Package ‘fbRads’

October 13, 2022

Type Package

Maintainer Gergely Daroczi <gergely.daroczi@card.com>

Author Ajaykumar Gopal <ajay@card.com>, Gergely Daroczi <gergely.daroczi@card.com>

Title Analyzing and Managing Facebook Ads from R

Description Wrapper functions around the Facebook Marketing 'API' to create, read, update and delete custom audiences, images, campaigns, ad sets, ads and related content.

Version 0.2

Date 2016-04-04

URL https://github.com/cardcorp/fbRads

License AGPL-3

Imports RCurl, jsonlite, digest, futile.logger, bit64, plyr, data.table

NeedsCompilation no

Repository CRAN

Date/Publication 2016-04-06 01:20:49

R topics documented:

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>fbad_add_audience</td>
<td>2</td>
</tr>
<tr>
<td>fbad_create_ad</td>
<td>3</td>
</tr>
<tr>
<td>fbad_create_adset</td>
<td>3</td>
</tr>
<tr>
<td>fbad_create_audience</td>
<td>5</td>
</tr>
<tr>
<td>fbad_create_campaign</td>
<td>5</td>
</tr>
<tr>
<td>fbad_create_creative</td>
<td>6</td>
</tr>
<tr>
<td>fbad_create_image</td>
<td>7</td>
</tr>
<tr>
<td>fbad_create_lookalike_audience</td>
<td>8</td>
</tr>
<tr>
<td>fbad_delete_audience</td>
<td>9</td>
</tr>
<tr>
<td>fbad_get_adaccount_details</td>
<td>9</td>
</tr>
<tr>
<td>fbad_get_search</td>
<td>10</td>
</tr>
<tr>
<td>fbad_init</td>
<td>11</td>
</tr>
<tr>
<td>fbad_list_ad</td>
<td>12</td>
</tr>
</tbody>
</table>
Add people to a custom FB audience

**fbad_add_audience**

**Description**

Add people to a custom FB audience

**Usage**

```r
fbad_add_audience(fbacc, audience_id, schema = c("EMAIL", "PHONE"), hashes)
```

**Arguments**

- **fbacc**: (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **audience_id**: string
- **schema**: only two schema are supported out of the four: you can add/remove persons to/from a custom audience by e-mail addresses or phone numbers
- **hashes**: character vector of e-mail addresses or phone numbers to be transformed to hashes

**References**

[https://developers.facebook.com/docs/marketing-api/custom-audience-targeting/v2.4#create](https://developers.facebook.com/docs/marketing-api/custom-audience-targeting/v2.4#create)
fbad_create_ad

Description
Create ad

Usage

```r
fbad_create_ad(fbacc, name, campaign_id, adset_id, creative_id,
    adgroup_status = c("ACTIVE", "PAUSED"), status = c("ACTIVE", "PAUSED"),
    ...)
```

Arguments

- **fbacc**: (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **name**: Ad group name
- **campaign_id**: Ad Set id (v2.4)
- **adset_id**: Ad Set id (v2.5)
- **creative_id**: creative ID
- **adgroup_status**: initial status of the Ad group (v2.4)
- **status**: initial status of the Ad group (v2.5)
- **...**: further parameters passed to the Facebook API

Value

- ad id

References

- [https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#Creating](https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#Creating)
fbad_create_adset

Usage

```r
fbad_create_adset(fbacc, name, optimization_goal = c("NONE", "APP_INSTALLS", "CLICKS", "ENGAGED_USERS", "EXTERNAL", "EVENT_RESPONSES", "IMPRESSIONS", "LINK_CLICKS", "OFFER_ClAIMS", "OFFSITE_CONVERSIONS", "PAGE_ENGAGEMENT", "PAGE_LIKES", "POST_ENGAGEMENT", "REACH", "SOCIAL_IMPRESSIONS", "VIDEO_VIEWS"), billing_event = c("APP_INSTALLS", "CLICKS", "IMPRESSIONS", "LINK_CLICKS", "OFFER_ClAIMS", "PAGE_LIKES", "POST_ENGAGEMENT", "VIDEO_VIEWS"), bid_amount, promoted_object, campaign_id, status = c("ACTIVE", "PAUSED", "ARCHIVED", "DELETED"), daily_budget, lifetime_budget, end_time, start_time, targeting, campaign_group_id, campaign_status = c("ACTIVE", "PAUSED", "ARCHIVED", "DELETED"), ...)```

Arguments

- `fbacc` (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`
- `name` name of the Ad Set
- `optimization_goal` optimization goal
- `billing_event` the billing event
- `bid_amount` integer
- `promoted_object` see at https://developers.facebook.com/docs/marketing-api/reference/ad-campaign/promoted-object/v2.4
- `campaign_id` parent Ad Campaign id (v2.5)
- `status` Ad Set status (v2.5)
- `daily_budget` using account currency
- `lifetime_budget` using account currency
- `end_time` UTC UNIX timestamp
- `start_time` UTC UNIX timestamp
- `targeting` list
- `campaign_group_id` parent Ad Campaign id (v2.4)
- `campaign_status` Ad Set status (v2.4)
- `...` further arguments passed to the API endpoint

Value

Ad Set id

References

https://developers.facebook.com/docs/marketing-api/reference/ad-campaign#Creating
**fbad_create_audience**  
*Create a new FB custom audience*

**Description**
Create a new FB custom audience

**Usage**

```r
fbad_create_audience(fbacc, name, description, opt_out_link)
```

**Arguments**

- **fbacc** (optional) FB_Ad_account object, which defaults to the last returned object of *fbad_init*.
- **name** string
- **description** optional string
- **opt_out_link** optional link

**Value**
custom audience ID

**References**
https://developers.facebook.com/docs/marketing-api/custom-audience-targeting/v2.4#create

**fbad_create_campaign**  
*Created Ad Campaign*

**Description**
Created Ad Campaign

**Usage**

```r
fbad_create_campaign(fbacc, buying_type = c("AUCTION", "FIXED_CPM", "RESERVED"), campaign_group_status = c("ACTIVE", "PAUSED"), campaign_status = c("ACTIVE", "PAUSED"), execution_options = NULL, name, objective, spend_cap = NULL)
```
**Arguments**

- **fbacc** (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **buying_type** Facebook optimization algorithm to delivery, pricing, and limits
- **campaign_group_status** initial status of the Ad Campaign (v2.4)
- **campaign_status** initial status of the Ad Campaign (v2.5)
- **execution_options** special execution settings passed to the API
- **name** Ad Campaign name
- **objective** the campaign’s objective
- **spend_cap** spend cap of the campaign

**Value**

- Ad Campaign id

**References**

https://developers.facebook.com/docs/marketing-api/reference/ad-campaign-group#

---

**fbad_create_creative** *Create an ad creative*

**Description**

Create an ad creative

**Usage**

```r
fbad_create_creative(fbacc, title, body, name, actor_id, follow_redirect, image_file, image_hash, image_crops, image_url, link_url, url_tags, object_id, object_story_id, object_story_spec, object_url, call_to_action_type = c("OPEN_LINK", "BOOK_TRAVEL", "SHOP_NOW", "PLAY_GAME", "LISTEN_MUSIC", "WATCH_VIDEO", "USE_APP"))
```

**Arguments**

- **fbacc** (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **title** string
- **body** string
**fbad_create_image**

- **name**: string
- **actor_id**: Facebook object ID reference
- **follow_redirect**: boolean
- **image_file**: local image passed to Facebook. You might first upload the image via fbad_create_image.
- **image_hash**: string
- **image_crops**: list
- **image_url**: string
- **link_url**: string
- **url_tags**: list
- **object_id**: Facebook object ID reference
- **object_story_id**: post ID reference
- **object_story_spec**: list
- **object_url**: string list
- **call_to_action_type**: string

**Value**
- creative id

**Note**
There are no checks done before passing provided arguments to Facebook. You have to know what you are up to. Read the Facebook docs.

**References**

https://developers.facebook.com/docs/marketing-api/adcreative/v2.4#create

---

**Description**
Upload image

**Usage**

```python
fbad_create_image(fbacc, img)
```
**fbad_create_lookalike_audience**

Create a new FB lookalike audience similar to an already existing custom audience

**Description**

Create a new FB lookalike audience similar to an already existing custom audience

**Usage**

```python
fbad_create_lookalike_audience(fbacc, name, origin_audience_id, ratio = 0.01, country = "US")
```

**Arguments**

- **fbacc** (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **name** string
- **origin_audience_id** numeric ID of origin custom audience
- **ratio** Between 0.01-0.20 and increments of 0.01. Indicates the top ratio percent of original audience in the selected country
- **country** Country name - the country to find the lookalike people.

**Value**

lookalike audience ID

**References**

https://developers.facebook.com/docs/marketing-api/adimage/v2.4#create
**fbad_delete_audience**  
*Delete a FB custom audience*

**Description**  
Delete a FB custom audience

**Usage**  
```python  
fbad_delete_audience(fbacc, audience_id)  
```

**Arguments**  
- **fbacc**  
  (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **audience_id**  
  numeric

**Value**  
custom audience ID

**References**  
https://developers.facebook.com/docs/marketing-api/custom-audience-targeting/v2.4#delete

---

**fbad_get_adaccount_details**  
*Get details for a Facebook Ads Account*

**Description**  
Get details for a Facebook Ads Account

**Usage**  
```python  
fbad_get_adaccount_details(accountid, token, version)  
```

**Arguments**  
- **accountid**  
  Ads account graph object id
- **token**  
  FB Ads API token
- **version**  
  Facebook Marketing API version
Value

list(s) containing account details

References

https://developers.facebook.com/docs/marketing-api/adaccount/v2.3

fbad_get_search  FB Search API Querying

Description

FB Search API Querying

Usage

fbad_get_search(fbacc, q, type = c("adeducationschool", "adeducationmajor", "adgeolocation", "adcountry", "adzipcode", "adgeolocationmeta", "adradiussuggestion", "adinterest", "adinterestsuggestion", "adinterestvalid", "adlocale", "adTargetingCategory", "adworkemployer"), ...)

Arguments

fbacc (optional) FB_Ad_account object, which defaults to the last returned object of fbad_init.
q string that is being searched for
type describes the type of search eg: adinterest, adeducationmajor etc
... other optional parameters accepted by the endpoint as key = value pairs eg: limit = 5000.

Value
data.frame containing results

References

https://developers.facebook.com/docs/marketing-api/targeting-search/v2.5

Examples

## Not run:
fbad_get_search(q = 'r programming language', type = 'adinterest')

fbad_get_search(q = c('dog', 'cat'), type = 'adinterestvalid')

## End(Not run)
Description

If you do not have a token, then register an (e.g. "Website") application at [https://developers.facebook.com/apps](https://developers.facebook.com/apps) and make a note of your "App ID" and "App Secret" at the "Dashboard" of your application. Then go to "Settings", click on "Add Platform", then "Website" and paste [http://localhost:1410](http://localhost:1410) as the "Site URL". Save, and then run the below example R commands to get your token. Please note that your app needs access to your ads as well, see [https://developers.facebook.com/docs/marketing-api/access](https://developers.facebook.com/docs/marketing-api/access) for more details.

Usage

`fbad_init(accountid, token, version = fb_api_most_recent_version())`

Arguments

- **accountid**: Facebook Ad account id without the `act_` prefix
- **token**: Facebook OAuth token as a string
- **version**: Facebook Marketing API version

Value

- list returned invisibly containing versioned base URL and relevant API parameters

Examples

```r
## Not run:
## You can generate a token for future use with the help of `httr`, e.g.
library(httr)
app <- oauth_app("facebook", your_app_id, your_app_secret)
oauth2.0_token(oauth_endpoints("facebook"), app,
  scope = '"',
  type = "application/x-www-form-urlencoded")$credentials$access_token

## Then pass this token with your account ID to fbad_init

## End(Not run)
```
fbad_list_ad

*List all Ads for current account, list of Ad Sets or Campaigns*

**Description**

List all Ads for current account, list of Ad Sets or Campaigns

**Usage**

`fbad_list_ad(fbacc, id, statuses, fields = "id")`

**Arguments**

- `fbacc` (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `id` will do the look-up for all Ads based on this ID. Defaults to current FB account. Can be a (vector of) Ad Set or Campaign id(s).
- `statuses` character vector of statuses to use as a filter. Defaults to none. Please refer to the Facebook documentation for a list of possible values.
- `fields` character vector of fields to get from the API, defaults to `id`. Please refer to the Facebook documentation for a list of possible values.

**Value**

data.frame

**Note**

Will do a batched request to the Facebook API if multiple ids are provided.

**References**

[https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#read-adaccount](https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#read-adaccount)

fbad_list_adset

*List all Ad Sets for current account or Ad Campaign(s)*

**Description**

List all Ad Sets for current account or Ad Campaign(s)

**Usage**

`fbad_list_adset(fbacc, id, statuses, fields = "id")`
Arguments

- **fbacc** (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **id** will do the look-up for all Ads based on this ID. Defaults to current FB account. Can be a (vector of) Ad Campaign id(s).
- **statuses** character vector of statuses to use as a filter. Defaults to none. Please refer to the Facebook documentation for a list of possible values.
- **fields** character vector of fields to get from the API, defaults to id. Please refer to the Facebook documentation for a list of possible values.

References

https://developers.facebook.com/docs/marketing-api/reference/ad-campaign#Reading

---

**fbad_list_campaign**

List all Ad Campaigns for current account

Description

List all Ad Campaigns for current account

Usage

```r
fbad_list_campaign(fbacc, id, statuses, fields = "id")
```

Arguments

- **fbacc** (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **id** not supported argument
- **statuses** character vector of statuses to use as a filter. Defaults to none. Please refer to the Facebook documentation for a list of possible values.
- **fields** character vector of fields to get from the API, defaults to id. Please refer to the Facebook documentation for a list of possible values.

References

https://developers.facebook.com/docs/marketing-api/reference/ad-campaign-group#Reading
Query for reach estimate for given targeting spec

Description
Query for reach estimate for given targeting spec

Usage
```r
fbad_reachestimate(fbacc, targeting_spec, currency = "USD",
optimize_for = c("NONE", "APP_INSTALLS", "CLICKS", "ENGAGED_USERS",
"EXTERNAL", "EVENT_responses", "IMPRESSIONS", "LINK_CLICKS", "OFFER_CLAIMS",
"OFFSITE_CONVERSIONS", "PAGE_ENGAGEMENT", "PAGELIKES", "POST_ENGAGEMENT",
"REACH", "SOCIAL_IMPRESSIOns", "VIDEO_VIEWS"))
```

Arguments
- **fbacc**: (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **targeting_spec**: lists of targeting spec characteristics as described at https://developers.facebook.com/docs/marketing-api/targeting-specs/v2.5
- **currency**: string
- **optimize_for**: what are you optimizing for in the planned Ad Set?

Value
list

References
https://developers.facebook.com/docs/marketing-api/reference/ad-account/reachestimate/

Examples
```r
## Not run:
targetspec <- list(
  age_min = unbox(24),
  age_max = unbox(55),
  geo_locations = list(countries = 'US'))
fbad_reachestimate(targeting_spec = targetspec)

## End(Not run)
```
**fbad_read_ad**  
*Read ad details*

**Description**  
Read ad details

**Usage**  
```r  
fbad_read_ad(fbacc, id, fields = "id")  
```

**Arguments**  
- **fbacc**  
  (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **id**  
  ad id(s)
- **fields**  
  character vector of fields to get from the API, defaults to `id`. Please refer to the Facebook documentation for a list of possible values.

**Value**  
`data.frame`

**Note**  
Will do a batched request to the Facebook API if multiple ids are provided.

**References**  
[https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#Reading](https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#Reading)

**Examples**  
```r  
## Not run:  
## get and Ad ID from yesterday  
adid <- fb_insights(date_preset = 'yesterday', level = 'ad')[[1]]$ad_id[1]  
## look for current status of the Ad  
fbad_read_ad(id = adid, fields = c('effective_status'))  
## End(Not run)  
```
Description

Read Ad Set details

Usage

`fbad_read_adset(fbacc, id, fields = "id")`

Arguments

- **fbacc**: (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **id**: ad set id(s)
- **fields**: character vector of fields to get from the API, defaults to id. Please refer to the Facebook documentation for a list of possible values.

References

[https://developers.facebook.com/docs/marketing-api/reference/ad-campaign#Reading](https://developers.facebook.com/docs/marketing-api/reference/ad-campaign#Reading)

Description

Read metadata on a FB custom audience

Usage

`fbad_read_audience(fbacc, audience_id, fields = c("id", "account_id", "approximate_count", "data_source", "delivery_status", "lookalike_audience_ids", "lookalike_spec", "name", "permission_for_actions", "operation_status", "subtype", "time_updated"))`

Arguments

- **fbacc**: (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- **audience_id**: numeric
- **fields**: character vector of fields to be returned
**fbad_read_campaign**

**Value**

custom audience ID

**References**

https://developers.facebook.com/docs/marketing-api/custom-audience-targeting/v2.4#read

---

**fbad_read_campaign**  
*Read Ad Campaign details*

**Description**

Read Ad Campaign details

**Usage**

`fbad_read_campaign(fbacc, id, fields = "id")`

**Arguments**

- **fbacc**  
  (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.  
- **id**  
  Ad Campaign id(s)  
- **fields**  
  character vector of fields to get from the API, defaults to id. Please refer to the Facebook documentation for a list of possible values.

**References**

https://developers.facebook.com/docs/marketing-api/reference/ad-campaign-group#Reading

---

**fbad_read_creative**  
*Read ad creative*

**Description**

Read ad creative

**Usage**

**fbad_remove_audience**

Add people from a custom FB audience

**Description**

Add people from a custom FB audience

**Usage**

`fbad_remove_audience(fbacc, audience_id, schema = c("EMAIL", "PHONE"), hashes)`

**Arguments**

- `fbacc`: (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`
- `audience_id`: string
- `schema`: only two schema are supported out of the four: you can add/remove persons to/from a custom audience by e-mail addresses or phone numbers
- `hashes`: character vector of e-mail addresses or phone numbers to be transformed to hashes

**References**

[https://developers.facebook.com/docs/marketing-api/reference/custom-audience/users#Deleting](https://developers.facebook.com/docs/marketing-api/reference/custom-audience/users#Deleting)
**fbad_share_audience**  
*Share a FB custom audience with other accounts*

**Description**

Share a FB custom audience with other accounts

**Usage**

```r
fbad_share_audience(fbacc, audience_id, adaccounts)
```

**Arguments**

- `fbacc`  
  (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `audience_id`  
  audience ID
- `adaccounts`  
  numeric vector of FB account IDs

**Note**

This throws error if you provide wrong account ids OR even valid account ids that were previously granted access to the given custom audience.

**References**

https://developers.facebook.com/docs/marketing-api/custom-audience-targeting/v2.3#sharing

---

**fbad_update_ad**  
*Update ad*

**Description**

Update ad

**Usage**

```r
fbad_update_ad(fbacc, id, ...)
```

**Arguments**

- `fbacc`  
  (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `id`  
  ad id
- `...`  
  parameters passed to Facebook API
Value

invisible TRUE

References

https://developers.facebook.com/docs/marketing-api/reference/adgroup/v2.5#Updating

`fbad_update_adset`  
*Update Ad Set*

**Description**

Update Ad Set

**Usage**

`fbad_update_adset(fbac, id, ...)`

**Arguments**

- `fbac` (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `id`  Ad Set id
- `...`  parameters passed to Facebook API

References

https://developers.facebook.com/docs/marketing-api/reference/ad-campaign#Updating

`fbad_update_campaign`  
*Update Ad Campaign*

**Description**

Update Ad Campaign

**Usage**

`fbad_update_campaign(fbac, id, ...)`

**Arguments**

- `fbac` (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `id`  Ad Campaign id
- `...`  parameters passed to Facebook API
fbRads

References
https://developers.facebook.com/docs/marketing-api/reference/ad-campaign-group#
Updating

---

**fbRads**

*fbRads package*

---

**Description**
This is a placeholder for storing import directives, please find more details in the README.md file of the package via system.file or on GitHub at https://github.com/cardcorp/fbRads.

---

**fb_api_most_recent_version**

*Returns the most recent version of the supported Facebook Marketing API*

---

**Description**
Returns the most recent version of the supported Facebook Marketing API

**Usage**

`fb_api_most_recent_version()`

**Value**

`string`

---

**fb_api_version**

*Returns the currently used version of the Facebook Marketing API*

---

**Description**
Returns the currently used version of the Facebook Marketing API

**Usage**

`fb_api_version()`

**Value**

`string`
fb_insights

Description

Insights

Usage

fb_insights(fbacc, target = fbacc$acct_path, job_type = c("sync", "async"), retries = 0, ...)

Arguments

fbacc (optional) FB_Ad_account object, which defaults to the last returned object of fbad_init.
target ad account id(s) (default), campaign id(s), adset id(s) or ad id(s)
job_type synchronous or asynchronous request. If the prior fails with "please reduce the amount of data", it will fall back to async request. Async query is possible with only one target.
retries number of times this query has been sent to Facebook previously and failed – to be used internally for error handling
... named arguments passed to the API, like time range, fields, filtering etc.

Value

list

References

https://developers.facebook.com/docs/marketing-api/insights/v2.5

Examples

## Not run:
fb_insights(fbacc)

## process results
l <- fb_insights(fbacc, date_preset = 'today', level = 'ad')
library(rlist)
list.stack(list.select(l, date_start, date_stop, ad_id, total_actions, total_unique_actions, total_action_value, impressions, unique_impressions, social_impressions, unique_social_impressions, clicks, unique_clicks, social_clicks, unique_social_clicks, spend, frequency, deeplink_clicks, app_store_clicks, website_clicks, reach, social_reach, ctr, unique_ctr, cpc, cpm, cpp, cost_per_total_action, cost_per_unique_click, relevance_score = relevance_score$score))

## End(Not run)
**fb_reportstats_ad**

Get Ad Report Stats

**Usage**

```r
fb_reportstats_ad(fbacc, ...)```

**Arguments**

- `fbacc` (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `...` named arguments passed to the API, like time range, fields, filtering etc.

**Value**

data.frame

**Note**

This API endpoint is be soon deprecated and only available with v2.3. Use the new Insight feature, if possible.

**References**

[https://developers.facebook.com/docs/marketing-api/adreportstats/v2.3](https://developers.facebook.com/docs/marketing-api/adreportstats/v2.3)

---

**fb_stats_ad**

Get Ad Stats

**Usage**

```r
fb_stats_ad(fbacc, ids)
```

**Arguments**

- `fbacc` (optional) FB_Ad_account object, which defaults to the last returned object of `fbad_init`.
- `ids` adgroup ids
print.FB_Ad_Account

Value

list (raw parsed JSON) to be post-processed

Note

This API endpoint is be soon deprecated and only available with v2.3. Use the new Insight feature, if possible.

References

https://developers.facebook.com/docs/marketing-api/adstatistics/v2.3#read

Examples

```r
## Not run:
res <- fb_stats_ad(...)  
library(rlist)
list.stack(list.select(res, adgroup_id, impressions, clicks, spent, unique_impressions))

## End(Not run)
```

print.FB_Ad_Account  

Description

Print method for custom fbRads class

Usage

```r
## S3 method for class 'FB_Ad_Account'
print(x, ...)
```

Arguments

- `x` R object with FB_Ad_Account class
- `...` further arguments passed to print (ignored)
Index

fb_api_most_recent_version, 21
fb_api_version, 21
fb_insights, 22
fb_reportstats_ad, 23
fb_stats_ad, 23
fbad_add_audience, 2
fbad_create_ad, 3
fbad_create_adset, 3
fbad_create_audience, 5
fbad_create_campaign, 5
fbad_create_creative, 6
fbad_create_image, 7
fbad_create_lookalike_audience, 8
fbad_delete_audience, 9
fbad_get_adaccount_details, 9
fbad_get_search, 10
fbad_init, 2–6, 8–10, 11, 12–20, 22, 23
fbad_list_ad, 12
fbad_list_adset, 12
fbad_list_campaign, 13
fbad_reachestimate, 14
fbad_read_ad, 15
fbad_read_adset, 16
fbad_read_audience, 16
fbad_read_campaign, 17
fbad_read_creative, 17
fbad_remove_audience, 18
fbad_share_audience, 19
fbad_update_ad, 19
fbad_update_adset, 20
fbad_update_campaign, 20
fbRads, 21
fbRads-package (fbRads), 21
print.FB_Ad_Account, 24