Package ‘genesysr’

February 23, 2024

Version 2.1.1

Title Genesys PGR Client

Description Access data on plant genetic resources from genebanks around the world published on Genesys (<https://www.genesys-pgr.org>).
Your use of data is subject to terms and conditions available at <https://www.genesys-pgr.org/content/legal/terms>.

License Apache License 2.0

URL https://gitlab.croptrust.org/genesys-pgr/genesysr

BugReports https://gitlab.croptrust.org/genesys-pgr/genesysr/-/issues

Language en-US

Encoding UTF-8

Maintainer Matija Obreza <matija.obreza@croptrust.org>

Depends R (>= 3.1.0)

Imports httr2 (>= 1.0.0), jsonlite, dplyr, readr, magrittr,
tidyselect,

RoxygenNote 7.2.3

VignetteBuilder knitr

Suggests knitr, rmarkdown

NeedsCompilation no

Author Global Crop Diversity Trust [cph],
Matija Obreza [aut, cre],
Nora Castaneda [ctb]

Repository CRAN

Date/Publication 2024-02-23 09:10:06 UTC

R topics documented:

`api1_url` *Get partial API v1 URL for the provided path*

Description

Get partial API v1 URL for the provided path

Usage

```
api1_url(path)
```

Arguments

`path`	relative path of the API v1 endpoint (e.g. `/me`)

Value

Returns "/api/v1" + path

Examples

```
api1_url("/me")
```

`api2_url` *Get partial API v2 URL for the provided path*

Description

Get partial API v2 URL for the provided path

Usage

```
api2_url(path)
```

Arguments

`path`	relative path of the API v2 endpoint (e.g. `/me`)

Value

Returns "/api/v2" + path

Examples

```
api2_url("/me")
```

`authorization` *Provide OAuth2 token to use for authorization with Genesys*

Description

Provide OAuth2 token to use for authorization with Genesys

Usage

```
authorization(authorization)
```

Arguments

`authorization`	OAuth2 Authorization header obtained from somewhere else (e.g. an ENV variable)

See Also

`user_login`, `client_login`

check_country — *Run Land-or-Sea check on MCPD data. Uploads only rows where ORIGCTY, DECLATITUDE and DECLONGITUDE are provided.*

Description

Run Land-or-Sea check on MCPD data. Uploads only rows where ORIGCTY, DECLATITUDE and DECLONGITUDE are provided.

Usage

```
check_country(mcpd)
```

Arguments

mcpd	Accession passport data in MCPD format

Value

Results from validator

Examples

```
## Not run:
  geoCheck <- genesysr::check_country(mcpd)

## End(Not run)
```

check_landorsea — *Run Land-or-Sea check on MCPD data using https://validator.genesys-pgr.org. Uploads only rows where DECLATITUDE and DECLONGITUDE are provided. In practice it is better to use ‘check_country‘ if ORIGCTY data exists.*

Description

Run Land-or-Sea check on MCPD data using https://validator.genesys-pgr.org. Uploads only rows where DECLATITUDE and DECLONGITUDE are provided. In practice it is better to use ‘check_country‘ if ORIGCTY data exists.

Usage

```
check_landorsea(mcpd)
```

Arguments

`mcpd` Accession passport data in MCPD format

Value

Results from validator

Examples

```
## Not run:
  waterCheck <- genesysr::check_landorsea(mcpd)

## End(Not run)
```

`check_taxonomy` *Check MCPD taxonomic data (GENUS, SPECIES, SPAUTHOR, SUBTAXA, SUBTAUTHOR) using https://validator.genesys-pgr.org.*

Description

Duplicate input rows are removed using dplyr::distinct() and results are returned for unique rows.

Usage

```
check_taxonomy(mcpd, toCurrentTaxa = FALSE)
```

Arguments

`mcpd` Accession passport data in MCPD format

`toCurrentTaxa` Should obsoleted names be reported?

Value

Results from validator

Examples

```
## Not run:
  taxaCheck <- genesysr::check_taxonomy(mcpd)

## End(Not run)
```

client_login *Login to Genesys as a service client (system-to-system)*

Description

The client must be enabled for Client Credential grant on Genesys.

Usage

```
client_login()
```

See Also

setup

download_mcpd *Download passport data for one genebank in Excel format and save it to disk*

Description

Download passport data for one genebank in Excel format and save it to disk

Usage

```
download_mcpd(instituteCode, file = NULL)
```

Arguments

instituteCode	FAO WIEWS institute code
file	Target file name. Defaults to Genesys-provided file name in the current working directory.

Value

The downloaded MCPD file name

Examples

```
## Not run:
  # Download MCPD passport data for NGA039
  excelFile <- download_mcpd("NGA039")

## End(Not run)
```

download_pdci *Download PDCI data for one genebank in Excel format and save it to disk.*

Description

Download PDCI data for one genebank in Excel format and save it to disk.

Usage

```
download_pdci(instituteCode, file = NULL)
```

Arguments

instituteCode	FAO WIEWS institute code
file	Target file name. Defaults to Genesys-provided file name in the current working directory.

Value

The downloaded PDCI file name

Examples

```
## Not run:
  # Download PDCI  data for NGA039
  excelData <- download_pdci("NGA039")

## End(Not run)
```

fetch_accessions *Fetch accession passport data and return the paged data structure for further processing. get_accessions might be more useful as it returns a data table.*

Description

Fetch accession passport data and return the paged data structure for further processing. get_accessions might be more useful as it returns a data table.

Usage

```
fetch_accessions(
  filters = list(),
  page = NULL,
  size = 1000,
  selector = NULL,
  at.least = NULL
)
```

Arguments

`filters`	an R `structure` with Genesys filters
`page`	the page index (0-based)
`size`	number of records to load per page (page size)
`selector`	NULL or a function to "select" variables of interest
`at.least`	stop fetching when at.least records are received from Genesys

Value

Paged data structure

See Also

`get_accessions`

Examples

```
## Not run:
  # Retrieve all accession data by country of origin
  accessions <- genesysr::fetch_accessions(mcpd_filter(ORIGCTY = c("DEU", "SVN")))

  # Fetch Musa
  musa <- genesysr::fetch_accessions(list(taxonomy.genus = c('Musa')))

  # Apply selector function
  accessions <- genesysr::fetch_accessions(
    mcpd_filter(ORIGCTY = c("DEU", "SVN")),
    selector = function(x) {
      list(id = x$id, acceNumb = x$acceNumb, instCode = x$institute$code)
    }
  )

## End(Not run)
```

`filter_DOI` *Add filter on accession DOI*

Description

Add filter on accession DOI

Usage

```
filter_DOI(filter = list(), DOI)
```

Arguments

`filter`	Existing filters (or blank list if not provided)
`DOI`	Accession DOI

`filter_GENUS` *Add filter by genus*

Description

Add filter by genus

Usage

```
filter_GENUS(filter = list(), GENUS)
```

Arguments

`filter`	Existing filters (or blank list if not provided)
`GENUS`	List of genera

`filter_INSTCODE` *Add filter by genus*

Description

Add filter by genus

Usage

```
filter_INSTCODE(filter = list(), INSTCODE)
```

Arguments

`filter`	Existing filters (or blank list if not provided)
`INSTCODE`	List of WIEWS institute codes

filter_ORIGCTY *Add filter on Country of origin of material*

Description

Add filter on Country of origin of material

Usage

```
filter_ORIGCTY(filter = list(), ORIGCTY)
```

Arguments

filter	Existing filters (or blank list if not provided)
ORIGCTY	Country of origin

filter_SAMPSTAT *Add filter on Biological status of sample*

Description

Add filter on Biological status of sample

Usage

```
filter_SAMPSTAT(filter = list(), SAMPSTAT)
```

Arguments

filter	Existing filters (or blank list if not provided)
SAMPSTAT	Biological status of sample

filter_SPECIES *Add filter on specific epithet*

Description

Add filter on specific epithet

Usage

```
filter_SPECIES(filter = list(), SPECIES)
```

Arguments

filter	Existing filters (or blank list if not provided)
SPECIES	List of specific epithets

get_accessions *Get accession passport data as a data table.*

Description

Get accession passport data as a data table.

Usage

```
get_accessions(
  filters = list(),
  page = 0,
  size = 1000,
  fields = NULL,
  exclude = NULL,
  selector = NULL,
  at.least = NULL
)
```

Arguments

filters	an R structure with Genesys filters
page	the page index (0-based)
size	number of records to load per page (page size)
fields	list of fields to fetch from Genesys
exclude	list of field prefixes to exclude from the Genesys response
selector	NULL or a function to "select" variables of interest
at.least	stop fetching when at.least records are received from Genesys

Value

Data table

See Also

mcpd_filter

Examples

```
## Not run:
  # Retrieve all accession data by country of origin (Slovenia, Ivory Coast)
 accessions <- genesysr::get_accessions(list(countryOfOrigin = list(code3 = c('SVN', 'CIV'))))

  # Fetch Musa, but only geographic data and accessionNumber
  musa <- genesysr::get_accessions(list(taxonomy = list(genus = c('Musa'))),
    fields = c("accessionNumber", "geo"))
  # Apply selector function
  accessions <- genesysr::get_accessions(mcpd_filter(ORIGCTY = c('DEU', 'SVN')),
    selector = function(x) {
      list(id = x$id, acceNumb = x$accessionNumber, instCode = x$instituteCode)
    }, at.least = 100)

## End(Not run)
```

list_crops *Fetch Genesys crops. Note that the list of Genesys crops does not fully correspond with various CROPNAME in MCPD provided by genebanks.*

Description

Fetch Genesys crops. Note that the list of Genesys crops does not fully correspond with various CROPNAME in MCPD provided by genebanks.

Usage

```
list_crops()
```

Value

Genesys crops

Examples

```
## Not run:
  # Retrieve all Genesys crops
  crops <- genesysr::list_crops()

## End(Not run)
```

list_institutes *List FAO WIEWS institutes.*

Description

Institute filters: - code: list of WIEWS institute codes - accessions: boolean, TRUE list only institutes with accessions in Genesys, FALSE without accessions - country$code3: list of ISO3166 country codes

Usage

```
list_institutes(filters = list(), at.least = NULL)
```

Arguments

`filters`	an R `structure` with Institute filters
`at.least`	stop fetching when at.least records are received from Genesys

Value

List of institutes

See Also

`mcpd_filter`

Examples

```
## Not run:
  # Retrieve taxa of selected accessions
  filters <- c();
  filters$accessions = TRUE; # Has accessions in Genesys
  institutes <- genesysr::list_institutes(filters)

## End(Not run)
```

`list_species` *Fetch taxonomic data of selected accessions.*

Description

Fetch taxonomic data of selected accessions.

Usage

```
list_species(filters = list())
```

Arguments

`filters`	an R `structure` with Genesys filters

Value

Taxonomic records of selected accessions

See Also

`mcpd_filter`

Examples

```
## Not run:
  # Retrieve taxa of selected accessions
  taxa <- genesysr::list_species(mcpd_filter(INSTCODE = c("LBN002", "MEX002")))

## End(Not run)
```

mcpd_filter *Make or adjust filter using MCPD terminology*

Description

See FAO/Bioversity Multi-Crop Passport Descriptors.

Usage

```
mcpd_filter(
  filter = list(),
  INSTCODE = NULL,
  DOI = NULL,
  ORIGCTY = NULL,
  SAMPSTAT = NULL,
  GENUS = NULL,
  SPECIES = NULL
)
```

Arguments

`filter`	Existing filters (or blank list if not provided)
`INSTCODE`	WIEWS Institute Code of the holding institute
`DOI`	Accession DOI
`ORIGCTY`	Country of origin
`SAMPSTAT`	Biological status of sample
`GENUS`	List of genera
`SPECIES`	List of specific epithets (within specified genera)

Examples

```
# Filter accessions from Mexico and Slovenia
mcpd_filter(ORIGCTY = c("MEX", "SVN"))
```

me *Who am i? Loads and prints the user profile from Genesys as JSON. You need to be logged in.*

Description

Who am i? Loads and prints the user profile from Genesys as JSON. You need to be logged in.

Usage

```
me()
```

Examples

```
## Not run:
  # Login
  setup_production()
  user_login()
  me()

## End(Not run)
```

print_setup *Print Genesys client configuration*

Description

Print Genesys client configuration

Usage

```
print_setup()
```

setup *Configure the Genesys environment*

Description

Configure the Genesys environment

Usage

```
setup(server = NULL, client_id = NULL, client_secret = NULL)
```

Arguments

`server`	Server base URL (e.g. "https://api.genesys-pgr.org" or "https://api.sandbox.genesys-pgr.org")
`client_id`	OAuth client ID
`client_secret`	OAuth client secret

See Also

See utility methods `setup_production`, `setup_sandbox`

Examples

```
# Link with sandbox
setup_sandbox()
```

setup_production *Setup for Genesys Production*

Description

Use the Genesys R Client with <https://api.genesys-pgr.org> requiring `user_login`

Usage

```
setup_production()
```

`setup_sandbox` *Setup for Genesys Sandbox*

Description

Use the Genesys R Client with <https://api.sandbox.genesys-pgr.org> requiring `user_login`

Usage

```
setup_sandbox()
```

`user_login` *Login to Genesys as a user*

Description

The authorization URL will open in a browser, ask the user to grant permissions to R. After successful authentication the browser will display a message:

Usage

```
user_login(redirect_uri = "http://127.0.0.1:48913")
```

Arguments

`redirect_uri`	a custom redirect_uri to submit as part of the authentication request. This is most useful if the default port is blocked and you wish to specify another port: redirect_uri = "http://127.0.0.1:44211" Note that "http://127.0.0.1" is requred and that "http://localhost" will not work.

Details

“‘ Authentication complete. Please close this page and return to R. “‘

Close the browser and return to R.

See Also

`setup`

Index