

Package ‘seekr’

July 14, 2021

Type Package

Title seekr: R interface for SEEK API (connection to FAIRDOMHub)

Version 0.1.0.9000

Description The package provides several functions for upload/download of data organized in pISA-tree to SEEK based cloud platform (specifically FAIRDOMHub).

License GPL-3

URL <https://github.com/NIB-SI/seekr>

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports knitr,

rio,
tools,
RCurl,
httr,
jsonlite,
pisar

VignetteBuilder knitr

Suggests knitr,
rmarkdown

R topics documented:

| | |
|--------------------------|-----------|
| contentType | 2 |
| print.seek_api | 2 |
| skCreate | 3 |
| skData | 5 |
| skFindId | 6 |
| skFindTitle | 7 |
| skGet | 8 |
| skIni | 9 |
| skLog | 10 |
| skParse | 11 |
| skSkeleton | 12 |
| skUpload | 13 |
| Index | 15 |

| | |
|-------------|--------------------------------------|
| contentType | <i>Determine MIME type for file.</i> |
|-------------|--------------------------------------|

Description

Determine MIME type for file.

Usage

```
contentType(x)
```

Arguments

| | |
|------|-------|
| File | name. |
|------|-------|

Value

MIME type string.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

Examples

```
contentType("bla.txt")
contentType("bla.pdf")
contentType("bla.tar.gz")
contentType("bla.bla")
```

| | |
|----------------|---|
| print.seek_api | <i>Print method for seek_api object</i> |
|----------------|---|

Description

Print method for seek_api object

Usage

```
## S3 method for class 'seek_api'
print(x, content = FALSE)
```

Arguments

| | |
|---------|--|
| x | Object of class seek_api. |
| content | If FALSE (default), content is no printed. |

Value

An object (list) of class seek_api.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[skParse](#)

Examples

```
## Not run:
skIni()
options()$seekr$myid
r <- skGet("people",options()$seekr$myid)
# Print contents
print( r, TRUE)
# Short version, default
r
r$response$status_code
status_code(r$response)

## End(Not run)
```

skCreate

Create pISA layer or *sk* component.

Description

Create pISA layer or *sk* component.

Usage

```
skCreate(type = "assays", meta = list(), class = "EXP", file = "NA.TXT")
```

Arguments

| | |
|-------|--|
| type | Component name (e.g. 'people', 'projets', ...). |
| meta | Data frame with pISA metadata or a list with minimal information (Title, Description, *ToDo: add fields*). |
| class | Assay class key string. Possible values are 'EXP' and 'MODEL'. |
| file | File name with path, relative to layer. |

Value

FAIRDOMhub created component.

Note

Upon success (status code 200) details of newly created component can be used. Check status code.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[skGet](#)

Examples

```
## Not run:
if(FALSE)
{
  skIni(prid = 26, test=TRUE)
  options("seekr")
  sp <- skCreate( type = "projects"
    , meta= list(
      Title=paste("Test project", Sys.time())
      , Description="Testing of upload")
    )
  str(sp)
  str(skData(sp))
  # Add member manually
  skIni(prid = 26, pid=104, test=TRUE)
  options("seekr")
  si <- skCreate( type = "investigations"
    , meta= list(
      Title=paste("Test investigation", Sys.time())
      , Description="Testing of upload")
    )
  si
  skData(si)$id

  iid=skData(si)$id
  iid <- 115
  skIni(prid = 26, pid=104, iid=iid, test=TRUE)
  options("seekr")
  ss <- skCreate( type = "studies"
    , meta= list(
      Title=paste("Test study", Sys.time())
      , Description="Testing of upload")
    )
  ss
  skData(ss)$id

  skIni(prid = 26
    , pid=104
    , iid=skData(si)$id
    , sid=skData(ss)$id
    , test=TRUE)
  options("seekr")
  sa <- skCreate( type = "assays"
    , meta= list(
      Title=paste("Test assay", Sys.time())
      , Description="Testing of upload")
      , class="EXP"
    )
}
```

```

#str(sa)
sa
}

# Type: data_file
astring <- "_p_Demo/_I_Test/_S_Show/_A_Work-R/"
oldwd <- setwd(system.file("extdata",astring,package="seekr"))
oldwd
.aname <- getLayer("A")
.aroot <- getRoot("A")
.ameta <- readMeta()
file <- "input/README.MD"
skIni(prid = 26, pid=104, iid=115 , sid=117 , aid=401, test=TRUE)
type <- "data_files"
type <- "documents"
sdatt <- skCreate( type = type
, meta= list(
  Title=paste("Test assay", Sys.time())
, Description="Testing of upload")
, file=file
)
#str(sdatt)
sdatt
skData(sdatt)$id
if(interactive()) setwd(oldwd)
getwd()
res <- sdatt$content
item_link <- file.path(res$meta$base_url,res$links$self)
if(interactive()) shell.exec(item_link)

## End(Not run)

```

skData

*Get content from an *sk* object.*

Description

Get content from an *sk* object.

Usage

```
skData(r, node, ...)
```

Arguments

| | |
|------|---|
| r | Object retrieved by skGet. |
| type | Name of the required element. If missing, a list with all relevant objects is returned. |

Value

File name (string).

Note

Parameter ... is ignored at this time.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[get](#)

Examples

```
## Not run:
skIni()
options()$seekr$myid
r <- skGet("people",options()$seekr$myid)
d <- skData(r,"attributes")
names(d)
d$last_name
skData(r)$tools
# Get list of people
r <- skGet("people")
d <- skData(r)
length(d)
names(d)
names(d[[1]])
titles <- sapply(d,function(x) x$attributes$title)
head(titles)
# Get FAIRDOMhub user id
myname <- titles[1]
myname
d[[pmatch(myname,titles)]]
id <- d[[pmatch(myname,titles)]]$id
id

## End(Not run)
```

skFindId

*Get details of component with id from an *sk* object.*

Description

Get details of component with id from an *sk* object.

Usage

```
skFindId(type, title)
```

Arguments

| | |
|-------|---|
| type | Components name (e.g. 'people', 'projects', ...). |
| title | Character string with the identifier of the component (title part). |

Value

FAIRDOMhub component identifier: id, type and title. If argument title is missing, a data frame with identifiers for all items is returned.

Note

If item is not found, value 0 is returned as id.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[skFindTitle](#)

Examples

```
## Not run:
skIni()
id <- skFindId("people", "Guest")
id
skFindTitle("people", id)
# does not exist
skFindId("people", "No User")
# List of projects
projects <- skFindId("projects")
head(projects)

## End(Not run)
```

skFindTitle

*Get details of component with id from an *sk* object.*

Description

Get details of component with id from an *sk* object.

Usage

```
skFindTitle(type, id)
```

Arguments

| | |
|------|--|
| type | Components name (e.g. 'people', 'projects', ...). |
| id | Character string with the identifier of the component (id part). |

Value

FAIRDOMhub component identifier: id, type and title. If argument title is missing, a data frame with identifiers for all items is returned. See note.

Note

If item is not found, empty string is returned as title.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[skFindTitle](#)

Examples

```
## Not run:
skIni()
id <- skFindId("people", "Guest")
id
idNew <- skFindTitle("people", id[1])
idNew
all.equal(id, idNew)
skFindTitle("people", 0)

## End(Not run)
```

skGet

Get information from repository.

Description

Get information from repository.

Usage

```
skGet(type, id, uri = options()$seekr$baseurl, ...)
```

Arguments

| | |
|------|--------------------------------------|
| type | Type of information (e.g. "person"). |
| id | Repository id of an item. |
| uri | Repository base address (URI).. |

Value

An object (list) of class seek_api.

Note

Parameter ... is ignored at this time.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also[get](#)**Examples**

```
## Not run:
skIni()
options()$seekr$myid
r <- skGet("people",options()$seekr$myid)
names(r)
r$response$status_code
status_code(r$response)
r
# Non existent user
skGet("people",0)

## End(Not run)
```

skIni

*Initialize FAIRDOMhub information***Description**

Define FAIRDOMhub url and user data

Usage

```
skIni(
  url = "https://www.fairdomhub.org/",
  usr = NULL,
  pwd = NULL,
  myid = NULL,
  prid = NULL,
  pid = NULL,
  iid = NULL,
  sid = NULL,
  aid = NULL
)
```

Arguments

| | |
|------|--|
| url | SEEK based server address. It can also be a list with four components interpreted as url, usr, pwd and myid. |
| usr | user name. |
| pwd | password. |
| myid | numeric user id. |
| prid | programme number. |
| pid | project number. |
| iid | investigation number. |
| sid | study number. |
| aid | assay number. |

Value

A list with URL and user information. For side effect see Notes.

Note

The returned list is added to the options() list under name 'seekr'.

At the moment two locations are available: main FAIRDOMHub (<https://www.fairdomhub.org/>) and testing site (<https://testing.sysmo-db.org>).

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

Examples

```
## Not run:
skIni(url="https://www.fairdomhub.org/", usr="username"
      , pwd="secret", myid=888)
options("seekr")
#
fh <- list(url="https://www.fairdomhub.org/", usr="username"
          , pwd="secret", myid=888)
skIni(fh)
options("seekr")

## End(Not run)
```

skLog

Writes a note to a log file.

Description

Writes a note to a log file.

Usage

```
skLog(..., file = "FAIRDOM.log", append = TRUE)
```

Arguments

| | |
|--------|------------------------------|
| ... | Objects to form a line. |
| file | Log file name. |
| append | Control append/rewrite mode. |

Value

System time of invoking..

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also[skGet](#)**Examples**

```
tst <- function(){
  skLog("Test", "writing to logfile", file="")
  fht <- system.time(Sys.sleep(1))
  skLog( "Time:", round(fht["elapsed"],2))
}
tst()
rm(tst)
```

skParse

*Parse the response from SEEK API***Description**

Parse the response from SEEK API

Usage

```
skParse(resp, ...)
```

Arguments

resp Response from SEEK API.

Value

An object (list) of class seek_api.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also[skGet](#)**Examples**

```
## Not run:
skIni()
options()$seekr$myid
r <- skGet("people",options()$seekr$myid)
names(r)
r$response$status_code
status_code(r$response)
names(r$content)
r
r <- skGet("people")
```

```
length(r$content)
names(r$content)
r$content[[1]]
names(r)
r

## End(Not run)
```

skSkeleton

*Create *sk* skeleton.*

Description

Creates *sk* object with required structure.

Usage

```
skSkeleton(type = "assay", meta, file)
```

Arguments

| | |
|------|--|
| type | Component name (e.g. 'people', 'projets', ...). |
| meta | Data frame with pISA metadata or a list with minimal information (Title, Description, *ToDo: add fields*). |

Value

A list with the minimal information structure.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[skCreate](#)

Examples

```
## Not run:
require(jsonlite)
meta= list(Title = "Test layer", Description = "Some description")
type = "projects"
sp <- skSkeleton( type = type
  , meta= meta
)
str(sp)
type = "investigations"
si <- skSkeleton( type = type
  , meta= meta
)
str(si)
type = "studies"
ss <- skSkeleton( type = type
```

```

    , meta= meta
  )
  str(ss)
  type = "assays"
  sa <- skSkeleton( type = type
    , meta= meta
  )
  str(sa)

  type = "data_files"
  file =
  sdata <- skSkeleton( type = type
    , meta= meta
  )
  str(sdata)

  ## End(Not run)

```

skUpload

Upload file.

Description

Upload file to a created object of type 'documents' or 'data_files'.

Usage

```
skUpload(object, file)
```

Arguments

| | |
|--------|--|
| object | A previously created SEEK component. Must be one of 'documents' or 'data_files'. |
| file | File name with path, relative to layer. |

Value

FAIRDOMhub created component.

Note

Upon success (status code 200) details of newly created component can be used. Check status code.

Author(s)

Andrej Blejec <andrej.blejec@nib.si>

See Also

[skCreate](#)

Examples

```
file <- dir()[1]
file
fhd <- skCreate("documents",meta,file=file)
fhd
sd <- skUpload(fhd, file)
```

Index

* **file**

- skData, [5](#)
- skGet, [8](#)
- skIni, [9](#)

* **pisa**

- contentType, [2](#)
- skCreate, [3](#)
- skFindId, [6](#)
- skFindTitle, [7](#)
- skLog, [10](#)
- skSkeleton, [12](#)
- skUpload, [13](#)

contentType, [2](#)

get, [6](#), [9](#)

print.seek_api, [2](#)

skCreate, [3](#), [12](#), [13](#)
skData, [5](#)
skFindId, [6](#)
skFindTitle, [7](#), [7](#), [8](#)
skGet, [4](#), [8](#), [11](#)
skIni, [9](#)
skLog, [10](#)
skParse, [3](#), [11](#)
skSkeleton, [12](#)
skUpload, [13](#)