began the previous calendar year. For example "2021-01" falls in season "2020/2021" and "2021-50" falls in season "2021/2022".

Value

Season as a character vector (e.g. "2020/2021").

Examples

```
isoyearweek_to_season_c(c("2021-01", "2021-50"))
```

```
isoyearweek_to_seasonweek_n
      ISO yearweek to season week (numeric)
```

Description

Maps an ISO yearweek to its position within the surveillance season, where season week 1 corresponds to ISO week 35.

Usage

```
isoyearweek_to_seasonweek_n(x)
```

Arguments

x ISO yearweek as a character string of the form "yyyy-ww", e.g. "2021-01".

Details

The ISO week is extracted from the yearweek string and then converted with `isoweek_to_seasonweek_n()`, so the same season-week numbering and the 18.5 half-step for ISO week 53 apply.

Value

Season week as a numeric vector.

Examples

```
isoyearweek_to_seasonweek_n(c("2021-01", "2021-35"))
```

```
keep_sundays            keep_sundays
```

Description

```
keep_sundays
```

Usage

```
keep_sundays(dates, format = "Uke isoweek_c-1/isoweek_c", keep_delete = TRUE)
```

Arguments

dates	dates (later)
format	format (later)
keep_delete	keep_delete (later)

Value

Label: delete or not

keep_sundays_and_latest_date
keep_sundays_and_latest_date

Description

keep_sundays_and_latest_date

Usage

```
keep_sundays_and_latest_date(  
  dates,  
  format = "Uke isoweek_c-1/isoweek_c",  
  keep_delete = TRUE  
)
```

Arguments

dates	dates (later)
format	format (later)
keep_delete	keep_delete (later)

Value

Date with level: delete or not

```
keep_sundays_and_latest_date_internal
      keep_sundays_and_latest_date_internal
```

Description

keep_sundays_and_latest_date_internal

Usage

```
keep_sundays_and_latest_date_internal(
  dates,
  format = "Uke isoweek_c-1/isoweek_c",
  keep_delete = TRUE,
  keep_latest_date = TRUE
)
```

Arguments

dates	dates (later)
format	format (later)
keep_delete	keep_delet (later)
keep_latest_date	keep_latest_date (later)

```
nor_workdays_by_date  Norwegian workdays and holidays by date
```

Description

A daily calendar flagging Norwegian public holidays, weekends and workdays for each date in the covered range.

Usage

```
nor_workdays_by_date
```

Format

A `data.table::data.table` with one row per date and the following columns:

date Date.

day_of_week Integer. 1 = Monday, 7 = Sunday.

mon_to_fri Integer. 1 if Monday to Friday, 0 if Saturday or Sunday.

sat_to_sun Integer. 1 if Saturday or Sunday, 0 if Monday to Friday.
public_holiday Integer. 1 if a public holiday (helligdag), 0 otherwise.
freeday Integer. 1 if a public holiday or Saturday/Sunday, 0 otherwise.
workday Integer. 1 if freeday is 0, 0 if freeday is 1.

Source

Derived from the Norwegian public-holiday list in `inst/rawdata/norway_holidays.xlsx`. See `data-raw/1_gen-data.R` in the package source.

Examples

```
head(nor_workdays_by_date)

# Count the workdays in the dataset
sum(nor_workdays_by_date$workday)
```

```
nor_workdays_by_isoyearweek
      Norwegian workdays and holidays by ISO yearweek
```

Description

For each complete (7-day) ISO yearweek, the proportion of days that are public holidays, free days and workdays in Norway.

Usage

```
nor_workdays_by_isoyearweek
```

Format

A `data.table::data.table` with one row per ISO yearweek and the following columns:

isoyearweek Character. ISO yearweek, "yyyy-ww".
public_holiday The proportion of days in the ISO yearweek that are public holidays.
freeday The proportion of days in the ISO yearweek that are public holidays or Saturday/Sunday.
workday 1 minus freeday.

Source

Aggregated from [nor_workdays_by_date](#). See `data-raw/1_gen-data.R` in the package source.

Examples

```
head(nor_workdays_by_isoyearweek)

# ISO yearweeks that contain at least one public holiday
head(nor_workdays_by_isoyearweek[nor_workdays_by_isoyearweek$public_holiday > 0, ])
```

now_c	<i>Current time as character</i>
-------	----------------------------------

Description

Returns the current system time formatted as a character string.

Usage

```
now_c(format = "%Y-%m-%d %H:%M:%S")
```

Arguments

format A format string passed to `base::format()`. Defaults to "%Y-%m-%d %H:%M:%S".

Details

The current time is taken from `base::Sys.time()` and formatted with the supplied format string, which uses the conversion codes documented in `base::strptime()`.

Value

The current time as a single character string.

Examples

```
now_c()
now_c(format = "%Y-%m-%d")
```

season_to_last_date	<i>Season to last date (Sunday)</i>
---------------------	-------------------------------------

Description

Returns the date of the Sunday that ends a given season.

Usage

```
season_to_last_date(x)

## Default S3 method:
season_to_last_date(x)

## S3 method for class 'character'
season_to_last_date(x)
```

Arguments

x Season as a character string of the form "yyyy/yyyy", e.g. "2019/2020".

Details

A season is written "yyyy/yyyy" where the two years are consecutive (for example "2019/2020"). Seasons are aligned to ISO weeks, with season week 1 starting at ISO week 35; the season therefore ends in late summer of the second year. The returned date is the Sunday of the final week of the season.

Value

A `base::Date` vector giving the last Sunday of each season.

Examples

```
season_to_last_date(c("2019/2020", "2020/2021"))
```

```
seasonweek_to_isoweek_c
```

Season week to ISO week (character)

Description

Maps a season week number back to its ISO week, returned as a zero-padded character string. This is the inverse of `isoweek_to_seasonweek_n()`.

Usage

```
seasonweek_to_isoweek_c(x)

## Default S3 method:
seasonweek_to_isoweek_c(x)

## S3 method for class 'numeric'
seasonweek_to_isoweek_c(x)
```

Arguments

x Season week as a number between 1 and 52.

Details

Season week 1 corresponds to ISO week 35, season week 2 to ISO week 36, and so on, wrapping around the new year. The ISO week is returned as two digits, e.g. "35" or "01".

Value

ISO week as a character vector (e.g. "35").

Examples

```
seasonweek_to_isoweek_c(1)
seasonweek_to_isoweek_c(c(31, 52))
```

```
seasonweek_to_isoweek_n
      Season week to ISO week (numeric)
```

Description

Maps a season week number back to its ISO week, returned as a number. This is the inverse of [isoweek_to_seasonweek_n\(\)](#).

Usage

```
seasonweek_to_isoweek_n(x)

## Default S3 method:
seasonweek_to_isoweek_n(x)

## S3 method for class 'numeric'
seasonweek_to_isoweek_n(x)
```

Arguments

x Season week as a number between 1 and 52.

Details

Season week 1 corresponds to ISO week 35, season week 2 to ISO week 36, and so on, wrapping around the new year.

Value

ISO week as an integer vector (1 to 53).

Examples

```
seasonweek_to_isoweek_n(1)
seasonweek_to_isoweek_n(c(31, 52))
```

Index

* datasets

- dates_by_isoyearweek, [16](#)
- nor_workdays_by_date, [30](#)
- nor_workdays_by_isoyearweek, [31](#)

- base::Date, [18](#), [27](#), [33](#)
- base::format(), [32](#)
- base::strptime(), [32](#)
- base::Sys.time(), [32](#)

- data.table::data.table, [16](#), [30](#), [31](#)
- date_to_calmonth_c, [3](#)
- date_to_calmonth_n, [3](#)
- date_to_calyear_c, [4](#)
- date_to_calyear_n, [5](#)
- date_to_calyearmonth_c, [6](#)
- date_to_isoquarter_c, [7](#)
- date_to_isoquarter_c(), [13](#)
- date_to_isoquarter_n, [8](#)
- date_to_isoquarter_n(), [7](#)
- date_to_isoweek_c, [9](#)
- date_to_isoweek_c(), [14](#)
- date_to_isoweek_n, [10](#)
- date_to_isoyear_c, [11](#)
- date_to_isoyear_c(), [5](#), [12–14](#)
- date_to_isoyear_n, [12](#)
- date_to_isoyear_n(), [5](#)
- date_to_isoyearquarter_c, [13](#)
- date_to_isoyearweek_c, [14](#)
- date_to_isoyearweek_c(), [15](#), [16](#)
- date_to_season_c, [15](#)
- date_to_seasonweek_n, [15](#)
- dates_by_isoyearweek, [16](#)

- isoweek_to_seasonweek_n, [17](#)
- isoweek_to_seasonweek_n(), [16](#), [28](#), [33](#), [34](#)
- isoyear_to_last_date, [18](#)
- isoyear_to_last_isoweek_n, [19](#)
- isoyear_to_last_isoyearweek_c, [20](#)
- isoyearweek_to_isoquarter_c, [21](#)
- isoyearweek_to_isoquarter_c(), [26](#)
- isoyearweek_to_isoquarter_n, [21](#)
- isoyearweek_to_isoquarter_n(), [21](#)
- isoyearweek_to_isoweek_c, [22](#)
- isoyearweek_to_isoweek_n, [23](#)
- isoyearweek_to_isoyear_c, [24](#)
- isoyearweek_to_isoyear_n, [25](#)
- isoyearweek_to_isoyearquarter_c, [25](#)
- isoyearweek_to_last_date, [26](#)
- isoyearweek_to_season_c, [27](#)
- isoyearweek_to_season_c(), [15](#)
- isoyearweek_to_seasonweek_n, [28](#)
- isoyearweek_to_seasonweek_n(), [16](#)

- keep_sundays, [28](#)
- keep_sundays_and_latest_date, [29](#)
- keep_sundays_and_latest_date_internal, [30](#)

- nor_workdays_by_date, [30](#), [31](#)
- nor_workdays_by_isoyearweek, [31](#)
- now_c, [32](#)

- season_to_last_date, [32](#)
- seasonweek_to_isoweek_c, [33](#)
- seasonweek_to_isoweek_n, [34](#)