

Package ‘climatekit’

September 9, 2026

Title Unified Climate Indices for Temperature, Precipitation, and Drought

Version 0.2.2

Description Compute the standard suite of climate indices from daily weather observations. Provides the canonical 'ETCCDI' 27 (Expert Team on Climate Change Detection and Indices), the 'ET-SCI' heatwave and cold-wave families plus the Excess Heat Factor of Nairn and Fawcett (2013), and agroclimatic, drought, and human-comfort families. Drought indices ('SPI', 'SPEI') accept a choice of distribution (gamma or Pearson III for SPI; log-logistic or generalised extreme value for SPEI). Reference evapotranspiration is available via Hargreaves and the FAO-56 Penman-Monteith method (Allen et al. 1998). Percentile-based indices support the Zhang (2005) in-base bootstrap. Daily inputs are numeric vectors plus a 'Date' vector; outputs are tidy data frames. Optional gridded support via 'terra' applies any index over a 'SpatRaster' and reads 'netCDF' input. No external API calls; pairs with data packages such as 'readnoaa'. References: Alexander et al. (2006) <[doi:10.1029/2005JD006290](https://doi.org/10.1029/2005JD006290)>; Zhang et al. (2011) <[doi:10.1002/wcc.147](https://doi.org/10.1002/wcc.147)>; Zhang et al. (2005) <[doi:10.1175/JCLI3366.1](https://doi.org/10.1175/JCLI3366.1)>.

Depends R (>= 4.1.0)

License MIT + file LICENSE

Language en-GB

Encoding UTF-8

RoxygenNote 7.3.3

Imports cli (>= 3.6.0), stats

Suggests testthat (>= 3.0.0), terra, ncdf4, knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

URL <https://charlescoverdale.github.io/climatekit/>,
<https://github.com/charlescoverdale/climatekit>

BugReports <https://github.com/charlescoverdale/climatekit/issues>

NeedsCompilation no

Author Charles Coverdale [aut, cre]

Maintainer Charles Coverdale <charlesfcoverdale@gmail.com>

Repository CRAN

Date/Publication 2026-09-09 08:30:11 UTC

Contents

ck_apply_grid	3
ck_available	4
ck_branas	5
ck_browse	6
ck_catalogue	6
ck_compute	7
ck_convert_temp	8
ck_cooling_degree_days	8
ck_csdi	9
ck_cwa	10
ck_cwd	11
ck_cwf	12
ck_cwm	12
ck_cwn	13
ck_diurnal_range	14
ck_dry_days	15
ck_ehf	16
ck_etccdi_27	17
ck_fire_danger	18
ck_first_frost	19
ck_from_netcdf	19
ck_frost_days	20
ck_growing_degree_days	21
ck_growing_season	21
ck_heating_degree_days	22
ck_heat_index	23
ck_heavy_precip	23
ck_huglin	24
ck_humidex	25
ck_hwa	25
ck_hwd	26
ck_hwf	27
ck_hwm	28
ck_hwn	29
ck_ice_days	30
ck_last_frost	30
ck_max_1day_precip	31

<code>ck_apply_grid</code>	3
----------------------------	---

<code>ck_max_5day_precip</code>	32
<code>ck_metadata</code>	32
<code>ck_pet</code>	33
<code>ck_pet_pm</code>	34
<code>ck_precip_intensity</code>	35
<code>ck_r95p</code>	36
<code>ck_r99p</code>	37
<code>ck_spei</code>	37
<code>ck_spi</code>	38
<code>ck_summer_days</code>	39
<code>ck_tn10p</code>	40
<code>ck_tn90p</code>	41
<code>ck_tnn</code>	42
<code>ck_tnx</code>	42
<code>ck_total_precip</code>	43
<code>ck_tropical_nights</code>	44
<code>ck_tx10p</code>	44
<code>ck_tx90p</code>	45
<code>ck_txn</code>	46
<code>ck_txx</code>	47
<code>ck_very_heavy_precip</code>	48
<code>ck_warm_spell</code>	48
<code>ck_wet_days</code>	49
<code>ck_wind_chill</code>	50
<code>ck_winkler</code>	51
<code>ck_wsdi</code>	51
<code>clear_cache</code>	52

Index	54
--------------	-----------

<code>ck_apply_grid</code>	<i>Apply a climatekit Index Function Over a SpatRaster</i>
----------------------------	--

Description

Compute a daily climatekit index function for every cell in a `SpatRaster` `x` whose layers represent successive days, and return the per-period results as a `SpatRaster`. The supplied function `fun` must accept a numeric vector and a Date vector and return a data frame with period and value columns (the standard climatekit shape).

Usage

```
ck_apply_grid(x, fun, dates, ...)
```

Arguments

x	A SpatRaster. Layers correspond one-to-one with dates.
fun	A ck_* function (or any function with the same signature: numeric vector + Date vector + optional named arguments, returning a data frame with period and value).
dates	Date vector of length terra::nlyr(x).
...	Additional named arguments forwarded to fun (for example period = "annual", ref_start, ref_end).

Details

All cells must share the same dates. Cells that are entirely NA are returned as NA. Run-time scales linearly with the number of cells; for very large grids consider sub-setting first.

Value

A SpatRaster with one layer per output period (layer names are the period labels).

Examples

```
if (requireNamespace("terra", quietly = TRUE)) {
  dates <- seq(as.Date("2024-01-01"), as.Date("2024-12-31"), by = "day")
  n <- length(dates)
  # Tiny 2x2 SpatRaster of synthetic daily Tmax
  r <- terra::rast(nrows = 2, ncols = 2, nlyrs = n,
                  xmin = 0, xmax = 2, ymin = 0, ymax = 2)
  set.seed(1)
  terra::values(r) <- matrix(rnorm(4 * n, 15, 5), nrow = 4)
  txx_r <- ck_apply_grid(r, ck_txx, dates = dates, period = "annual")
  terra::nlyr(txx_r)
}
```

ck_available

List All Available Climate Indices

Description

Returns a data frame listing every index that climatekit can compute, along with its category, unit, and a short description.

Usage

```
ck_available()
```

Value

A data frame with columns index, category, unit, and description.

Examples

```
ck_available()
```

ck_branas

Branas Hydrothermal Index

Description

The Branas index combines temperature and precipitation during the growing season to estimate disease pressure (especially downy mildew) in vineyards. It is the sum of (monthly mean temperature) times (monthly precipitation total) over the five months of the growing season: April-August in the Northern Hemisphere; October-February in the Southern Hemisphere. The Southern Hemisphere season spans two calendar years and is reported under the year in which it starts.

Usage

```
ck_branas(precip, tavg, dates, lat = 50)
```

Arguments

precip	Numeric vector of daily precipitation (mm).
tavg	Numeric vector of daily mean temperatures (degrees C).
dates	Date vector of the same length as precip.
lat	Numeric. Latitude in decimal degrees, used to select the hemisphere convention. Default 50 (Northern Hemisphere).

Value

A data frame with columns period, value, index, and unit.

References

Branas, J., Bernon, G., & Levadoux, L. (1946). Elements de viticulture generale.

Examples

```
dates <- seq(as.Date("2024-04-01"), as.Date("2024-08-31"), by = "day")
set.seed(42)
tavg <- rnorm(length(dates), mean = 12, sd = 3)
precip <- rgamma(length(dates), shape = 0.5, rate = 0.2)
ck_branas(precip, tavg, dates)
```

ck_browse	<i>Browse the climatekit Index Catalogue</i>
-----------	--

Description

Filter the comprehensive climatekit index catalogue (see [ck_catalogue\(\)](#)) by sector, applicable standard (ETCCDI / ET-SCI / agroclimatic / comfort / drought / energy), or a free-text search across the function name, full name, and ETCCDI code.

Usage

```
ck_browse(sector = NULL, standard = NULL, search = NULL)
```

Arguments

sector	Character (length 1) or NULL. Filter to indices tagged with this sector. Common values: "agriculture", "health", "water", "energy". NULL (default) returns all rows.
standard	Character (length 1) or NULL. Filter to indices under this standard. Common values: "ETCCDI", "ET-SCI", "ETCCDI-approx", "agroclimatic", "comfort", "drought", "energy". NULL (default) returns all rows.
search	Character (length 1) or NULL. Free-text search; rows are kept where the term appears (case-insensitive) in the function name, the full name, or the ETCCDI code.

Value

The catalogue, filtered to matching rows. Same column structure as [ck_catalogue\(\)](#).

Examples

```
ck_browse(standard = "ETCCDI")
ck_browse(sector = "agriculture")
ck_browse(search = "heat")
```

ck_catalogue	<i>climatekit Index Catalogue</i>
--------------	-----------------------------------

Description

Returns the complete catalogue of climate indices implemented by climatekit, with one row per ck_* function and columns covering the canonical short code (where applicable), the full name, the index family, the relevant sector, the unit, the source standard, and the principal citation key.

Usage

ck_catalogue()

Details

Use [ck_browse\(\)](#) to filter by sector or standard.

Value

A data frame with columns ck_function, code, name, category, sector, unit, standard, and citation_key.

Examples

```
tab <- ck_catalogue()
head(tab)
# Tally indices by standard:
table(tab$standard)
```

ck_compute	<i>Compute a Climate Index by Name</i>
------------	--

Description

A generic dispatcher that calls the appropriate ck_* function based on a string index name. Useful for programmatic workflows where the index is selected at runtime.

Usage

ck_compute(data, index, ...)

Arguments

data	A named list or data frame containing the required input vectors. Column names should match function argument names (e.g. tmin, tmax, precip, dates).
index	Character. Name of the index to compute (e.g. "frost_days"). Use ck_available() to see valid names. Where ETCCDI and ET-SCI share an acronym, climatekit follows the meaning its own ck_* function carries: "cwd" is ET-SCI cold-wave duration (ck_cwd()), not ETCCDI consecutive wet days, and "cdd" is ETCCDI consecutive dry days (ck_dry_days()), not cooling degree days. The unambiguous aliases "consecutive_wet_days", "consecutive_dry_days" and "cold_wave_duration" are also accepted.
...	Additional arguments passed to the underlying function (e.g. period, threshold, base).

Value

A data frame as returned by the underlying ck_* function.

Examples

```
d <- data.frame(
  dates = as.Date("2024-01-01") + 0:9,
  tmin = c(-2, 3, -1, 5, -3, 0, 2, -4, 1, -1)
)
ck_compute(d, "frost_days")
```

ck_convert_temp	<i>Convert Temperature Units</i>
-----------------	----------------------------------

Description

Convert between Celsius, Fahrenheit, and Kelvin.

Usage

```
ck_convert_temp(x, from, to)
```

Arguments

x	Numeric vector of temperatures.
from	Character. Source unit: "C", "F", or "K".
to	Character. Target unit: "C", "F", or "K".

Value

Numeric vector of converted temperatures.

Examples

```
ck_convert_temp(c(0, 100), from = "C", to = "F")
ck_convert_temp(32, from = "F", to = "C")
```

ck_cooling_degree_days	<i>Cooling Degree Days</i>
------------------------	----------------------------

Description

Sum of (Tavg - base) for all days where daily average temperature is above the base temperature (default 18 degrees C).

Usage

```
ck_cooling_degree_days(tavg, dates, base = 18, period = "annual")
```

Arguments

- tavg Numeric vector of daily average temperatures (degrees C).
- dates Date vector of the same length as tavg.
- base Numeric. Base temperature in degrees C (default 18).
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tavg <- c(25, 30, 22, 20, 28, 19, 32, 17, 35, 27)
ck_cooling_degree_days(tavg, dates)
```

ck_csd	<i>Cold Spell Duration Index (CSDI)</i>
--------	---

Description

ETCCDI canonical index CSDI. Annual count of days in spans of at least six consecutive days where daily Tmin falls below the 10th percentile of the calendar-day distribution from a reference period (default 1961 to 1990). Cold-side counterpart to [ck_wsdi\(\)](#).

Usage

```
ck_csd(tmin, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 6L)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 6, the ETCCDI standard).

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_csdi(tmin, dates))
```

ck_cwa	<i>Cold-Wave Amplitude (CWA)</i>
--------	----------------------------------

Description

ET-SCI cold-wave family index. `mode = "excess"` (default) returns the peak (threshold - Tmin) across cold-wave days, expressed as a positive magnitude. `mode = "absolute"` returns the minimum raw Tmin across cold-wave days (the coldest event-day value). Returns NA for years with no cold waves.

Usage

```
ck_cwa(
  tmin,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  min_spell = 3L,
  mode = c("excess", "absolute")
)
```

Arguments

- `tmin` Numeric vector of daily minimum temperatures (degrees C).
- `dates` Date vector of the same length as `tmax`. Must contain data covering the reference period.
- `ref_start, ref_end` Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- `min_spell` Integer. Minimum spell length in days (default 3, the ET-SCI standard).
- `mode` One of "excess" (default) or "absolute". See details.

Value

A data frame with columns `period`, `value`, `index`, and `unit`.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_cwa(tmin, dates))
```

ck_cwd	<i>Cold-Wave Duration (CWD, ET-SCI)</i>
--------	---

Description

ET-SCI cold-wave family index. Length in days of the longest cold-wave event in each year (see [ck_cwn\(\)](#) for the cold-wave definition).

Usage

```
ck_cwd(tmin, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 3L)
```

Arguments

tmin	Numeric vector of daily minimum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
min_spell	Integer. Minimum spell length in days (default 3, the ET-SCI standard).

Details

Note: the same letters CWD also denote the ETCCDI **Consecutive Wet Days** precipitation index, which is unrelated and is implemented in [ck_wet_days\(\)](#). These are two different indices that share an acronym in the climate-extremes literature.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_cwd(tmin, dates))
```

ck_cwf	<i>Cold-Wave Frequency (CWF)</i>
--------	----------------------------------

Description

ET-SCI cold-wave family index. Annual total number of days inside any cold-wave event (see [ck_cwn\(\)](#) for the cold-wave definition).

Usage

```
ck_cwf(tmin, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 3L)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 3, the ET-SCI standard).

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_cwf(tmin, dates))
```

ck_cwm	<i>Cold-Wave Magnitude (CWM)</i>
--------	----------------------------------

Description

ET-SCI cold-wave family index. mode = "excess" (default) returns the mean of (threshold - Tmin) across cold-wave days, expressed as a positive magnitude. mode = "absolute" returns the mean raw Tmin across cold-wave days. Returns NA for years with no cold waves.

Usage

```
ck_cwm(  
  tmin,  
  dates,  
  ref_start = 1961L,  
  ref_end = 1990L,  
  min_spell = 3L,  
  mode = c("excess", "absolute")  
)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 3, the ET-SCI standard).
- mode One of "excess" (default) or "absolute". See details.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)  
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")  
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +  
  rnorm(length(dates))  
tail(ck_cwm(tmin, dates))
```

ck_cwn	<i>Cold-Wave Number (CWN)</i>
--------	-------------------------------

Description

ET-SCI cold-wave family index. Annual count of distinct cold-wave events, where a cold wave is a span of at least three consecutive days with daily Tmin below the 10th percentile of the calendar-day distribution from a reference period (default 1961 to 1990).

Usage

```
ck_cwn(tmin, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 3L)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 3, the ET-SCI standard).

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_cwn(tmin, dates))
```

ck_diurnal_range	<i>Diurnal Temperature Range</i>
------------------	----------------------------------

Description

Mean daily temperature range (Tmax - Tmin) per period.

Usage

```
ck_diurnal_range(tmin, tmax, dates, period = "annual")
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmin and tmax.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tmin <- c(-2, 3, -1, 5, -3, 0, 2, -4, 1, -1)
tmax <- c(5, 10, 6, 12, 4, 8, 9, 3, 7, 6)
ck_diurnal_range(tmin, tmax, dates)
```

ck_dry_days	<i>Maximum Consecutive Dry Days</i>
-------------	-------------------------------------

Description

Maximum number of consecutive days with precipitation below a threshold. This is the ETCCDI **CDD** index.

Usage

```
ck_dry_days(precip, dates, threshold = 1, period = "annual")
```

Arguments

- precip Numeric vector of daily precipitation (mm).
- dates Date vector of the same length as precip.
- threshold Numeric. Dry day threshold in mm (default 1).
- period Character. Aggregation period: "annual" (default) or "monthly".

Details

Note: CDD also denotes climpact’s cooling degree days, which is unrelated and is implemented in [ck_cooling_degree_days\(\)](#). In [ck_compute\(\)](#) the name "cdd" resolves here, to consecutive dry days; "consecutive_dry_days" is accepted as an unambiguous alias.

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 0, 5, 0, 0, 0, 2, 0, 0, 0)
ck_dry_days(precip, dates)
```

ck_ehf

*Excess Heat Factor (EHF, Nairn & Fawcett 2013)***Description**

Annual summary of the daily Excess Heat Factor heatwave intensity metric. EHF combines a 3-day mean daily temperature anomaly above the 95th percentile of the reference period with an acclimatisation term (3-day mean minus previous 30-day mean). Positive EHF days indicate heatwave conditions; larger values indicate more severe or less-acclimatised events. This is the operational heatwave metric used by the Australian Bureau of Meteorology.

Usage

```
ck_ehf(
  tmax,
  tmin,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  stat = c("max", "n_positive", "sum_positive")
)
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
tmin	Numeric vector of daily minimum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
stat	One of "max", "n_positive", or "sum_positive".

Details

Three annual summaries are exposed via the stat argument:

- "max" (default): peak EHF in the year. Strongest single-day intensity.
- "n_positive": count of days with EHF > 0. A frequency-of-heatwave-conditions measure.
- "sum_positive": sum of EHF on days with EHF > 0. A severity-weighted total.

Value

A data frame with columns period, value, index, and unit.

References

- Nairn, J. R., & Fawcett, R. J. B. (2013). Defining heatwaves: heatwave defined as a heat-impact event servicing all community and business sectors in Australia. *CAWCR Technical Report No. 060*.
- Perkins, S. E., & Alexander, L. V. (2013). On the measurement of heatwaves. *Journal of Climate*, 26(13), 4500-4517. doi:10.1175/JCLID1200383.1.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
s <- 2 * pi * as.integer(format(dates, "%j")) / 365
tmax <- 20 + 10 * sin(s) + rnorm(length(dates))
tmin <- 10 + 8 * sin(s) + rnorm(length(dates))
tail(ck_ehf(tmax, tmin, dates))
```

ck_etccdi_27

Canonical ETCCDI 27 Indices

Description

Returns the 27 canonical Expert Team on Climate Change Detection and Indices (ETCCDI) indices as documented by Alexander et al. (2006) and Zhang et al. (2011), with each row showing the canonical short code, full name, input variable, unit, definition, and the corresponding climatekit function (or NA where the index is not yet implemented).

Usage

```
ck_etccdi_27()
```

Details

Use this table to audit coverage, locate the ck_* function for a given ETCCDI code, or filter to indices that climatekit currently implements: `subset(ck_etccdi_27(), !is.na(ck_function))`.

Value

A data frame with one row per canonical ETCCDI index and columns code, name, variable, unit, definition, ck_function, and status.

References

- Alexander, L. V. et al. (2006). Global observed changes in daily climate extremes of temperature and precipitation. *Journal of Geophysical Research: Atmospheres*, 111(D5). doi:10.1029/2005JD006290.
- Zhang, X. et al. (2011). Indices for monitoring changes in extremes based on daily temperature and precipitation data. *Wiley Interdisciplinary Reviews: Climate Change*, 2(6), 851-870. doi:10.1002/wcc.147.

Examples

```

tab <- ck_etccdi_27()
head(tab)
# Indices currently implemented in climatekit:
subset(tab, !is.na(ck_function))[, c("code", "ck_function")]
# Coverage:
table(tab$status)

```

ck_fire_danger	<i>Fire Danger Index (Simplified)</i>
----------------	---------------------------------------

Description

A simplified fire danger proxy based on temperature, humidity, wind speed, and recent precipitation. This is NOT the Canadian Forest Fire Weather Index (Van Wagner 1987); for the full FWI system, use the cffdrs package.

Usage

```
ck_fire_danger(tavg, humidity, wind_speed, precip)
```

Arguments

tavg	Numeric vector of temperatures (degrees C).
humidity	Numeric vector of relative humidity (percent, 0-100).
wind_speed	Numeric vector of wind speeds (km/h).
precip	Numeric vector of daily precipitation (mm).

Value

A data frame with columns value, index, and unit.

Examples

```

ck_fire_danger(
  tavg = c(30, 25, 35),
  humidity = c(20, 40, 15),
  wind_speed = c(25, 10, 30),
  precip = c(0, 5, 0)
)

```

ck_first_frost	<i>First Frost Date</i>
----------------	-------------------------

Description

Day of year of the first autumn frost ($T_{min} < 0$ degrees C) in each year. Hemisphere is selected by lat: in the Northern Hemisphere ($lat \geq 0$) the search starts at July 1 (DOY 183); in the Southern Hemisphere it starts at March 1 (DOY 60), matching the autumn entry for each.

Usage

```
ck_first_frost(tmin, dates, lat = 50)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmin.
- lat Numeric. Latitude in decimal degrees, used to select the hemisphere convention. Default 50 (Northern Hemisphere).

Value

A data frame with columns period, value (day of year), date (the frost date), index, and unit.

Examples

```
dates <- seq(as.Date("2024-07-01"), as.Date("2024-12-31"), by = "day")
set.seed(42)
tmin <- 15 - seq_along(dates) * 0.15 + rnorm(length(dates), sd = 3)
ck_first_frost(tmin, dates)
```

ck_from_netcdf	<i>Read a netCDF File as a SpatRaster</i>
----------------	---

Description

Convenience wrapper that delegates to `terra::rast()`. Reads the file at path and returns a SpatRaster, optionally restricted to a single variable. terra and ncdf4 must be installed (both are listed in Suggests:).

Usage

```
ck_from_netcdf(path, var = NULL)
```

Arguments

- path Character. Path to a netCDF file.
- var Character or NULL. Variable to extract. If NULL, the default behaviour of `terra::rast()` applies.

Value

A `SpatRaster` (one layer per time step in the netCDF file).

Examples

```
## Not run:
r <- ck_from_netcdf("tas_day.nc", var = "tas")
terra::nlyr(r) # number of daily layers

## End(Not run)
```

ck_frost_days	<i>Frost Days</i>
---------------	-------------------

Description

Count the number of days where minimum temperature is below 0 degrees C.

Usage

```
ck_frost_days(tmin, dates, period = "annual")
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmin.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns `period`, `value`, `index`, and `unit`.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tmin <- c(-2, 3, -1, 5, -3, 0, 2, -4, 1, -1)
ck_frost_days(tmin, dates)
```

ck_growing_degree_days	<i>Growing Degree Days</i>
------------------------	----------------------------

Description

Sum of (Tavg - base) for all days where daily average temperature is above the base temperature (default 10 degrees C).

Usage

ck_growing_degree_days(tavg, dates, base = 10, period = "annual")

Arguments

- tavg Numeric vector of daily average temperatures (degrees C).
- dates Date vector of the same length as tavg.
- base Numeric. Base temperature in degrees C (default 10).
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tavg <- c(15, 20, 8, 12, 25, 9, 30, 11, 22, 18)
ck_growing_degree_days(tavg, dates)
```

ck_growing_season	<i>Growing Season Length</i>
-------------------	------------------------------

Description

Compute the growing season length following the ETCCDI definition: the number of days between the first occurrence of at least 6 consecutive days with daily mean temperature above 5 degrees C and the first span of 6 consecutive days with Tmean below 5 degrees C after July 1 (Northern Hemisphere) or January 1 (Southern Hemisphere). Calculated per year.

Usage

ck_growing_season(tavg, dates, lat = 50)

Arguments

tavg	Numeric vector of daily mean temperatures (degrees C).
dates	Date vector of the same length as tavg.
lat	Numeric. Latitude in decimal degrees (used to determine hemisphere for end-of-season rule). Default 50 (Northern Hemisphere).

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:364
set.seed(42)
tavg <- sin(seq(0, 2 * pi, length.out = 365)) * 15 + 5
ck_growing_season(tavg, dates)
```

ck_heating_degree_days
<i>Heating Degree Days</i>

Description

Sum of (base - Tavg) for all days where daily average temperature is below the base temperature (default 18 degrees C).

Usage

```
ck_heating_degree_days(tavg, dates, base = 18, period = "annual")
```

Arguments

tavg	Numeric vector of daily average temperatures (degrees C).
dates	Date vector of the same length as tavg.
base	Numeric. Base temperature in degrees C (default 18).
period	Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tavg <- c(5, 10, 15, 20, 8, 12, 18, 3, 25, 7)
ck_heating_degree_days(tavg, dates)
```

ck_heat_index	<i>Heat Index</i>
---------------	-------------------

Description

Compute the heat index (apparent temperature) using the Rothfusz regression equation used by the US National Weather Service.

Usage

```
ck_heat_index(tavg, humidity)
```

Arguments

tavg	Numeric vector of temperatures (degrees C).
humidity	Numeric vector of relative humidity (percent, 0-100).

Value

A data frame with columns value, index, and unit.

References

Rothfusz, L. P. (1990). The heat index equation. NWS Technical Attachment SR 90-23.

Examples

```
ck_heat_index(tavg = c(30, 35, 40), humidity = c(60, 70, 50))
```

ck_heavy_precip	<i>Heavy Precipitation Days</i>
-----------------	---------------------------------

Description

Count of days with precipitation at or above a threshold (default 10 mm).

Usage

```
ck_heavy_precip(precip, dates, threshold = 10, period = "annual")
```

Arguments

precip	Numeric vector of daily precipitation (mm).
dates	Date vector of the same length as precip.
threshold	Numeric. Threshold in mm (default 10).
period	Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 5, 12, 0, 15, 2, 0, 11, 4, 0)
ck_heavy_precip(precip, dates)
```

ck_huglin	<i>Huglin Heliothermal Index</i>
-----------	----------------------------------

Description

The Huglin index is used in viticulture to characterise the thermal potential of a region for grape growing. It is computed over the growing season (April 1 to September 30 in the Northern Hemisphere; October 1 to March 31 in the Southern Hemisphere).

Usage

```
ck_huglin(tmin, tmax, dates, lat)
```

Arguments

tmin	Numeric vector of daily minimum temperatures (degrees C).
tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmin.
lat	Numeric. Latitude in decimal degrees (used to determine hemisphere and day-length coefficient).

Value

A data frame with columns period, value, index, and unit.

References

Huglin, P. (1978). Nouveau mode d’évaluation des possibilites heliothermiques d’un milieu viticole. *Comptes Rendus de l’Academie d’Agriculture de France*, 64, 1117-1126.

Examples

```
dates <- seq(as.Date("2024-04-01"), as.Date("2024-09-30"), by = "day")
set.seed(42)
tmin <- rnorm(length(dates), mean = 12, sd = 3)
tmax <- tmin + runif(length(dates), 8, 15)
ck_huglin(tmin, tmax, dates, lat = 45)
```

ck_humidex	<i>Humidex</i>
------------	----------------

Description

Compute the Canadian humidex from temperature and dewpoint.

Usage

ck_humidex(tavg, dewpoint)

Arguments

- tavg Numeric vector of temperatures (degrees C).
- dewpoint Numeric vector of dewpoint temperatures (degrees C).

Value

A data frame with columns value, index, and unit.

References

Masterson, J., & Richardson, F. A. (1979). Humidex: A method of quantifying human discomfort due to excessive heat and humidity. Environment Canada.

Examples

ck_humidex(tavg = c(30, 35), dewpoint = c(20, 25))

ck_hwa	<i>Heatwave Amplitude (HWA)</i>
--------	---------------------------------

Description

ET-SCI heatwave family index. Reports the peak magnitude of daily Tmax across all heatwave days in the year. mode = "excess" (default) gives the maximum of (Tmax - threshold). mode = "absolute" gives the maximum raw Tmax across heatwave days (matching Perkins-Alexander 2013). Returns NA for years with no heatwaves.

Usage

```
ck_hwa(  
  tmax,  
  dates,  
  ref_start = 1961L,  
  ref_end = 1990L,  
  min_spell = 3L,  
  mode = c("excess", "absolute")  
)
```

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 3, the ET-SCI standard).
- mode One of "excess" (default) or "absolute". See details.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)  
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")  
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +  
  rnorm(length(dates))  
tail(ck_hwa(tmax, dates))
```

ck_hwd	<i>Heatwave Duration (HWD)</i>
--------	--------------------------------

Description

ET-SCI heatwave family index. Length in days of the longest heatwave event in each year (see [ck_hwn\(\)](#) for the heatwave definition).

Usage

```
ck_hwd(tmax, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 3L)
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
min_spell	Integer. Minimum spell length in days (default 3, the ET-SCI standard).

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_hwd(tmax, dates))
```

ck_hwf	<i>Heatwave Frequency (HWF)</i>
--------	---------------------------------

Description

ET-SCI heatwave family index. Annual total number of days inside any heatwave event (see [ck_hwn\(\)](#) for the heatwave definition).

Usage

```
ck_hwf(tmax, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 3L)
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
min_spell	Integer. Minimum spell length in days (default 3, the ET-SCI standard).

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_hwf(tmax, dates))
```

ck_hwm	<i>Heatwave Magnitude (HWM)</i>
--------	---------------------------------

Description

ET-SCI heatwave family index. Reports the mean magnitude of daily Tmax across all heatwave days in the year. mode = "excess" (default) gives the mean of (Tmax - threshold), matching the ET-SCI / climapct convention. mode = "absolute" gives the mean raw Tmax across heatwave days, matching Perkins-Alexander (2013). Returns NA for years with no heatwaves.

Usage

```
ck_hwm(
  tmax,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  min_spell = 3L,
  mode = c("excess", "absolute")
)
```

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 3, the ET-SCI standard).
- mode One of "excess" (default) or "absolute". See details.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_hwn(tmax, dates))
```

ck_hwn	<i>Heatwave Number (HWN)</i>
--------	------------------------------

Description

ET-SCI heatwave family index. Annual count of distinct heatwave events, where a heatwave is a span of at least three consecutive days with daily Tmax above the 90th percentile of the calendar-day distribution from a reference period (default 1961 to 1990).

Usage

```
ck_hwn(tmax, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 3L)
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
min_spell	Integer. Minimum spell length in days (default 3, the ET-SCI standard).

Details

Single-threshold definition (TX-only). For the dual-threshold Perkins-Alexander variant (TX and TN both above 90th percentile) see `climpact`.

Value

A data frame with columns `period`, `value`, `index`, and `unit`.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_hwn(tmax, dates))
```

ck_ice_days	<i>Ice Days</i>
-------------	-----------------

Description

Count the number of days where maximum temperature is below 0 degrees C.

Usage

```
ck_ice_days(tmax, dates, period = "annual")
```

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tmax <- c(-2, 3, -1, 5, -3, 0, 2, -4, 1, -1)
ck_ice_days(tmax, dates)
```

ck_last_frost	<i>Last Frost Date</i>
---------------	------------------------

Description

Day of year of the last spring frost ($T_{min} < 0$ degrees C) in each year. Hemisphere is selected by lat: in the Northern Hemisphere the search runs up to July 1 (DOY 183); in the Southern Hemisphere up to October 1 (DOY 274), matching the spring boundary for each.

Usage

```
ck_last_frost(tmin, dates, lat = 50)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmin.
- lat Numeric. Latitude in decimal degrees, used to select the hemisphere convention. Default 50 (Northern Hemisphere).

Value

A data frame with columns period, value (day of year), date (the frost date), index, and unit.

Examples

```
dates <- seq(as.Date("2024-01-01"), as.Date("2024-06-30"), by = "day")
set.seed(42)
tmin <- -10 + seq_along(dates) * 0.12 + rnorm(length(dates), sd = 3)
ck_last_frost(tmin, dates)
```

ck_max_1day_precip	Maximum 1-Day Precipitation
--------------------	-----------------------------

Description

Maximum precipitation recorded in a single day per period.

Usage

```
ck_max_1day_precip(precip, dates, period = "annual")
```

Arguments

- precip Numeric vector of daily precipitation (mm).
- dates Date vector of the same length as precip.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 5, 22, 0, 15, 25, 0, 11, 4, 30)
ck_max_1day_precip(precip, dates)
```

ck_max_5day_precip	<i>Maximum 5-Day Precipitation</i>
--------------------	------------------------------------

Description

Maximum precipitation total over any 5 consecutive days per period.

Usage

```
ck_max_5day_precip(precip, dates, period = "annual")
```

Arguments

- precip Numeric vector of daily precipitation (mm).
- dates Date vector of the same length as precip.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 5, 22, 0, 15, 25, 0, 11, 4, 30)
ck_max_5day_precip(precip, dates)
```

ck_metadata	<i>Get Metadata for a Climate Index</i>
-------------	---

Description

Returns metadata (unit, category, description, reference) for a named climate index.

Usage

```
ck_metadata(index)
```

Arguments

- index Character string. The index name (e.g. "frost_days"). Use [ck_available\(\)](#) to see valid names.

Value

A list with elements index, category, unit, description, and reference.

Examples

```
ck_metadata("frost_days")
```

ck_pet	<i>Potential Evapotranspiration (Hargreaves Method)</i>
--------	---

Description

Estimate daily PET using the Hargreaves-Samani equation, which requires only daily temperature extremes and latitude.

Usage

```
ck_pet(tmin, tmax, lat, dates)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- tmax Numeric vector of daily maximum temperatures (degrees C).
- lat Numeric. Latitude in decimal degrees.
- dates Date vector of the same length as tmin.

Value

A data frame with columns date, value, index, and unit.

References

Hargreaves, G. H., & Samani, Z. A. (1985). Reference crop evapotranspiration from temperature. *Applied Engineering in Agriculture*, 1(2), 96-99.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tmin <- c(15, 16, 14, 17, 15, 13, 16, 14, 15, 16)
tmax <- c(30, 32, 28, 33, 31, 27, 34, 29, 30, 32)
ck_pet(tmin, tmax, lat = 45, dates = dates)
```

ck_pet_pm

*Reference Evapotranspiration (FAO-56 Penman-Monteith)***Description**

Compute reference evapotranspiration ETo using the FAO-56 Penman-Monteith equation (Allen et al. 1998), the international standard for ETo estimation. Required inputs are daily Tmin and Tmax; optional inputs (humidity, wind speed, incoming solar radiation, elevation) increase accuracy. Where humidity, wind, or solar radiation are missing, FAO-56 fallback estimators are used.

Usage

```
ck_pet_pm(
  tmin,
  tmax,
  lat,
  dates,
  elev = 0,
  wind = 2,
  rh_min = NULL,
  rh_max = NULL,
  rs = NULL,
  albedo = 0.23,
  krs = 0.16
)
```

Arguments

tmin	Numeric vector of daily minimum temperatures (degrees C).
tmax	Numeric vector of daily maximum temperatures (degrees C), same length as tmin.
lat	Numeric. Latitude in decimal degrees.
dates	Date vector of the same length as tmin.
elev	Numeric. Elevation above sea level in metres (default 0).
wind	Numeric vector or single value. 2-m wind speed (m/s). Default 2 (FAO-56 fallback for unmeasured wind).
rh_min, rh_max	Optional numeric vectors of daily minimum and maximum relative humidity in percent. Both must be supplied together; otherwise vapour pressure falls back to $e_0(tmin)$.
rs	Optional numeric vector of daily incoming solar radiation ($MJ\ m^{-2}\ day^{-1}$). If NULL, Hargreaves-Samani estimate is used.
albedo	Numeric. Surface albedo (default 0.23).
krs	Numeric. Hargreaves-Samani coefficient for the Rs fallback (default 0.16 for inland sites; 0.19 for coastal).

Details

Inputs and units:

- tmin, tmax: daily minimum and maximum temperature (degrees C).
- lat: latitude in decimal degrees, used for the extraterrestrial-radiation calculation.
- elev: elevation above sea level in metres (default 0).
- wind: 2-metre wind speed (m/s). Default 2 m/s (the FAO-56 fallback value when wind data are unavailable).
- rh_min, rh_max: minimum and maximum daily relative humidity (\ pressure is estimated as $e_0(tmin)$ (FAO-56 Eq. 48).
- rs: daily incoming solar radiation ($MJ\ m^{-2}\ day^{-1}$). If NULL, estimated by Hargreaves-Samani: $R_s = krs * R_a * \sqrt{T_{max} - T_{min}}$ with $krs = 0.16$ for inland sites.
- albedo: surface albedo (default 0.23, the FAO grass reference).

Value

A data frame with columns date, value, index ("pet_pm"), and unit.

References

Allen, R. G., Pereira, L. S., Raes, D., & Smith, M. (1998). Crop evapotranspiration: guidelines for computing crop water requirements. *FAO Irrigation and Drainage Paper 56*.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tmin <- c(15, 16, 14, 17, 15, 13, 16, 14, 15, 16)
tmax <- c(30, 32, 28, 33, 31, 27, 34, 29, 30, 32)
ck_pet_pm(tmin, tmax, lat = 45, dates = dates)
```

ck_precip_intensity	Precipitation Intensity (SDII)
---------------------	--------------------------------

Description

Mean precipitation on wet days (days with precipitation ≥ 1 mm). Also known as the Simple Daily Intensity Index.

Usage

```
ck_precip_intensity(precip, dates, period = "annual")
```

Arguments

- | | |
|--------|---|
| precip | Numeric vector of daily precipitation (mm). |
| dates | Date vector of the same length as precip. |
| period | Character. Aggregation period: "annual" (default) or "monthly". |

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 5, 12, 0, 15, 2, 0, 11, 4, 0)
ck_precip_intensity(precip, dates)
```

ck_r95p	<i>Very Wet Days Total (R95p)</i>
---------	-----------------------------------

Description

ETCCDI canonical index R95p. Annual total precipitation on days where daily precipitation exceeds the 95th percentile of wet-day precipitation in a reference period (default 1961 to 1990). A wet day is one with precipitation at or above 1 mm. The threshold is a single value derived from all wet days in the reference period (not calendar-day specific).

Usage

```
ck_r95p(precip, dates, ref_start = 1961L, ref_end = 1990L, period = "annual")
```

Arguments

- precip Numeric vector of daily precipitation (mm).
- dates Date vector of the same length as precip. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
precip <- pmax(rgamma(length(dates), shape = 0.4, scale = 8) - 1, 0)
tail(ck_r95p(precip, dates))
```

ck_r99p	<i>Extremely Wet Days Total (R99p)</i>
---------	--

Description

ETCCDI canonical index R99p. Annual total precipitation on days where daily precipitation exceeds the 99th percentile of wet-day precipitation in a reference period (default 1961 to 1990). Same convention as [ck_r95p\(\)](#).

Usage

```
ck_r99p(precip, dates, ref_start = 1961L, ref_end = 1990L, period = "annual")
```

Arguments

precip	Numeric vector of daily precipitation (mm).
dates	Date vector of the same length as precip. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
period	Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
precip <- pmax(rgamma(length(dates), shape = 0.4, scale = 8) - 1, 0)
tail(ck_r99p(precip, dates))
```

ck_spei	<i>Standardised Precipitation-Evapotranspiration Index (SPEI)</i>
---------	---

Description

Compute the SPEI by fitting a log-logistic distribution to the monthly climatic water balance (precipitation minus potential evapotranspiration) accumulated over a rolling window.

Usage

```
ck_spei(precip, pet, dates, scale = 3, distribution = c("log-logistic", "gev"))
```

Arguments

precip	Numeric vector of daily precipitation (mm).
pet	Numeric vector of daily potential evapotranspiration (mm).
dates	Date vector of the same length as precip and pet.
scale	Integer. Accumulation period in months (default 3).
distribution	Character. Either "log-logistic" (default, Vicente-Serrano et al. 2010) or "gev" (Generalised Extreme Value, fitted by L-moments; preferred for water-balance series with heavy upper or lower tails). The three-parameter log-logistic is fitted by L-moments, which requires an L-skewness in (0, 1). A calendar month whose water balance is symmetric or left-skewed falls outside that range; the fit warns and returns NA for that month rather than reporting a value from a distribution that does not describe the data. Use distribution = "gev", which accommodates both tails, when this happens often in your series.

Value

A data frame with columns period, value, index, and unit.

References

Vicente-Serrano, S. M., Begueria, S., & Lopez-Moreno, J. I. (2010). A multiscalar drought index sensitive to global warming: the Standardized Precipitation Evapotranspiration Index. *Journal of Climate*, 23(7), 1696-1718.

Examples

```
dates <- seq(as.Date("2020-01-01"), as.Date("2023-12-31"), by = "day")
set.seed(42)
precip <- rgamma(length(dates), shape = 0.5, rate = 0.1)
pet <- rep(3, length(dates))
ck_spei(precip, pet, dates, scale = 3)
```

ck_spi	<i>Standardised Precipitation Index (SPI)</i>
--------	---

Description

Compute the SPI by fitting a parametric distribution to rolling monthly precipitation accumulations and transforming to standard normal deviates. Two distributions are supported: the two-parameter gamma (default; WMO-1090 standard) and the three-parameter Pearson III. The Pearson III tail is heavier and is preferred in arid regions where the wet-day distribution is highly skewed (Stagge et al. 2015).

Usage

```
ck_spi(precip, dates, scale = 3, distribution = c("gamma", "pearsonIII"))
```

Arguments

precip	Numeric vector of daily precipitation (mm).
dates	Date vector of the same length as precip.
scale	Integer. Accumulation period in months (default 3).
distribution	Character. Either "gamma" (default) or "pearsonIII".

Value

A data frame with columns period, value, index, and unit.

References

McKee, T. B., Doesken, N. J., & Kleist, J. (1993). The relationship of drought frequency and duration to time scales.

Stagge, J. H., Tallaksen, L. M., Gudmundsson, L., Van Loon, A. F., & Stahl, K. (2015). Candidate distributions for climatological drought indices (SPI and SPEI). *International Journal of Climatology*, 35(13), 4027-4040. doi:10.1002/joc.4267.

Examples

```
dates <- seq(as.Date("2020-01-01"), as.Date("2023-12-31"), by = "day")
set.seed(42)
precip <- rgamma(length(dates), shape = 0.5, rate = 0.1)
ck_spi(precip, dates, scale = 3)
```

ck_summer_days	<i>Summer Days</i>
----------------	--------------------

Description

Count the number of days where maximum temperature exceeds 25 degrees C.

Usage

```
ck_summer_days(tmax, dates, period = "annual")
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmax.
period	Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tmax <- c(22, 26, 28, 24, 30, 25, 27, 23, 31, 29)
ck_summer_days(tmax, dates)
```

ck_tn10p	<i>Percentage of Cool Nights (TN10p)</i>
----------	--

Description

ETCCDI canonical index TN10p. Percentage of days where daily Tmin falls below the 10th percentile of the calendar-day distribution from a reference period (default 1961 to 1990). Computation follows the same convention as `ck_tx10p()` and supports the same bootstrap argument.

Usage

```
ck_tn10p(
  tmin,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  period = "annual",
  bootstrap = FALSE
)
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- period Character. Aggregation period: "annual" (default) or "monthly".
- bootstrap Logical. If TRUE, apply the Zhang (2005) in-base bootstrap correction. Default FALSE for backward compatibility and speed.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_tn10p(tmin, dates))
```

ck_tn90p	<i>Percentage of Warm Nights (TN90p)</i>
----------	--

Description

ETCCDI canonical index TN90p. Percentage of days where daily Tmin exceeds the 90th percentile of the calendar-day distribution from a reference period (default 1961 to 1990). Computation follows the same convention as [ck_tx10p\(\)](#) and supports the same bootstrap argument.

Usage

```
ck_tn90p(
  tmin,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  period = "annual",
  bootstrap = FALSE
)
```

Arguments

tmin	Numeric vector of daily minimum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
period	Character. Aggregation period: "annual" (default) or "monthly".
bootstrap	Logical. If TRUE, apply the Zhang (2005) in-base bootstrap correction. Default FALSE for backward compatibility and speed.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmin <- 5 + 8 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_tn90p(tmin, dates))
```

ck_tnn	<i>Annual or Monthly Minimum of Daily Minimum Temperature (TNn)</i>
--------	---

Description

ETCCDI canonical index TNn. The minimum value of daily minimum temperature (Tmin) within each reporting period (coldest night).

Usage

```
ck_tnn(tmin, dates, period = "annual")
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmin.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tmin <- c(-2, 3, -1, 5, -8, 0, 2, -12, 1, -1)
ck_tnn(tmin, dates)
```

ck_tnx	<i>Annual or Monthly Maximum of Daily Minimum Temperature (TNx)</i>
--------	---

Description

ETCCDI canonical index TNx. The maximum value of daily minimum temperature (Tmin) within each reporting period (warmest night).

Usage

```
ck_tnx(tmin, dates, period = "annual")
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmin.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tmin <- c(15, 18, 22, 19, 14, 21, 23, 17, 20, 19)
ck_tnx(tmin, dates)
```

ck_total_precip	Total Wet-Day Precipitation (PRCPTOT)
-----------------	---------------------------------------

Description

Annual or monthly total precipitation in wet days, where a wet day is a day with precipitation at or above wet_day_threshold mm (default 1 mm, the ETCCDI standard). This is the canonical ETCCDI 'PRCPTOT' definition (Alexander et al. 2006; Zhang et al. 2011).

Usage

```
ck_total_precip(precip, dates, period = "annual", wet_day_threshold = 1)
```

Arguments

- precip Numeric vector of daily precipitation (mm).
- dates Date vector of the same length as precip.
- period Character. Aggregation period: "annual" (default) or "monthly".
- wet_day_threshold Numeric (mm). Days with precipitation strictly below this threshold are excluded from the sum. Default 1.

Details

Sub-threshold trace amounts are excluded. Pass wet_day_threshold = 0 to recover the previous behaviour of summing all daily values.

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 5, 3, 0, 8, 2, 0, 1, 4, 0)
ck_total_precip(precip, dates)
```

ck_tropical_nights	<i>Tropical Nights</i>
--------------------	------------------------

Description

Count the number of days where minimum temperature exceeds 20 degrees C.

Usage

```
ck_tropical_nights(tmin, dates, period = "annual")
```

Arguments

- tmin Numeric vector of daily minimum temperatures (degrees C).
- dates Date vector of the same length as tmin.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-07-01") + 0:9
tmin <- c(18, 21, 22, 19, 25, 20, 23, 17, 24, 21)
ck_tropical_nights(tmin, dates)
```

ck_tx10p	<i>Percentage of Cool Days (TX10p)</i>
----------	--

Description

ETCCDI canonical index TX10p. Percentage of days where daily Tmax falls below the 10th percentile of the calendar-day distribution from a reference period (default 1961 to 1990). The threshold is computed using a 5-day window centred on each calendar day, pooled across the reference period.

Usage

```
ck_tx10p(
  tmax,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  period = "annual",
  bootstrap = FALSE
)
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmax. Must contain data covering the reference period.
ref_start, ref_end	Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
period	Character. Aggregation period: "annual" (default) or "monthly".
bootstrap	Logical. If TRUE, apply the Zhang (2005) in-base bootstrap correction. Default FALSE for backward compatibility and speed.

Details

Set bootstrap = TRUE to apply the Zhang et al. (2005) in-base leave-one-out bootstrap, which removes the self-inclusion bias for years inside the reference period. The bootstrap is computationally expensive (roughly N^2 percentile fits for an N-year reference) but is the canonical climdex.pcic / climact behaviour and is required for climate-change attribution work that spans the base.

Value

A data frame with columns period, value, index, and unit.

References

Zhang, X., Hegerl, G. C., Zwiers, F. W., & Kenyon, J. (2005). Avoiding inhomogeneity in percentile-based indices of temperature extremes. *Journal of Climate*, 18(11), 1641-1651. doi:10.1175/JCLI3366.1.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_tx10p(tmax, dates))
```

ck_tx90p	<i>Percentage of Warm Days (TX90p)</i>
----------	--

Description

ETCCDI canonical index TX90p. Percentage of days where daily Tmax exceeds the 90th percentile of the calendar-day distribution from a reference period (default 1961 to 1990). Computation follows the same convention as [ck_tx10p\(\)](#) and supports the same bootstrap argument.

Usage

```
ck_txn90p(
  tmax,
  dates,
  ref_start = 1961L,
  ref_end = 1990L,
  period = "annual",
  bootstrap = FALSE
)
```

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- period Character. Aggregation period: "annual" (default) or "monthly".
- bootstrap Logical. If TRUE, apply the Zhang (2005) in-base bootstrap correction. Default FALSE for backward compatibility and speed.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_txn90p(tmax, dates))
```

ck_txn	<i>Annual or Monthly Minimum of Daily Maximum Temperature (TXn)</i>
--------	---

Description

ETCCDI canonical index TXn. The minimum value of daily maximum temperature (Tmax) within each reporting period (coldest day).

Usage

```
ck_txn(tmax, dates, period = "annual")
```

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tmax <- c(5, 10, -3, 12, 4, 8, 22, -8, 7, 6)
ck_txn(tmax, dates)
```

ck_txx	<i>Annual or Monthly Maximum of Daily Maximum Temperature (TXx)</i>
--------	---

Description

ETCCDI canonical index TXx. The maximum value of daily maximum temperature (Tmax) within each reporting period.

Usage

```
ck_txx(tmax, dates, period = "annual")
```

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax.
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
tmax <- c(5, 10, 18, 12, 4, 8, 22, 3, 7, 6)
ck_txx(tmax, dates)
```

ck_very_heavy_precip	<i>Very Heavy Precipitation Days</i>
----------------------	--------------------------------------

Description

Count of days with precipitation at or above a threshold (default 20 mm).

Usage

```
ck_very_heavy_precip(precip, dates, threshold = 20, period = "annual")
```

Arguments

- precip Numeric vector of daily precipitation (mm).
- dates Date vector of the same length as precip.
- threshold Numeric. Threshold in mm (default 20).
- period Character. Aggregation period: "annual" (default) or "monthly".

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(0, 5, 22, 0, 15, 25, 0, 11, 4, 30)
ck_very_heavy_precip(precip, dates)
```

ck_warm_spell	<i>Warm Spell Days (Series-Quantile Approximation)</i>
---------------	--

Description

Count the number of days in warm spells, where a warm spell is defined as at least six consecutive days with Tmax above the threshold quantile of the full input series. This is a quick approximation driven by a single series-wide quantile, and does not require a reference period.

Usage

```
ck_warm_spell(tmax, dates, threshold = 0.9, period = "annual")
```

Arguments

tmax	Numeric vector of daily maximum temperatures (degrees C).
dates	Date vector of the same length as tmax.
threshold	Numeric. Quantile threshold (default 0.9, i.e. 90th percentile).
period	Character. Aggregation period: "annual" (default) or "monthly".

Details

For the canonical ETCCDI WSDI definition (1961-1990 calendar-day base, six-day spell rule), use `ck_wsdi()`.

The quantile is estimated with the Hyndman and Fan type 8 estimator, matching every other percentile in `climatekit`. Before v0.2.1 this function alone used R's default type 7. The two estimators differ by order 1/n, so counts rarely change on multi-year daily series.

Value

A data frame with columns `period`, `value`, `index`, and `unit`.

Examples

```
dates <- as.Date("2024-01-01") + 0:364
set.seed(42)
tmax <- rnorm(365, mean = 20, sd = 5)
ck_warm_spell(tmax, dates)
```

ck_wet_days	<i>Maximum Consecutive Wet Days</i>
-------------	-------------------------------------

Description

Maximum number of consecutive days with precipitation at or above a threshold. This is the ETC-CDI CWD index.

Usage

```
ck_wet_days(precip, dates, threshold = 1, period = "annual")
```

Arguments

precip	Numeric vector of daily precipitation (mm).
dates	Date vector of the same length as precip.
threshold	Numeric. Wet day threshold in mm (default 1).
period	Character. Aggregation period: "annual" (default) or "monthly".

Details

Note: the same letters CWD also denote the ET-SCI **cold-wave duration** index, which is unrelated and is implemented in `ck_cwd()`. In `ck_compute()` the name "cwd" resolves to cold-wave duration, not to this function; use "wet_days" or the unambiguous alias "consecutive_wet_days" for consecutive wet days.

Value

A data frame with columns period, value, index, and unit.

Examples

```
dates <- as.Date("2024-01-01") + 0:9
precip <- c(5, 3, 0, 2, 8, 1, 0, 0, 4, 6)
ck_wet_days(precip, dates)
```

ck_wind_chill	<i>Wind Chill Temperature</i>
---------------	-------------------------------

Description

Compute wind chill using the North American Wind Chill Index formula (Environment Canada / US NWS). Valid for temperatures at or below 10 degrees C and wind speeds above 4.8 km/h.

Usage

```
ck_wind_chill(tavg, wind_speed)
```

Arguments

- tavg Numeric vector of temperatures (degrees C).
- wind_speed Numeric vector of wind speeds (km/h).

Value

A data frame with columns value, index, and unit.

Examples

```
ck_wind_chill(tavg = c(-5, -10, 0), wind_speed = c(20, 30, 15))
```

ck_winkler	<i>Winkler Index</i>
------------	----------------------

Description

The Winkler index (also called growing degree days for viticulture) accumulates daily mean temperature above 10 degrees C during the growing season (April-October in NH, October-April in SH).

Usage

ck_winkler(tavg, dates)

Arguments

tavg Numeric vector of daily average temperatures (degrees C).
dates Date vector of the same length as tavg.

Value

A data frame with columns period, value, index, and unit.

References

Amerine, M. A., & Winkler, A. J. (1944). Composition and quality of musts and wines of California grapes.

Examples

```
dates <- seq(as.Date("2024-04-01"), as.Date("2024-10-31"), by = "day")
set.seed(42)
tavg <- rnorm(length(dates), mean = 18, sd = 4)
ck_winkler(tavg, dates)
```

ck_wsdi	<i>Warm Spell Duration Index (WSDI)</i>
---------	---

Description

ETCCDI canonical index WSDI. Annual count of days in spans of at least six consecutive days where daily Tmax exceeds the 90th percentile of the calendar-day distribution from a reference period (default 1961 to 1990). The threshold is computed using a 5-day window centred on each calendar day, pooled across the reference period.

Usage

ck_wsdi(tmax, dates, ref_start = 1961L, ref_end = 1990L, min_spell = 6L)

Arguments

- tmax Numeric vector of daily maximum temperatures (degrees C).
- dates Date vector of the same length as tmax. Must contain data covering the reference period.
- ref_start, ref_end Integer. Reference period boundary years (inclusive). Defaults to 1961 and 1990.
- min_spell Integer. Minimum spell length in days (default 6, the ETCCDI standard).

Details

Days inside qualifying spells are counted into the year they fall in; a spell that crosses a year boundary contributes to both years proportionally. Annual aggregation only.

Value

A data frame with columns period, value, index, and unit.

Examples

```
set.seed(1)
dates <- seq(as.Date("1961-01-01"), as.Date("1991-12-31"), by = "day")
tmax <- 15 + 10 * sin(2 * pi * as.integer(format(dates, "%j")) / 365) +
  rnorm(length(dates))
tail(ck_wsgi(tmax, dates))
```

clear_cache	<i>Clear Cache (Deprecated)</i>
-------------	---------------------------------

Description

Deprecated in v0.2.1 and scheduled for removal in v0.4.0. climatekit performs no I/O and never writes a cache, so this function has never had anything to clear. It remains only so that existing scripts keep running.

Usage

```
clear_cache()
```

Details

If you are caching downloaded weather data, clear it with the tools of whichever data package fetched it (for example readnoaa), not here.

Value

Invisibly returns FALSE. Prior to v0.2.1 this returned TRUE when a directory set through `options(climatekit.cache_dir = ...)` contained files, which only ever happened if something outside `climatekit` had written there.

Examples

```
# Deprecated: this warns and does nothing.  
suppressWarnings(clear_cache())
```

Index

ck_apply_grid, 3
ck_available, 4
ck_available(), 7, 32
ck_branas, 5
ck_browse, 6
ck_browse(), 7
ck_catalogue, 6
ck_catalogue(), 6
ck_compute, 7
ck_compute(), 15, 50
ck_convert_temp, 8
ck_cooling_degree_days, 8
ck_cooling_degree_days(), 15
ck_csdi, 9
ck_cwa, 10
ck_cwd, 11
ck_cwd(), 7, 50
ck_cwf, 12
ck_cwm, 12
ck_cwn, 13
ck_cwn(), 11, 12
ck_diurnal_range, 14
ck_dry_days, 15
ck_dry_days(), 7
ck_ehf, 16
ck_etccdi_27, 17
ck_fire_danger, 18
ck_first_frost, 19
ck_from_netcdf, 19
ck_frost_days, 20
ck_growing_degree_days, 21
ck_growing_season, 21
ck_heat_index, 23
ck_heating_degree_days, 22
ck_heavy_precip, 23
ck_huglin, 24
ck_humidex, 25
ck_hwa, 25
ck_hwd, 26
ck_hwf, 27
ck_hwm, 28
ck_hwn, 29
ck_hwn(), 26, 27
ck_ice_days, 30
ck_last_frost, 30
ck_max_1day_precip, 31
ck_max_5day_precip, 32
ck_metadata, 32
ck_pet, 33
ck_pet_pm, 34
ck_precip_intensity, 35
ck_r95p, 36
ck_r95p(), 37
ck_r99p, 37
ck_spei, 37
ck_spi, 38
ck_summer_days, 39
ck_tn10p, 40
ck_tn90p, 41
ck_tnn, 42
ck_tnx, 42
ck_total_precip, 43
ck_tropical_nights, 44
ck_tx10p, 44
ck_tx10p(), 40, 41, 45
ck_tx90p, 45
ck_txn, 46
ck_txx, 47
ck_very_heavy_precip, 48
ck_warm_spell, 48
ck_wet_days, 49
ck_wet_days(), 11
ck_wind_chill, 50
ck_winkler, 51
ck_wsdi, 51
ck_wsdi(), 9, 49
clear_cache, 52
terra::rast(), 19, 20