

Package ‘hlidacr’

July 22, 2025

Type Package

Title Access Data from the 'Hlídač Státu' API

Version 0.2.0

Date 2021-09-16

Description

Provides access to datasets published by 'Hlídač státu' <<https://www.hlidacstatu.cz/>>, a Czech watchdog, via their API.

License MIT + file LICENSE

URL <https://github.com/skvrnami/hlidacr>

BugReports <https://github.com/skvrnami/hlidacr/issues>

Encoding UTF-8

Suggests testthat, covr, knitr, rmarkdown, vcr

RoxygenNote 7.1.1

Imports httr, jsonlite, glue, purrr, stringr, curl, usethis, urltools

VignetteBuilder knitr

NeedsCompilation no

Author Michael Škvrňák [aut, cre] (ORCID:
<<https://orcid.org/0000-0003-4647-4095>>)

Maintainer Michael Škvrňák <michael.skvrnak@protonmail.com>

Repository CRAN

Date/Publication 2021-09-16 13:40:02 UTC

Contents

check_connection	2
check_token	2
check_types	3
create_type	3
get_company	4

get_contract	4
get_contract_text	5
get_datasets	6
get_dataset_data	7
get_dataset_metadata	8
get_dataset_record_detail	8
get_person	9
get_person_social	10
get_subsidy	11
get_websites	12
get_website_detail	12
handle_error_response	13
search_contracts	14
search_person	15
search_person_by_parameters	16
search_subsidies	17
Index	18

check_connection	<i>Check internet connection</i>
------------------	----------------------------------

Description

Check internet connection

Usage

check_connection()

Value

No return value, called for side effects

check_token	<i>Check API token</i>
-------------	------------------------

Description

Check API token

Usage

check_token(token)

Arguments

token API token to Hlidac statu

Value

No return value, called for side effects

check_types	<i>Check if all social media account types are valid</i>
-------------	--

Description

Check if all social media account types are valid

Usage

```
check_types(types)
```

Arguments

types Vector with types of social media accounts

Value

No return value, called for side effects

create_type	<i>Create query string for osoby/social API endpoint</i>
-------------	--

Description

Create query string for osoby/social API endpoint

Usage

```
create_type(types)
```

Arguments

types Vector with types of social media accounts

Value

String for querying API

get_company	<i>Get company details</i>
-------------	----------------------------

Description

Get company details

Usage

```
get_company(company_name, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

company_name	Name of the company
token	Access token

Value

list with 3 elements: - ico: ID number of the company - jmeno: Name of the company - datoveSchranky: Data post box - electronic repository for communication between public institutions and private companies/people

See Also

<https://www.hlidacstatu.cz/swagger/index.html>
<https://www.hlidacstatu.cz/api/v1/doc>

Examples

```
## Not run:  
get_company("Agrofert")  
  
## End(Not run)
```

get_contract	<i>Get contract details</i>
--------------	-----------------------------

Description

Get contract details

Usage

```
get_contract(id, token = NULL)
```

Arguments

id	ID of contract
token	Authorization token

Value

list with data

See Also

<https://www.hlidacstatu.cz/swagger/index.html>
<https://www.hlidacstatu.cz/api/v1/doc>
<https://smlouvy.gov.cz/>

Examples

```
## Not run:  
get_contract(id = "1086905")  
  
## End(Not run)
```

get_contract_text	<i>Get contract text</i>
-------------------	--------------------------

Description

Get contract text

Usage

```
get_contract_text(id, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

id	ID of contract
token	Authorization token

Value

Character vector containing text of the contract

See Also

<https://www.hlidacstatu.cz/swagger/index.html>
<https://www.hlidacstatu.cz/api/v1/doc>
<https://smlouvy.gov.cz/>

Examples

```
## Not run:  
get_contract_text(id = "1086905")  
  
## End(Not run)
```

get_datasets	<i>Get list of datasets</i>
--------------	-----------------------------

Description

Get list of datasets available via 'Hlídač státu' API

Usage

```
get_datasets(token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

token	Authorization token
-------	---------------------

Value

Response from the server containing a list of all datasets available. The list contains 3 elements: - total: Total number of datasets available (integer) - page: Page of the result (integer) - results: data concerning the datasets (data.frame)

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Datasets: [get_dataset_data\(\)](#), [get_dataset_metadata\(\)](#), [get_dataset_record_detail\(\)](#)

Examples

```
## Not run:  
get_datasets()  
  
## End(Not run)
```

get_dataset_data	<i>Get data from the dataset</i>
------------------	----------------------------------

Description

Get data from the dataset

Usage

```
get_dataset_data(  
    dataset_id,  
    token = Sys.getenv("HLIDAC_TOKEN"),  
    query = NULL,  
    page = 1,  
    sort = NULL,  
    desc = NULL  
)
```

Arguments

dataset_id	ID of dataset
token	Authorization token
query	Search query
page	Number of page (Please note that if the page parameter exceeds 200, the function returns error)
sort	Sort by column
desc	Descending or ascending sorting

Value

list containing 3 elements: - total: Total number of datasets available (integer) - page: Page of the result (integer), equal to the 'page' argument of the function - results: Data.frame with data, columns vary depending on the dataset

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Datasets: [get_dataset_metadata\(\)](#), [get_dataset_record_detail\(\)](#), [get_datasets\(\)](#)

Examples

```
## Not run:  
get_dataset_data("ministri")  
get_dataset_data("ministri", query = "Zeman", page = 1)  
  
## End(Not run)
```

get_dataset_metadata *Get dataset's metadata*

Description

Get metadata related to specified dataset.

Usage

```
get_dataset_metadata(dataset_id, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

dataset_id	ID of dataset
token	Authorization token

Value

a list containing metadata related to the specified dataset

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Datasets: [get_dataset_data\(\)](#), [get_dataset_record_detail\(\)](#), [get_datasets\(\)](#)

Examples

```
## Not run:  
get_dataset_metadata("ministri")  
  
## End(Not run)
```

get_dataset_record_detail
 Get record detail

Description

Get record detail

Usage

```
get_dataset_record_detail(  
  dataset_id,  
  item_id,  
  token = Sys.getenv("HLIDAC_TOKEN")  
)
```

Arguments

dataset_id	ID of dataset
item_id	ID of record
token	Authorization token

Value

list of vectors containing data related to the item

See Also

[get_dataset_data](https://www.hlidacstatu.cz/swagger/index.html) <https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Datasets: [get_dataset_data\(\)](#), [get_dataset_metadata\(\)](#), [get_datasets\(\)](#)

Examples

```
## Not run:  
get_dataset_record_detail("ministri", item_id = 1)  
  
## End(Not run)
```

get_person

Get person details

Description

Get person details

Usage

```
get_person(person_id, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

person_id	Person's ID
token	Access token

Value

list with the following elements: - titulPred: academic titles written before person's name - jmeno: given name - prijmeni: surname - titulPo: academic titles written after person's name (if applicable) - narozeni: date of birth - nameId: person's id - profile: URL of profile at hlidacstatu.cz - sponzoring: data.frame with data on sponsorship of political parties by the person - udalosti: events related to the person such as party membership and running in election, serving in political bodies and relations to private companies (shareholder, serving on a board) - socialniSite: data.frame with data on social media accounts

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Persons: [get_person_social\(\)](#), [search_person_by_parameters\(\)](#), [search_person\(\)](#)

Examples

```
## Not run:
get_person("andrej-babis")

## End(Not run)
```

get_person_social	<i>Get person social media accounts</i>
-------------------	---

Description

Get person social media accounts

Usage

```
get_person_social(types, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

types	Types of social media accounts, only the following types are allowed: 'Twitter', 'Facebook_page', 'Facebook_profile', 'Instagram', 'WWW', 'Youtube'
token	Access token

Value

data.frame with data on persons' social accounts with the following columns: - titulPred: academic titles written before person's name - jmeno: given name - prijmeni: surname - titulPo: academic titles written after person's name - nameId: person's ID - profile: URL of profile at hlidacstatu.cz - socialniSite: data.frame with data on social accounts (Type of social media, Id of the social media account, URL)

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Persons: [get_person\(\)](#), [search_person_by_parameters\(\)](#), [search_person\(\)](#)

Examples

```
## Not run:
get_person_social(types = c("Twitter", "Instagram"))

## End(Not run)
```

get_subsidy

Get subsidy

Description

Get subsidy

Usage

```
get_subsidy(id, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

id	Subsidy ID
token	Authorization token

Value

list with details of the subsidy

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Subsidies: [search_subsidies\(\)](#)

Examples

```
## Not run:
get_subsidy("deminimis-1000229862")

## End(Not run)
```

`get_websites`*Get websites*

Description

Get websites

Usage

```
get_websites(token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

token Authorization token

Value

data.frame with data on websites run by the state with the following columns: - hostid: ID of website - host: host of website - url: URL of website - opendataUrl: URL to open data on website's availability - pageUrl: URL to website's profile at Hlidac statu - urad: the institution running the website - publicname: name of the website - popis: description of the website - hash

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Examples

```
## Not run:  
get_websites()  
  
## End(Not run)
```

`get_website_detail`*Get website detail*

Description

Get website detail

Usage

```
get_website_detail(id, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

id	Website id
token	Authorization token

Value

list with 2 elements: - availability: data on availability of the website including response time and HTTP status code every minute - ssl: assessment of the website's SSL server configuration

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Examples

```
## Not run:  
get_website_detail(id = "10107")  
  
## End(Not run)
```

handle_error_response *Handle response other than 200*

Description

Handle response other than 200

Usage

```
handle_error_response(response)
```

Arguments

response	Response from the server
----------	--------------------------

Value

No return value, called for side effects

search_contracts	<i>Search contract</i>
------------------	------------------------

Description

Search contract

Usage

```
search_contracts(query, token = Sys.getenv("HLIDAC_TOKEN"), page = 1, sort = 0)
```

Arguments

query	Full-text search query
token	Access token
page	Page of results (Please note that if the page parameter exceeds 200, the function returns error)
sort	Ordering of the results, the available options are the following: - 0: sort by relevance - 1: recently added contracts first - 2: recently added contracts last - 3: the cheapest contracts first - 4: the most expensive contracts first - 5: recently concluded contracts first - 6: recently concluded contracts last - 7: the most defective contracts first - 8: sort by purchaser - 9: sort by supplier

Value

list containing 3 elements: - Total: Total number of datasets available (integer) - Page: Page of the result (integer) - Results: Data.frame with data concerning the datasets

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

<https://www.hlidacstatu.cz/napoveda>

<https://smlouvy.gov.cz/>

Examples

```
## Not run:  
search_contracts(query = "golf")  
  
## End(Not run)
```

search_person	<i>Search person</i>
---------------	----------------------

Description

Search person

Usage

```
search_person(query, page = NULL, token = Sys.getenv("HLIDAC_TOKEN"))
```

Arguments

query	Search query for a person, required parameter
page	Page number
token	Access token

Value

data.frame with found persons with the following columns: - jmeno: given name - prijmeni: sur-name - narozeni: date of birth - nameId: person's ID - profile: URL of profile at hlidacstatu.cz - titulPred: academic titles written before person's name - titulPo: academic titles written after person's name

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Persons: [get_person_social\(\)](#), [get_person\(\)](#), [search_person_by_parameters\(\)](#)

Examples

```
## Not run:  
search_person(query = "Kalousek")  
  
## End(Not run)
```

`search_person_by_parameters`*Search person by name and date of birth*

Description

Search person by name and date of birth

Usage

```
search_person_by_parameters(  
    first_name = NULL,  
    last_name = NULL,  
    birth_date = NULL,  
    ignore_diacritics = TRUE,  
    token = Sys.getenv("HLIDAC_TOKEN")  
)
```

Arguments

<code>first_name</code>	Person's first name
<code>last_name</code>	Person's last name
<code>birth_date</code>	Person's date of birth (in YYYY-MM-DD format)
<code>ignore_diacritics</code>	Parameter indicating whether diacritics in names should be ignored (boolean)
<code>token</code>	Access token

Value

data.frame with found persons with the following columns: - jmeno: given name - prijmeni: sur-name - narozeni: date of birth - nameId: person's ID - profile: URL of profile at hlidacstatu.cz - titulPred: academic titles written before person's name - titulPo: academic titles written after person's name

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Persons: [get_person_social\(\)](#), [get_person\(\)](#), [search_person\(\)](#)

Examples

```
## Not run:  
search_person_by_parameters(first_name = "Miroslav", last_name = "Kalousek",  
    birth_date = "1960-12-17")  
  
## End(Not run)
```

search_subsidies	<i>Search subsidies</i>
------------------	-------------------------

Description

Search subsidies

Usage

```
search_subsidies(  
    query,  
    page = 1,  
    sort = NULL,  
    token = Sys.getenv("HLIDAC_TOKEN")  
)
```

Arguments

query	Search query
page	Page number (Please note that if the page parameter exceeds 200, the function returns error)
sort	Sorting of results, the available options are the following: - 0: order by relevance - 1: order by the date of signature, the most recent first - 2: order by the date of signature, the most recent last - 3: order by the size of the subsidy, the largest first - 4: order by the size of the subsidy, the largest last - 5: order by ICO in a descending order - 6: order by ICO in an ascending order
token	Authorization token

Value

list containing 3 elements: - total: Total number of datasets available (integer) - page: Page of the result (integer), equal to the 'page' argument of the function - results: Data.frame with data, columns vary depending on the dataset

See Also

<https://www.hlidacstatu.cz/swagger/index.html>

<https://www.hlidacstatu.cz/api/v1/doc>

Other Subsidies: [get_subsidy\(\)](#)

Examples

```
## Not run:  
search_subsidies("golf")  
  
## End(Not run)
```

Index

- * **Companies**
 - get_company, 4
 - * **Contracts**
 - search_contracts, 14
 - * **Datasets**
 - get_dataset_data, 7
 - get_dataset_metadata, 8
 - get_dataset_record_detail, 8
 - get_datasets, 6
 - * **Persons**
 - get_person, 9
 - get_person_social, 10
 - search_person, 15
 - search_person_by_parameters, 16
 - * **Subsidies**
 - get_subsidy, 11
 - search_subsidies, 17
- check_connection, 2
- check_token, 2
- check_types, 3
- create_type, 3
-
- get_company, 4
- get_contract, 4
- get_contract_text, 5
- get_dataset_data, 6, 7, 8, 9
- get_dataset_metadata, 6, 7, 8, 9
- get_dataset_record_detail, 6–8, 8
- get_datasets, 6, 7–9
- get_person, 9, 11, 15, 16
- get_person_social, 10, 10, 15, 16
- get_subsidy, 11, 17
- get_website_detail, 12
- get_websites, 12
-
- handle_error_response, 13
-
- search_contracts, 14
- search_person, 10, 11, 15, 16
- search_person_by_parameters, 10, 11, 15, 16
- search_subsidies, 11, 17