# Package 'czso'

July 21, 2024

**Title** Use Open Data from the Czech Statistical Office in R

**Version** 0.4.1

**Description** Get programmatic access to the open data provided by the Czech Statistical Office (CZSO, <https://czso.cz>).

**License** MIT + file LICENSE

**URL** https://github.com/petrbouchal/czso, https://petrbouchal.xyz/czso/

**BugReports** https://github.com/petrbouchal/czso/issues

**Imports** cli, curl (>= 4.3), dplyr (>= 0.8.3), httr (>= 1.4.1), jsonlite (>= 1.6), lifecycle, magrittr, readr (>= 1.3.1), rlang (>= 0.4.4), stringi, tibble (>= 2.1.3), tools (>= 3.6.0), utils (>= 3.6.0)

**Suggests** testthat

**RdMacros** lifecycle

**Encoding** UTF-8

**Language** en

**RoxygenNote** 7.3.2

**NeedsCompilation** no

**Author** Petr Bouchal [aut, cre] (<https://orcid.org/0000-0002-0471-716X>), Jindra Lacko [ctb]

**Maintainer** Petr Bouchal <pbouchal@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-07-20 23:30:01 UTC

## Contents

czso_filter_catalogue . . . . . 2
czso_get_catalogue . . . . . 3
czso_get_codelist . . . . . 4
czso_get_dataset_doc . . . . . 5
czso_get_dataset_metadata . . . . . 6

czso_get_table . . . . . 7
czso_get_table_schema . . . . . 9
monterey . . . . . 10

**Index** **11**

---

`czso_filter_catalogue` *Filter the catalogue using a set of keywords*

---

## Description

Filter the catalogue using a set of keywords

## Usage

```
czso_filter_catalogue(catalogue, search_terms)
```

## Arguments

| | |
|---|---|
| `catalogue` | a catalogue as returned by `czso_get_catalogue()` |
| `search_terms` | #' A regex pattern (incl. plain text), or a vector of regex patterns, to filter the catalogue by. A case-insensitive filter is performed on the title, description and keywords. The search returns only catalogue entries where all the patterns are matched anywhere within the title, description or keywords. |

## Value

A tibble with the filtered catalogue.

## Examples

```
ctlg <- czso_get_catalogue()
czso_filter_catalogue(ctlg, search_terms = c("kraj", "me?zd"))
czso_filter_catalogue(ctlg, search_terms = c("úmrt", "orp"))
czso_filter_catalogue(ctlg, search_terms = c("kraj", "vazba", "orp"))
czso_filter_catalogue(ctlg, search_terms = c("ISCO", "číselník"))
czso_filter_catalogue(ctlg, search_terms = c("zaměstnání", "číselník"))
```

`czso_get_catalogue` *Get catalogue of open CZSO datasets*

## Description

Retrieves a list of all CZSO's open datasets available from the Czech Open data catalogue.

## Usage

```
czso_get_catalogue(search_terms = NULL)
```

## Arguments

| | |
|---|---|
| `search_terms` | a regex pattern, or a vector of regex patterns, to filter the catalogue by. A case-insensitive filter is performed on the title, description and keywords. The search returns only catalogue entries where all the patterns are matched anywhere within the title, description or keywords. |

## Details

Pass the string in the `dataset_id` column to `get_czso_table()`. `dataset_iri` is the unique identifier of the dataset in the national catalogue and also the URL containing all metadata for the dataset.

## Value

a data frame with details on all CZSO datasets available in the Czech National Open Data Catalogue. The columns are fairly well described by their names, except:

- some columns contain IRIs instead of human readable text; still you can deduce the content from the IRI.
- the `spatial` columns contains an IRI ending in the pattern `{unit_type}/{unit_code}`. The unit_type denotes what unit the data covers (scope/domain not granularity) and the second identifies the unit covered. The unit_type will usually be `"stat"` for "state" and the unit_code will be 1. The unit_type can also be `"KR"` for region or `"OB"` for municipality, or `"OK"` for district. In that case, the unit_code will be a code of that unit.
- `page` points to the documentation, i.e. methodology notes for the dataset.

## See Also

Other Core workflow: `czso_get_codelist()`, `czso_get_table()`

## Examples

```
czso_get_catalogue()
czso_get_catalogue(search_terms = c("kraj", "me?zd"))
```

`czso_get_codelist` *Get CZSO codelist (registry / číselník)*

## Description

Downloads codelist (registry table) and returns it in a tibble. Codelists are canonical lists of entities, their names and IDs. See Details. Codelists are included in catalogue which can be retrieved using `czso_get_catalogue()`. Their IDs start with `"cis"` followed by a two- to three-digit number.

## Usage

```
czso_get_codelist(
  codelist_id,
  language = c("cs", "en"),
  dest_dir = NULL,
  resource_num = NULL,
  force_redownload = F
)
```

## Arguments

| | |
|---|---|
| `codelist_id` | character or numeric of length 1 or 2; ID of codelist to download. See Details. |
| `language` | language, either "cs" (the default) or "en", which is available for some codelists. |
| `dest_dir` | character. Directory in which downloaded files will be stored. If left unset, will use the `czso.dest_dir` option if the option is set, and `tempdir()` otherwise. Will be created if it does not exist. |
| `resource_num` | integer, order of resource. Only override if you need a different format. |
| `force_redownload` | whether to download even if a cached local file is available. |

## Details

### Codelists:

Codelists are canonical registries of entities: things, statistical areas and aggregates, concepts, categorisations. A codelist typically contains IDs and names of all the entities fitting into a certain category.

The most commonly used codelists are geographical, e.g. lists of regions or municipalities.

In the world of the CZSO, each codelist has a numeric ID of two to four digits. You can pass this number to the function (even as a string), or you can pass the dataset ID found in the catalogue; the latter will have the form of e.g. `"cisNN"`.

*Relationships between codelists ("vazba mezi číselníky"):*
The CZSO data store also holds tables describing relations between codelists. This is especially useful for spatial hierarchies (e.g. which towns belong to which region), or for converting between categorisations (e.g. two different sets of IDs for regions.)

You can pass a vector of two IDs (numeric or character) and if the relational table for these two exists, it will be returned. (If it does not work, try flipping them around). The equivalent dataset ID, as found in the catalogue, is `"cisXXvazYY"`.

**Columns in output:**

For single-codelist files, see below for the most commonly included columns. For relational tables, you will see each column twice, each time with a suffix of 1 or 2.

- AKRCIS: codelist abbreviation
- KODCIS: codelist ID
- CHODNOTA: entity ID
- TEXT: entity name
- ZKRTEXT: entity name abbreviated
- ADMPLOD: valid from
- ADMNEPO: invalid after
- KOD_RUIAN: for geographical entites, RUIAN code (different master registry run by the geodesists)
- CZNUTS: for geographical entities, NUTS code

## Value

a tibble All columns except dates kept as character. See Details for the columns.

## See Also

Other Core workflow: `czso_get_catalogue()`, `czso_get_table()`

## Examples

```
czso_get_codelist("cis100")

# equivalent
czso_get_codelist(100)

# get a table of relations between two codelists
czso_get_codelist(c(100, 43))

# equivalent
czso_get_codelist("cis100vaz43")
```

---

# czso_get_dataset_doc *Get documentation for CZSO dataset*

---

## Description

Retrieves the URL/downloads the file containing the documentation of the dataset, in the required format.

### Usage

```
czso_get_dataset_doc(
  dataset_id,
  action = c("return", "open", "download"),
  destfile = NULL,
  format = c("html", "pdf", "word")
)
```

### Arguments

| | |
|---|---|
| dataset_id | Dataset ID |
| action | Whether to return URL (the default), download the file, or open the URL in the default web browser. |
| destfile | Where to save the file. Only used if if action = download. |
| format | What file format to access: html (the default), pdf, or word. |

### Details

The document to which this functions provides access contains methodological background on the specified dataset and is identified by the schema field in the list returned by czso_get_dataset_metadata().

### Value

if action = download, the path to the downloaded file; file URL otherwise.

### See Also

Other Additional tools: czso_get_dataset_metadata(), czso_get_table_schema()

### Examples

```
czso_get_dataset_doc("110080")
```

---

## czso_get_dataset_metadata

*Get dataset metadata*

---

### Description

Get metadata from CZSO API, which can be somewhat more detailed/readable than what is provided in the dataset's entry in the output of czso_get_dataset().

### Usage

```
czso_get_dataset_metadata(dataset_id)
```

### Arguments

| | |
|---|---|
| dataset_id | Dataset ID |

### Details

As far as I can tell there is no way to get the metadata in English, though some key datasets, such as codelists, do have English-language documentation. See `czso_get_table()` for how to access English-language codelists (registers).

### Value

a list with elements named in English, where the names are mostly self-explanatory. So are the contents where these are dates; title, description, notes and tags only exist in Czech as far as I know. Some fields merit explanation:

- `resources`: a list of files available to download in this dataset
- `frequency`: see https://project-open-data.cio.gov/iso8601_guidance/ for a key
- `ruian_type`: what type of spatial unit the data covers (spatial domain/extent/scope, not granularity). `ST` means "state" (this is almost always the case), `"KR"` means region (kraj), `"OK"` district (okres), `"OB"` municipality (obec); `"RS"` cohesion region (region soudržnosti, larger than region)
- `ruian_code`: the code of the unit the data covers as per the RUIAN taxonomy
- `schema` points to documentation while `describedBy` points to the technical schema in JSON or XML.

### See Also

Other Additional tools: `czso_get_dataset_doc()`, `czso_get_table_schema()`

### Examples

```
czso_get_dataset_metadata("110080")
```

---

## czso_get_table *Retrieve and read dataset from CZSO*

---

### Description

Downloads and reads dataset identified by `dataset_id`. Unzips if necessary, but only loads CSV files, otherwise returns the path to the downloaded file. Converts types of columns where known, e.g. value columns to numeric.

## Usage

```
czso_get_table(
  dataset_id,
  dest_dir = NULL,
  force_redownload = FALSE,
  resource_num = 1
)
```

## Arguments

| | |
|---|---|
| dataset_id | a character. Found in the czso_id column of data frame returned by get_catalogue(). |
| dest_dir | character. Directory in which downloaded files will be stored. If left unset, will use the czso.dest_dir option if the option is set, and tempdir() otherwise. Will be created if it does not exist. |
| force_redownload | integer. Whether to redownload data source file even if already cached. Defaults to FALSE. |
| resource_num | integer. Order of resource in resource list for the given dataset. Defaults to 1, the normal value for CZSO datasets. |

## Details

### Structure of the output tibble:

CZSO provides its open data as tidy data, so each row only contains one value in the hodnota column and the remaining columns give details on how that value is defined. See "Included columns" below on how these work.

### Data types:

The schema of the dataset is not yet used, so some columns may be mistyped and are by default returned as character vectors.

### Included columns:

The range of columns present in the output varies from one dataset to another, so the package does not attempt to provide English-language names for the known subset, as that would result in a jumble of Czenglish.

Instead, here is a guide to some of the common column names you will encounter:

- idhod: a unique ID of the value in the CZSO database. This does not allow you to link to any other (meta)data as far as I know, but it does provide unique identification should you need it.
- hodnota: the value.
- stapro_kod: code of the statistic/indicator/variable as listed. in the SMS UKAZ register (https://www.czso.cz/csu/czso/statistical-variables-indicators); this one has Czech-English documentation - access this by clicking the UK flag top right. You can also get a data table with the definitions, if you search for "statistické proměnné" in the title field of the catalogue. Last I checked, the ID of this table was "990124-17".
- rok denotes year as YYYY.
- ctvrtleti denotes quarter if available.

Other metadata will come in the form `{variable}_[txt|cis|kod]`. The `_txt` column holds the Czech text name for the category. The `_cis` column holds the ID of the codelist (register) you need to decode the code in `_kod`. The English codelists are at http://apl.czso.cz/iSMS/en/cislist.jsp, Czech ones at http://apl.czso.cz/iSMS/cs/cislist.jsp. You can find the Czech-language codelists in the catalogue retrieved with `czso_get_catalogue()`, where their IDs begin with `"cis"` followed by the number; the English ones can also be retrieved from the link above using a permalink URL.

More conveniently, you can use the `czso_get_codelist()` function to retrieve the codelist.

Units are denoted in a separate column.

A helper on common breakdowns with their associated columns:

- `uzemi`: territory
- `vek`: age
- `pohlavi`: gender

`NA`s in "breakdown" columns (e.g. gender or age) denote the total.

## Value

a tibble, or vector of file paths if file is not CSV or if there are multiple files in the dataset. See Details on the columns contained in the tibble

## Note

Do not use this for harvesting datasets from CZSO en masse.

## See Also

Other Core workflow: `czso_get_catalogue()`, `czso_get_codelist()`

## Examples

```
czso_get_table("110080")
```

---

`czso_get_table_schema` *Get CZSO table schema*

---

## Description

Retrieves and parses the schema for the table identified by dataset_id and resource_num.

## Usage

```
czso_get_table_schema(dataset_id, resource_num = 1)
```

## Arguments

| | |
|---|---|
| `dataset_id` | Dataset ID |
| `resource_num` | Resource number, typically 1 in CZSO (the default) |

## Details

Currently only handles JSON schema files for CSV files. If the schema is a different format, an error is returned pointing the user to the URL of the file.

## Value

a tibble with a description of the table columns, with the following items:

- `name`: the column name.
- `titles`: usually the duplicate of `name`
- `dc:description`: a Czech-language description of the column
- `required`: whether the column is required
- `datatatype`: the data type of the column; either "number" or "string"

## See Also

Other Additional tools: `czso_get_dataset_doc()`, `czso_get_dataset_metadata()`

## Examples

```
czso_get_table_schema("110080")
```

---

# monterey *{czso} on MacOS Monterey*

---

## Description

Explanation of how and why extra setup steps are needed to use {czso} on MacOS Monterey

## Details

Some early versions of MacOS Monterey ship with an old version of Libre SSL. This causes an otherwise rare bug where‘ the certificates on CZSO servers cannot be decrypted, so R cannot communicate with the server. Because R relies on the default system SSL library, this in fact affects even curl on the system command line.

The solution in R is to put

`CURL_SSL_BACKEND=SecureTransport` into .Renviron. (Don’t forget to add a newline if this is the last line in the file). Setting this anywhere after R startup may not work, hence the .Renviron solution is recommended.

You can also put this in your system environent e.g. by setting the system variable in the .profile file.

The issue no longer exists on MacOS Monterey 12.3 Beta, so presumably will also not exist on Monterey 12.3.

# Index

∗ **Additional tools**
    czso_get_dataset_doc, 5
    czso_get_dataset_metadata, 6
    czso_get_table_schema, 9
∗ **Core workflow**
    czso_get_catalogue, 3
    czso_get_codelist, 4
    czso_get_table, 7

czso_filter_catalogue, 2
czso_get_catalogue, 3, *5*, *9*
czso_get_codelist, *3*, 4, *9*
czso_get_dataset_doc, 5, *7*, *10*
czso_get_dataset_metadata, *6*, 6, *10*
czso_get_table, *3*, *5*, 7
czso_get_table_schema, *6, 7*, 9

monterey, 10

tibble, *5*, *9*