Package 'fitbitr'

March 25, 2023

Type Package

Title Interface with the 'Fitbit' API

Version 0.3.0

Description Many 'Fitbit' users, and R-friendly 'Fitbit' users
especially, have found themselves curious about their 'Fitbit' data.
'Fitbit' aggregates a large amount of personal data, much of which is
interesting for personal research and to satisfy curiosity, and is
even potentially useful in medical settings. The goal of 'fitbitr' is
to make interfacing with the 'Fitbit' API as streamlined as possible,
to make it simple for R users of all backgrounds and comfort levels to
analyze their 'Fitbit' data and do whatever they want with it!
Currently, 'fitbitr' includes methods for pulling data on activity,
sleep, and heart rate, but this list is likely to grow in the future
as the package gains more traction and more requests for new methods
to be implemented come in. You can find details on the 'Fitbit' API
at <https://dev.fitbit.com/build/reference/web-api/>.

License GPL (>= 3)

URL https://github.com/mrkaye97/fitbitr,
https://matthewrkaye.com/fitbitr/

BugReports https://github.com/mrkaye97/fitbitr/issues

Imports dplyr, httr, janitor, jsonlite, lubridate, magrittr, purrr,
rlang, tibble (>= 2.0.0), tidyr

Suggests checkmate (>= 2.0.0), devtools, spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.2.3

NeedsCompilation no

Author Matt Kaye [aut, cre]

Maintainer Matt Kaye <mrkaye97@gmail.com>

Repository CRAN

Date/Publication 2023-03-25 16:10:02 UTC

R topics documented:

generate_fitbitr_token . . . 2
get_active_zone_minutes_intraday . . . 4
get_activity_calories . . . 5
get_activity_summary . . . 5
get_calories . . . 6
get_calories_bmr . . . 7
get_calories_intraday . . . 7
get_distance . . . 8
get_distance_intraday . . . 9
get_elevation . . . 10
get_elevation_intraday . . . 11
get_floors . . . 12
get_floors_intraday . . . 12
get_heart_rate_intraday . . . 13
get_heart_rate_zones . . . 14
get_intraday_time_series . . . 15
get_lifetime_bests . . . 16
get_lifetime_totals . . . 16
get_minutes_fairly_active . . . 17
get_minutes_lightly_active . . . 17
get_minutes_sedentary . . . 18
get_minutes_very_active . . . 19
get_sleep_stage_granular . . . 19
get_sleep_stage_summary . . . 20
get_sleep_summary . . . 21
get_steps . . . 22
get_steps_intraday . . . 22
get_tracker_bests . . . 23
get_tracker_totals . . . 24
perform_get . . . 24

Index **26**

`generate_fitbitr_token` *Generate an Oauth token*

Description

Performs the OAuth 2.0 dance to create a token to use with the Fitbit API.

Usage

```
generate_fitbitr_token(
  app_name = "fitbitr",
  client_id = Sys.getenv("FITBIT_CLIENT_ID"),
  client_secret = Sys.getenv("FITBIT_CLIENT_SECRET"),
  callback = Sys.getenv("FITBIT_CALLBACK", "https://localhost:1410/"),
 scope = c("activity", "cardio_fitness", "electrocardiogram", "heartrate", "location",
  "nutrition", "oxygen_saturation", "profile", "respiratory_rate", "settings", "sleep",
    "social", "temperature", "weight"),
  cache = TRUE,
  use_basic_auth = TRUE,
  ...
)
```

Arguments

app_name	The name of your OAuth app. Default: fitbitr
client_id	Your Fitbit client ID
client_secret	Your Fitbit client secret
callback	Your Fitbit redirect URL
scope	The scopes to enable
cache	Do you want to cache your token? See oauth2.0_token for details
use_basic_auth	A boolean for whether or not to use basic auth in oauth2.0_token. Defaults to TRUE
...	additional arguments to be passed to oauth2.0_token

Details

Saves a token as .fitbitr_token which can then be used in the background to authorize requests

Value

Returns an OAuth 2.0 token (invisibly) that can be used to authorize requests to the Fitbit API. Also saves the token to .fitbitr_token.

Examples

```
## Not run:
generate_fitbitr_token(
  client_id = <YOUR-CLIENT-ID>
  client_secret = <YOUR-CLIENT-SECRET>,
  cache = TRUE
)

## End(Not run)
```

`get_active_zone_minutes_intraday` *Get intraday active zone minutes time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_active_zone_minutes_intraday(
  date = lubridate::today(),
  detail_level = c("1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

`date`	A date to get data for
`detail_level`	The detail level. One of `"1min"`, `"5min"`, or `"15min"`
`start_time`	The start time of the time window. Default: `NULL` gets the whole day
`end_time`	The end time of the time window. Default: `NULL` gets the whole day

See Also

Other intraday: `get_calories_intraday()`, `get_distance_intraday()`, `get_elevation_intraday()`, `get_floors_intraday()`, `get_heart_rate_intraday()`, `get_steps_intraday()`

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_active_zone_minutes_intraday(detail_level = "15min")

## get more granular data
get_active_zone_minutes_intraday(detail_level = "1min")

## End(Not run)
```

get_activity_calories *Activity Calories Time Series*

Description

Resource path /activities/activityCalories

Usage

```
get_activity_calories(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: `date` and `activity_calories`

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_activity_calories(date)

## End(Not run)
```

get_activity_summary *Activity Summary*

Description

See https://dev.fitbit.com/build/reference/web-api/activity/ for more details.

Usage

```
get_activity_summary(date)
```

Arguments

date	The date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with the date and a number of activity summary metrics for the day.

Examples

```
## Not run:
date <- lubridate::today()
get_activity_summary(date)

## End(Not run)
```

---

get_calories *Calories Time Series*

Description

Resource path /activities/calories

Usage

```
get_calories(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: date and calories

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
calories(date)

## End(Not run)
```

get_calories_bmr *Calories BMR Time Series*

Description

Resource path /activities/caloriesBMR

Usage

```
get_calories_bmr(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: date and calories_bmr

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_calories_bmr(date)

## End(Not run)
```

get_calories_intraday *Get intraday calories time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_calories_intraday(
  date = lubridate::today(),
  detail_level = c("1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

date	A date to get data for
detail_level	The detail level. One of "1min", "5min", or "15min"
start_time	The start time of the time window. Default: NULL gets the whole day
end_time	The end time of the time window. Default: NULL gets the whole day

Value

A tibble with two columns: time and calories

See Also

Other intraday: get_active_zone_minutes_intraday(), get_distance_intraday(), get_elevation_intraday(), get_floors_intraday(), get_heart_rate_intraday(), get_steps_intraday()

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_calories_intraday(detail_level = "15min")

## get more granular data
get_calories_intraday(detail_level = "1min")

## End(Not run)
```

---

get_distance *Distance Time Series*

Description

Resource path /activities/distance

Usage

```
get_distance(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: date and distance

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_distance(date)

## End(Not run)
```

`get_distance_intraday` *Get intraday distance time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_distance_intraday(
  date = lubridate::today(),
  detail_level = c("1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

`date`	A date to get data for
`detail_level`	The detail level. One of `"1min"`, `"5min"`, or `"15min"`
`start_time`	The start time of the time window. Default: `NULL` gets the whole day
`end_time`	The end time of the time window. Default: `NULL` gets the whole day

Value

A tibble with two columns: `time` and `distance`

See Also

Other intraday: `get_active_zone_minutes_intraday()`, `get_calories_intraday()`, `get_elevation_intraday()`, `get_floors_intraday()`, `get_heart_rate_intraday()`, `get_steps_intraday()`

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_distance_intraday(detail_level = "15min")

## get more granular data
get_distance_intraday(detail_level = "1min")

## End(Not run)
```

`get_elevation` *Elevation Time Series*

Description

Resource path /activities/elevation

Usage

```
get_elevation(start_date, end_date)
```

Arguments

`start_date`	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
`end_date`	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: `date` and `elevation`

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_elevation(date)

## End(Not run)
```

get_elevation_intraday *Get intraday elevation time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_elevation_intraday(
  date = lubridate::today(),
  detail_level = c("1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

date	A date to get data for
detail_level	The detail level. One of "1min", "5min", or "15min"
start_time	The start time of the time window. Default: NULL gets the whole day
end_time	The end time of the time window. Default: NULL gets the whole day

Value

A tibble with two columns: time and elevation

See Also

Other intraday: get_active_zone_minutes_intraday(), get_calories_intraday(), get_distance_intraday(), get_floors_intraday(), get_heart_rate_intraday(), get_steps_intraday()

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_elevation_intraday(detail_level = "15min")

## get more granular data
get_elevation_intraday(detail_level = "1min")

## End(Not run)
```

get_floors *Floors Time Series*

Description

Resource path /activities/floors

Usage

```
get_floors(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: date and floors

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_floors(date)

## End(Not run)
```

get_floors_intraday *Get intraday floors time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_floors_intraday(
  date = lubridate::today(),
  detail_level = c("1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

date	A date to get data for
detail_level	The detail level. One of "1min", "5min", or "15min"
start_time	The start time of the time window. Default: NULL gets the whole day
end_time	The end time of the time window. Default: NULL gets the whole day

Value

A tibble with two columns: time and floors

See Also

Other intraday: get_active_zone_minutes_intraday(), get_calories_intraday(), get_distance_intraday(), get_elevation_intraday(), get_heart_rate_intraday(), get_steps_intraday()

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_floors_intraday(detail_level = "15min")

## get more granular data
get_floors_intraday(detail_level = "1min")

## End(Not run)
```

---

get_heart_rate_intraday

Get intraday heart time series

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_heart_rate_intraday(
  date = lubridate::today(),
  detail_level = c("1sec", "1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

date	A date to get data for
detail_level	The detail level. One of "1sec", "1min", "5min", or "15min"
start_time	The start time of the time window. Default: NULL gets the whole day
end_time	The end time of the time window. Default: NULL gets the whole day

See Also

Other intraday: get_active_zone_minutes_intraday(), get_calories_intraday(), get_distance_intraday(), get_elevation_intraday(), get_floors_intraday(), get_steps_intraday()

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_heart_rate_intraday(detail_level = "15min")

## get more granular data
get_heart_rate_intraday(detail_level = "1min")

## End(Not run)
```

---

get_heart_rate_zones *Heart Rate Zones*

Description

See https://dev.fitbit.com/build/reference/web-api/activity/ for more details.

Usage

```
get_heart_rate_zones(start_date, end_date = start_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble of the date, the heart rate zone (zone), the minimum heart rate in that zone (min_hr), the maximum heart rate in that zone (max_hr), the minutes in that zone (minutes_in_zone), and the calories burned in that zone (calories_out)

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()

get_heart_rate_zones(start_date, end_date)

## End(Not run)
```

`get_intraday_time_series` *Get an intraday time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_intraday_time_series(
  resource = c("active-zone-minutes", "calories", "distance", "elevation", "floors",
    "heart", "steps"),
  date,
  detail_level,
  start_time,
  end_time
)
```

Arguments

`resource`	The resource to get
`date`	A date to get data for
`detail_level`	The detail level. One of `"1min"`, `"5min"`, or `"15min"`
`start_time`	The start time of the time window. Default: `NULL` gets the whole day
`end_time`	The end time of the time window. Default: `NULL` gets the whole day

Value

A tibble with two columns: `time` and `{{resource}}`

get_lifetime_bests *Lifetime Bests*

Description

Retrieve lifetime best distance, floors, and steps

Usage

```
get_lifetime_bests()
```

Value

A tibble the best `distance`, `floors`, and `steps` (by date) tracked on any of your trackers

Examples

```
## Not run:
get_lifetime_bests()

## End(Not run)
```

get_lifetime_totals *Lifetime Totals*

Description

Retrieve lifetime total distance, floors, and steps

Usage

```
get_lifetime_totals()
```

Value

A tibble of all-time totals across trackers (i.e. the total `distance`, `floors`, and `steps` tracked across all of your trackers)

Examples

```
## Not run:
get_lifetime_totals()

## End(Not run)
```

get_minutes_fairly_active

Minutes Fairly Active Time Series

Description

Resource path /activities/minutesFairlyActive

Usage

```
get_minutes_fairly_active(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: `date` and `minutes_fairly_active`

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_minutes_fairly_active(date)

## End(Not run)
```

get_minutes_lightly_active

Minutes Lightly Active Time Series

Description

Resource path /activities/minutesLightlyActive

Usage

```
get_minutes_lightly_active(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: `date` and `minutes_lightly_active`

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_minutes_lightly_active(date)

## End(Not run)
```

`get_minutes_sedentary` *Minutes Sedentary Time Series*

Description

Resource path /activities/minutesSedentary

Usage

```
get_minutes_sedentary(start_date, end_date)
```

Arguments

`start_date`	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
`end_date`	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: `date` and `minutes_sedentary`

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_minutes_sedentary(date)

## End(Not run)
```

get_minutes_very_active *Minutes Very Active Time Series*

Description

Resource path /activities/minutesVeryActive

Usage

```
get_minutes_very_active(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: `date` and `minutes_very_active`

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_minutes_very_active(date)

## End(Not run)
```

get_sleep_stage_granular *Granular Sleep Stage Data*

Description

Returns a tibble of nightly sleep stage data. Very granular. Returns blocks of time spent in each phase.

Usage

```
get_sleep_stage_granular(start_date, end_date = start_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble of granular sleep stage data. This method is more granular than get_sleep_stage_summary, and returns blocks of time that you spent in each zone throughout the night.

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()

get_sleep_stage_granular(start_date, end_date)

## End(Not run)
```

---

get_sleep_stage_summary

Nightly Sleep Stage Summary Data

Description

Returns a tibble of nightly sleep stage data. Minutes in each stage, count of times in each stage, and a thirty day average for the number of minutes in each stage.

Usage

```
get_sleep_stage_summary(start_date, end_date = start_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble of a variety of sleep stage summary data, by day

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()

get_sleep_stage_summary(start_date, end_date)

## End(Not run)
```

`get_sleep_summary` *Nightly Sleep Summary*

Description

Returns a tibble of summary by night

Usage

```
get_sleep_summary(start_date, end_date = start_date)
```

Arguments

`start_date`	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
`end_date`	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble of a variety of sleep summary data by day

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()

get_sleep_summary(start_date, end_date)

## End(Not run)
```

get_steps *Steps Time Series*

Description

Resource path /activities/steps

Usage

```
get_steps(start_date, end_date)
```

Arguments

start_date	The start date of records to be returned in "yyyy-mm-dd" or date(time) format
end_date	The end date of records to be returned in "yyyy-mm-dd" or date(time) format

Value

A tibble with two columns: date and steps

Examples

```
## Not run:
start_date <- lubridate::today() - lubridate::weeks(1)
end_date <- lubridate::today()
get_steps(date)

## End(Not run)
```

get_steps_intraday *Get intraday steps time series*

Description

See the API documentation for more detailed explanations of parameters and more usage information and examples.

Usage

```
get_steps_intraday(
  date = lubridate::today(),
  detail_level = c("1min", "5min", "15min"),
  start_time = NULL,
  end_time = NULL
)
```

Arguments

date	A date to get data for
detail_level	The detail level. One of "1min", "5min", or "15min"
start_time	The start time of the time window. Default: NULL gets the whole day
end_time	The end time of the time window. Default: NULL gets the whole day

Value

A tibble with two columns: time and steps

See Also

Other intraday: get_active_zone_minutes_intraday(), get_calories_intraday(), get_distance_intraday(), get_elevation_intraday(), get_floors_intraday(), get_heart_rate_intraday()

Examples

```
## Not run:
date <- lubridate::today()

## get minute by minute data
get_steps_intraday(detail_level = "15min")

## get more granular data
get_steps_intraday(detail_level = "1min")

## End(Not run)
```

---

get_tracker_bests *Tracker Bests*

Description

Retrieve tracker best distance, floors, and steps

Usage

```
get_tracker_bests()
```

Value

A tibble the best distance, floors, and steps (by date) tracked on your tracker

Examples

```
## Not run:
get_tracker_bests()

## End(Not run)
```

`get_tracker_totals` *Tracker Totals*

Description

Retrieve tracker total distance, floors, and steps

Usage

```
get_tracker_totals()
```

Value

A tibble of all-time tracker totals (i.e. the total `distance`, `floors`, and `steps` tracked by your tracker)

Examples

```
## Not run:
get_tracker_totals()

## End(Not run)
```

`perform_get` *Perform a GET request*

Description

Perform a GET request

Usage

```
perform_get(url, ...)
```

Arguments

`url`	The URL to make the request to
`...`	Additional arguments (not currently used)

Value

The response

Index

∗ **intraday**
  get_active_zone_minutes_intraday, 4
  get_calories_intraday, 7
  get_distance_intraday, 9
  get_elevation_intraday, 11
  get_floors_intraday, 12
  get_heart_rate_intraday, 13
  get_steps_intraday, 22

generate_fitbitr_token, 2
get_active_zone_minutes_intraday, 4, *8, 9*, *11*, *13, 14*, *23*
get_activity_calories, 5
get_activity_summary, 5
get_calories, 6
get_calories_bmr, 7
get_calories_intraday, *4*, 7, *9*, *11*, *13, 14*, *23*
get_distance, 8
get_distance_intraday, *4*, *8*, 9, *11*, *13, 14*, *23*
get_elevation, 10
get_elevation_intraday, *4*, *8, 9*, 11, *13, 14*, *23*
get_floors, 12
get_floors_intraday, *4*, *8, 9*, *11*, 12, *14*, *23*
get_heart_rate_intraday, *4*, *8, 9*, *11*, *13*, 13, *23*
get_heart_rate_zones, 14
get_intraday_time_series, 15
get_lifetime_bests, 16
get_lifetime_totals, 16
get_minutes_fairly_active, 17
get_minutes_lightly_active, 17
get_minutes_sedentary, 18
get_minutes_very_active, 19
get_sleep_stage_granular, 19
get_sleep_stage_summary, *20*, 20
get_sleep_summary, 21
get_steps, 22
get_steps_intraday, *4*, *8, 9*, *11*, *13, 14*, 22
get_tracker_bests, 23
get_tracker_totals, 24

oauth2.0_token, *3*

perform_get, 24