Package 'cimir'

June 11, 2019

Title Interface to the CIMIS Web API

Version 0.3-0

Description Connect to the California Irrigation Management Information System (CIMIS) Web API. See the CIMIS main page <https://cimis.water.ca.gov/> and web API documentation <https://et.water.ca.gov> for more information.

License GPL (>= 3)

URL https://github.com/mkoohafkan/cimir

BugReports https://github.com/mkoohafkan/cimir/issues

Depends R (>= 3.4)

Imports curl (>= 3.3), glue (>= 1.3), stringr (>= 1.3), dplyr (>= 0.7), tidyr (>= 0.8), jsonlite (>= 1.5), purrr (>= 0.2), rlang (>= 0.3)

Encoding UTF-8

LazyData true

Suggests knitr (>= 1.21), rmarkdown (>= 1.11)

VignetteBuilder knitr

RoxygenNote 6.1.1

NeedsCompilation no

Author Michael Koohafkan [aut, cre]

Maintainer Michael Koohafkan <michael.koohafkan@gmail.com>

Repository CRAN

Date/Publication 2019-06-10 23:00:07 UTC

R topics documented:

`cimir` *cimir: Interface to CIMIS*

Description

This package provides an R interface to the California Irrigation Management Information System (CIMIS) Web API. In order to use this package, you will need to create a CIMIS account and request a web services AppKey.

Package options

cimir uses the following `options()` to configure behavior:

- `cimir.appkey`: The CIMIS AppKey to use for queries.
- `cimir.timeout`: The maximum time to wait for a response from the CIMIS Web API.

Alternatively, the CIMIS App Key can be saved to an environment variable `CIMIS_APPKEY`.

Author(s)

Maintainer: Michael Koohafkan <michael.koohafkan@gmail.com>

See Also

Useful links:

- https://github.com/mkoohafkan/cimir
- Report bugs at https://github.com/mkoohafkan/cimir/issues

cimis_compass_to_degrees *Compass Direction To Degrees*

Description

Convert the Compass direction labels to degrees.

Usage

```
cimis_compass_to_degrees(x)
```

Arguments

x	A vector of compass directions, i.e. the data item labels "DayWindNnw", "DayWindSse", etc. Recognized directions are North-northeast (NNE), East-northeast (ENE), East-southeast (ESE), South-southeast (SSE), South-southwest (SSW), West-southwest (WSW), West-northwest (WNW), and North-northwest (NNW).

Value

A numeric vector of degrees corresponding to the middle azimumth of the corresponding compass direction.

Examples

```
cimis_compass_to_degrees("day-wind-nne")
cimis_compass_to_degrees(c("SSE", "SSW", "wsw", "Wnw", "nnw"))
```

cimis_data *Query CIMIS Data*

Description

Query CIMIS data using the Web API.

Usage

```
cimis_data(targets, start.date, end.date, items, measure.unit = c("E",
  "M"), prioritize.SCS = TRUE)
```

Arguments

targets	geographies or weather stations of interest. This parameter may specify one or many stations, zip codes, coordinates, or street addresses; however, you are not allowed to mix values from different categories. This means the targets parameter must contain only stations, only zip codes, only coordinates, or only street addresses. You will receive an error if you attempt to mix different category types. The formats are accepted: • A comma delimited list of WSN station numbers • A comma delimited list of California zip codes • A semicolon delimited list of decimal - degree coordinates • A semicolon delimited list of street addresses
start.date	Specifies the start date. The data format is "yyyy-mm-dd".
end.date	Specifies the end date. The data format is "yyyy-mm-dd".
items	specifies one or more comma-delimited data elements to include in your response. See data_items() for a complete list of possible data element values. Default: day-asce-eto, day-precip, day-sol-rad-avg, day-vap-pres-avg, day-air-tmp-max, day-air-tmp-min, day-air-tmp-avg, day-rel-hum-max, day-rel-hum-min, day-rel-hum-avg, day-dew-pnt, day-wind-spd-avg, day-wind-run, day-soil-tmp-avg.
measure.unit	The unit of measure may be either "E" for English units or "M" for metric units. The value of this parameter will affect data values in the response. For example, designating English units will result in temperature values being returned in Fahrenheit rather than Celsius.
prioritize.SCS	This parameter is relevant only when the targets parameter contains zip code(s). If TRUE, the Spatial CIMIS System (SCS) will be used as the preferred data provider.

Value

A tibble object.

Examples

```
if(is_key_set()) {
  cimis_data(targets = 170, start.date = Sys.Date() - 4,
    end.date = Sys.Date() - 1)
}
```

---

cimis_format_location *Format CIMIS Station Location*

Description

Format the latitude and longitude of station in Decimal Degrees (DD) or Hour Minutes Seconds (HMS).

Usage

```
cimis_format_location(d, format = c("DD", "HMS"))
```

Arguments

d	A data frame of CIMIS data results.
format	The format to use, either Decimal Degrees ("DD") or Hour Minutes Seconds ("HMS").

Value

The data frame, with a new "Latitude" and "Longitude" columns replacing the "HmsLatitude" and "HmsLongitude" columns.

Examples

```
if(is_key_set()) {
  d = cimis_station(170)
  cimis_format_location(d, "DD")
  cimis_format_location(d, "HMS")
}
```

---

cimis_items *CIMIS Data Items*

Description

List CIMIS data items.

Usage

```
cimis_items(type = c("Daily", "Hourly"))
```

Arguments

type	The type of data item, i.e. "Daily" or "Hourly".

Value

a dataframe of data items.

Examples

```
cimis_items()
```

`cimis_split_query` *Split CIMIS Query*

Description

Split a large CIMIS query into multiple smaller queries based on a time interval.

Usage

```
cimis_split_query(targets, start.date, end.date, items,
  max.records = 1750L)
```

Arguments

`targets`	geographies or weather stations of interest. This parameter may specify one or many stations, zip codes, coordinates, or street addresses; however, you are not allowed to mix values from different categories. This means the targets parameter must contain only stations, only zip codes, only coordinates, or only street addresses. You will receive an error if you attempt to mix different category types. The formats are accepted: • A comma delimited list of WSN station numbers • A comma delimited list of California zip codes • A semicolon delimited list of decimal - degree coordinates • A semicolon delimited list of street addresses
`start.date`	Specifies the start date. The data format is `"yyyy-mm-dd"`.
`end.date`	Specifies the end date. The data format is `"yyyy-mm-dd"`.
`items`	specifies one or more comma-delimited data elements to include in your response. See `data_items()` for a complete list of possible data element values. Default: day-asce-eto, day-precip, day-sol-rad-avg, day-vap-pres-avg, day-air-tmp-max, day-air-tmp-min, day-air-tmp-avg, day-rel-hum-max, day-rel-hum-min, day-rel-hum-avg, day-dew-pnt, day-wind-spd-avg, day-wind-run, day-soil-tmp-avg.
`max.records`	The maximum number of records returned by a query. The default value is the the maximum data limit allowed by the CIMIS Web API (1,750 records).

Details

Queries are not split by `targets` or `items`, i.e. each resulting query will include all targets and items.

Value

A data frame with columns "targets", "start.date", "end.date", and "items".

Examples

```
cimis_split_query(170, "2000-01-01", "2010-12-31", "day-air-tmp-avg")
cimis_split_query(c(149, 170), "2018-01-01", "2018-12-31",
  c("day-air-tmp-avg", "hly-air-tmp", "hly-rel-hum"))
```

---

cimis_station *Query CIMIS Station Metadata*

Description

Query CIMIS station metadata.

Usage

```
cimis_station(station)

cimis_spatial_zipcode(zipcode)

cimis_zipcode(zipcode)
```

Arguments

station	The station ID. If missing, metadata for all stations is returned.
zipcode	The (spatial) zip code. If missing, metadata for all stations is returned.

Value

A tibble object.

Examples

```
if(is_key_set()) {
  cimis_station()
  cimis_zipcode()
  cimis_spatial_zipcode()
}
```

cimis_to_datetime *To Datetime*

Description

Collapse The Date and Hour columns to a single DateTime Column.

Usage

```
cimis_to_datetime(d)
```

Arguments

d	A data frame of CIMIS data results.

Details

According to the CIMIS Report FAQs, all CIMIS data is based on Pacific Standard Time (PST).

Value

The data frame, with a new "Datetime" column replacing the "Date" and "Hour" columns.

Examples

```
if(is_key_set()) {
  d = cimis_data(targets = 170, start.date = Sys.Date() - 4,
    end.date = Sys.Date() - 1, items = "hly-air-tmp")
  cimis_to_datetime(d)
}
```

set_key *Specify CIMIS API key*

Description

Enter your CIMIS AppKey for web API data access.

Usage

```
set_key(key)

remove_key()

is_key_set()
```

Arguments

key	A CIMIS AppKey.

Examples

```
## Not run:
set_key("YOUR-APP-KEY")
is_key_set()
remove_key()

## End(Not run)
```

Index