Package ‘RCzechia’

Type Package
Title Spatial Objects of the Czech Republic
Version 1.9.2
Date 2022-05-14
Author Jindra Lacko
Maintainer Jindra Lacko <jindra.lacko@gmail.com>
Description Administrative regions and other spatial objects of the Czech Republic.
URL https://github.com/jlacko/RCzechia
BugReports https://github.com/jlacko/RCzechia/issues
License MIT + file LICENSE
Encoding UTF-8
Depends R (>= 3.3.0), sf
Imports curl, httr, jsonlite, magrittr
Suggests raster, readxl, units, ggplot2, stringr, rvest, xml2, scales,
       knitr, testthat, roxygen2, devtools, rmarkdown, dplyr, lwgeom,
       s2, covR
SystemRequirements GDAL (>= 2.2.3), GEOS (>= 3.6.2), PROJ (>= 4.9.3)
RoxygenNote 7.2.0
VignetteBuilder knitr
NeedsCompilation no
Repository CRAN
Date/Publication 2022-05-14 20:30:02 UTC

R topics documented:

   RCzechia-package .................................................. 2
casti ................................................................. 3
chr_uzemi ............................................................ 4
geocode ............................................................... 5
RCzechia-package

Description

A selection of spatial objects relevant to the Czech Republic. Due to CRAN package size requirements (5 MB) the objects are stored externally (on Amazon S3) - and therefore could not be implemented as datasets. They are functions returning data frames instead.

Details

To save time (and bandwidth) the downloaded objects are saved locally in ‘tempdir’ directory when requested, and downloaded at most once per R session; out of respect to CRAN Repository Policy a more permanent caching on user’s side is not attempted.

This means that:

• a working internet connection is required to use the full resolution objects
• all objects need to be called with (possibly empty) braces

For the most frequently used objects - **republika, kraje** and **okresy** - a low resolution version is also implemented. The low resolution data sets are stored locally (and working internet connection is not necessary to use them).

All objects are implemented as sf data frames.
Administrative regions

- republika - borders of the Czech Republic
- kraje - regions / NUTS3 units
- okresy - districts / LAU1 units
- orp_polygony - municipalities with extended powers (obce s rozšířenou působností)
- obce_polygony - municipalities as polygons
- obce_body - municipalities as centroids (points)
- casti - city districts (for cities that implement them)

Natural objects

- reky - rivers
- plochy - water bodies
- lesy - woodland areas (more than 30 ha in area)

Other objects

- silnice - roads
- zeleznice - railroads
- KFME_grid - grid cells (faunistické čtverce) according to Kartierung der Flora Mitteleuropas methodology
- chr_uzemi - protected natural areas ( chráněná území)

Utility functions

In addition three utility functions are implemented to support spatial workflow:

- geocode - geocoding (from address to coordinates)
- revgeo - reverse geocoding (from coordinates to address)

---

casti          City Parts

Description

Function taking no parameters and returning data frame of districts of Prague and other major cities as sf polygons.

Usage

casti()
Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).
The data is current to June 2021. Downloaded size is 1.5 MB.

Value

sf data frame with 142 rows of 4 variables + geometry

KOD Code of the city part / kod mestske casti
NAZEV Name of the city part / nazev mestske casti
KOD_OBEC Code of the city
NAZ_OBEC Name of the city

Source

© ČÚZK, 2021 https://vdp.cuzk.cz/

chr_uzemi Protected Natural Areas

Description

Function returning data frame of protected natural areas (Chráněná území) of the Czech Republic as sf polygons. It has no obligatory parameters.

Usage

chr_uzemi()

Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).
The data is current to September 2020. Downloaded size is 7 MB (so use with caution, and patience).

Value

sf data frame with 2677 rows of 3 variables + geometry

TYP Type of protected area
NAZEV Name, with Czech accents
PLOCHA type of protected area: large or small

Source

© AOPK ČR, 2020 https://data.nature.cz/
**Description**

This function connects to Czech State Administration of Land Surveying and Cadastre (https://www.cuzk.cz/en) API to geocode an address. As consequence it is implemented only for Czech addresses.

**Usage**

```r
geocode(address, crs = 4326)
```

**Arguments**

- `address`: point to be geocoded, as character (vector)
- `crs`: coordinate reference system of output; default = WGS84

**Details**

Input of the function are an address to geocode (or a vector of addresses) and expected Coordinate Reference System of output (default is WGS84 = EPSG:4326, but in some use cases inž. Křovák = EPSG:5514 may be more relevant).

NA's in input are considered an error.

Output is a `sf` data frame of spatial points.

Depending on the outcome of matching the address to RÚIAN data there is a number of possible outcomes:

- All items were *matched exactly*: the returned `sf` data frame has the same number of rows as there were elements in vector to be geocoded. The field `target` will have zero duplicates.
- Some items had *multiple matches*: the returned `sf` data frame has more rows than the there were elements in vector to be geocoded. In the field `target` will be duplicate values. Note that the RÚIAN API limits multiple matches to 10.
- Some (but not all) items had *no match* in RÚIAN data: the returned `sf` data frame will have fewer rows than the vector sent. to be geocoded elements. Some values will be missing from field `target`.
- No items were matched at all: the function returns empty data frame and a message.
- The CUZK API is down or overloaded: the function returns empty data frame and a message.

Note that character encoding is heavily platform dependent, and you may need to convert to UTF-8, e.g. by running `address <- iconv(address, from = "windows-1250", to = "UTF-8")` before calling the function.

Value

sf data frame with 3 variables + geometry

address the address searched (address input)
type type of record matched by API
result address as returned by API / recorded in RÚIAN
geometry hidden column with spatial point data

Examples

```
asdf <- geocode("Gogolova 212, Praha 1")
print(asdf)
```

KFME_grid

KFME grid cells (faunistické čtverce) of the Czech Republic

Description

Function returning grid covering the Czech Republic according to the Kartierung der Flora Mitteleuropas methodology.

Usage

```
KFME_grid(resolution = "low")
```

Arguments

resolution Should the function return high or low resolution shapefile? Allowed values are "low" and "high". Default is "low".

Details

The function returns a sf data frame of grid cells. Depending on the value of parameter ‘resolution’ either low resolution (26x42 cells - default) with labels in 4 digit format (e.g. Hrčava = 6479) or high resolution (104x168 cells) with labels in 4 digit + 1 letter format (e.g Hrčava = 6479c).

Value

sf data frame with 1092 rows in low resolution and 4368 rows in high resolution

cfverec KFME code of the grid cell; depending on value of ‘resolution’ parameter either 4 digits, or 4 digits + 1 letter
**Examples**

```r
library(ggplot2)

ggplot() + geom_sf(data = republika("low")) + geom_sf(data = KFME_grid("low"), fill = NA)
```

---

**kraje  
Regions (kraje) of the Czech Republic**

**Description**

Function returning data frame of NUTS3 administrative units for the Czech Republic as sf polygons. It takes a single parameter resolution - high res (default) or low res polygons.

**Usage**

```r
kraje(resolution = "high")
```

**Arguments**

- `resolution` Should the function return high or low resolution shapefile? Allowed values are "high" (default) and "low". This parameter affects only the geometry column, all other fields remain the same.

**Details**

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to June 2021 (i.e changes introduced by act 51/2020 Sb. are reflected). Downloaded size of high resolution shapefile is <1 MB.

**Value**

sf data frame with 14 rows of 3 variables + geometry

- `KOD_KRAJ` Code of the region.
- `KOD_CZNUTS3` Code of the region as NUTS3 (kraj).
- `NAZ_CZNUTS3` Name of the region as NUTS3 (kraj).

**Source**

© ČÚZK, 2021 [https://vdp.cuzk.cz/](https://vdp.cuzk.cz/)
Examples

```r
library(sf)

hranice <- kraje("low")
plot(hranice, col = "white", max.plot = 1)
```

---

**lesy**  
*Woodland Areas*

**Description**

Function returning data frame of woodland areas (lesy) of more than 30 hectares in area of the Czech Republic as `sf` polygons. It has no obligatory parameters.

**Usage**

```r
lesy()
```

**Details**

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package). The data is current to January 2014. Downloaded size is 2.1 MB.

**Value**

`sf` data frame with 2,366 rows of geometry variable only

**Source**

© ArcČR, ARCDATA PRAHA, ZÚ, ČSÚ, 2016 [https://www.arcdata.cz/produkty/geograficka-data/arccr-4-0](https://www.arcdata.cz/produkty/geograficka-data/arccr-4-0)

---

**obce_body**  
*Municipalities / communes (obce) as centerpoints*

**Description**

Function returning data frame of LAU2 administrative units for the Czech Republic as `sf` points. It takes no parameters.

**Usage**

```r
obce_body()
```
Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to June 2021 (i.e changes introduced by act 51/2020 Sb. are reflected). Downloaded size is <1 MB.

Value

sf data frame with 6,258 rows of 14 variables + geometry

- **KOD_OBEC** Code of the level I commune (obec).
- **NAZ_OBEC** Name of the level I commune (obec).
- **KOD_POU** Code of the level II commune (obec s poverenym uradem).
- **NAZ_POU** Name of the level II commune (obec s poverenym uradem).
- **KOD_ORP** Code of the level III commune (obec s rozsirenom pusuobnosti).
- **NAZ_ORP** Name of the level III commune (obec s rozsirenom pusuobnosti).
- **KOD_OKRES** Code of the district (okres).
- **KOD_LAU1** Code of the LAU1 administrative unit (okres).
- **NAZ_LAU1** Name of the LAU1 administrative unit (okres).
- **KOD_KRAJ** Code of the region (kraj).
- **KOD_CZNUTS3** Code of the NUTS3 unit (kraj).
- **NAZ_CZNUTS3** Name of the NUTS3 unit (kraj).

Source

© ČÚZK, 2021 [https://vdp.cuzk.cz/](https://vdp.cuzk.cz/)

---

**obce_polygony**

Municipalities / communes (obce) as polygons

Description

Function returning data frame of LAU2 administrative units for the Czech Republic as sf polygons. It takes no parameters.

Usage

obce_polygony()

Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to June 2021 (i.e changes introduced by act 51/2020 Sb. are reflected). Downloaded size is 13.3 MB (so use with caution, and patience).
Value

| sf data frame with 6,258 rows of 14 variables + geometry |

| KOD_OBEC | Code of the level I commune (obec). |
| NAZ_OBEC | Name of the level I commune (obec). |
| KOD_POU  | Code of the level II commune (obec s poverenym uradem). |
| NAZ_POU  | Name of the level II commune (obec s poverenym uradem)). |
| KOD_ORP  | Code of the level III commune (obec s rozsirenou pusuobnosti). |
| NAZ_ORP  | Name of the level III commune (obec s rozsirenou pusuobnosti). |
| KOD_OKRES| Code of the district (okres). |
| KOD_LAU1 | Code of the LAU1 administrative unit (okres). |
| NAZ_LAU1 | Name of the LAU1 administrative unit (okres). |
| KOD_KRAJ | Code of the region (kraj). |
| KOD_CZNUTS3 | Code of the NUTS3 unit (kraj) |
| NAZ_CZNUTS3 | Name of the NUTS3 unit (kraj) |

Source

© ČÚZK, 2021 https://vdp.cuzk.cz/

Examples

```r
library(sf)
library(dplyr)

praha <- obce_polygony() %>%
  filter(NAZ_LAU1 == "Praha")

plot(praha, max.plot = 1)
```

---

<table>
<thead>
<tr>
<th>okresy</th>
<th>Districts (okresy)</th>
</tr>
</thead>
</table>

Description

Function returning data frame of LAU1 administrative units for the Czech Republic as sf polygons. It takes a single parameter resolution - high res (default) or low res polygons.

Usage

```r
okresy(resolution = "high")
```
Arguments

resolution Should the function return high or low resolution shapefile? Allowed values are "high" (default) and "low". This parameter affects only the geometry column, all other fields remain the same.

Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to June 2021 (i.e changes introduced by act 51/2020 Sb. are reflected). Downloaded size of high resolution shapefile 2.1 MB.

Value

sf data frame with 77 rows of 6 variables + geometry

KOD_OKRES Code of the district (okres).
KOD_LAU1 Code of the district as LAU1 unit (okres).
NAZ_LAU1 Name of the district as LAU1 unit (okres).
KOD_KRAJ Code of the region.
KOD_CZNUTS3 Code of the region as NUTS3 (kraj).
NAZ_CZNUTS3 Name of the region (kraj).

Source

© ČÚZK, 2021 https://vdp.cuzk.cz/

Examples

library(sf)

hranice <- okresy()
plot(st_geometry(hranice), col = "white")

object.size(okresy("low"))
object.size(okresy("high"))
orp_polygony  
*Obce s rozsířenou působností*

**Description**

Function returning data frame of municipalities with extended powers (obce s rozšířenou působností) as sf polygons. It takes no parameters.

**Usage**

orp_polygony()

**Details**

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to June 2021 (i.e changes introduced by act 51/2020 Sb. are reflected). Downloaded size is 3.1 MB.

**Value**

sf data frame with 206 rows of 5 variables + geometry

- **KOD_ORP** Code of the level III commune (obec s rozsířenou působností).
- **NAZ_ORP** Full name of the level III commune (obec s rozsířenou působností).
- **KOD_KRAJ** Code of the region (kraj).
- **KOD_CZNUTS3** Code of the NUTS3 unit (kraj)
- **NAZ_CZNUTS3** Name of the NUTS3 unit (kraj)

**Source**

© ČÚZK, 2021 [https://vdp.cuzk.cz/](https://vdp.cuzk.cz/)

---

**plochy**  
*Water Bodies*

**Description**

Function returning data frame of water bodies of the Czech Republic as sf polygons. It takes no parameters.

**Usage**

plochy()
Details
Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).
The data is current to December 2020. Downloaded size is 1.5 MB.

Value
sf data frame with 1.769 rows of 2 variables + geometry

NAZEV Name, with Czech accents
VYSKA water level, meters above sea level

Source

Description
Function returning data frame of rivers of the Czech Republic as sf lines. It takes a single parameter `scope` with default "global".

Usage
reky(scope = "global", resolution = "high")

Arguments
<table>
<thead>
<tr>
<th>scope</th>
<th>Should the function return all rivers, or just Vltava in Prague / Svitava &amp; Svratka in Brno?</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolution</td>
<td>Should the function return high or low resolution shapefile? Allowed values are &quot;high&quot; (default) and &quot;low&quot;. This parameter affects only the geometry column, all other fields remain the same.</td>
</tr>
</tbody>
</table>

Details
Two special case scopes are defined: Praha (returning the part of Vltava in and around Prague) and Brno (returning Svitava and Svratka near and around Brno).

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).
The data is current to December 2020. Downloaded size is 4.4 MB.
Value

sf data frame with 3.616 rows of 4 variables + geometry:

**TYP** Type of river
**NAZEV** Name, with Czech accents
**Navigable** Boolean indicating navigability of river.
**Major** Boolean indicating one of the major rivers.

Source


Examples

```r
library(sf)
plot(st_geometry(subset(okresy(), KOD_LAU1 == "CZ0642"))) # Brno city
plot(reky("Brno"), add = TRUE) # Svitava & Svratka added to Brno my city plot
```

---

**republika** | **Republika**
---

Description

Boundaries of the Czech Republic as sf polygon.

Usage

```r
republika(resolution = "high")
```

Arguments

**resolution** Should the function return high or low resolution shapefile? Allowed values are "high" (default) and "low". This parameter affects only the geometry column, all other fields remain the same.

Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to June 2021. Downloaded size of high resolution shapefile is <1 MB.

Value

sf data frame with 1 row of 1 variable + geometry:
revgeo

Source
© ČÚZK, 2021 https://vdp.cuzk.cz/

Examples

library(sf)

hranice <- republika("low")
plot(hranice, col = "white")

revgeo

Reversely Geocode a Czech Address

Description
This function connects to Czech State Administration of Land Surveying and Cadastre (https://www.cuzk.cz/en) API to reversely geocode an address. As consequence it is implemented only for Czech addresses.

Usage

revgeo(coords)

Arguments

coords coordinates to be reverse geocoded; expected as sf data frame of spatial points

Details
Input of the function is a sf data frame of spatial points, and output a vector of characters.
The function returns the same sf data frame as input, with added field revgeocoded; it contains the result of operation. Should the data frame contain a column named revgeocoded it will be overwritten.

In case of reverse geocoding failures (e.g. coordinates outside of the Czech Republic and therefore scope of ČÚZK) NA is returned.
In case of API failures (CUZK down) the function returns NAs again, with a message.

Value
sf data frame as input, with column revgeocoded added (or overwritten)
Examples

```r
library(dplyr)
library(sf)

brno <- obce_polygony() %>% # shapefile of Brno
    filter(NAZ_OBEC == "Brno") %>%
    st_transform(5514) # planar CRS (eastings & northings)

pupek_brna <- st_centroid(brno) # calculate centroid

adresa_pupku <- revgeo(pupek_brna)$revgeocoded # address of the center
```

---

**senat_obvody**  
*Senate Districts (Volební obvod pro volby do Senátu) of the Czech Republic*

**Description**

Function returning data frame of the 81 Senate Districts for the Czech Republic as sf polygons. It takes a single parameter resolution - high res (default) or low res polygons.

**Usage**

```r
senat_obvody(resolution = "high")
```

**Arguments**

- `resolution` Should the function return high or low resolution shapefile? Allowed values are "high" (default) and "low". This parameter affects only the geometry column, all other fields remain the same.

**Details**

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to February 2021 (last update was in 2016). Downloaded size of high resolution shapefile is 10 MB, size of the low res object is negligible (but a working internet is still required, as the object is not internal).

**Value**

- sf data frame with 81 rows of 3 variables + geometry

  - **OBVOD** Code of the district; left padded with zero in case of districts one to nine.
  - **SIDLO** Seat of the senator.
  - **NAZEV_VO** Formal name of the district.
silnice

Source

ČSÚ [https://www.czso.cz/csu/czso/podminky_pro_vyuzivani_a_dalsi_zverejnovani_statistickych_udaju_csu](https://www.czso.cz/csu/czso/podminky_pro_vyuzivani_a_dalsi_zverejnovani_statistickych_udaju_csu)

Examples

```r
library(sf)

senat <- senat_obvody("low")
plot(st geometry(senat), col = "white")
nrow(senat) # 81, because the Constitution says so...
```

<table>
<thead>
<tr>
<th>silnice</th>
<th>Road Network</th>
</tr>
</thead>
</table>

Description

Function returning data frame of roads of the Czech Republic as sf lines. It has no obligatory parameters.

Usage

```r
silnice()
```

Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to December 2020. Downloaded size is 6 MB.

Value

sf data frame with 59,594 rows of 3 variables + geometry:

- **TRIDA** Class of the road: highway = dálnice, speedway = rychlostní silnice, 1st class road = silnice I. třídy, 2nd class road = silnice II. třídy, 3rd class road = silnice III. třídy, other road = neevidovaná silnice
- **CISLO_SILNICE** Local road code
- **MEZINARODNI_OZNACENI** International road code

Source

Description

Function returning data frame of the local election districts for the Czech Republic as sf polygons. It takes a single parameter resolution - high res (default) or low res polygons.

Usage

volebni_okrsky(resolution = "high")

Arguments

resolution Should the function return high or low resolution shapefile? Allowed values are "high" (default) and "low". This parameter affects only the geometry column, all other fields remain the same.

Details

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to February 2021 (reflecting the freeze between announcing the date of the general elections by the President in December 2020 and certifying the results sometime in October 2021). Downloaded size of high resolution shapefile is 76 MB, size of the low res object is 5 MB (so proceed with caution, and patience).

Value

sf data frame with 14 761 rows of 6 variables + geometry

Kod Unique id of the district.
Cislo Id of the district within a given Obec / not globally unique.
ObecKod Id of obec - maps to obce_polygony()$SKOD_OBEC.
MomeKod Id of městská část - maps to casti()$SKOD.
KOD_LAU1 Id of okres - maps to okresy()$KOD_LAU1.
KOD_CZNUTS3 Id of kraj - maps to kraje()$KOD_CZNUTS3.

Source

© ČÚZK, 2021 https://vdp.cuzk.cz/
**Examples**

```r
library(sf)

prazske_okrsky <- subset(volebni_okrsky("low"), ObecKod == "554782")
plot(prazske_okrsky) # the districts of Prague
```

---

**Description**

Terrain of the Czech Republic as a raster package object.

**Usage**

```r
vyskopis(format = "rayshaded")
```

**Arguments**

- `format` Should the function return actual relief (meters above sea level) or shaded relief (rayshaded). Allowed values are "actual" and "rayshaded".

**Details**

The function returns a raster file of either actual relief (values are meters above sea level) or rayshaded relief (created via highly recommended rayshader package).

The raster is 5084 by 3403 cells, meaning each pixel is about 90 × 90 meters. It works the best at level of country or regions, at the level of a city or lower it may be somewhat grainy.

Due to package size constraints both versions are stored externally (and a working internet connection is required to use the package).

The data is current to September 2016. Downloaded size of the rayshaded raster is 8.4 MB, actual raster is 31.4 MB.

**Value**

`raster` package `RasterLayer`.

**Source**

© ArcČR, ARCDATA PRAHA, ZÚ, ČSÚ, 2016 [https://www.arcdata.cz/produkty/geograficka-data/arccr-4-0](https://www.arcdata.cz/produkty/geograficka-data/arccr-4-0)
Examples

```r
library(raster)
relief <- vyskopis("rayshaded")
plot(relief, col = gray.colors(16))
```

---

zeleznice  

**Railroad Network**

**Description**

Function returning data frame of railroads of the Czech Republic as \texttt{sf} lines. It has no obligatory parameters.

**Usage**

```r
zeleznice()
```

**Details**

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to December 2020. Downloaded size is <1 MB.

**Value**

\texttt{sf} data frame with 9,957 rows of 3 variables + geometry:

- **ELEKTRIFIKACE** is the railroad electrified?
- **KOLEJNOST** track: single = jednokolejní, double = dvojkolejní, more = tří a vícekolejní
- **ROZCHODNOST** gauge: standard = normální, narrow = úzkokolejka

**Source**

zip_codes

ZIP Codes of the Czech Republic

Description

Function returning data frame of the 2 671 ZIP Code Areas for the Czech Republic as sf polygons. It takes a single parameter resolution - high res (default) or low res polygons.

Usage

zip_codes(resolution = "high")

Arguments

resolution  Should the function return high or low resolution shapefile? Allowed values are "high" (default) and "low". This parameter affects only the geometry column, all other fields remain the same.

Details

Note that there are a number of special ZIP Codes - e.g. 118 01 for Government of the Czech Republic (Strakova akademie / Nábřeží Edvarda Beneše 4). These are not listed here, as they do not relate to a specific delivery area but are in essence private.

The geometry type is MULTIPOLYGON, as there are a number of non continuous areas of delivery.

Due to package size constraints the data are stored externally (and a working internet connection is required to use the package).

The data is current to February 2021 (last update was in January 2020). Downloaded size of high resolution shapefile is 45 MB, size of the low res object is 2 MB.

Value

sf data frame with 2 671 rows of 2 variables + geometry

PSC ZIP Code as string in format NNNNN.

NAZ_POSTA Responsible Post Office

Source

ČSÚ https://www.czso.cz/csuz/czso/podminky_pro_vyuzivani_a_dalsi_zverejnovani_statistickyh_u_daju_csu
Examples

library(sf)
library(dplyr)

# residence of the Czech Prime Minister
kramarova_vila <- RCzechia::geocode("Gogolova 212, Praha 1")

# ZIP code of the PM residence
kramarova_vila %>%
  st_join(RCzechia::zip_codes("low"), left = FALSE) %>%
pull(PSC)
Index

casti, 3
chr uzemi, 4

geocode, 5
KFME_grid, 6
kraje, 7
lesy, 8
obce_body, 8
obce_polygony, 9
okresy, 10
orp_polygony, 12
plochy, 12
RCzechia-package, 2
reky, 13
republika, 14
revgeo, 15
senat_obvody, 16
silnice, 17
volebni okrsky, 18
vyskopis, 19
zeleznice, 20
zip_codes, 21