Package 'khisr'

April 14, 2024

Title Retrieve Data from Kenya Health Information System (KHIS)

Description Simplify data retrieval from Kenya's health system with a powerful interface for efficient data retrieval from the Kenya Health Information System (KHIS) <https://hiskenya.org>. Empower researchers, analysts, and healthcare professionals to access critical health data efficiently.

Version 1.0.2

License MIT + file LICENSE

URL https://khisr.damurka.com, https://github.com/damurka/khisr

BugReports https://github.com/damurka/khisr/issues

Depends R (>= 3.3)

Imports cli, curl, dplyr, httr2, jsonlite, magrittr, purrr, rlang, stringr, tidyr, utils, withr

Suggests knitr, rmarkdown, spelling, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

RoxygenNote 7.3.1

NeedsCompilation no

Author David Kariuki [aut, cre, cph] (<https://orcid.org/0009-0003-6159-1107>)

Maintainer David Kariuki <hello@damurka.com>

Repository CRAN

Date/Publication 2024-04-14 11:10:02 UTC

R topics documented:

analytics-dimension 2
get_analytics 3
get_metadata 4

khis-configuration 6
khis_base_url 7
khis_cred 7
khis_cred_clear 8
khis_has_cred 9
khis_username 10
metadata-filter 10
metadata-helpers 13

Index **16**

`analytics-dimension` *Analytics Data Dimensions*

Description

Constructs a dimensions expression for analytics queries based on specified property, operator, and values.

Usage

```
analytics_dimension(property, operator, values)

operator %.d% values

operator %.f% values
```

Arguments

`property`	A character string representing whether its dimension or filter. It only accepts `'dimension'`, `'filter'`.
`operator`	A character string representing the property to filter on (e.g., `'dx'`, `'pe'`, `'ou'`).
`values`	A vector of values or semi-colon separated string items.

Details

DHIS2 organizes data using multiple dimensions, each with a unique identifier and a set of items that represent specific data points within that dimension.

- Data elements (dx): Indicators, data set reporting rate metrics, data element operands, program indicators, program data elements, program attributes, validation rules.
- Periods (pe): ISO periods (e.g., 202401) and relative periods (e.g., LAST_WEEK).
- Organisation unit hierarchy (ou): Specific health facilities, districts, countries, and keywords for user location or its sub-units.
- Category option combinations (co): Category option combo identifiers.
- Attribute option combinations (ao): Category option combo identifiers.

- Categories: Category option identifiers.
- Data element group sets: Data element group identifiers.
- Organisation unit group sets: Organisation unit group identifiers.

The infix operator used for filter and dimension includes:

- `%.d%`: Infix operator for constructing dimension filters. Equivalent to calling `analytics_dimension("dimension", ...)`.
- `%.f%`: Infix operator for constructing filter filters. Equivalent to calling `analytics_dimension("filter", ...)`.

Value

A spliced list with filter in the format property=operator:value

Examples

```
# Create a dimension for data element "DE_1234"
analytics_dimension('dimension', "dx", "DE_1234")

# Equivalent to the expression above
dx %.d% "DE_1234"

# Create a filter dimension for the period of January 2024
pe %.f% "202401"

# Create filter dimension for periods "202401" and "202402":
analytics_dimension("filter", "pe", c("202401", "202402"))
```

---

get_analytics *Retrieves Analytics Table Data from KHIS*

Description

`get_analytics()` fetches data from the KHIS analytics data tables for a given period and data element(s), without performing any aggregation.

Usage

```
get_analytics(
  ...,
  return_type = c("uid", "name"),
  retry = 2,
  verbosity = 0,
  timeout = 60
)
```

Arguments

`...`	One or more `analytics_dimension()` in key-value pairs.
`return_type`	The type to be return names of uid.
`retry`	Number of times to retry the API call in case of failure (defaults to 2).
`verbosity`	Level of HTTP information to print during the call: • 0: No output • 1: Show headers • 2: Show headers and bodies • 3: Show headers, bodies, and CURL status message.
`timeout`	Maximum number of seconds to wait for the API response.

Details

- Retrieves data directly from KHIS analytics tables.
- Allows specifying KHIS session objects, retry attempts, and logging verbosity.

Value

A tibble with detailed information

Examples

```
# Clinical Breast Examination data elements
# XEX93uLsAm2 = CBE Abnormal
# cXe64Yk0QMY = CBE Normal
element_id = c('cXe64Yk0QMY', 'XEX93uLsAm2')

# Download data from February 2023 to current date
data <- get_analytics(dx %.d% element_id, pe %.d% 'LAST_MONTH')
data
```

`get_metadata` *Get Metadata from a KHIS*

Description

`get_metadata` retrieves metadata for a specified endpoint of KHIS

Usage

```
get_metadata(
  endpoint,
  ...,
  fields = c("id", "name"),
  retry = 2,
  verbosity = 0,
  timeout = 60
)
```

Arguments

endpoint	The KHIS API endpoint for the metadata of interest (e.g. dataElements, organisationUnits).
...	One or more metadata_filter() params in key-value pairs.
fields	The specific columns to be returned in the tibble.
retry	Number of times to retry the API call in case of failure (defaults to 2).
verbosity	Level of HTTP information to print during the call: • 0: No output • 1: Show headers • 2: Show headers and bodies • 3: Show headers, bodies, and CURL status message.
timeout	Maximum number of seconds to wait for the API response.

Value

A tibble with the KHIS metadata response.

Examples

```
# Get the categories metadata
get_metadata('categories')

# Get the datasets metadata with fields 'id,name,organisationUnits' and filter
# only the datasets with id 'WWh5hbCmvND'
get_metadata('dataSets',
             fields = 'id,name,organisationUnits[id,name,path]',
             id %.eq% 'WWh5hbCmvND')

# Get data elements filtered by dataElementgroups id
get_metadata('dataElements',
             dataElementGroups.id %.eq% 'IXd7DXxZqzL',
             fields = ':all')
```

khis-configuration *khisr Configuration*

Description

Some aspects of khisr behaviour can be controlled via an option.

Usage

```
with_khis_quiet(code)

local_khis_quiet(env = parent.frame())
```

Arguments

code	Code to execute quietly
env	The environment to use for scoping

Value

No return value, called for side effects

No return value, called for side effects

No return value, called for side effects

Messages

The `khis_quiet` option can be used to suppress messages form khisr. By default, khisr always messages, i.e. it is *not* quiet.

set `khis_quiet` to `TRUE` to suppress message, by one of these means, in order of decreasing scope:

- Put `options(khis_quiet = TRUE)` in the start-up file, such as `.Rprofile`, or in your R script.
- Use `local_khis_quiet()` to silence khisr in a specific scope.
- Use `with_khis_quite` to run small bit of code silently.

`local_khis_quiet` and `with_khis` follow the conventions of the withr package.

Examples

```
# message: "The credentials have been set."
khis_cred(username = 'username', password = 'password')

# suppress messages for a small amount of code
with_khis_quiet(
  khis_cred(username = 'username', password = 'password')
)
# message: "The credentials have been set."
khis_cred(username = 'username', password = 'password')
```

```
# suppress messages for a in a specific scope
local_khis_quiet()

# no message
khis_cred(username = 'username', password = 'password')

# clear credentials
khis_cred_clear()
```

`khis_base_url` *Produces the Configured KHIS API Base URI*

Description

Produces the Configured KHIS API Base URI

Usage

```
khis_base_url()
```

Value

the KHIS base URI

Examples

```
# Set the credentials
khis_cred(username = 'KHIS username', password = 'KHIS password')

# View the username expect 'KHIS username'
khis_base_url()
```

`khis_cred` *Sets the KHIS Credentials*

Description

Sets the KHIS Credentials

Usage

```
khis_cred(
  config_path = NULL,
  username = NULL,
  password = NULL,
  base_url = NULL
)
```

Arguments

config_path	An optional parameters that contains the path to configuration file with username and password.
username	The KHIS username. Can be optional if config_path is already provided
password	The KHIS password. Can be optional if config_path is already provided.
base_url	The base url used for the KHIS instance. The default is NULL which is the https://hiskenya.org/api

Details

The credentials can be provided using a configuration file (more secure) or providing username and password arguments. The config_path is considered more secure since credentials will not appear in the code.

Value

No return value

See Also

Other credential functions: khis_cred_clear(), khis_has_cred(), khis_username()

Examples

```
# Load username and password
khis_cred(username = 'khis_username', password = 'PASSWORD')
```

---

khis_cred_clear *Clear the Credentials from Memory*

Description

Clear the Credentials from Memory

Usage

```
khis_cred_clear()
```

Value

No return value

See Also

Other credential functions: khis_cred(), khis_has_cred(), khis_username()

Examples

```
khis_cred_clear()
```

`khis_has_cred` *Are There Credentials on Hand?*

Description

Are There Credentials on Hand?

Usage

```
khis_has_cred()
```

Value

a boolean value indicating if the credentials are available

See Also

Other credential functions: `khis_cred()`, `khis_cred_clear()`, `khis_username()`

Examples

```
# Set the credentials
khis_cred(username = 'KHIS username', password = 'KHIS password')

# Check if credentials available. Expect TRUE
khis_has_cred()

# Clear credentials
khis_cred_clear()

# Check if credentials available. Expect FALSE
khis_has_cred()
```

khis_username *Produces the Configured Username*

Description

Produces the Configured Username

Usage

```
khis_username()
```

Value

the username of the user credentials

See Also

Other credential functions: `khis_cred()`, `khis_cred_clear()`, `khis_has_cred()`

Examples

```
# Set the credentials
khis_cred(username = 'KHIS username', password = 'KHIS password')

# View the username expect 'KHIS username'
khis_username()

# Clear credentials
khis_cred_clear()

# View the username expect 'NULL'
khis_username()
```

metadata-filter *Metadata Filter*

Description

Formats a metadata filter to DHIS 2 comparison operators.

Usage

```
metadata_filter(property, operator, values, call = caller_env())

property %.eq% values

property %.ieq% values

property %.~eq% values

property %.ne% values

property %.Like% values

property %.~Like% values

property %.^Like% values

property %.~^Like% values

property %.Like$% values

property %.~Like$% values

property %.like% values

property %.~like% values

property %.^like% values

property %.~^like% values

property %.like$% values

property %.~like$% values

property %.gt% values

property %.ge% values

property %.lt% values

property %.le% values

property %.token% values

property %.~token% values

property %.in% values
```

```
property %.~in% values
```

Arguments

property	The property on the metadata you want to filter on
operator	The comparison operator you want to perform
values	The value to check against
call	description

Details

To filter the metadata there are several filter operations that can be applied to the returned list of metadata. The format of the filter itself is straight-forward and follows the pattern property:operator:value, where property is the property on the metadata you want to filter on, operator is the comparison operator you want to perform and value is the value to check against (not all operators require value).
To view the operator see DHIS 2 Operator

- `%.eq%` - Equality
- `%.ieq%` - Case insensitive string, match exact
- `%.~eq%` - Inequality
- `%.ne%` - Inequality
- `%.Like%` - Case sensitive string, match anywhere
- `%.~Like%` - Case sensitive string, not match anywhere
- `%.^Like%` - Case sensitive string, match start
- `%.~^Like%` - Case sensitive string, not match start
- `%.Like$%` - Case sensitive string, match end
- `%.~Like$%` - Case sensitive string, not match end
- `%.like%` - Case insensitive string, match anywhere
- `%.~like%` - Case insensitive string, not match anywhere
- `%.^like%` - Case insensitive string, match start
- `%.~^like%` - Case insensitive string, not match start
- `%.like$%` - Case insensitive string, match end
- `%.~like$%` - Case insensitive string, not match end
- `%.gt%` - Greater than
- `%.ge%` - Greater than or equal
- `%.lt%` - Less than
- `%.le%` - Less than or equal
- `%.token%` - Match on multiple tokens in search property
- `%.~token%` - Not match on multiple tokens in search property
- `%.in%` - Find objects matching 1 or more values
- `%.~in%` - Find objects not matching 1 or more values

Value

A spliced list with filter in the format property:operator:value

Examples

```
# Generate an equality filter
id %.eq% 'element_id'

# Finding multiple ids
'id' %.in% c('id1', 'id2', 'id3')

# Get all data elements which have a data set with id ID1
'dataSetElements.dataSet.id' %.eq% 'ID1'

# get data elements which are members of the ANC data element group
'dataElementGroups.id' %.eq% 'qfxEYY9xAl6'

# Get data elements which have any option set
metadata_filter('optionSet', '!null', NULL)
```

`metadata-helpers` *Metadata Helper Functions*

Description

Wrappers for `get_metadata()` that retrieves data from a specific KHIS API endpoint.

Usage

```
get_categories(..., fields = c("id", "name"))

get_category_combos(..., fields = c("id", "name"))

get_category_option_combos(..., fields = c("id", "name"))

get_category_option_group_sets(..., fields = c("id", "name"))

get_category_option_groups(..., fields = c("id", "name"))

get_category_options(..., fields = c("id", "name"))

get_data_element_group_sets(..., fields = c("id", "name"))

get_data_element_groups(..., fields = c("id", "name"))

get_data_elements(..., fields = c("id", "name"))

get_data_sets(..., fields = c("id", "name"))

get_user_groups(..., fields = c("id", "name"))
```

```
get_indicator_group_sets(..., fields = c("id", "name"))

get_indicator_groups(..., fields = c("id", "name"))

get_indicators(..., fields = c("id", "name"))

get_option_group_sets(..., fields = c("id", "name"))

get_option_groups(..., fields = c("id", "name"))

get_option_sets(..., fields = c("id", "name"))

get_options(..., fields = c("id", "name"))

get_organisation_unit_groupsets(..., fields = c("id", "name"))

get_organisation_unit_groups(..., fields = c("id", "name"))

get_organisation_units(..., fields = c("id", "name"))

get_dimensions(..., fields = c("id", "name"))

get_period_types(..., fields = c("id", "name"))
```

Arguments

`...`	Arguments passed on to `get_metadata` `retry` Number of times to retry the API call in case of failure (defaults to 2). `verbosity` Level of HTTP information to print during the call: • 0: No output • 1: Show headers • 2: Show headers and bodies • 3: Show headers, bodies, and CURL status message. `timeout` Maximum number of seconds to wait for the API response.
`fields`	The specific columns to be returned in the tibble.

Value

A tibble with the KHIS metadata response.

Examples

```
# Get all organisation units
get_organisation_units()
```

```
# Get all data elements
get_data_elements()

# Get data elements by element ids
get_data_elements(id %.in% c('VR7vdS7P0Gb', 'gQro1y7Rsbq'))

# Get datasets by name with the word 'MOH 705'
get_data_sets(name %.like% 'MOH 705')
```

Index

∗ **credential functions**
  khis_cred, 7
  khis_cred_clear, 8
  khis_has_cred, 9
  khis_username, 10
%.Like$% *(*metadata-filter*)*, 10
%.Like% *(*metadata-filter*)*, 10
%.^Like% *(*metadata-filter*)*, 10
%.^like% *(*metadata-filter*)*, 10
%.~Like$% *(*metadata-filter*)*, 10
%.~Like% *(*metadata-filter*)*, 10
%.~^Like% *(*metadata-filter*)*, 10
%.~^like% *(*metadata-filter*)*, 10
%.~eq% *(*metadata-filter*)*, 10
%.~in% *(*metadata-filter*)*, 10
%.~like$% *(*metadata-filter*)*, 10
%.~like% *(*metadata-filter*)*, 10
%.~token% *(*metadata-filter*)*, 10
%.d% *(*analytics-dimension*)*, 2
%.eq% *(*metadata-filter*)*, 10
%.f% *(*analytics-dimension*)*, 2
%.ge% *(*metadata-filter*)*, 10
%.gt% *(*metadata-filter*)*, 10
%.ieq% *(*metadata-filter*)*, 10
%.in% *(*metadata-filter*)*, 10
%.le% *(*metadata-filter*)*, 10
%.like$% *(*metadata-filter*)*, 10
%.like% *(*metadata-filter*)*, 10
%.lt% *(*metadata-filter*)*, 10
%.ne% *(*metadata-filter*)*, 10
%.token% *(*metadata-filter*)*, 10

analytics-dimension, 2
analytics_dimension *(*analytics-dimension*)*, 2
analytics_dimension(), *4*

get_analytics, 3
get_categories *(*metadata-helpers*)*, 13
get_category_combos *(*metadata-helpers*)*, 13
get_category_option_combos *(*metadata-helpers*)*, 13
get_category_option_group_sets *(*metadata-helpers*)*, 13
get_category_option_groups *(*metadata-helpers*)*, 13
get_category_options *(*metadata-helpers*)*, 13
get_data_element_group_sets *(*metadata-helpers*)*, 13
get_data_element_groups *(*metadata-helpers*)*, 13
get_data_elements *(*metadata-helpers*)*, 13
get_data_sets *(*metadata-helpers*)*, 13
get_dimensions *(*metadata-helpers*)*, 13
get_indicator_group_sets *(*metadata-helpers*)*, 13
get_indicator_groups *(*metadata-helpers*)*, 13
get_indicators *(*metadata-helpers*)*, 13
get_metadata, 4, *14*
get_metadata(), *13*
get_option_group_sets *(*metadata-helpers*)*, 13
get_option_groups *(*metadata-helpers*)*, 13
get_option_sets *(*metadata-helpers*)*, 13
get_options *(*metadata-helpers*)*, 13
get_organisation_unit_groups *(*metadata-helpers*)*, 13
get_organisation_unit_groupsets *(*metadata-helpers*)*, 13
get_organisation_units *(*metadata-helpers*)*, 13
get_period_types *(*metadata-helpers*)*, 13
get_user_groups *(*metadata-helpers*)*, 13

khis-configuration, 6
khis_base_url, 7

khis_cred, 7, *8–10*
khis_cred_clear, *8*, 8, *9, 10*
khis_has_cred, *8*, 9, *10*
khis_username, *8, 9*, 10

local_khis_quiet *(*khis-configuration*)*, 6

metadata-filter, 10
metadata-helpers, 13
metadata_filter *(*metadata-filter*)*, 10
metadata_filter(), *5*

with_khis_quiet *(*khis-configuration*)*, 6