

Package ‘plug’

July 23, 2025

Type Package

Title Secure and Intuitive Access to 'Plug' Interface

Version 0.1.0

Date 2025-01-09

Description Provides a secure and user-friendly interface to interact with the 'Plug' <<https://plugbytpf.com.br>> 'API'. It enables developers to store and manage tokens securely using the 'keyring' package, retrieve data from 'API' endpoints with the 'httr2' package, and handle large datasets with chunked data fetching. Designed for simplicity and security, the package facilitates seamless integration with 'Plug' ecosystem.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Imports tibble, httr2, glue, keyring

URL <<https://github.com/StrategicProjects/plug>>

Config/Needs/website tidyverse/tidytemplate

NeedsCompilation no

Author Andre Leite [aut, cre],
Felipe Ferreira [aut],
Hugo Vaconcelos [aut],
Diogo Bezerra [aut],
Roger Azevedo [aut]

Maintainer Andre Leite <leite@castlab.org>

Repository CRAN

Date/Publication 2025-01-13 17:50:07 UTC

Contents

plug_download_base	2
plug_execute_query	3
plug_get_valid_token	3

plug_list_credentials	4
plug_list_tokens	5
plug_store_credentials	5
Index	7

plug_download_base	<i>Download all data from a specific base</i>
--------------------	---

Description

This function downloads all data from a specified base using the query `SELECT * FROM base_name`.

Usage

```
plug_download_base(  
  base_name,  
  endpoint = "https://plug.der.pe.gov.br/MadrixApi/executeQuery",  
  verbosity = 0  
)
```

Arguments

- base_name The name of the base from which to download all data.
- endpoint The endpoint URL for executing queries.
- verbosity The verbosity level of the API request (0 = none, 1 = minimal, 2 = detailed).

Value

A tibble containing all data from the specified base.

Examples

```
## Not run:  
data <- plug_download_base(  
  base_name = "Contratos_VIEW"  
)  
  
## End(Not run)
```

plug_execute_query	<i>Execute a custom SQL query on the Plug database</i>
--------------------	--

Description

This function executes a user-defined SQL query on the Plug database, with safe query construction using `glue_sql`.

Usage

```
plug_execute_query(  
  sql_template,  
  endpoint = "https://plug.der.pe.gov.br/MadrixApi/executeQuery",  
  verbosity = 0,  
  ...  
)
```

Arguments

<code>sql_template</code>	A SQL query template with placeholders for variables.
<code>endpoint</code>	The endpoint URL for executing queries.
<code>verbosity</code>	The verbosity level of the API request (0 = none, 1 = minimal, 2 = detailed).
<code>...</code>	Named arguments to replace placeholders in the SQL template.

Value

A tibble containing the query results.

Examples

```
## Not run:  
data <- plug_execute_query(sql_template = "SELECT TOP 1 * FROM Contratos_VIEW")  
  
## End(Not run)
```

plug_get_valid_token	<i>Get a valid token for Plug API</i>
----------------------	---------------------------------------

Description

This function checks if a valid global token exists for the Plug API. If no valid token is found, it generates a new token using the stored global credentials by sending a properly formatted request. If it fails to retrieve the token (for example, due to missing credentials or network issues), it will not throw an error but will display a message in English ("No valid credentials found.") and return `NULL`.

Usage

```
plug_get_valid_token(  
  validity_time = 3600,  
  endpoint = "https://plug.der.pe.gov.br/MadrixApi/authenticate/"  
)
```

Arguments

`validity_time` The validity period of the token in seconds. Default is 3600 (1 hour).
`endpoint` The endpoint URL for generating the token.

Value

The valid token as a string, or NULL if no valid credentials were found or an error occurred.

Examples

```
## Not run:  
token <- plug_get_valid_token(validity_time = 3600)  
  
## End(Not run)
```

`plug_list_credentials` *List registered credentials for Plug API*

Description

This function lists all globally stored credentials (username and password) for the Plug API. If none are found or an error occurs, it displays a message in English ("No credentials found for Plug API.") and returns an empty list.

Usage

```
plug_list_credentials()
```

Value

A named list with username and password fields if credentials are found, or an empty list if no credentials are stored.

Examples

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

plug_list_tokens	<i>List registered tokens for Plug API</i>
------------------	--

Description

This function lists the stored API token and its expiration time for the Plug API. If none are found or an error occurs, it displays a message in English ("No token found for Plug API.") and returns an empty list.

Usage

```
plug_list_tokens()
```

Value

A named list with token and expiration fields if a token is found, or an empty list if no token is stored.

Examples

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

plug_store_credentials	<i>Store user credentials securely for Plug API</i>
------------------------	---

Description

This function securely stores the global username and password required to authenticate with the Plug application.

Usage

```
plug_store_credentials(username, password)
```

Arguments

username	The username for the Plug.
password	The password for the Plug.

Value

No return value. The credentials are securely stored.

Examples

```
plug_store_credentials("myusername", "mypassword")
```

Index

plug_download_base, [2](#)
plug_execute_query, [3](#)
plug_get_valid_token, [3](#)
plug_list_credentials, [4](#)
plug_list_tokens, [5](#)
plug_store_credentials, [5](#)