Package 'saqgetr'

May 9, 2026

Type Package

Title Import Air Quality Monitoring Data in a Fast and Easy Way

Version 0.2.21

Date 2021-01-12

Description A collection of tools to access prepared air quality monitoring data files from web servers with ease and speed. Air quality data are sourced from open and publicly accessible repositories and can be found in these locations: <https://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-8> and <https://discomap.eea.europa.eu/map/fme/AirQualityExport.htm>. The web server space has been provided by Ricardo Energy & Environment.

Maintainer Stuart K. Grange <stuart.grange@york.ac.uk>

URL https://github.com/skgrange/saqgetr

BugReports https://github.com/skgrange/saqgetr/issues

License GPL-3 | file LICENSE

ByteCompile true

Depends R (>= 3.2.0)

Imports stringr, lubridate, dplyr, purrr, readr, tidyr, magrittr, httr

Suggests openair

Encoding UTF-8

LazyLoad true

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Author Stuart K. Grange [cre, aut] (ORCID: <https://orcid.org/0000-0003-4093-3596>)

Repository CRAN

Date/Publication 2021-01-12 09:20:02 UTC

Contents

`get_saq_observations` *Function to get* **saqgetr** *air quality observations.*

Description

Function to get **saqgetr** air quality observations.

Usage

```
get_saq_observations(
  site,
  variable = NA,
  start = NA,
  end = NA,
  valid_only = FALSE,
  tz = "UTC",
  verbose = FALSE
)
```

Arguments

`site`	A vector of sites to import. Use get_saq_sites to find what sites are available.
`variable`	An optional variable vector. If not used, all variables will be returned.
`start`	Start date for returned observations. Can either be a date string in `"yyyy-mm-dd"` format, or a year integer.
`end`	End date for returned observations. Can either be a date string in `"yyyy-mm-dd"` format, or a year integer.
`valid_only`	Should only valid observations be kept?
`tz`	Time zone for the observations' dates.
`verbose`	Should the function give messages?

Value

Tibble.

Author(s)

Stuart K. Grange.

See Also

get_saq_sites, saq_clean_observations

Examples

```
# Load a site's data
data_hafodyrynys <- get_saq_observations(
  site = "gb1038a",
  start = 2018,
  end = 2018
)

# Print tibble
print(data_hafodyrynys)



# Get multiple sites nox and ozone data for between a date range
data_many <- get_saq_observations(
  site = c("gb1014a", "gb1044a", "gb1060a"),
  variable = c("nox", "no2", "o3"),
  start = 2018,
  end = 2022,
  verbose = TRUE
)

# Print tibble
print(data_many)

# Sites and site names
data_many %>%
  dplyr::distinct(site)
```

---

get_saq_processes *Function to get air quality time series processes serviced by the* **saqgetr** *package.*

Description

Function to get air quality time series processes serviced by the **saqgetr** package.

Usage

```
get_saq_processes(file = NA)
```

Arguments

file	File of processes helper table.

Value

Tibble.

Author(s)

Stuart K. Grange.

Examples

```
# Get processes
data_processes <- get_saq_processes()
```

---

get_saq_simple_summaries

Function to get simple summaries of air quality observations.

Description

Function to get simple summaries of air quality observations.

Usage

```
get_saq_simple_summaries(file = NA, summary = "annual_means", tz = "UTC")
```

Arguments

file	File of simple summaries table.
summary	Summary period to import. Can either be "annual_means" or "monthly_means".
tz	Time zone for the observations' dates.

Value

Tibble.

Author(s)

Stuart K. Grange.

Examples

```
# Import annual means
data_annual <- get_saq_simple_summaries(summary = "annual_means")



## Not run:

# Import monthly means, quite a large request so will take some time
data_month <- get_saq_simple_summaries(summary = "monthly_means")


## End(Not run)
```

`get_saq_sites` *Function to import information for monitoring sites/stations/facilities serviced by the* **saqgetr** *package.*

Description

Function to import information for monitoring sites/stations/facilities serviced by the **saqgetr** package.

Usage

```
get_saq_sites(file = NA)
```

Arguments

`file`	File of sites helper table.

Value

Tibble.

Author(s)

Stuart K. Grange.

Examples

```
# Load sites table
data_sites <- get_saq_sites()
```

`get_saq_summaries` *Function to import summary integers for used in the* **saqgetr** *package.*

Description

Function to import summary integers for used in the **saqgetr** package.

Usage

```
get_saq_summaries(file = NA)
```

Arguments

`file`	File of summary helper table.

Value

Tibble.

Author(s)

Stuart K. Grange.

Examples

```
# Get summary integers
data_summary_integers <- get_saq_summaries()
```

`get_saq_validity` *Function to import validity integers for used in the* **saqgetr** *package.*

Description

Function to import validity integers for used in the **saqgetr** package.

Usage

```
get_saq_validity(file = NA)
```

Arguments

file	File of validity helper table.

Value

Tibble.

Author(s)

Stuart K. Grange.

Examples

```
# Get validity integers
data_validity_integers <- get_saq_validity()
```

---

saq_clean_observations

Function to clean and format observational data from **saqgetr***'s* get_saq_observations *function.*

Description

Function to clean and format observational data from **saqgetr**'s get_saq_observations function.

Usage

```
saq_clean_observations(df, summary = "hour", valid_only = TRUE, spread = FALSE)
```

Arguments

df	Tibble/data frame from get_saq_observations.
summary	Summary to filter data to. Default is "hour".
valid_only	Should only valid observations be kept?
spread	Should the data be "spread" where the data frame is reshaped so pollutants form variables/columns. This format is usually what is desired when using **openair**.

Value

Tibble.

Author(s)

Stuart K. Grange

Examples

```
# Load a site's data
data_hafodyrynys <- get_saq_observations(
  site = "gb1038a",
  start = 2018,
  end = 2018
)

# Keep only valid and hourly data
data_hafodyrynys_hourly <- data_hafodyrynys %>%
  saq_clean_observations(summary = "hour", valid_only = TRUE) %>%
  print()

# Spread hourly data, a different table format here
data_hafodyrynys_hourly_spread <- data_hafodyrynys %>%
  saq_clean_observations(summary = "hour", valid_only = TRUE, spread = TRUE) %>%
  print()
```

`tz_central` *Functions for time zone strings.*

Description

Functions for time zone strings.

Usage

```
tz_central()

tz_eastern()
```

Author(s)

Stuart K. Grange

`%>%` *Pseudo-function to re-export* **magrittr***'s pipe.*

Description

Pseudo-function to re-export **magrittr**'s pipe.

Index