Package: CzechData (via r-universe)

September 12, 2024

Title Download various datasets (including spatial data) for the Czech Republic

Version 0.6.1

Description Download various datasets (including spatial data) for the Czech Republic.

License MIT + file LICENSE

URL https://jancaha.github.io/CzechData/index.html, https://github.com/JanCaha/CzechData

Depends R (>= 3.0), sf (>= 0.7.2)

Imports curl (>= 3.3), dplyr (>= 0.7.8), glue (>= 1.3.0), janitor (>= 1.1.1), lifecycle, lubridate (>= 1.7.4), magrittr (>= 1.5), purrr (>= 0.3.2), raster (>= 2.8.19), readr (>= 1.3.1), readxl (>= 1.2.0), rlang (>= 0.3.3), stringr (>= 1.3.1), tibble (>= 2.0.1), usethis, httr, memoise, utils, fs

Suggests DT (>= 0.5), htmltools (>= 0.3.6), htmlwidgets (>= 1.3), knitr (>= 1.21), leaflet (>= 2.0.2), RColorBrewer (>= 1.1.2), rmapshaper (>= 0.4.1), rmarkdown (>= 1.11), markdown, tidyverse (>= 1.2.1), testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

RdMacros lifecycle

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

RemoteUrl https://github.com/JanCaha/CzechData

RemoteRef HEAD

RemoteSha cbbfe649bf6d622c1aa41871d4d8a077a76c2d0b

Contents

`ciselnik_CSU` *data.frame linking codes of CSU to RUIAN codes*

Description

Czech statistical office uses different codes to identify RUIAN elemnts. This dataset should act as converter of codes from data from CSU to link them with spatial data of RUIAN.

Usage

```
ciselnik_CSU
```

Format

A data frame with 206 rows and 4 variables:

typ type of spatail unit

kod_csu id of spatial unit used by csu

kod_ruian id of spatial unit used by ruian

Source

https://www.czso.cz/csu/czso/ciselniky

generate_Data200_citation *Generate attribution for dataset Data50 od Data200*

Description

Create citation string as per terms of use (https://geoportal.cuzk.cz/Dokumenty/Podminky_EN.pdf).

Usage

```
generate_Data200_citation()

generate_Data50_citation()
```

Value

`character` with citation.

Functions

- `generate_Data200_citation()`: Generate citation for Data200 datasource.
- `generate_Data50_citation()`: Generate citation for Data50 datasource.

Examples

```
generate_Data50_citation()
generate_Data200_citation()
```

katastralni_uzemi *data.frame of all cadastral territories in Czech Republic*

Description

A dataset containing the names and other attributes of all 13,078 cadastral territories in Czech Republic. The codes (every column with string kod in name) are treated as character strings even though that some of them are numbers. These codes, however, serve only as IDs. Colums with suffix _kod are various levels of self-government units in Czech Republic.

Usage

```
katastralni_uzemi
```

Format

A data frame with 13078 rows and 9 variables:

kod id of the cadastral territory

nazev name of the cadastral territory

obec_kod

pou_kod

orp_kod

okres_kod

lau1_kod

vusc_kod

prares_kod

Source

http://services.cuzk.cz/shp/stat/epsg-5514/1.zip

---

kraje *data.frame of all regions(NUTS3) in Czech Republic*

Description

A dataset containing the names and other attributes of all 14 regions in Czech Republic. The codes (every column with string kod in name) are treated as character strings even though that some of them are numbers. These codes, however, serve only as IDs. Colums with suffix _kod are various levels of self-government units in Czech Republic.

Usage

```
kraje
```

Format

A data frame with 14 rows and 4 variables:

kod id of the region

nazev name of the region

sou_kod

nuts3_kod

Source

http://services.cuzk.cz/shp/stat/epsg-5514/1.zip

load_average_salary *Defunct: Load average salary for specific spatial units*

Description

[Defunct]

Load average salary for specific spatial units for years 2011 to 2017.

Use package czso and specifically function `czso::czso_get_table(dataset_id = "110080")` to obtain the data and `czso::czso_get_table_schema(dataset_id = "110080")` to get the columns description.

Usage

```
load_average_salary()

load_average_salary_col_explanations()
```

Value

data.frame containg the requested data

Functions

- `load_average_salary_col_explanations()`: Load description for columns

load_cadastral_territory *Extract data from Cadastral map*

Description

Extract specific layer, in form of spatial data, from cadastral map for given cadastral territory in Czech Republic. Checks are performed to find out if the provided `id` is valid for some cadastral territory in Czech Republic.

Usage

```
load_cadastral_territory(id, layer = "katastralni uzemi", WGS84 = FALSE)
```

Arguments

`id`	id of cadastral territory as character
`layer`	identification of data to extract as character, see details. Default value is `"katastralní území"`
`WGS84`	convert data to WGS-84 coordinate system? Default `FALSE`.

Details

The `layer` can have values from following set, the value in brackets is alias to full layer name:

1. `"BODOVE_POLE_B"`
2. `"BODOVE_POLE_T"`
3. `"BUDOVY_B"` (`"budovy body"`)
4. `"BUDOVY_DEF"`
5. `"BUDOVY_P"` (`"budovy"`)
6. `"DALSI_PRVKY_MAPY_B"`
7. `"DALSI_PRVKY_MAPY_L"`
8. `"DALSI_PRVKY_MAPY_T"`
9. `"HRANICE_PARCEL_L"` (`"hranice parcel"`)
10. `"KATASTRALNI_UZEMI_DEF"`
11. `"KATASTRALNI_UZEMI_L"`
12. `"KATASTRALNI_UZEMI_P"` (`"katastralni uzemi"`)
13. `"PARCELY_KN_B"`
14. `"PARCELY_KN_DEF"`
15. `"PARCELY_KN_L"`
16. `"PARCELY_KN_P"` (`"parcely"`)
17. `"PRVKY_ORIENT_MAPY_B"`
18. `"PRVKY_ORIENT_MAPY_L"`
19. `"PRVKY_ORIENT_MAPY_T"`
20. `"VB_P"`

So the codes `layer = "BUDOVY_B"` and `layer = "budovy body"` are equal.

The values of `id` follow general pattern of six number with first number being 6,7 or 9.

Value

`data.frame` with spatail objects (`sf`) of the specified layer

Information about dataset

More detailed information about data can be found at the provider's website http://atom.cuzk.cz/.

Examples

```
## Not run:
    parcely_vyskov <- load_cadastral_territory("788571", layer = "parcely")

## End(Not run)
```

load_Data200 *Load or save from Data200*

Description

Load data from Data200 data source (https://geoportal.cuzk.cz/(S(ijginumejzilvacbfijkylwj)/Default.aspx?mode=TextMeta&side=mapy_data200&text=dSady_mapyData200&head_tab=sekce-02-gp&menu=229). The data can be used only after correctly citing the creator (as per terms of use https://geoportal.cuzk.cz/Dokumenty/Podminky_EN.pdf). The citation is in form "Mapový podklad – Data200, insert year © Český úřad zeměměřický a katastrální, www.cuzk.cz".

Usage

```
load_Data200(layer, WGS84 = FALSE)

save_Data200(path, layer = NULL, type = NULL)

load_Data200_info(english_names = FALSE)
```

Arguments

layer	identification of data to extract as character, see details.
WGS84	convert data to WGS-84 coordinate system? Default FALSE.
path	character path to store the files to.
type	character type of layers to save. See details, types are listed in brackets.
english_names	change the names of the columns to English. Default FALSE.

Details

The layer can have values from following set, in the bracket is the name of general category (can be used as type in saving the data):

1. "AdministrativniHraniceLinie" ("Hranice")
2. "AdministrativniUzemiCentroid" ("Hranice")
3. "AdministrativniUzemiUTJ" ("Hranice")
4. "AdministrativniUzemiObce" ("Hranice")
5. "AdministrativniUzemiOkresy" ("Hranice")
6. "AdministrativniUzemiKraje" ("Hranice")
7. "HrazJezNad50m" ("Vodstvo")
8. "HrazJezPod50m" ("Vodstvo")
9. "JezeroRybnikVodniNadrz" ("Vodstvo")
10. "VodniTokPod50m" ("Vodstvo")
11. "VodniTokNad50m" ("Vodstvo")

12. "Ostrovy" ("Vodstvo")
13. "MokrinaBazina" ("Vodstvo")
14. "Vodopad" ("Vodstvo")
15. "Prameny1" ("Vodstvo")
16. "Prameny2" ("Vodstvo")
17. "OrografickeNazvy" ("Popis")
18. "GeomorfologickeOblasti" ("Popis")
19. "GeomorfologickeCelky" ("Popis")
20. "GeomorfologickePodcelky" ("Popis")
21. "NarodniParkPrirodniRezervace" ("RuzneObjekty")
22. "Produktovod" ("RuzneObjekty")
23. "Vysilac" ("RuzneObjekty")
24. "VyznamneObjekty" ("RuzneObjekty")
25. "ProduktovodVyznamneBody" ("RuzneObjekty")
26. "Vez" ("RuzneObjekty")
27. "DulLom" ("RuzneObjekty")
28. "Budova" ("RuzneObjekty")
29. "ElektrickeVedeni" ("RuzneObjekty")
30. "Elektrarna" ("RuzneObjekty")
31. "ObceBody" ("Sidla")
32. "ObcePolygony" ("Sidla")
33. "Privoz" ("Doprava")
34. "PrivozStanice" ("Doprava")
35. "ZeleznicniPrejezd" ("Doprava")
36. "Heliport" ("Doprava")
37. "LanovaDraha" ("Doprava")
38. "DalnicniOdpocivka" ("Doprava")
39. "KrizovatkaMimourovnova" ("Doprava")
40. "LetisteNad40Ha" ("Doprava")
41. "LetisteNad40HaBod" ("Doprava")
42. "ZelezniceZastavky" ("Doprava")
43. "LetistePod40Ha" ("Doprava")
44. "LodniPristav" ("Doprava")
45. "PristavaciDraha" ("Doprava")
46. "Zeleznice" ("Doprava")
47. "Silnice" ("Doprava")
48. "LesyPlantaze" ("Vegetace")

49. "KotovaneBody" ("Relief")
50. "Vrstevnice" ("Relief")
51. "SkalniStenaSraz" ("Relief")
52. "Jeskyne" ("Relief")
53. "DMR" ("Relief")
54. "DMRShaded" ("Relief")

Value

"load_Data200" - data.frame with spatial objects (sf) of the specified layer. For layer either "DMR" or "DMRShaded" the output is actually a (raster). "save_Data200" - path to the unzipped files (for layer) or folder (for type), the zipped file is also stored at path (mainly for further use)

Functions

- load_Data200(): Loads single dataset
- save_Data200(): Download and store layer (and zipped general category) or complete category
- load_Data200_info(): Load information about layers in Data200.

Examples

```
## Not run:
waterfalls <- load_Data200(layer = "Vodopad")

## End(Not run)
## Not run:
folder_water_objects <- save_Data200("~/data/water", type = "Vodopad")

## End(Not run)
```

---

load_Data50 *Load or save from Data50*

Description

Load data from Data50 data source (https://geoportal.cuzk.cz/(S(xbw0cmgh1cve4bciko2oo4e2))/Default.aspx?lng=EN&mode=TextMeta&side=mapy_data50&text=dSady_mapyData50&head_tab=sekce-02-gp&menu=2290). The data can be used only after correctly citing the creator (as per terms of use https://geoportal.cuzk.cz/Dokumenty/Podminky_EN.pdf). The citation is in form "Mapový podklad – Data50, insert year © Český úřad zeměměřický a katastrální, www.cuzk.cz".

Some basic description of the dataset Data50 or Data200. Most importantly names of layers and sizes of files that need to be downloaded.

Usage

```
load_Data50(layer, WGS84 = FALSE)

save_Data50(path, layer = NULL, type = NULL)

load_Data50_info(english_names = FALSE)
```

Arguments

layer	identification of data to extract as character, see details.
WGS84	convert data to WGS-84 coordinate system? Default FALSE.
path	character path to store the files to.
type	character type of layers to save. See details, types are listed in brackets.
english_names	change the names of the columns to English. Default FALSE.

Details

The layer can have values from following set, in the bracket is the name of general category (can be used as type in saving the data):

1. "BlokBudov" ("sidelniKulturniHospodarskeObjekty")
2. "Budova" ("sidelniKulturniHospodarskeObjekty")
3. "Hrad" ("sidelniKulturniHospodarskeObjekty")
4. "Hrbitov" ("sidelniKulturniHospodarskeObjekty")
5. "ChatovaKolonie" ("sidelniKulturniHospodarskeObjekty")
6. "Kostel" ("sidelniKulturniHospodarskeObjekty")
7. "LyzarskyMustek" ("sidelniKulturniHospodarskeObjekty")
8. "RozhlednaVysilac" ("sidelniKulturniHospodarskeObjekty")
9. "Rozvalina" ("sidelniKulturniHospodarskeObjekty")
10. "Stadion" ("sidelniKulturniHospodarskeObjekty")
11. "UsazovaciNadrzOdkaliste" ("sidelniKulturniHospodarskeObjekty")
12. "VetrnyMotor" ("sidelniKulturniHospodarskeObjekty")
13. "VezovitaStavba" ("sidelniKulturniHospodarskeObjekty")
14. "VodojemVezovy" ("sidelniKulturniHospodarskeObjekty")
15. "Zamek" ("sidelniKulturniHospodarskeObjekty")
16. "Zricenina" ("sidelniKulturniHospodarskeObjekty")
17. "Cesta" ("komunikace")
18. "LanovaDraha" ("komunikace")
19. "Lavka" ("komunikace")
20. "Letiste" ("komunikace")
21. "LetisteObvodovaLinie" ("komunikace")

22. "Most" ("komunikace")
23. "Pesina" ("komunikace")
24. "Pristav" ("komunikace")
25. "Privoz" ("komunikace")
26. "SilniceDalnice" ("komunikace")
27. "SilniceVeVystavbe" ("komunikace")
28. "Tunel" ("komunikace")
29. "Ulice" ("komunikace")
30. "ZeleznicniStanice" ("komunikace")
31. "ZeleznicniTrat" ("komunikace")
32. "ZeleznicniVlecka" ("komunikace")
33. "ElektrickeVedeni" ("produktovodyElektrickeVedeni")
34. "Produktovod" ("produktovodyElektrickeVedeni")
35. "Akvadukt" ("vodstvo")
36. "Hraz" ("vodstvo")
37. "Jez" ("vodstvo")
38. "Shybka" ("vodstvo")
39. "VodniPlocha" ("vodstvo")
40. "VodniTok" ("vodstvo")
41. "HraniceSpravniJednotkyaKU" ("hraniceUzemnichJednotek")
42. "ChraneneUzemi" ("hraniceUzemnichJednotek")
43. "Les" ("vegetacePovrch")
44. "LoukaPastvina" ("vegetacePovrch")
45. "RaselinisteMocalBazina" ("vegetacePovrch")
46. "ZahradaSadParkViniceChmelnice" ("vegetacePovrch")
47. "Jeskyne" ("terenniRelief")
48. "KotovanyBod" ("terenniRelief")
49. "SkalnatySraz" ("terenniRelief")
50. "Skaly" ("terenniRelief")
51. "TerenniStupen" ("terenniRelief")
52. "TerenniStupenSpadnice" ("terenniRelief")
53. "Vrstevnice" ("terenniRelief")
54. "DefinicniBodCastiObce" ("popis")
55. "DefinicniBodSpravnihoCelku" ("popis")
56. "Jmeno_B" ("popis")
57. "Jmeno_L" ("popis")
58. "Jmeno_P" ("popis")
59. "ObjektRuzny" ("popis")

Value

"load_Data50" - data.frame with spatial objects (sf) of the specified layer. "save_Data50" - path to the unzipped files (for layer) or folder (for type), the zipped file is also stored at path (mainly for further use)

data.frame with description of layers.

Functions

- load_Data50(): Loads single dataset
- save_Data50(): Download and store layer (and zipped general category) or complete category
- load_Data50_info(): Load information about layers in Data50.

Examples

```
## Not run:
    rivers <- load_Data50(layer = "VodniTok")

## End(Not run)
## Not run:
    folder_communications <- save_Data50("~/data/coomunications", type = "komunikace")

## End(Not run)
## Not run:
    info <- load_Data50_info(english_names = TRUE)
    info <- load_Data200_info(english_names = TRUE)

## End(Not run)
```

---

load_financial_indicators

Defunct: Load financial indicators for specific spatial units

---

Description

[Defunct]

Load financial indicators for specific spatial units (NUTS2 and NUTS3) for years 1995 to 2018.

Use package czso and specifically function czso::czso_get_table(dataset_id = "050101") to obtain the data and czso::czso_get_table_schema(dataset_id = "050101") to get the columns description.

Usage

```
load_financial_indicators()

load_financial_indicators_col_explanations()
```

Value

data.frame containg the requested data

Functions

- `load_financial_indicators_col_explanations()`: Load description for columns

`load_population_age` *Defunct: Load population by age*

Description

[Defunct]

Load population by sex and five-year age categories for years 2010 to 2018.

Use package `czso` and specifically function `czso::czso_get_table(dataset_id = "130142")` to obtain the data and `czso::czso_get_table_schema(dataset_id = "130142")` to get the columns description.

Usage

```
load_population_age(year = NA, area_type = NA)

load_population_age_col_explanations()
```

Arguments

`year`	for which the data should be obtained. Default value is `NA`, which means all the years. Values from range (including both limits) 2010 - 2018 are accepted.
`area_type`	type of are for which the data should be obtained. Default value is `NA`, which means all areas. Accepted values are in Description.

Details

The `area_type` can have values from following set:

1. `"okresy"`
2. `"kraje"`
3. `"republika"`

Value

data.frame containg the requested data

Functions

- `load_population_age()`: Load the data
- `load_population_age_col_explanations()`: Load description for columns

Information about dataset

More detailed information about data can be found at the provider's website https://www.czso.cz/csu/czso/obyvatelstvo-k-3112-podle-pohlavi-v-obcich.

`load_population_settlements`

Defunct: Load populations from settlements

Description

[Defunct]

Load population by sex for settlements for each years from 2000 to 2018.

Use package `czso` and specifically function `czso::czso_get_table(dataset_id = "130149")` to obtain the data and `czso::czso_get_table_schema(dataset_id = "130149")` to get the columns description.

Usage

```
load_population_settlements(year = NA)

load_population_settlements_col_explanations()
```

Arguments

`year`	year for which the data should be obtained. Default value is `NA`, which means all the years. Values from range (including both limits) 2000 - 2018 are accepted.

Value

data.frame containg the requested data

Functions

- `load_population_settlements()`: Load the data
- `load_population_settlements_col_explanations()`: Load description for columns

Information about dataset

More detailed information about data can be found at the provider's website https://www.czso.cz/csu/czso/obyvatelstvo-podle-petiletych-vekovych-skupin-a-pohlavi-v-krajich-a-okresech.

load_RUIAN_settlement *Extract data from RUIAN*

Description

Extract specific layer, in form of spatial data, from RUIAN for given settlement in Czech Republic. Checks are performed to find out if the provided `id` is valid for some settlement in Czech Republic.

Usage

```
load_RUIAN_settlement(id, layer = "obec", WGS84 = FALSE)
```

Arguments

`id`	id of settlement as character
`layer`	identification of data to extract as character, see details. Default value is `"obec"`
`WGS84`	convert data to WGS-84 coordinate system? Default `FALSE`.

Details

In case of adress places (using ADRM_B or adresni mista as layer) are checked, the csv file with more attributes is also downloaded and linked to the spatial data layer.

The `layer` can have values from following set, the value in brackets is alias to full layer name:

1. `"ADRM_B"` (`"adresni mista"`)
2. `"CO_B"` (`"casti obce"`)
3. `"KATUZE_P"` (`"katastralni uzemi"`)
4. `"OBEC_P"` (`"obec"`)
5. `"SO_B"` (`"stavebni objekty"`)
6. `"UL_L"` (`"ulice"`)
7. `"VO_P"` (`"volebni okrsky"`)
8. `"ZSJ_P"` (`"zakladni sidelni jednotky"`)
9. `"MOMC_P"`
10. `"MOP_P"`
11. `"SOP_P"`

So the codes `layer = "CO_B"` and `layer = "casti obce"` are equal.

The values of `id` follow general pattern of six number with first number being 5.

Value

`data.frame` with spatial objects (`sf`) of the specified layer

Information about dataset

More detailed information about data can be found at the provider's website http://atom.cuzk.cz/.

Examples

```
## Not run:
    adresy_vyskov <- load_RUIAN_settlement("592889", layer = "adresni mista")

## End(Not run)
```

---

load_RUIAN_state *Extract data from RUIAN for whole Czech Republic*

Description

Extract specific layer, in form of spatial data, from RUIAN for whole Czech Republic. The minor issue with these data is the size, the datasets that needs to be downloaded is roughly 190 MB.

Usage

```
load_RUIAN_state(layer = "stat", WGS84 = FALSE)
```

Arguments

layer	identification of data to extract as character, see details. Default value is "stát"
WGS84	convert data to WGS-84 coordinate system? Default FALSE.

Details

The layer can have values from following set, the value in brackets is alias to full layer name:

1. "KATUZE_P" ("katastralni uzemi")
2. "OBCE_P" ("obce")
3. "OKRESY_P" ("okresy")
4. "ORP_P" ("orp")
5. "POU_P" ("pou")
6. "PRARES_P"
7. "REGION_P" ("regiony")
8. "STATY_P" ("stat")
9. "STU_P" ("stavebni úrady")
10. "VO_P" ("volebni okrsky")
11. "VUSC_P" ("kraje")

So the codes layer = "OKRESY_P" and layer = "okresy" are equal.

Value

`data.frame` with spatail objects (`sf`) of the specified layer

Information about dataset

More detailed information about data can be found at the provider's website `http://atom.cuzk.cz/`.

Examples

```
## Not run:
    obce_CR <- load_RUIAN_state(layer = "obce")

## End(Not run)
```

---

load_SLDB_2011 *Deprecated: Get information from Czech census in 2011*

Description

[Soft-deprecated]

Data from Czech census in year 2011 by four main topics. The data are provided at various aggregation levels that can be filtered.

Use package `czso` and specifically function `czso::czso_get_table(dataset_id)` to obtain the data and `czso::czso_get_table_schema(dataset_id)` to get the columns description. The values for specific datasets of census are `"SLDB-VYBER"`, `"sldbdomy"`, `"sldbdomac"`, `"sldbvyjizdka"`.

Usage

```
load_SLDB_2011(type = "obyvatelstvo", load_names = TRUE)

load_SLDB_2011_col_explanations(type = "obyvatelstvo")
```

Arguments

`type`	type of requested information as character. Default value is "obyvatelstvo". See details for more.
`load_names`	boolean value if the column names should be loaded from external source. Default TRUE as the column names do no make any sense otherwise.

Details

Types of data that can be downloaded, and used as `type` parameter in the function call:

1. `"obyvatelstvo"`
2. `"domy-byty"`
3. `"domacnosti"`
4. `"vyjizdka"`

Value

data.frame containg the requested data

Functions

- `load_SLDB_2011()`: Load the data
- `load_SLDB_2011_col_explanations()`: Get names of columns for SLDB of specific type as data.frame

Information about dataset

More detailed information about data can be found at the provider's website https://www.czso.cz/csu/sldb.

Examples

```
## Not run:
    sldb <- load_SLDB_2011(type = "obyvatelstvo")

## End(Not run)
```

---

obce *data.frame of all settlements in Czech Republic*

Description

A dataset containing the names and other attributes of all 6,258 settlements (villages and cities) in Czech Republic. The codes (every column with string kod in name) are treated as character strings even though that some of them are numbers. These codes, however, serve only as IDs. Colums with suffix _kod are various levels of self-government units in Czech Republic.

Usage

```
obce
```

Format

A data frame with 6258 rows and 7 variables:

kod id of the settlement

nazev name of the settlement

pou_kod

orp_kod

okres_kod

lau1_kod

vusc_kod

Source

http://services.cuzk.cz/shp/stat/epsg-5514/1.zip

---

okresy *data.frame of all districts (LAU1) in Czech Republic*

Description

A dataset containing the names and other attributes of all 77 districts in Czech Republic. The codes (every column with string kod in name) are treated as character strings even though that some of them are numbers. These codes, however, serve only as IDs. Colums with suffix _kod are various levels of self-government units in Czech Republic.

Usage

```
okresy
```

Format

A data frame with 77 rows and 5 variables:

kod id of the district

nazev name of the district

lau1_kod

vusc_kod

nuts3_kod

Source

http://services.cuzk.cz/shp/stat/epsg-5514/1.zip

---

orp *data.frame of all settlements of type III (orp) in Czech Republic*

Description

A dataset containing the names and other attributes of all 206 settlements of type III (orp) in Czech Republic. The codes (every column with string kod in name) are treated as character strings even though that some of them are numbers. These codes, however, serve only as IDs. Colums with suffix _kod are various levels of self-government units in Czech Republic.

Usage

```
orp
```

Format

A data frame with 206 rows and 4 variables:

kod id of the settlements of type III (orp)

nazev name of the settlements of type III (orp)

vusc_kod

nuts3_kod

Source

http://services.cuzk.cz/shp/stat/epsg-5514/1.zip

---

pou *data.frame of all settlements of type II (pou) in Czech Republic*

Description

A dataset containing the names and other attributes of all 393 settlements of type II (pou) in Czech Republic. The codes (every column with string kod in name) are treated as character strings even though that some of them are numbers. These codes, however, serve only as IDs. Colums with suffix _kod are various levels of self-government units in Czech Republic.

Usage

```
pou
```

Format

A data frame with 393 rows and 5 variables:

kod id of the settlements of type II (pou)

nazev name of the settlements of type II (pou)

orp_kod

vusc_kod

nuts3_kod

Source

http://services.cuzk.cz/shp/stat/epsg-5514/1.zip

set_cache_length *Set and get cache validity time*

Description

Set and get how long the cache is valid and the files are downloaded only once and then returned from the cache.

Usage

```
set_cache_length(length)

get_cache_length()
```

Arguments

length	numeric time in seconds how long the cache is valid.

Value

`get_cache_length()` numeric length of cache validity (default value is 7 days)

Functions

- `set_cache_length()`: Set cache length
- `get_cache_length()`: Get cache length

Index