

Package: educabR (via r-universe)

October 29, 2026

Title Download and Process Brazilian Education Data from INEP

Version 1.2.0.9000

Description Download and process public education data from INEP (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira). Provides functions to access microdata from the School Census (Censo Escolar), ENEM (Exame Nacional do Ensino Médio), SAEB (Sistema de Avaliação da Educação Básica), Higher Education Census (Censo da Educação Superior), ENADE (Exame Nacional de Desempenho dos Estudantes), ENCCEJA (Exame Nacional para Certificação de Competências de Jovens e Adultos), IDD (Indicador de Diferença entre os Desempenhos Observado e Esperado), CPC (Conceito Preliminar de Curso), IGC (Índice Geral de Cursos), CAPES graduate education data, FUNDEB (Fundo de Manutenção e Desenvolvimento da Educação Básica), IDEB (Índice de Desenvolvimento da Educação Básica), and other educational datasets. Returns data in tidy format ready for analysis. Data source: INEP Open Data Portal
<<https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos>>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

URL <https://github.com/SidneyBissoli/educabR>,
<https://sidneybissoli.github.io/educabR/>

BugReports <https://github.com/SidneyBissoli/educabR/issues>

Depends R (>= 4.1.0)

Imports cli, dplyr, httr2, lifecycle, purrr, readr, rlang, stringi, stringr, tidyr, tools

Suggests ggplot2, jsonlite, kableExtra, knitr, readxl, rmarkdown, systemfonts, testthat (>= 3.0.0), textshaping, tibble, withr

Config/testthat/edition 3

VignetteBuilder knitr

Config/roxygen2/version 8.1.1

Config/pak/sysreqs libicu-dev libssl-dev libx11-dev

Repository <https://sidneybissoli.r-universe.dev>

Date/Publication 2026-10-29 15:40:29 UTC

RemoteUrl <https://github.com/sidneybissoli/educabr>

RemoteRef HEAD

RemoteSha da9fe0184ac83b6198453228b2d2b13a72377663

Contents

available_years	3
clear_cache	3
enem_summary	4
get_cache_dir	5
get_capes	5
get_censo_escolar	7
get_censo_superior	9
get_cpc	11
get_enade	12
get_encceja	13
get_enem	15
get_enem_escola	16
get_enem_itens	18
get_fundeb_distribution	18
get_fundeb_enrollment	20
get_idd	22
get_ideb	23
get_ideb_series	26
get_igc	27
get_saeb	28
list_cache	30
list_censo_files	31
list_censo_superior_files	32
list_ideb_available	32
set_cache_dir	33

Index	34
--------------	-----------

available_years	<i>Check available years for a dataset</i>
-----------------	--

Description

Returns the years available for a given dataset. On the first call in a session, queries the data source to discover which years are actually available (requires internet). Results are cached for the session. Falls back to a known list if discovery fails.

Usage

```
available_years(dataset)
```

Arguments

dataset	The dataset name.
---------	-------------------

Value

An integer vector of available years.

Examples

```
## Not run:
available_years("enem")
available_years("enade")
available_years("fundebr_enrollment")

## End(Not run)
```

clear_cache	<i>Clear the educabR cache</i>
-------------	--------------------------------

Description

Removes all cached files from the educabR cache directory.

Usage

```
clear_cache(dataset = NULL)
```

Arguments

dataset	Optional. A character string specifying which dataset cache to clear. If NULL, clears all caches.
---------	---

Value

Invisibly returns TRUE if successful.

See Also

Other cache functions: [get_cache_dir\(\)](#), [list_cache\(\)](#), [set_cache_dir\(\)](#)

Examples

```
## Not run:  
# clear all cached data  
clear_cache()  
  
# clear only ENEM cache  
clear_cache("enem")  
  
## End(Not run)
```

enem_summary

Summary statistics for ENEM scores

Description

Calculates summary statistics for ENEM scores, optionally grouped by demographic variables.

Usage

```
enem_summary(data, by = NULL)
```

Arguments

data	A tibble with ENEM data (from get_enem()).
by	Optional grouping variable(s) as character vector.

Value

A tibble with summary statistics for each score area.

See Also

Other ENEM functions: [get_enem\(\)](#), [get_enem_escola\(\)](#), [get_enem_itens\(\)](#)

Examples

```
## Not run:
enem <- get_enem(2023, n_max = 10000)

# overall summary
enem_summary(enem)

# summary by sex
enem_summary(enem, by = "tp_sexo")

## End(Not run)
```

get_cache_dir	<i>Get the current cache directory</i>
---------------	--

Description

Returns the current cache directory used by educabR.

Usage

```
get_cache_dir()
```

Value

A character string with the path to the cache directory.

See Also

Other cache functions: [clear_cache\(\)](#), [list_cache\(\)](#), [set_cache_dir\(\)](#)

Examples

```
get_cache_dir()
```

get_capes	<i>Get CAPES graduate education data</i>
-----------	--

Description

Downloads and processes data from CAPES (Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior) on Brazilian graduate programs (stricto sensu). Data is retrieved from the CAPES Open Data Portal via the CKAN API.

Usage

```
get_capes(
  year,
  type = c("programas", "discentes", "docentes", "cursos", "catalogo"),
  n_max = Inf,
  keep_file = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the data (2013-2024).
type	The type of data to download. One of: <ul style="list-style-type: none"> • "programas": Graduate programs • "discentes": Students • "docentes": Faculty • "cursos": Courses • "catalogo": Theses and dissertations catalog
n_max	Maximum number of rows to read. Default is Inf (all rows).
keep_file	Logical. If TRUE, keeps the downloaded file in cache. Default is TRUE.
quiet	Logical. If TRUE, suppresses progress messages.

Details

CAPES is the federal agency responsible for evaluating and regulating graduate programs in Brazil. The data covers stricto sensu programs (master's and doctoral).

The data types include:

- **programas**: Program identifiers, area, evaluation scores
- **discentes**: Student enrollment, demographics, scholarship status
- **docentes**: Faculty information, qualifications, employment
- **cursos**: Course details, modality, start dates
- **catalogo**: Catalog of theses and dissertations

Important notes:

- Data is sourced from the CAPES Open Data Portal (CKAN), not INEP.
- Files are large CSV files. Downloading may take several minutes.
- Column names are standardized to lowercase with underscores.
- Internet connection is required to discover download URLs via the CKAN API before downloading.

Value

A tibble with CAPES data in tidy format.

Data source

<https://dadosabertos.capes.gov.br>

Examples

```
## Not run:
# get graduate programs for 2023
programas <- get_capes(2023, type = "programas")

# get student data for 2022 with limited rows
discentes <- get_capes(2022, type = "discentes", n_max = 1000)

## End(Not run)
```

get_censo_escolar	<i>Get School Census (Censo Escolar) data</i>
-------------------	---

Description

Downloads and processes microdata from the Brazilian School Census (Censo Escolar), conducted annually by INEP. Returns school-level data with information about infrastructure, location, and administrative details.

Usage

```
get_censo_escolar(
  year,
  file = NULL,
  uf = NULL,
  n_max = Inf,
  keep_zip = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the census (1995-2025).
file	Optional. Name (or partial name) of a specific CSV file to load. By default, loads the main school data file. Use <code>list_censo_files()</code> to see available files for a given year. <ul style="list-style-type: none">• 1995-2006: Multiple legacy files (e.g. "EDUCPROF", "DADOSCURSO").• 2007-2024: Single file with all data (escola, matrícula, docente, turma).• 2025+: Data split into separate tables. Use file to select: "Escola" (default), "Matricula", "Docente", "Turma", "Gestor", "Curso_Tecnico". Non-escola tables lack CO_UF, so the uf filter does not apply to them.
uf	Optional. Filter by state (UF code or abbreviation).

n_max	Maximum number of rows to read. Default is Inf (all rows).
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

The School Census is the main statistical survey on basic education in Brazil. It collects data from all public and private schools offering basic education (early childhood, elementary, and high school).

Important notes:

- The microdata contains one row per school (~217,000 schools in 2023).
- Column names are standardized to lowercase with underscores.
- Use the `uf` parameter to filter by state for faster processing.
- Older years (1995-2006) contain multiple CSV files with different data. Use `list_censo_files()` to discover available files, then pass the desired file name to the `file` parameter.

Value

A tibble with school data in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/censo-escolar>

See Also

Other School Census functions: `list_censo_files()`

Examples

```
## Not run:
# get schools data for 2023
escolas <- get_censo_escolar(2023)

# get schools from Sao Paulo state only
escolas_sp <- get_censo_escolar(2023, uf = "SP")

# read only first 1000 rows for exploration
escolas_sample <- get_censo_escolar(2023, n_max = 1000)

# list available files for an older year
list_censo_files(1995)
# [1] "CENSOESC_1995.CSV" "DADOS_DESP_1995.CSV" "DADOSCURSO_1995.CSV"

# load a specific file from an older year
cursos <- get_censo_escolar(1995, file = "DADOSCURSO")

# 2025: data is split into separate tables
```

```
list_censo_files(2025)
escolas_2025 <- get_censo_escolar(2025)
matriculas_2025 <- get_censo_escolar(2025, file = "Matricula")
docentes_2025 <- get_censo_escolar(2025, file = "Docente")
turmas_2025 <- get_censo_escolar(2025, file = "Turma")

## End(Not run)
```

get_censo_superior	<i>Get Higher Education Census (Censo da Educação Superior) data</i>
--------------------	--

Description

Downloads and processes microdata from the Brazilian Higher Education Census (Censo da Educação Superior), conducted annually by INEP. Returns data on institutions, courses, students, or faculty.

Usage

```
get_censo_superior(
  year,
  type = c("ies", "cursos", "alunos", "docentes"),
  uf = NULL,
  n_max = Inf,
  keep_zip = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the census (2009-2024).
type	Type of data to load. Options: <ul style="list-style-type: none"> • "ies": Higher education institutions (default) • "cursos": Undergraduate courses • "alunos": Student enrollment • "docentes": Faculty/professors
uf	Optional. Filter by state (UF code or abbreviation).
n_max	Maximum number of rows to read. Default is Inf (all rows). Consider using a smaller value for exploration.
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

The Higher Education Census is the most comprehensive statistical survey on higher education institutions (HEIs) in Brazil. It collects data from all HEIs offering undergraduate and graduate programs.

Data types:

- "ies": One row per institution — administrative data, location, academic organization, funding type.
- "cursos": One row per undergraduate course — area of study, modality (in-person/distance), enrollment counts.
- "alunos": One row per student enrollment — demographics, program, admission type, enrollment status.
- "docentes": One row per faculty member — education level, employment type, teaching regime.

Important notes:

- Student files ("alunos") can be very large (several GB). Use `n_max` to read a sample first.
- Column names are standardized to lowercase with underscores.
- Use the `uf` parameter to filter by state for faster processing.

Value

A tibble with Higher Education Census microdata in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/censo-da-educacao-superior>

See Also

Other Higher Education Census functions: [list_censo_superior_files\(\)](#)

Examples

```
## Not run:
# get institution data for 2023
ies <- get_censo_superior(2023)

# get course data for Sao Paulo
cursos_sp <- get_censo_superior(2023, type = "cursos", uf = "SP")

# get a sample of student data
alunos <- get_censo_superior(2023, type = "alunos", n_max = 10000)

# get faculty data
docentes <- get_censo_superior(2023, type = "docentes")

## End(Not run)
```

get_cpc

*Get CPC (Conceito Preliminar de Curso) data***Description**

Downloads and processes CPC data from INEP. The CPC is a quality indicator for undergraduate courses in Brazil, composed of ENADE scores, IDD, faculty qualifications, pedagogical resources, and other institutional factors.

Usage

```
get_cpc(year, n_max = Inf, keep_file = TRUE, quiet = FALSE)
```

Arguments

year	The year of the indicator (2007-2019, 2021-2023). Note: there is no 2020 edition. Years 2004-2006 used a different indicator ("Conceito Enade").
n_max	Maximum number of rows to read. Default is Inf (all rows).
keep_file	Logical. If TRUE, keeps the downloaded file in cache. Default is TRUE.
quiet	Logical. If TRUE, suppresses progress messages.

Details

CPC is calculated by INEP as part of the higher education quality assessment system (SINAES). It serves as a preliminary indicator used to determine which courses require on-site evaluation.

The data includes:

- Course and institution identifiers
- CPC scores (continuous and categorical/faixa)
- Component scores (ENADE, IDD, faculty, infrastructure, etc.)
- Number of students evaluated

Important notes:

- CPC follows ENADE's rotating cycle of course areas, so each year covers a specific set of fields.
- There is no 2020 edition (COVID-19 suspension).
- Column names are standardized to lowercase with underscores.
- Files are in Excel format (xls/xlsx), not CSV.

Value

A tibble with CPC data in tidy format.

Data dictionary

For detailed information about variables, see INEP’s documentation: <https://www.gov.br/inep/pt-br/areas-de-atuacao/pesquisas-estatisticas-e-indicadores/indicadores-de-qualidade-da-educacao-s>

See Also

Other CPC/IGC functions: `get_igc()`

Examples

```
## Not run:
# get CPC data for 2023
cpc <- get_cpc(2023)

# get CPC data for 2021 with limited rows
cpc_2021 <- get_cpc(2021, n_max = 1000)

## End(Not run)
```

get_enade	<i>Get ENADE (Exame Nacional de Desempenho dos Estudantes) data</i>
-----------	---

Description

Downloads and processes microdata from ENADE, the Brazilian National Student Performance Exam. ENADE evaluates the performance of undergraduate students in higher education.

Usage

```
get_enade(year, n_max = Inf, keep_zip = TRUE, quiet = FALSE)
```

Arguments

year	The year of the exam (2004-2024).
n_max	Maximum number of rows to read. Default is Inf (all rows). Consider using a smaller value for exploration.
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

ENADE is conducted annually by INEP and evaluates undergraduate students nearing the end of their programs. Each year, a different set of course areas is assessed on a rotating cycle (typically every 3 years per area).

The microdata includes:

- Student performance scores (general and specific knowledge)

- Socioeconomic questionnaire responses
- Course and institution identifiers

Important notes:

- ENADE files can be large (several hundred MB for recent years).
- Use `n_max` to read a sample first for exploration.
- Column names are standardized to lowercase with underscores.
- Not all course areas are assessed every year due to the rotating cycle.

Value

A tibble with ENADE microdata in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/enade>

Examples

```
## Not run:
# get ENADE data for 2023
enade <- get_enade(2023, n_max = 10000)

# get full dataset for 2021
enade_2021 <- get_enade(2021)

## End(Not run)
```

get_encceja

Get ENCCEJA (Exame Nacional para Certificação de Competências de Jovens e Adultos) data

Description

Downloads and processes microdata from ENCCEJA, the Brazilian National Exam for Youth and Adult Education Certification. ENCCEJA assesses competencies of young people and adults who did not complete basic education at the regular age.

Usage

```
get_encceja(
  year,
  type = c("regular", "ppl"),
  n_max = Inf,
  keep_zip = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the exam (2014, 2017-2020, 2022-2025; INEP published no microdata for 2015, 2016 and 2021).
type	Which participants to load. INEP ships two microdata files per edition: <ul style="list-style-type: none"> • "regular": the national regular exam (REG_NAC / REGULAR file), with almost all participants (default) • "ppl": the exam applied to people deprived of liberty (PPL_NAC / PPL file)
n_max	Maximum number of rows to read. Default is Inf (all rows). Consider using a smaller value for exploration.
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

ENCCEJA is conducted by INEP and provides certification for elementary and high school equivalency for youth and adults (EJA). The exam covers four knowledge areas:

- Natural Sciences (Ciências Naturais)
- Mathematics (Matemática)
- Portuguese Language (Língua Portuguesa)
- Social Sciences (Ciências Humanas)

Important notes:

- ENCCEJA files can be large (several hundred MB).
- Use n_max to read a sample first for exploration.
- Column names are standardized to lowercase with underscores.

Value

A tibble with ENCCEJA microdata in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/encceja>

Examples

```
## Not run:
# get ENCCEJA data for 2023
encceja <- get_encceja(2023, n_max = 10000)

# get full dataset for 2022
encceja_2022 <- get_encceja(2022)

# participants deprived of liberty (PPL)
```

```
encceja_ppl <- get_encceja(2023, type = "ppl")

## End(Not run)
```

get_enem	<i>Get ENEM (Exame Nacional do Ensino Médio) data</i>
----------	---

Description

Downloads and processes microdata from ENEM, the Brazilian National High School Exam. ENEM is used for university admissions and as a high school equivalency exam.

Usage

```
get_enem(
  year,
  type = "participantes",
  n_max = Inf,
  keep_zip = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the exam (1998-2025).
type	Type of data to load. Only used for ENEM 2024+, where microdata is split into separate files. Options: "participantes" (demographics and socioeconomic data, default), "resultados" (scores). Ignored for years before 2024 (single file with all data).
n_max	Maximum number of rows to read. Default is Inf (all rows). Consider using a smaller value for exploration, as ENEM files contain millions of rows.
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

ENEM is conducted annually by INEP and is the largest exam in Brazil, with millions of participants. The microdata includes:

- Participant demographics (age, sex, race, etc.)
- Socioeconomic questionnaire responses
- Scores for each test area
- Essay scores
- School information (when applicable)

Important notes:

- ENEM files are very large (several GB when extracted).
- Use `n_max` to read a sample first for exploration.
- Column names are standardized to lowercase with underscores.
- Score variables start with `nu_nota_` prefix.
- From 2024 onwards, INEP split the microdata into separate files. Use the `type` parameter to choose which file to load.

Value

A tibble with the ENEM microdata in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/microdados/enem>

See Also

Other ENEM functions: [enem_summary\(\)](#), [get_enem_escola\(\)](#), [get_enem_itens\(\)](#)

Examples

```
## Not run:
# get a sample of 10000 rows for exploration
enem_sample <- get_enem(2023, n_max = 10000)

# get full data (warning: large file)
enem_2023 <- get_enem(2023)

# ENEM 2024+: choose data type
participantes <- get_enem(2024, type = "participantes", n_max = 1000)
resultados <- get_enem(2024, type = "resultados", n_max = 1000)

## End(Not run)
```

get_enem_escola

Get ENEM por Escola (ENEM by School) data

Description

Downloads and processes ENEM results aggregated by school. This dataset contains average ENEM scores, participation rates, and other indicators for each school in Brazil.

Usage

```
get_enem_escola(n_max = Inf, keep_zip = TRUE, quiet = FALSE)
```

Arguments

n_max	Maximum number of rows to read. Default is Inf (all rows).
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

ENEM por Escola is a single bundled dataset covering years 2005 to 2015. It was discontinued by INEP after 2015 and no per-year files exist.

The data includes:

- School identification (code, name, municipality, state)
- Average ENEM scores by knowledge area
- Number of participants and participation rates
- School-level indicators

Important notes:

- This is a single file covering all years (2005-2015), not per-year.
- Column names are standardized to lowercase with underscores.
- Data was discontinued after 2015.

Value

A tibble with ENEM by School data in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/enem-por-escola>

See Also

Other ENEM functions: [enem_summary\(\)](#), [get_enem\(\)](#), [get_enem_itens\(\)](#)

Examples

```
## Not run:  
# get all ENEM by School data (2005-2015)  
enem_escola <- get_enem_escola()  
  
# read only first 1000 rows for exploration  
enem_escola_sample <- get_enem_escola(n_max = 1000)  
  
## End(Not run)
```

get_enem_itens	<i>Get ENEM item response data</i>
----------------	------------------------------------

Description

Downloads and processes ENEM item response (gabarito) data, which contains detailed information about each question.

Usage

```
get_enem_itens(year, n_max = Inf, keep_zip = TRUE, quiet = FALSE)
```

Arguments

year	The year of the exam (1998-2025).
n_max	Maximum number of rows to read.
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Value

A tibble with item response data.

See Also

Other ENEM functions: [enem_summary\(\)](#), [get_enem\(\)](#), [get_enem_escola\(\)](#)

Examples

```
## Not run:
# get item data for 2023
itens <- get_enem_itens(2023)

## End(Not run)
```

get_fundeb_distribution	<i>Get FUNDEB distribution data</i>
-------------------------	-------------------------------------

Description

Downloads and processes FUNDEB resource distribution data from STN (Secretaria do Tesouro Nacional). Each year's Excel file contains multiple sheets with monthly transfer data by state/municipality, broken down by funding source.

Usage

```
get_fundeb_distribution(
  year,
  uf = NULL,
  source = NULL,
  destination = NULL,
  n_max = Inf,
  keep_file = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the data (2007-2026).
uf	Optional. A UF code (e.g., "SP", "RJ") to filter by state. Default is NULL (all states).
source	Optional. The funding source to filter by. One of: "FPE", "FPM", "IPI", "ITR", "VAAF", "VAAT", "VAAR", "ICMS", "IPVA", "ITCMD". Default is NULL (all sources).
destination	Optional. The transfer destination. One of: <ul style="list-style-type: none"> "uf": Transfers to states and the Federal District "municipio": Transfers to municipalities Default is NULL (both).
n_max	Maximum number of rows to return. Default is Inf (all rows).
keep_file	Logical. If TRUE, keeps the downloaded file in cache. Default is TRUE.
quiet	Logical. If TRUE, suppresses progress messages.

Details

FUNDEB (Fundo de Manutencao e Desenvolvimento da Educacao Basica e de Valorizacao dos Profissionais da Educacao) is the main funding mechanism for basic education in Brazil.

Each Excel file from STN contains ~20 data sheets named with a prefix indicating the destination (E_ for states, M_ for municipalities) and a suffix indicating the funding source (e.g., E_FPE, M_ICMS). Each sheet contains two tables: the main FUNDEB transfers and a FUNDEB adjustment table.

Important notes:

- Data is sourced from STN (Tesouro Nacional), not INEP.
- Files are in Excel format (XLS) — requires the readxl package.
- Column names are standardized to lowercase with underscores.
- Summary sheets (Resumo, Total, etc.) are automatically excluded.

Value

A tibble in tidy (long) format with columns:

estados State name
uf State code (UF)
mes_ano Date (last day of the month)
origem Funding source (FPE, FPM, ICMS, etc.)
destino Transfer destination ("UF" or "Município")
tabela Table type ("Fundeb" or "Ajuste Fundeb")
valor Transfer amount in BRL (numeric)

Data source

<https://www.tesourotransparente.gov.br>

See Also

Other FUNDEB functions: [get_fundeb_enrollment\(\)](#)

Examples

```
## Not run:
# get all FUNDEB distribution data for 2023
dist_2023 <- get_fundeb_distribution(2023)

# get only FPE transfers to states
fpe_estados <- get_fundeb_distribution(2023, source = "FPE",
                                     destination = "uf")

# get data for Sao Paulo only
sp <- get_fundeb_distribution(2023, uf = "SP")

## End(Not run)
```

get_fundeb_enrollment *Get FUNDEB enrollment data*

Description

Downloads and processes FUNDEB enrollment data from FNDE's OData API. These are the enrollment counts considered for FUNDEB funding calculation.

Usage

```
get_fundeb_enrollment(
  year,
  uf = NULL,
  n_max = Inf,
  keep_file = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the data (2007-2026).
uf	Optional. A UF code (e.g., "SP", "RJ") to filter by state. The complete dataset is always cached first, then filtered locally. Default is NULL (all states).
n_max	Maximum number of rows to read. Default is Inf (all rows).
keep_file	Logical. If TRUE, caches the API result as a local CSV file. Default is TRUE.
quiet	Logical. If TRUE, suppresses progress messages.

Details

Enrollment data comes from FNDE (Fundo Nacional de Desenvolvimento da Educacao) via its OData API. It includes the number of enrollments considered for FUNDEB funding, broken down by state, municipality, education type, school network, class type, and location.

Important notes:

- Data is sourced from FNDE, not INEP.
- Requires the `jsonlite` package.
- Results are cached locally as CSV after first download.
- Column names are standardized to lowercase with underscores.
- When `uf` is used with a cached file, filtering is done locally.

Value

A tibble with columns:

ano_censo	Census year
uf	State code (UF)
municipio	Municipality name
tipo_rede_educacao	Education network type
descricao_tipo_educacao	Education type description
descricao_tipo_ensino	Teaching type description
descricao_tipo_turma	Class type description
descricao_tipo_carga_horaria	Class hours type description
descricao_tipo_localizacao	Location type description
qtd_matricula	Number of enrollments

Data source

FNDE: <https://www.fnde.gov.br>

See Also

Other FUNDEB functions: [get_fundeb_distribution\(\)](#)

Examples

```
## Not run:
# get FUNDEB enrollment data for 2023
mat_2023 <- get_fundeb_enrollment(2023)

# get enrollment data for Sao Paulo only
mat_sp <- get_fundeb_enrollment(2023, uf = "SP")

# get enrollment data with limited rows
mat_sample <- get_fundeb_enrollment(2023, n_max = 1000)

## End(Not run)
```

get_idd

Get IDD (Indicador de Diferença entre os Desempenhos Observado e Esperado) data

Description

Downloads and processes microdata from IDD, an indicator that measures the value added by an undergraduate course to student performance. It compares ENADE scores with expected performance based on students' prior achievement (ENEM scores at admission).

Usage

```
get_idd(year, n_max = Inf, keep_zip = TRUE, quiet = FALSE)
```

Arguments

year	The year of the indicator (2014-2019, 2021-2023). Note: there is no 2020 edition.
n_max	Maximum number of rows to read. Default is Inf (all rows).
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

IDD is calculated by INEP as part of the higher education quality assessment system. It complements ENADE by isolating the contribution of the course itself to student learning, controlling for student input quality.

The data includes:

- Course and institution identifiers
- IDD scores (continuous and categorical)
- Number of students considered in the calculation
- Related ENADE and ENEM metrics

Important notes:

- IDD is published alongside ENADE results, following the same rotating cycle of course areas.
- Column names are standardized to lowercase with underscores.
- Not all courses have IDD values (minimum sample requirements apply).

Value

A tibble with IDD data in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/idd>

Examples

```
## Not run:  
# get IDD data for 2023  
idd <- get_idd(2023)  
  
# get IDD data for 2021 with limited rows  
idd_2021 <- get_idd(2021, n_max = 1000)  
  
## End(Not run)
```

get_ideb

Get IDEB (Índice de Desenvolvimento da Educação Básica) data

Description

Downloads and processes IDEB data from INEP in tidy (long) format. IDEB is the main indicator of education quality in Brazil, combining student performance (from SAEB) with grade promotion rates.

Usage

```
get_ideb(level, stage, metric, year = NULL, quiet = FALSE)
```

Arguments

level	<p>The geographic level. Required.</p> <ul style="list-style-type: none"> • "escola": School level • "municipio": Municipality level • "estado": State level • "regiao": Region level (Norte, Nordeste, Sudeste, Sul, Centro-Oeste) • "brasil": National level
stage	<p>The education stage. Required.</p> <ul style="list-style-type: none"> • "anos_iniciais": Early elementary (1st-5th grade) • "anos_finais": Late elementary (6th-9th grade) • "ensino_medio": High school • "ensino_medio_integrado": High school <i>including</i> students of technical-professional education integrated to high school (EPT integrada), a cut first published with IDEB 2025. Only available for level = "brasil" or "estado", and has no metric = "meta".
metric	<p>The type of data to return. Required.</p> <ul style="list-style-type: none"> • "indicador": IDEB components (rendimento, nota padronizada, ideb) • "aprovacao": Approval rates by school year • "nota": SAEB scores by subject (math/portuguese) • "meta": IDEB targets/projections (not published for stage = "ensino_medio_integrado")
year	<p>Optional. Integer vector of IDEB editions to filter (e.g., c(2021, 2023, 2025)). NULL returns all available editions. Note: official targets (metric = "meta") are only defined through 2021.</p>
quiet	<p>Logical. If TRUE, suppresses progress messages.</p>

Details

IDEB is calculated every two years since 2005 based on:

- **Learning:** Average scores in Portuguese and Mathematics from SAEB
- **Flow:** Grade promotion rate (inverse of repetition/dropout)

The index ranges from 0 to 10. Brazil's national goal is to reach 6.0 by 2022 (the level of developed countries in PISA).

The function always downloads the most recent IDEB file available from INEP, which contains the full historical series (2005-2025).

stage = "ensino_medio_integrado" reads the "Ensino médio mais educação profissional técnica integrada" spreadsheets first published with IDEB 2025. Unlike the traditional high school IDEB — which excludes students of integrated technical-professional education (EPT integrada) from the SAEB performance component to preserve historical comparability — this cut includes them. INEP publishes it only at the national and state levels (state networks only), covering editions 2017-2025 for states and 2019-2025 for Brazil, with no official targets.

Value

A tibble in tidy (long) format. Columns vary by level and metric:

ID columns (vary by level):

- escola: uf_sigla, municipio_codigo, municipio_nome, escola_id, escola_nome, rede
- municipio: uf_sigla, municipio_codigo, municipio_nome, rede
- brasil: rede
- estado: uf_nome, uf_sigla, rede
- regioao: regioao, rede

Value columns (vary by metric):

- indicador: ano, indicador, valor
- aprovacao: ano, ano_escolar or serie (ensino_medio), taxa_aprovacao
- nota: ano, disciplina, nota
- meta: ano, meta

Data source

Official IDEB portal: <https://www.gov.br/inep/pt-br/areas-de-atuacao/pesquisas-estatisticas-e-indicadores/ideb>

See Also

Other IDEB functions: [get_ideb_series\(\)](#), [list_ideb_available\(\)](#)

Examples

```
## Not run:
# school-level IDEB indicators for early elementary
ideb <- get_ideb("escola", "anos_iniciais", "indicador")

# municipality-level approval rates, only 2023 and 2025
aprov <- get_ideb("municipio", "anos_finais", "aprovacao", year = c(2023, 2025))

# national IDEB targets
metas <- get_ideb("brasil", "ensino_medio", "meta")

# state-level SAEB scores
notas <- get_ideb("estado", "anos_iniciais", "nota")

# region-level IDEB indicators
regioes <- get_ideb("regiao", "anos_finais", "indicador")

# state-level IDEB including EPT integrada students (2025+ cut)
emi <- get_ideb("estado", "ensino_medio_integrado", "indicador")

## End(Not run)
```

get_ideb_series	<i>Get IDEB historical series</i>
-----------------	-----------------------------------

Description

[Deprecated]

`get_ideb_series()` is deprecated because the new `get_ideb()` already returns data in long format with all historical editions. Use `get_ideb(level, stage, metric)` instead.

Usage

```
get_ideb_series(  
  years = NULL,  
  level = c("escola", "municipio"),  
  stage = c("anos_iniciais", "anos_finais", "ensino_medio"),  
  uf = NULL,  
  quiet = FALSE  
)
```

Arguments

years	Vector of years to include (default: all available).
level	The aggregation level.
stage	The education stage.
uf	Optional. Filter by state.
quiet	Logical. If TRUE, suppresses progress messages.

Value

A tibble with IDEB data.

See Also

Other IDEB functions: [get_ideb\(\)](#), [list_ideb_available\(\)](#)

Examples

```
## Not run:  
# deprecated: use get_ideb() instead  
ideb <- get_ideb("municipio", "anos_iniciais", "indicador")  
  
## End(Not run)
```

`get_igc`*Get IGC (Índice Geral de Cursos) data*

Description

Downloads and processes IGC data from INEP. The IGC is a quality indicator for higher education institutions in Brazil, calculated as a weighted average of CPC scores across all evaluated courses plus CAPES scores for graduate programs.

Usage

```
get_igc(year, n_max = Inf, keep_file = TRUE, quiet = FALSE)
```

Arguments

<code>year</code>	The year of the indicator (2007-2019, 2021-2023). Note: there is no 2020 edition. Years 2004-2006 used a different indicator ("Conceito Enade").
<code>n_max</code>	Maximum number of rows to read. Default is Inf (all rows).
<code>keep_file</code>	Logical. If TRUE, keeps the downloaded file in cache. Default is TRUE.
<code>quiet</code>	Logical. If TRUE, suppresses progress messages.

Details

IGC is calculated by INEP as part of the higher education quality assessment system (SINAES). It provides an overall quality measure for institutions, considering both undergraduate and graduate programs.

The data includes:

- Institution identifiers (code, name, organization type)
- IGC scores (continuous and categorical/faixa)
- Number of courses and students considered
- Component breakdown (undergraduate CPC average, graduate CAPES scores)

Important notes:

- IGC is published annually based on the last three ENADE cycles.
- There is no 2020 edition (COVID-19 suspension).
- Column names are standardized to lowercase with underscores.
- Files are in Excel format (xls/xlsx), except 2007 which is 7z.

Value

A tibble with IGC data in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/areas-de-atuacao/pesquisas-estatisticas-e-indicadores/indicadores-de-qualidade-da-educacao-s>

See Also

Other CPC/IGC functions: [get_cpc\(\)](#)

Examples

```
## Not run:
# get IGC data for 2023
igc <- get_igc(2023)

# get IGC data for 2021 with limited rows
igc_2021 <- get_igc(2021, n_max = 1000)

## End(Not run)
```

get_saeb

Get SAEB (Sistema de Avaliação da Educação Básica) data

Description

Downloads and processes microdata from SAEB, the Brazilian Basic Education Assessment System. SAEB evaluates educational quality through student performance assessments in Portuguese and Mathematics.

Usage

```
get_saeb(
  year,
  type = c("aluno", "escola", "diretor", "professor"),
  level = c("fundamental_medio", "educacao_infantil"),
  serie = NULL,
  n_max = Inf,
  keep_zip = TRUE,
  quiet = FALSE
)
```

Arguments

year	The year of the assessment (2011, 2013, 2015, 2017, 2019, 2021, 2023).
type	Type of data to load. Options: <ul style="list-style-type: none"> • "aluno": Student results (default) • "escola": School questionnaire • "diretor": Principal questionnaire

	<ul style="list-style-type: none"> • "professor": Teacher questionnaire
level	<p>For 2021 only, SAEB was split into two files:</p> <ul style="list-style-type: none"> • "fundamental_medio": Elementary and High School (default) • "educacao_infantil": Early Childhood Education Ignored for other years.
serie	<p>For type = "aluno", the grade to load. Since 2013 INEP publishes one student file per grade (TS_ALUNO_<serie>.csv), so get_saeb() reads one grade per call. Use INEP's codes, in any case:</p> <ul style="list-style-type: none"> • "2ef": 2nd grade of elementary school (2019, 2021, 2023) • "5ef": 5th grade of elementary school (2013-2023) • "9ef": 9th grade of elementary school (2013-2023) • "3em": 3rd grade of high school (2013, 2015) • "3em_ag", "3em_esc": 3rd grade of high school, aggregated and school samples (2017) • "34em": 3rd and 4th grades of high school (2019, 2021, 2023) <p>The numbers 2, 5 and 9 are accepted as shortcuts for "2ef", "5ef" and "9ef". Required when the edition has more than one student file; the error message lists the grades available for the year. Ignored for 2011 (a single results file covers all grades) and for the 2021 early childhood file.</p>
n_max	Maximum number of rows to read. Default is Inf (all rows). Consider using a smaller value for exploration.
keep_zip	Logical. If TRUE, keeps the downloaded ZIP file in cache.
quiet	Logical. If TRUE, suppresses progress messages.

Details

SAEB is conducted biennially by INEP and assesses students in grades 5 and 9 of elementary school, and grade 3 of high school. The data includes:

- Student performance scores in Portuguese and Mathematics
- School infrastructure and management questionnaires
- Teacher and principal profiles

Important notes:

- SAEB files can be large (several hundred MB).
- Use n_max to read a sample first for exploration.
- Column names are standardized to lowercase with underscores.
- In 2021, INEP split SAEB into two separate downloads (elementary/high school and early childhood). Use the level parameter to choose.
- Student files are split by grade from 2013 on; choose one with serie. The 5th and 9th grade files are about 1 GB of CSV each. To combine grades, call once per grade, e.g. `purrr::map(c("5ef", "9ef"), function(s) get_saeb(2023, serie = s))`.
- For 2011, type = "aluno" reads TS_RESULTADO_ALUNO (proficiency results), which covers all grades in one file.

Value

A tibble with SAEB microdata in tidy format.

Data dictionary

For detailed information about variables, see INEP's documentation: <https://www.gov.br/inep/pt-br/aceso-a-informacao/dados-abertos/microdados/saeb>

Examples

```
## Not run:
# get 5th grade student results for 2023
saeb_5ef <- get_saeb(2023, serie = "5ef", n_max = 10000)

# 5th and 9th grades, one call per grade
saeb_ef <- purrr::map(c("5ef", "9ef"), function(s) get_saeb(2023, serie = s))

# get school questionnaire data
saeb_escola <- get_saeb(2023, type = "escola")

# SAEB 2021: early childhood education
saeb_infantil <- get_saeb(2021, level = "educacao_infantil", n_max = 1000)

## End(Not run)
```

list_cache

List cached files

Description

Lists all files currently in the educabR cache.

Usage

```
list_cache(dataset = NULL)
```

Arguments

dataset Optional. Filter by dataset name.

Value

A tibble with information about cached files.

See Also

Other cache functions: [clear_cache\(\)](#), [get_cache_dir\(\)](#), [set_cache_dir\(\)](#)

Examples

```
## Not run:  
list_cache()  
  
## End(Not run)
```

list_censo_files	<i>List available Censo Escolar files</i>
------------------	---

Description

Lists the data files available in a downloaded School Census. Use this to discover which files are available for a given year, then pass the desired file name to [get_censo_escolar\(\)](#)'s file parameter.

Usage

```
list_censo_files(year)
```

Arguments

year	The year of the census.
------	-------------------------

Value

A character vector of file names found.

See Also

Other School Census functions: [get_censo_escolar\(\)](#)

Examples

```
## Not run:  
# first download the data  
get_censo_escolar(1995)  
  
# then see what files are available  
list_censo_files(1995)  
# [1] "CENSOESC_1995.CSV" "DADOS_DESP_1995.CSV" "DADOSCURSO_1995.CSV"  
  
# load a specific file  
cursos <- get_censo_escolar(1995, file = "DADOSCURSO")  
  
## End(Not run)
```

`list_censo_superior_files`*List available Higher Education Census files*

Description

Lists the data files available in a downloaded Higher Education Census. Useful for exploring the contents of the ZIP file.

Usage

```
list_censo_superior_files(year)
```

Arguments

<code>year</code>	The year of the census.
-------------------	-------------------------

Value

A character vector of file names found.

See Also

Other Higher Education Census functions: [get_censo_superior\(\)](#)

Examples

```
## Not run:  
list_censo_superior_files(2023)  
  
## End(Not run)
```

`list_ideb_available` *List available IDEB data*

Description

Lists the IDEB data combinations available for download.

Usage

```
list_ideb_available()
```

Value

A tibble with available IDEB datasets (level, stage, metric).

See Also

Other IDEB functions: [get_ideb\(\)](#), [get_ideb_series\(\)](#)

Examples

```
list_ideb_available()
```

set_cache_dir

Set the cache directory for educabR

Description

Sets the directory where downloaded files will be cached. This avoids repeated downloads of the same data.

Usage

```
set_cache_dir(path = NULL, persistent = FALSE)
```

Arguments

path	A character string with the path to the cache directory. If NULL, uses a temporary directory (default).
persistent	Logical. If TRUE, the cache directory setting is saved to the user's R profile for future sessions.

Value

Invisibly returns the cache directory path.

See Also

Other cache functions: [clear_cache\(\)](#), [get_cache_dir\(\)](#), [list_cache\(\)](#)

Examples

```
## Not run:  
# set a custom cache directory (use tempdir() in examples)  
set_cache_dir(file.path(tempdir(), "educabR_cache"))  
  
## End(Not run)
```

Index

- * **CAPES functions**
 - get_capes, 5
 - * **CPC/IGC functions**
 - get_cpc, 11
 - get_igc, 27
 - * **ENADE functions**
 - get_enade, 12
 - * **ENCCEJA functions**
 - get_encceja, 13
 - * **ENEM functions**
 - enem_summary, 4
 - get_enem, 15
 - get_enem_escola, 16
 - get_enem_itens, 18
 - * **FUNDEB functions**
 - get_fundeb_distribution, 18
 - get_fundeb_enrollment, 20
 - * **Higher Education Census functions**
 - get_censo_superior, 9
 - list_censo_superior_files, 32
 - * **IDD functions**
 - get_idd, 22
 - * **IDEB functions**
 - get_ideb, 23
 - get_ideb_series, 26
 - list_ideb_available, 32
 - * **SAEB functions**
 - get_saeb, 28
 - * **School Census functions**
 - get_censo_escolar, 7
 - list_censo_files, 31
 - * **cache functions**
 - clear_cache, 3
 - get_cache_dir, 5
 - list_cache, 30
 - set_cache_dir, 33
- available_years, 3
- clear_cache, 3
- clear_cache(), 5, 30, 33
- enem_summary, 4
- enem_summary(), 16–18
- get_cache_dir, 5
- get_cache_dir(), 4, 30, 33
- get_capes, 5
- get_censo_escolar, 7
- get_censo_escolar(), 31
- get_censo_superior, 9
- get_censo_superior(), 32
- get_cpc, 11
- get_cpc(), 28
- get_enade, 12
- get_encceja, 13
- get_enem, 15
- get_enem(), 4, 17, 18
- get_enem_escola, 16
- get_enem_escola(), 4, 16, 18
- get_enem_itens, 18
- get_enem_itens(), 4, 16, 17
- get_fundeb_distribution, 18
- get_fundeb_distribution(), 22
- get_fundeb_enrollment, 20
- get_fundeb_enrollment(), 20
- get_idd, 22
- get_ideb, 23
- get_ideb(), 26, 33
- get_ideb_series, 26
- get_ideb_series(), 25, 33
- get_igc, 27
- get_igc(), 12
- get_saeb, 28
- list_cache, 30
- list_cache(), 4, 5, 33
- list_censo_files, 31
- list_censo_files(), 7, 8
- list_censo_superior_files, 32

`list_censo_superior_files()`, [10](#)

`list_ideb_available`, [32](#)

`list_ideb_available()`, [25](#), [26](#)

`set_cache_dir`, [33](#)

`set_cache_dir()`, [4](#), [5](#), [30](#)