Package 'ffmetadata'

November 19, 2018

Type Package

Title Access to Fragile Families Metadata

Version 1.0.0

Description A collection of functions that allows users to retrieve
metadata for the Fragile Families challenge via a Web API
(<http://api.metadata.fragilefamilies.princeton.edu>).
Users can select and search metadata for relevant variables by
filtering on different attribute names.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports jsonlite, httr

RoxygenNote 6.1.1

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

URL https://github.com/fragilefamilieschallenge/ffmetadata

BugReports https://github.com/fragilefamilieschallenge/ffmetadata/issues

NeedsCompilation no

Author Ryan Vinh [aut, cre],
Ian Fellows [aut],
Will Lowe [ctb]

Maintainer Ryan Vinh <rvinh@princeton.edu>

Repository CRAN

Date/Publication 2018-11-19 19:00:11 UTC

R topics documented:

---

ffmetadata   *ffmetadata*

Description

This package provides functions to access to metadata for data from the Fragile Families Project.

Search functions

To search or filter variables by string:

- search_metadata

Information functions

To get more detail on a variable:

- select_metadata

---

search_metadata   *Search Metadata*

Description

search_metadata allows users to retrieve a list of variable names based on whether or not those variables contain a given query or set of queries within a given field or set of fields

Usage

```
search_metadata(filter_list = list(), ..., operation = "eq")
```

Arguments

filter_list	an optional named list that can be passed to search_metadata in addition to explicitly named parameters
...	any valid field name
operation	optional parameter that allows one to specify the given the method of comparison for search, "equals" is the default

Details

List of valid field names:

- data_source
- data_type
- fp_PCG
- fp_father
- fp_fchild
- fp_mother
- fp_other
- fp_partner
- group_id
- group_subid
- id
- label
- leaf
- measures
- name
- old_name
- probe
- qText
- respondent
- scope
- section
- warning
- wave

Value

returns list of names of all variables that match specified parameter values

Examples

```
search_test1 <- search_metadata(wave = "Year 1")

search_test2 <- search_metadata(wave = "Year 1", respondent = "Mother")

search_test3 <- search_metadata(wave = "Year 1", name = "f%", operation = c("eq", "like"))

search_test4 <- search_metadata(name = "f1%", operation = "like")

# Note that when using operators checking for null values (or lack thereof), the operation
# parameter should not be included in the call. It should be formatted instead like so:
search_test5 <- search_metadata(qtext = "is_null")
```

---

select_metadata   *Select Metadata*

Description

select_metadata allows users to retrieve the value of a specific field or fields of a given variable, or
the entire variable metadata if no field is specified

Usage

```
select_metadata(variable_name = NULL, fields = NULL,
  returnDataFrame = TRUE)
```

Arguments

variable_name	name of variable
fields	specific field or list of fields of variable to be accessed. See details for valid field names
returnDataFrame	
	optional parameter set to TRUE by default, output is named list if set to FALSE

Details

List of valid field names:

- data_source
- data_type
- fp_PCG
- fp_father
- fp_fchild
- fp_mother
- fp_other
- fp_partner
- group_id
- group_subid
- id
- label
- leaf
- measures
- name
- old_name
- probe

- qText
- respondent
- scope
- section
- warning
- wave

Value

returns string with value of a given field if field is specified, returns all metadata for variable as data frame if field is unspecified

Examples

```
select1 <- select_metadata(variable_name = "ce3agefc")

select2 <- select_metadata(variable_name = "ce3agefc", fields = "data_type")

select3 <- select_metadata(variable_name = "ce3agefc", fields = c("data_type", "data_source"))
```

Index