Package ‘genesysr’

February 15, 2023

Version 1.1.0
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 httr, jsonlite, dplyr, readr
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 2023-02-15 13:10:02 UTC

R topics documented:

api1_url ................................................................. 2
api2_url ................................................................. 3
authorization ............................................................. 3
api1_url

Get full Genesys API v1 URL for a specific path

Description

Get full Genesys API v1 URL for a specific path

Usage

api1_url(path)

Arguments

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

Value

Absolute URL to an API call

Examples

api1_url("/me")
api2_url

| api2_url | Get full Genesys API v1 URL for a specific path |

**Description**

Get full Genesys API v1 URL for a specific path

**Usage**

api2_url(path)

**Arguments**

path

**Value**

Absolute URL to an API call

**Examples**

api2_url("/me")

-------------

authorization

| 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

**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.

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)
check_taxonomy

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

Description
Download all 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
HTTP response data

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

## End(Not run)
**download_pdcie**

*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**

```r
download_pdcie(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**

HTTP response data

**Examples**

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

---

**fetch_accessions**

*Fetch accession passport data*

**Description**

Fetch accession passport data

**Usage**

```r
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

Examples

```r
## Not run:

# Retrieve all accession data by country of origin
accessions <- fetch_accessions(mcpd_filter(ORIGCTY = c("DEU", "SVN")))

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

# Apply selector function
accessions <- 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

Fetch accession passport data

Description
Fetch accession passport data

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

  Paged data structure

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 <- 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**

```r
list_crops()
```

**Value**

Genesys crops

**Examples**

```r
## 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
list_institutes

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
mcpd_filter

Make or adjust filter using MCPD terminology

Description

See FAO/Biodiversity Multi-Crop Passport Descriptors.

Usage

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

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter</td>
<td>Existing filters (or blank list if not provided)</td>
</tr>
<tr>
<td>INSTCODE</td>
<td>WIEWS Institute Code of the holding institute</td>
</tr>
<tr>
<td>DOI</td>
<td>Accession DOI</td>
</tr>
<tr>
<td>ORIGCTY</td>
<td>Country of origin</td>
</tr>
<tr>
<td>SAMPSTAT</td>
<td>Biological status of sample</td>
</tr>
<tr>
<td>GENUS</td>
<td>List of genera</td>
</tr>
<tr>
<td>SPECIES</td>
<td>List of specific epithets (within specified genera)</td>
</tr>
</tbody>
</table>

Examples

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

*Who am i?*

**Description**

Who am i?

**Usage**

`me()`

---

**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 \texttt{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 \texttt{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 and the verification code must be copy-pasted after you grant access to the client.

Usage

user_login()

See Also

setup
Index

api1_url, 2
api2_url, 3
authorization, 3
check_country, 4
check_landorsea, 4
check_taxonomy, 5
client_login, 3, 6

download_mcpd, 6
download_pdc, 7

fetch_accessions, 7
filter_DOI, 8
filter_GENUS, 9
filter_INSTCODE, 9
filter_ORIGCTY, 9
filter_SAMPSTAT, 10
filter_SPECIES, 10

gt_accessions, 11

list_crops, 12
list_institutes, 12
list_species, 13

mcpd_filter, 11, 13, 14
me, 15

print_setup, 15

setup, 6, 15, 16
setup_production, 15, 16
setup_sandbox, 15, 16

user_login, 3, 16, 16