Package: vsezved (via r-universe)

August 11, 2024

Title Get data on Czech schools from <https://stistko.uiv.cz/registr/> and <https://data.msmt.cz/>

Version 0.2.0

Description Get access to data on Czech schools: the register open data provided by the Ministry of Education at <https://data.msmt.cz/> and non-open web database at <http://stistko.uiv.cz/registr/> This is mostly organisatoinal data on primary and secondary schools.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports magrittr, tibble, rvest (>= 0.99.0.9000), xml2 (>= 1.3.2), purrr (>= 0.3.4), usethis (>= 2.0.0), dplyr (>= 1.0.2), tidyr (>= 1.1.2), janitor (>= 2.0.1), lifecycle (>= 0.2.0), httr (>= 1.4.2), curl (>= 4.3), prettyunits, readr, stringr, lubridate

RdMacros lifecycle

Suggests testthat (>= 3.0.0), withr (>= 2.3.0)

Config/testthat/edition 3

URL https://github.com/petrbouchal/vsezved, https://petrbouchal.xyz/vsezved/

BugReports https://github.com/petrbouchal/vsezved/issues

Repository https://petrbouchal.r-universe.dev

RemoteUrl https://github.com/petrbouchal/vsezved

RemoteRef HEAD

RemoteSha d731aed49162b070594436c95455dd538fe70977

Contents

`vz_download_codelist` *Download Stistko Ciselnik*

Description

Downloads the HTML page from the given URL.

Usage

```
vz_download_codelist(url, dest_dir = NULL)
```

Arguments

`url`	A character string representing the URL to download the HTML page from.
`dest_dir`	A character string specifying the destination directory. Defaults to tempdir().

Value

A character string containing the path to the downloaded HTML file.

Examples

```
vz_download_codelist("http://stistko.uiv.cz/katalog/ciselnik11x.asp?idc=BASO&aap=on")
```

vz_get_codelist *Get Stistko Ciselnik (df from code)*

Description

Reads and processes the HTML file of a Stistko ciselnik based on a code

Usage

```
vz_get_codelist(code, dest_dir = NULL)
```

Arguments

code	A character string representing the code of the codelist.
dest_dir	Where to save the downloaded file. Defaults to `tempdir()`.

Value

A data frame containing the processed data from the ciselnik.

Examples

```
vz_get_codelist("BASO")
```

vz_get_codelist_url *Get URL for Stistko Ciselnik*

Description

Constructs the URL for the specified Stistko codelist code.

Usage

```
vz_get_codelist_url(code)
```

Arguments

code	A character string representing the ciselnik code.

Value

A character string containing the URL for the specified ciselnik code.

Examples

```
vz_get_codelist_url("BASO")
```

`vz_get_directory` *Get school directory*

Description

[Experimental] This function performs a search on the school directory at uiv.cz and returns the resulting export - either the XLS file or the data, or both. The school directory is a version of the school register: unlike the core register, it contains contact information but lacks some other information (such as unique address identification.) Use `vz_get_register()` for the core register.

Usage

```
vz_get_directory(
  tables = c("addresses", "schools", "locations", "specialisations"),
  ...,
  return_tibbles = FALSE,
  write_files = TRUE,
  dest_dir = getwd()
)
```

Arguments

`tables`	a character vector of tables to retrieve. See ** Tables** below.
`...`	key-value pairs of search fields. Use `vz_get_search_fields()` to see a list of fields and their potential values.
`return_tibbles`	Whether to return the data (if TRUE) or only download the files (if FALSE).
`write_files`	Whether to write the XLS files locally.
`dest_dir`	Directory in which to write XLS files. Defaults to working directory.

Value

A list of a tibbles if return_tibbles = TRUE, a single tibble if only one table name is passed `tables`, otherwise a character vector of paths to the downloaded *.xls files.

if return_tibbles is TRUE, a named list of tibbles, with a tibble for each table in `tables` with the corresponding name, unless the function was called with a `tables` parameter of length one, in which case the result is a tibble; if return_tibbles is FALSE, the result is a character vector of file paths. Note that the downloaded XLS files are in fact HTML files and you are best off loading them using `vz_load_directory()` and tidying with `vz_load_directory`, though they can be opened in Excel too.

Tables

Tables can include "addresses", "schools", "locations", "specialisations". If you need more tables based on the same query (fields), pass them into a single function call in order to avoid burdening the data provider's server (the server needs to perform a search for each function call; there is no caching and no data dumps are made available).

What this does

The function

- performs a search on the school directory at uiv.cz
- by default the search is for all schools, unless ... params are set to narrow down the search
- traverses the results to the export links
- downloads the XLS files
- loads them into tibbles if return_tibbles is TRUE

This is the only way to get to the data - there are no static dumps available. At the same time, no intense web scraping takes place - only individual export files (max 4 per call) are downloaded the same way as it would be done manually.

Note

To avoid blitzing the data provider's server with many heavy requests:

1. If you need more tables based on the same search, pass it in one call, using the `tables` argument. This means that only one initial search is peformed.
2. Only ask for the tables you need.
3. If you need a subset of the data, use the `fields` (...) argument
4. If you need multiple subsets of the data, try to do that via the `fields` (...) argument too, though that may not always be possible.
5. If you are downloading a large dump and reusing it in a pipeline, keep the downloaded XLS files (or your own export) locally (setting `write_files` to TRUE), use caching and avoid calling this function repeatedly (ideally make any reruns conditional on the age of the stored export or use a pipeline management framework such as targets.

Examples

```
vz_get_directory("addresses", uzemi = "CZ010", return_tibbles = TRUE, write_files = TRUE)
```

---

vz_get_directory_responses

Get school directory responses

Description

Key low-level code for getting school directory data: crawl through layers of forms and return HTTP response containing quasi-XLS attachments with data exports.

Usage

```
vz_get_directory_responses(
  tables = c("addresses", "schools", "locations", "specialisations"),
  ...
)
```

Arguments

tables	a character vector of tables to retrieve. See ** Tables** below.
...	key-value pairs of search fields. Use vz_get_search_fields() to see a list of fields and their potential values.

Value

HTTP response parsable with response_to_quasixls or generally with httr.

Tables

Tables can include "addresses", "schools", "locations", "specialisations". If you need more tables based on the same query (fields), pass them into a single function call in order to avoid burdening the data provider's server (the server needs to perform a search for each function call; there is no caching and no data dumps are made available).

---

vz_get_register *Download and read school register*

Description

This is the high-level function for getting data from the online XML export of the school register. It downloads the file (whole country by default) and turns it into a tibble, cleaning up names and dropping some uninteresting columns (this may change as the package matures.)

Usage

```
vz_get_register(
  nuts3_kod = NULL,
  url = NULL,
  tables = c("organisations", "schools", "locations", "specialisations"),
  write_file = TRUE,
  dest_dir = getwd()
)
```

Arguments

nuts3_kod	used to point to per-region datasets; if left unset, defaults to state-wide data
url	URL; if left to NULL, will use internal default
tables	Which tables to return. Can be one or more of "organisations", "schools", "locations" or "specialisations" (specialisations not yet available via the package).
write_file	Whether to keep the downloaded XML file. Currently only writing to the working directory is supported.
dest_dir	Where to write the resulting XML

Value

a tibble or list of tibbles if multiple table names are passed to `tables`.

`vz_get_register_xml` *Download (XML) file of register*

Description

Uses CKAN to find the correct URL in the education ministry's open data catalogue and retrieve the file.

Usage

```
vz_get_register_xml(
  url = NULL,
  nuts3_kod = NULL,
  write_file = F,
  dest_dir = getwd()
)
```

Arguments

`url`	URL; if left to NULL, will use internal default
`nuts3_kod`	used to point to per-region datasets; if left unset, defaults to state-wide data
`write_file`	Whether to keep the downloaded XML file. Currently only writing to the working directory is supported.
`dest_dir`	Where to write the resulting XML

Value

Path to downloaded (XML) file.

`vz_get_search_form` *Get search form for school directory*

Description

Get search form for school directory

Usage

```
vz_get_search_form(search_page = NULL)
```

Arguments

search_page	search page session as returned by vz_get_search_page()

Value

An rvest_form object to be passed on to vz_get_directory_responses().

---

vz_get_search_page *Get search page for directory search*

Description

Get search page for directory search

Usage

```
vz_get_search_page(base_url = NULL)
```

Arguments

base_url	If left unset, defaults to internally recorded base URL

Value

an rvest_session object containing the session for the search page. Can be passed on to vz_get_search_form().

---

vz_get_xml_url *Get URL of file in MSMT data store*

Description

Currently assumes we are getting register XML data

Usage

```
vz_get_xml_url(nuts3_kod = NULL, base_url = NULL)
```

Arguments

nuts3_kod	NUTS code for region, e.g. CZ010 for Prague. Leave as NULL for whole-country school register.
base_url	Base URL. Leave as NULL for MSMT data store URL.

Value

a URL, character of length 1

vz_grab_codelist *Get Stistko Ciselnik*

Description

Downloads the HTML page for the specified Stistko codelist code.

Usage

```
vz_grab_codelist(code, dest_dir = NULL)
```

Arguments

code	A character string representing the ciselnik code.
dest_dir	A character string specifying the destination directory. Defaults to tempdir().

Value

A character string containing the path to the downloaded HTML file.

Examples

```
vz_grab_codelist("BASO")
```

vz_load_directory *Load directory XLS file*

Description

Read and clean up quasi-XLSX file retrieved by `vz_write_directory_quasixls`

Usage

```
vz_load_directory(path)
```

Arguments

path	Path to .xls file retrieved by `vz_write_directory_quasixls`

Value

a tibble

vz_load_register *Load register*

Description

Read XML register and return tibble(s) with the register tables.

Usage

```
vz_load_register(
  dl_path,
  tables = c("organisations", "schools", "locations", "specialisations")
)
```

Arguments

dl_path	Path to XML file output by `vz_get_register_xml()`.
tables	Which tables to return. Can be one or more of "organisations", "schools", "locations" or "specialisations" (specialisations not yet available via the package).

Value

a tibble or list of tibbles if multiple table names are passed to `tables`.

vz_read_codelist *Read Stistko Ciselnik*

Description

Reads and processes the HTML file of a Stistko ciselnik.

Usage

```
vz_read_codelist(path)
```

Arguments

path	A character string representing the path to the HTML file.

Value

A data frame containing the processed data from the ciselnik.

`vz_write_directory_quasixls` *Turn a httr response created by* `vz_get_directory_responses()` *into and XLS file*

Description

Turn a httr response created by `vz_get_directory_responses()` into and XLS file

Usage

```
vz_write_directory_quasixls(response, write_file = FALSE, dest_dir = getwd())
```

Arguments

`response`	a httr respons returned by `vz_get_directory_responses()`
`write_file`	Whether to write the XLS files locally.
`dest_dir`	Directory in which to write XLS files. Defaults to working directory.

Value

character of length 1: path to XLS file

Index