Package 'pocketapi'

October 14, 2022

Title Wrapper Around the 'Pocket' API

Version 0.1

Description Functions that interface with the 'Pocket' API (<https://getpocket.com/developer/>). Allows the user to get, add, and modify items in their own 'Pocket' account.

Depends R (>= 3.5.0)

Imports httr, purrr, tibble, dplyr, glue, jsonlite, magrittr, usethis

License MIT + file LICENSE

Encoding UTF-8

LazyData true

BugReports <https://github.com/CorrelAid/pocketapi//issues>

URL <https://github.com/CorrelAid/pocketapi/>

RoxygenNote 7.1.1

Suggests testthat, mockery, covr, httptest, knitr, rmarkdown, ggplot2

VignetteBuilder knitr

NeedsCompilation no

Author Frie Preu [aut, cre],
Yannik Buhl [aut],
Max Alletsee [aut],
Dorian Le Jeune [ctb],
Christoph Dworschak [ctb],
Sarah Jukna [ctb],
CorrelAid e.V. [cph]

Maintainer Frie Preu <fr1e@pm.me>

Repository CRAN

Date/Publication 2020-11-20 10:20:02 UTC

R topics documented:

---

create_authorize_url   *create_authorize_url*

Description

Creates the URL the user needs to enter into her browser in order to authorize the Pocket application/the request token.

Usage

```
create_authorize_url(request_token)
```

Arguments

request_token   Character string. Pocket request token generated by the function get_request_token.

Details

See the GitHub README for details on Authentication.

Value

Character string. The URL for authorizing the Pocket application for which the request token was requested. This is what you enter into your browser.

See Also

Other authentication functions: get_access_token(), get_request_token()

---

get_access_token *get_access_token*

Description

Creates the URL the user needs to enter into her browser in order to authorize the Pocket application/the request token.

Usage

```
get_access_token(consumer_key, request_token)
```

Arguments

consumer_key   Character string. Here goes your Pocket consumer key.

request_token   Character string. Pocket request token generated by get_request_token.

Details

See the GitHub README for details on Authentication.

Value

Character string. Returns the access token.

See Also

Other authentication functions: create_authorize_url(), get_request_token()

---

get_request_token *get_request_token*

Description

Requests a request_token for the Pocket application with a given consumer key from the Pocket Authentication API. The request token can then be used in create_authorize_url and get_access_token.

Usage

```
get_request_token(consumer_key)
```

Arguments

consumer_key   Character string. Here goes your Pocket consumer key.

Details

See the [GitHub README](#) for details on Authentication.

Value

Character string. Request token for the Pocket Application corresponding to your consumer_key.

See Also

Other authentication functions: `create_authorize_url`(), `get_access_token`()

---

pocket_add   *pocket_add*

Description

Add one or more items to your Pocket account.

Usage

```
pocket_add(
  add_urls,
  item_ids = "",
  tags = NULL,
  success = TRUE,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

add_urls	Character vector. The URL or URLs you want to add to your Pocket list.
item_ids	Character vector. The item_ids of the items you want to add. Defaults to empty character vector.
tags	Character vector. One or more tags to be applied to all of the newly added URLs. Defaults to NULL.
success	Logical. Enables success/failure messages for every URL. Defaults to TRUE. Needs GET permission if TRUE.
consumer_key	Character string. Your Pocket consumer key. Defaults to `Sys.getenv("POCKET_CONSUMER_KEY")`.
access_token	Character string. Your Pocket request token. Defaults to `Sys.getenv("POCKET_ACCESS_TOKEN")`.

Details

This function uses the `modify` endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

Value

the response from the httr call, invisibly

---

pocket_archive   *pocket_archive*

Description

archive items from your Pocket list.

Usage

```
pocket_archive(
  item_ids,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

item_ids	character vector. Pocket item ids you want to archive.
consumer_key	character. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
access_token	character. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

This function uses the modify endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

Value

Invisibly returns a list containing information on whether the action failed or succeeded, including the respective item ID.

---

pocket_delete   *pocket_delete*

Description

delete items from your Pocket list.

Usage

```
pocket_delete(
  item_ids,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

item_ids	character vector. Pocket item ids you want to delete from your list.
consumer_key	character. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
access_token	character. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

This function uses the modify endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

Value

Invisibly returns a list containing information on whether the action failed or succeeded, including the respective item ID.

---

pocket_favorite   *pocket_favorite*

Description

favorite items from your Pocket list.

Usage

```
pocket_favorite(
  item_ids,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

`item_ids`	character vector. Pocket item ids you want to favorite
`consumer_key`	character. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
`access_token`	character. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

This function uses the `modify` endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

Value

Invisibly returns a list containing information on whether the action failed or succeeded, including the respective item ID.

`pocket_get`	*pocket_get*

Description

Get a data frame with your pocket data.

Usage

```
pocket_get(
  favorite = NULL,
  item_type = NULL,
  tag = NULL,
  state = "all",
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

`favorite`	boolean. Default NULL. Allows to filter for favorited items. If TRUE, only favorited items will be returned. If FALSE, only un-favorited items will be returned.
`item_type`	character. Default NULL. Allows to filter for content type of items. Valid values are: "image", "article", "video". Please note that there might be Pocket items that do not belong to any of those types. The Pocket API documentation only mentions those three.
`tag`	character. Default NULL. Only one tag can be filtered at a time. Set to '_untagged_' if you only want to get untagged items.

state	character. Default "all". Allows to filter on unread/archived items or return all. Valid values are "unread", "archive", "all".
consumer_key	character. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
access_token	character. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

See https://getpocket.com/developer/docs/v3/retrieve for the meaning of certain variable values.

Value

tibble. Tibble with one row for each Pocket item.

---

pocket_modify   *pocket_modify*

Description

Function that sends a request with a list of actions to the 'modify' Pocket API endpoint.

Usage

```
pocket_modify(
  actions,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

actions	List. List of lists where each element is an action object. See https://getpocket.com/developer/docs/v3/mod
consumer_key	Character string. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
access_token	Character string. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

see https://getpocket.com/developer/docs/v3/modify. This function uses the modify endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

---

pocket_tag *pocket_tag*

Description

modify the tags of the items in pocket.

Usage

```
pocket_tag(
  action_name = c("tags_replace", "tags_remove", "tags_add", "tags_clear",
    "tag_rename", "tag_delete"),
  item_ids = NULL,
  tags = NULL,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

action_name	character vector. The kind of tag action you want to undertake. Possible values: 'tags_add', 'tags_remove', 'tags_replace', 'tags_clear', 'tag_rename', or 'tag_delete'.
item_ids	character vector. Pocket item ids you want to modify the tags for.
tags	character vector. The names of the tags to work with the chosen action.
consumer_key	character. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
access_token	character. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

This function uses the `modify` endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

---

pocket_unfavorite *pocket_unfavorite*

Description

Unfavorite items from your Pocket list.

Usage

```
pocket_unfavorite(
  item_ids,
  consumer_key = Sys.getenv("POCKET_CONSUMER_KEY"),
  access_token = Sys.getenv("POCKET_ACCESS_TOKEN")
)
```

Arguments

`item_ids`	Character vector. Pocket item IDs you want to unfavorite. Get them, e.g., via `pocket_get`.
`consumer_key`	character. Your Pocket consumer key. Defaults to Sys.getenv("POCKET_CONSUMER_KEY").
`access_token`	character. Your Pocket request token. Defaults to Sys.getenv("POCKET_ACCESS_TOKEN").

Details

This function uses the `modify` endpoint of the Pocket API which exhibits some weird behaviour. For example, even if a 'modify' action is not successful, the API will still return "success". See [issue [#26](https://github.com/CorrelAid/pocketapi/issues/26) for a discussion.

Value

Invisibly returns a list containing information on whether the action failed or succeeded, including the respective item ID.

Index