

Package ‘BLSloadR’

February 14, 2026

Type Package

Title Download Time Series Data from the U.S. Bureau of Labor Statistics

Version 0.4

Description These functions provide a convenient interface for downloading data from the U.S. Bureau of Labor Statistics <<https://www.bls.gov>>. The functions in this package utilize flat files produced by the Bureau of Labor Statistics, which contain full series history. These files include employment, unemployment, wages, prices, industry and occupational data at a national, state, and sub-state level, depending on the series. Individual functions are included for those programs which have data available at the state level. The core functions provide direct access to the Current Employment Statistics (CES) <<https://www.bls.gov/ces/>>, Local Area Unemployment Statistics (LAUS) <<https://www.bls.gov/lau/>>, Occupational Employment and Wage Statistics (OEWS) <<https://www.bls.gov/oes/>> and Alternative Measures of Labor Underutilization (SALT) <<https://www.bls.gov/lau/stalt.htm>> data produced by the Bureau of Labor Statistics.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

Imports data.table (>= 1.16), httr (>= 1.4.7), dplyr (>= 1.1), stringr (>= 1.5), lubridate (>= 1.9), rvest (>= 1.0.4), readxl (>= 1.4.5), sf (>= 1.0), tidyselect (>= 1.2), tigris (>= 2.0), zoo (>= 1.8), htmltools, rstudioapi

RoxygenNote 7.3.3

Suggests ggplot2, knitr, mapgl, rmarkdown, tidyr, gt

VignetteBuilder knitr

URL <https://schmidtdetr.github.io/BLSloadR/>

Config/Needs/website rmarkdown

LazyData true

NeedsCompilation no

Author Nevada Department of Employment, Training, and Rehabilitation [cph],
David Schmidt [aut, cre],
Mark Rembert [aut]

Maintainer David Schmidt <deschmidt@detr.nv.gov>

Repository CRAN

Date/Publication 2026-02-13 23:40:02 UTC

Contents

area_lookup	3
bls_get_cache_dir	4
bls_overview	4
create_bls_object	5
display_in_console	6
display_in_popup	6
display_in_viewer	7
download_bls_files	7
fread_bls	8
get_bls_data	8
get_bls_diagnostics	9
get_bls_summary	9
get_ces	10
get_jolts	12
get_laus	14
get_national_ces	16
get_oews	19
get_oews_areas	21
get_qcew	22
get_salt	27
has_bls_issues	28
ind_lookup	29
list_ces_industries	30
list_ces_states	31
list_national_ces_options	31
load_bls_dataset	32
print_bls_warnings	34
show_ces_options	34
show_national_ces_options	35
smart_bls_download	35

Index

37

area_lookup	<i>Area Lookup Tables (QCEW)</i>
-------------	----------------------------------

Description

A data frame containing area codes, titles, and additional geographic information about valid areas for the Quarterly Census of Employment and Wages (QCEW).

This table is primarily used internally by 'BLSloadR' functions to add human-readable area titles or define valid area lookups.

Usage

```
data(area_lookup)
```

Format

A data frame with 4649 rows and 5 columns:

- `area_fips` - Character. The area FIPS code. When all numeric characters, it represents either a state or a county definition.
- `area_title` - Character. The description of the area code provided by the Bureau of Labor Statistics.
- `area_type` - Character. A description of the type of area defined. Values are National, State, County, County Unknown or Undefined, National Subgroup, Combined Statistical Area, Metropolitan Statistical Area, or Micropolitan Statistical Area.
- `stfips` - Character. For state or counties, the two-digit FIPS code of the associated state. For national areas, or those areas which may span multiple states the value is "00".
- `specified_region` - Either the two-character US Postal abbreviation for a state or group of states, or "No region" for other areas.

Details

Area codes are five characters long. When all numeric characters, this is a state-county FIPS, with statewide data using a "000" as the county FIPS. Other aggregations include various alphabetic characters to aid in the classification of different regions.

Source

<https://www.bls.gov/cew/classifications/areas/qcew-area-titles.htm>

Examples

```
# Load the lookup table
data(area_lookup)

# Find the area codes for all Statewide areas
state_codes <- area_lookup[area_lookup$area_type == "State", ]
```

```
# Get all Metropolitan Statistical Area codes including Arkansas.
ar_codes <- area_lookup |>
dplyr::filter(grepl("AR", specified_region) &
  area_type == "Metropolitan Statistical Area")
```

bls_get_cache_dir	<i>Get the current BLSloadR Cache Directory</i>
-------------------	---

Description

Displays the path currently being used for caching. This will check the 'BLS_CACHE_DIR' environment variable, falling back to the default system cache directory if the variable is not set.

Usage

```
bls_get_cache_dir()
```

Value

A character string of the cache path.

bls_overview	<i>Display BLS Dataset Overview</i>
--------------	-------------------------------------

Description

Fetches and displays the overview text file for a BLS dataset. This provides a convenient reference within the R environment without needing to manually find and review the text file on the BLS website.

Usage

```
bls_overview(
  series_id,
  display_method = "viewer",
  base_url = "https://download.bls.gov/pub/time.series"
)
```

Arguments

series_id	Character string. The BLS series identifier (e.g., "ln", "cu", "ap")
display_method	Character string. How to display the overview: "viewer" (default), "console", or "popup"
base_url	Character string. Base URL for BLS data (default uses official BLS site)

Value

Invisibly returns the text content. Function is called to use the viewer, console, or as a popup, depending on the 'display_method' argument.

Examples

```
# Display Average Price Data overview
bls_overview("ap")

# Display consumer price index overview
bls_overview("cu")

# Display in console instead of viewer
bls_overview("ap", display_method = "console")
```

create_bls_object	<i>Create a BLS data object with diagnostics</i>
-------------------	--

Description

This is a helper function to create a list with the additional class 'bls_data_collection' containing data downloaded from the U.S. Bureau of Labor Statistics as well as diagnostic details about the download. It is used invisibly in the package to bundle information about file downloads.

Usage

```
create_bls_object(
  data,
  downloads,
  data_type = "BLS",
  processing_steps = character(0)
)
```

Arguments

data	The processed data (data.table/data.frame)
downloads	List of download results from fread_bls()
data_type	Character string describing the type of BLS data (e.g., "CES", "JOLTS", "CPS")
processing_steps	Character vector describing processing steps applied

Value

A bls_data_collection object

display_in_console *Display text content in console.*

Description

Helper function used to display content from 'bls_overview' in the console..

Usage

```
display_in_console(content, series_id)
```

Arguments

content	Character. Text content to display
series_id	Two-letter series ID for a BLS time series to render in the display.

Value

No object returned, called to render content in console only..

display_in_popup *Display text content in popup window.*

Description

Helper function used to display content from 'bls_overview' in a popup window.

Usage

```
display_in_popup(content, series_id)
```

Arguments

content	Character. Text content to display
series_id	Two-letter series ID for a BLS time series to render in the display.

Value

No object returned, called to render content in popup.

display_in_viewer	<i>Display text content in Viewer window.</i>
-------------------	---

Description

Helper function used to display content from 'bls_overview' in the HTML viewer.

Usage

```
display_in_viewer(content, series_id)
```

Arguments

content	Character. Text content to display
series_id	Two-letter series ID for a BLS time series to render in the display.

Value

No object returned, called to render content in HTML viewer.

download_bls_files	<i>Helper function for downloading and tracking BLS files</i>
--------------------	---

Description

This function is used to pass multiple URLs at the Bureau of Labor Statistics into 'fread_bls()'

Usage

```
download_bls_files(urls, suppress_warnings = TRUE, cache = FALSE)
```

Arguments

urls	Named character vector of URLs to download
suppress_warnings	Logical. If TRUE, suppress individual download warnings
cache	Logical. If TRUE, download and cache local copy of files.

Value

Named list of bls_data objects

fread_bls	<i>Download BLS Time Series Data</i>
-----------	--------------------------------------

Description

This function downloads a tab-delimited BLS flat file, incorporating diagnostic information about the file and returning an object with the `bls_data` class that can be used in the `BLSloadR` package.

Usage

```
fread_bls(url, verbose = FALSE, cache = check_bls_cache_env())
```

Arguments

<code>url</code>	Character string. URL to the BLS flat file
<code>verbose</code>	Logical. If TRUE, prints additional messages during file read and processing.
<code>cache</code>	Logical. If TRUE, uses local persistent caching.

Value

A named list with the data and diagnostics.

get_bls_data	<i>Extract data from BLS data object</i>
--------------	--

Description

This is a helper function to extract the data element of a `'bls_data_collection'` object.

Usage

```
get_bls_data(bls_obj)
```

Arguments

<code>bls_obj</code>	A <code>bls_data_collection</code> object or raw data
----------------------	---

Value

The data component of a `'bls_data_collection'` object as a data frame.

get_bls_diagnostics *Get download diagnostics from BLS data object*

Description

This is a helper function to extract the download diagnostics element of a 'bls_data_collection' object.

Usage

```
get_bls_diagnostics(bls_obj)
```

Arguments

bls_obj A bls_data_collection object

Value

List of download diagnostics from a bls_data_collection object.

get_bls_summary *Get summary information from BLS data object*

Description

This is a helper function to extract the summary element of a 'bls_data_collection' object. This contains the number of files downloaded, the number of files with potential warnings, and the total number of warnings.

Usage

```
get_bls_summary(bls_obj)
```

Arguments

bls_obj A bls_data_collection object

Value

List of summary information

 get_ces

Download Current Employment Statistics (CES) Data

Description

This function downloads Current Employment Statistics data from the Bureau of Labor Statistics. The data includes national, regional, state, and substate employment statistics. By default, all available areas, data types, and periods are included.

Usage

```
get_ces(
  states = NULL,
  industry_filter = NULL,
  current_year_only = FALSE,
  transform = TRUE,
  monthly_only = TRUE,
  simplify_table = TRUE,
  suppress_warnings = TRUE,
  return_diagnostics = FALSE,
  cache = check_bls_cache_env()
)
```

Arguments

- | | |
|-------------------|--|
| states | Character vector of state abbreviations to download (e.g., c("MA", "NY", "CA")). If specified, downloads only these states (all industries, all years). Cannot be combined with industry_filter or current_year_only. Use 'list_ces_states()' to see all available states. |
| industry_filter | Character string specifying industry category to download. If specified, downloads this industry for all states (2007-present). Cannot be combined with states or current_year_only. Use 'list_ces_industries()' to see all available industry filters. |
| current_year_only | Logical. If TRUE, downloads the current year file which contains all states and industries for recent years (2006-present). Cannot be combined with states or industry_filter. If FALSE (default), uses other parameters. |
| transform | Logical. If TRUE (default), converts employment values from thousands to actual counts by multiplying by 1000 for specific data types (codes 1, 6, 26) and removes ", In Thousands" from data type labels. |
| monthly_only | Logical. If TRUE (default), filters out annual data (period M13). |
| simplify_table | Logical. If TRUE (default), removes excess columns and creates a date column from Year and Period in the original data. |

suppress_warnings	Logical. If TRUE (default), suppress individual download warnings and diagnostic messages for cleaner output during batch processing. If FALSE, returns the data and prints warnings and messages to the console.
return_diagnostics	Logical. If FALSE (default), returns only the data. If TRUE, returns the full <code>bls_data_collection</code> object with diagnostics.
cache	Logical. Uses <code>USE_BLS_CACHE</code> environment variable, or defaults to FALSE. If TRUE, will download a cached file from BLS server and update cache if BLS server indicates an updated file.

Details

****Performance Notes:**** The default behavior downloads a very large file (~500MB+) containing all states and industries, which can take several minutes. For faster downloads, consider:

- Use `states = c("MA", "NY")` to download only specific states
- Use `industry_filter = "total_nonfarm"` for summary employment data only
- Use `current_year_only = TRUE` for recent data only (2006-present)

****State Codes:**** Use standard two-letter state abbreviations (e.g., "MA", "CA", "NY"). Puerto Rico = "PR", Virgin Islands = "VI", District of Columbia = "DC".

****Industry Filters:**** Available options include:

- "total_nonfarm" - Total non-farm employment summary
- "total_private" - Private sector totals (2007-present)
- "manufacturing" - Manufacturing sector (2007-present)
- "construction" - Construction sector (2007-present)
- "retail_trade" - Retail trade sector (2007-present)
- "government" - Government sector (2007-present)
- And others - see BLS documentation for full list

Value

By default, returns a `data.table` with CES data. If `return_diagnostics = TRUE`, returns a `bls_data_collection` object containing data and comprehensive diagnostics.

See Also

[list_ces_states\(\)](#) to see available states, [list_ces_industries\(\)](#) to see available industry filters, [show_ces_options\(\)](#) for a comprehensive overview of filtering options.

Examples

```

# Fast download: Massachusetts and Connecticut data only (all industries)
ces_states <- get_ces(states = c("MA", "CT"))

# Fast download: Manufacturing data for all states
ces_manufacturing <- get_ces(industry_filter = "manufacturing")

# Fast download: Current year data for all states and industries
ces_current <- get_ces(current_year_only = TRUE)

# Complete dataset (slower - all states, industries, and years)
ces_all <- get_ces()

# Download with full diagnostics if needed
ces_result <- get_ces(states = "MA", return_diagnostics = TRUE)
ces_data <- get_bls_data(ces_result)

# Check for download issues
if (has_bls_issues(ces_result)) {
  print_bls_warnings(ces_result)
}

```

get_jolts

Download Job Openings and Labor Turnover Survey (JOLTS) Data

Description

This function downloads Job Openings and Labor Turnover data from the U.S. Bureau of Labor Statistics. JOLTS data provides insights into job market dynamics including job openings, hires, separations, quits, and layoffs. Data is available at national, regional, and state levels with various industry and size class breakdowns.

Usage

```

get_jolts(
  monthly_only = TRUE,
  remove_regions = TRUE,
  remove_national = TRUE,
  suppress_warnings = TRUE,
  return_diagnostics = FALSE,
  cache = check_bls_cache_env()
)

```

Arguments

monthly_only Logical. If TRUE (default), excludes annual data (period M13) and includes only monthly observations.

<code>remove_regions</code>	Logical. If TRUE (default), excludes regional aggregates (Midwest, Northeast, South, West) identified by state codes MW, NE, SO, WE.
<code>remove_national</code>	Logical. If TRUE (default), excludes national-level data (state code 00). Set to FALSE to include national data with industry and size class breakdowns.
<code>suppress_warnings</code>	Logical. If TRUE (default), suppress individual download warnings and diagnostic messages for cleaner output during batch processing. If FALSE, returns the data and prints warnings and messages to the console.
<code>return_diagnostics</code>	Logical. If TRUE, returns a <code>bls_data_collection</code> object with full diagnostics. If FALSE (default), returns just the data table.
<code>cache</code>	Logical. Uses <code>USE_BLS_CACHE</code> environment variable, or defaults to FALSE. If TRUE, will download a cached file from BLS server and update cache if BLS server indicates an updated file.

Details

The function performs several data transformations:

- Converts rate values to proportions (divides by 100) except for Unemployed to Job Opening ratio.
- Converts level values to actual counts (multiplies by 1000)
- Creates a proper date column from year and period
- Adds readable month names

Value

By default, returns a `data.table` with JOLTS data. If `return_diagnostics = TRUE`, returns a `bls_data_collection` object containing JOLTS data with the following key columns:

series_id	BLS series identifier
year	Year of observation
period	Time period (M01-M12 for months)
value	JOLTS statistic value (transformed based on data type)
date	Date of observation
state_text	State name
dataelement_text	Type of JOLTS measure (job openings, hires, separations, etc.)
area_text	Geographic area description
sizeclass_text	Establishment size class
industry_text	Industry classification
ratelevel_code	Whether the value is a "Level" (count) or "Rate" (percentage)
periodname	Month name

Examples

```
# Download state-level JOLTS data (default - returns data directly)
jолts_data <- get_jolts()

# Include national data with industry breakdowns
jолts_national <- get_jolts(remove_national = FALSE)

# Get full diagnostic object if needed
jолts_with_diagnostics <- get_jolts(return_diagnostics = TRUE)
print_bls_warnings(jолts_with_diagnostics)

# View job openings by state for latest period
job_openings <- jолts_data[dataelement_text == "Job openings" &
                           date == max(date)]
```

get_laus

*Download Local Area Unemployment Statistics (LAUS) Data***Description**

This function downloads Local Area Unemployment Statistics data from the U.S. Bureau of Labor Statistics. Due to the large size of some LAUS datasets (county and city files are >300MB), users must specify which geographic level to download. The function provides access to both seasonally adjusted and unadjusted data at various geographic levels. Additional datasets provide comprehensive non-seasonally-adjusted data for all areas broken out in 5-year increments

Usage

```
get_laus(
  geography = "state_adjusted",
  monthly_only = TRUE,
  transform = TRUE,
  suppress_warnings = TRUE,
  return_diagnostics = FALSE,
  cache = check_bls_cache_env()
)
```

Arguments

geography Character string specifying the geographic level and adjustment type. Default is "state_adjusted". Valid options are:

- "state_current_adjusted" - Current seasonally adjusted state data
- "state_unadjusted" - All historical unadjusted state data
- "state_adjusted" - All historical seasonally adjusted state data (default)
- "region_unadjusted" - Unadjusted regional and division data
- "region_adjusted" - Seasonally adjusted regional and division data

- "metro" - Metropolitan statistical area data
- "division" - Division-level data
- "micro" - Micropolitan statistical area data
- "combined" - Combined statistical area data
- "county" - County-level data (large file >300MB)
- "city" - City and town data (large file >300MB)
- "1990-1994" - Comprehensive unadjusted data for 1990-1994
- "1995-1999" - Comprehensive unadjusted data for 1995-1999
- "2000-2004" - Comprehensive unadjusted data for 2000-2004
- "2005-2009" - Comprehensive unadjusted data for 2005-2009
- "2010-2014" - Comprehensive unadjusted data for 2010-2014
- "2015-2019" - Comprehensive unadjusted data for 2015-2019
- "2020-2024" - Comprehensive unadjusted data for 2020-2024
- "2025-2029" - Comprehensive unadjusted data for 2025-2029
- "ST" - Any state two-character USPS abbreviation, plus DC and PR

monthly_only	Logical. If TRUE (default), excludes annual data (period M13) and creates a date column from year and period.
transform	Logical. If TRUE (default), converts rate and ratio measures from percentages to proportions by dividing by 100. Unemployment rates will be expressed as decimals (e.g., 0.05 for 5% unemployment) rather than as whole numbers (e.g. 5).
suppress_warnings	Logical. If TRUE (default), suppress individual download warnings and diagnostic messages for cleaner output during batch processing. If FALSE, returns the data and prints warnings and messages to the console.
return_diagnostics	Logical. If TRUE, returns a bls_data_collection object with full diagnostics. If FALSE (default), returns just the data table.
cache	Logical. Uses USE_BLS_CACHE environment variable, or defaults to FALSE. If TRUE, will download a cached file from BLS server and update cache if BLS server indicates an updated file.

Details

The function joins data from multiple BLS files:

- Main data file (varies by geography selection)
- Series definitions (la.series)
- Area codes and names (la.area)
- Measure definitions (la.measure)

Value

By default, returns a `data.table` with LAUS data. If `return_diagnostics = TRUE`, returns a `bls_data_collection` object containing LAUS data with the following key columns:

series_id BLS series identifier

year Year of observation

period Time period (M01-M12 for months, M13 for annual)

value Employment statistic value (transformed if `transform = TRUE`)

date Date of observation (if `monthly_only = TRUE`)

area_text Geographic area name

area_type_code Code indicating area type

measure_text Type of measure (unemployment rate, labor force, employment, etc.)

seasonal Seasonal adjustment status

Examples

```
# Download state-level seasonally adjusted data (default operation)
laus_states <- get_laus()

# View unemployment rates by state for latest period
unemployment <- laus_states[grep("unemployment rate", measure_text) & date == max(date)]

# Download unadjusted state data
laus_states_unadjusted <- get_laus(geography = "state_unadjusted")

# Get full diagnostic object if needed
laus_with_diagnostics <- get_laus(return_diagnostics = TRUE)
print_bls_warnings(laus_with_diagnostics)
```

get_national_ces

Get National Current Employment Statistics (CES) Data from BLS

Description

This function downloads and processes national Current Employment Statistics (CES) data from the Bureau of Labor Statistics (BLS). It retrieves multiple related datasets and joins them together to create a comprehensive employment statistics dataset with industry classifications, data types, and time period information.

Usage

```
get_national_ces(
  dataset_filter = "all_data",
  monthly_only = TRUE,
  simplify_table = TRUE,
  suppress_warnings = TRUE,
  return_diagnostics = FALSE,
  cache = check_bls_cache_env()
)
```

Arguments

- `dataset_filter` Character string specifying which dataset to download. Options include:
- "all_data" (default) - Complete dataset with all series
 - "current_seasonally_adjusted" - Only seasonally adjusted all-employee series
 - "real_earnings_all_employees" - Real earnings data for all employees
 - "real_earnings_production" - Real earnings data for production employees
- `monthly_only` Logical. If TRUE (default), excludes annual averages (period "M13") and returns only monthly data. If FALSE, includes all periods including annual averages.
- `simplify_table` Logical. If TRUE (default), removes several metadata columns (`series_title`, `begin_year`, `begin_period`, `end_year`, `end_period`, `naics_code`, `publishing_status`, `display_level`, `selectable`, `sort_sequence`) and adds a formatted date column. If FALSE, returns the full dataset with all available columns.
- `suppress_warnings` Logical. If TRUE (default), suppresses download warnings and diagnostics. If FALSE, displays warning output and diagnostic information.
- `return_diagnostics` Logical. If TRUE, returns a `bls_data_collection` object with full diagnostics. If FALSE (default), returns just the data table.
- `cache` Logical. Uses `USE_BLS_CACHE` environment variable, or defaults to FALSE. If TRUE, will download a cached file from BLS server and update cache if BLS server indicates an updated file.

Details

The function can download one of four specialized national CES datasets based on the `dataset_filter` parameter:

- `all_data`: Complete dataset (`ce.data.0.AllCESSeries`) - contains entire history of all series currently published by the CES program
- `current_seasonally_adjusted`: (`ce.data.01a.CurrentSeasAE`) - contains every seasonally adjusted all employee series and complete history
- `real_earnings_all_employees`: (`ce.data.02b.AllRealEarningsAE`) - contains real earnings data (1982-84 dollars) for all employees

- `real_earnings_production`: (`ce.data.03c.AllRealEarningsPE`) - contains real earnings data (1982-84 dollars) for production/nonsupervisory employees

Additional metadata files are always downloaded and joined:

- `ce.series` - Series metadata
- `ce.industry` - Industry classifications
- `ce.datatype` - Data type definitions
- `ce.period` - Time period definitions
- `ce.supersector` - Supersector classifications

These datasets are joined together to provide context and labels for the employment statistics. The function uses the enhanced `'download_bls_files()'` helper function for robust downloads with diagnostic reporting.

Performance Note: Using specialized datasets (other than "all_data") can significantly reduce download time and file size while still providing comprehensive employment statistics.

Value

By default, returns a `data.table` with CES data. If `return_diagnostics = TRUE`, returns a `bls_data_collection` object containing data and comprehensive diagnostics.

Note

This function requires the following packages: `dplyr`, `data.table`, `httr`, and `lubridate` (for date formatting when `simplify_table=TRUE`). The `'fread_bls()'` and `'create_bls_object()'` helper functions must be available in your environment.

See Also

Please visit the Bureau of Labor Statistics at <https://www.bls.gov/ces/> for more information about CES data

Examples

```
# Get complete monthly CES data with simplified table structure (default)
ces_monthly <- get_national_ces()

# Get only seasonally adjusted data (faster download)
ces_seasonal <- get_national_ces(dataset_filter = "current_seasonally_adjusted")

# Get real earnings data for all employees
ces_real_earnings <- get_national_ces(dataset_filter = "real_earnings_all_employees")

# Get all data including annual averages with full metadata
ces_full <- get_national_ces(dataset_filter = "all_data",
                             monthly_only = FALSE, simplify_table = FALSE)

# Get data with warnings and diagnostic information displayed
ces_with_warnings <- get_national_ces(suppress_warnings = FALSE)
```

```
# Get full diagnostic object if needed
data_with_diagnostics <- get_national_ces(return_diagnostics = TRUE)
print_bls_warnings(data_with_diagnostics)
```

get_oews	<i>Download Occupational Employment and Wage Statistics (OEWS) Data</i>
----------	---

Description

This function downloads and joins together occupational employment and wage data from the Bureau of Labor Statistics OEWS program. The data includes employment and wage estimates by occupation and geographic area. Note that OEWS is a large data set (over 6 million rows), so it will require longer to download.

Usage

```
get_oews(
  simplify_table = TRUE,
  suppress_warnings = TRUE,
  return_diagnostics = FALSE,
  fast_read = TRUE,
  cache = check_bls_cache_env()
)
```

Arguments

simplify_table	Logical. If TRUE (default), remove columns from the result that are internal BLS references or can be derived from other elements in the table.
suppress_warnings	Logical. If TRUE (default), suppress individual download warnings and diagnostic messages for cleaner output during batch processing. If FALSE, returns the data and prints warnings and messages to the console.
return_diagnostics	Logical. If TRUE, returns a <code>bls_data_collection</code> object with full diagnostics. If FALSE (default), returns just the data table.
fast_read	Logical. If TRUE (default), derives lookup values directly from <code>series_id</code> to avoid reading the series file, to speed download process. With <code>fast_read</code> , the data can download in 17 seconds (depending on bandwidth). Without <code>fast_read</code> , the same download takes 57 seconds.
cache	Logical. Uses <code>USE_BLS_CACHE</code> environment variable, or defaults to FALSE. If TRUE, will download a cached file from BLS server and update cache if BLS server indicates an updated file.

Value

By default, returns a `data.table` with OEWS data. If `return_diagnostics = TRUE`, returns a `bls_data_collection` object containing data and comprehensive diagnostics. The columns in the returned data frame when `'simplify_table = TRUE'` are listed below. Unless otherwise specified, all data is returned as a character string to preserve the value of leading and trailing zeroes.

- `series_id` - The unique OEWS series identifier.
- `year` - The year to which the estimate refers. Because OEWS is not time series data, this is always the most recent year.
- `value` - Numeric. The value of the given data type, for the given area, in the given industry and occupation.
- `seasonal` - Whether or not the data is seasonally adjusted.
- `areatype_code` - Code representing the type of area (National ("N"), Statewide ("S"), or Local ("M")).
- `industry_code` - NAICS code of the industry.
- `occupation_code` - SOC code of the occupation. Description given by `occupation_name`.
- `datatype_code` - Lookup code for the data type of a given row. Description given by `datatype_name`.
- `state_code` - Two-digit FIPS code for the state.
- `area_code` - The unique OEWS code for a substate area. Description given by `area_name`.
- `series_title` - Descriptive title of the full series ID.
- `occupation_name` - The text description of the occupation.
- `occupation_description` - More detailed description of the tasks associated with the occupation.
- `area_name` - The text description of the area.
- `datatype_name` - The text description of the type of data represented by 'value'.

Examples

```
# Download current OEWS data
oevs_data <- get_oevs()

# View available occupations
unique(oevs_data$occupation_name)

# Filter for specific occupation
software_devs <- oevs_data[grepl("Software", occupation_name)]

# Get full diagnostic object if needed
oevs_with_diagnostics <- get_oevs(return_diagnostics = TRUE)
print_bls_warnings(oevs_with_diagnostics)
```

get_oevs_areas *Download OEWS Area Definitions*

Description

Download OEWS Area Definitions

Usage

```
get_oevs_areas(ref_year, silent = TRUE, geometry = TRUE)
```

Arguments

ref_year	Four-digit year (converted to integer). The year for which to retrieve OEWS area definitions. Valid values are 2024 through current release year. Prior years included Township codes, which change the structure of the file.
silent	Logical. If TRUE (default), suppress console output
geometry	Logical. If TRUE (default), downloads shapefiles for OEWS area definitions using 'tigris::counties()' and 'tigris::shift_geometry()' to render Alaska, Hawaii, and Puerto Rico with a focus on the area of the continental United States.

Value

Data table which maps individual counties to OEWS area definitions.

- fips_code - The State FIPS code
- state_name - The state name
- state_abb - The state two-character postal abbreviation
- oews_area_code - The OEWS area code defining the metropolitan area or nonmetropolitan area the county belongs to.
- oews_area_name - The OEWS area name
- county_code - The FIPS code for the county
- county_name - The county name

Examples

```
# Get OEWS area definitions without shapefiles and with processing messages.  
test <- get_oevs_areas(ref_year = 2024, geometry = FALSE, silent = FALSE)
```

get_qcew

*Get QCEW Data Slices***Description**

This function pulls data from the BLS QCEW Open Data Access CSV Data Slices. It iterates over specified years and quarters (or annual data) to retrieve industry-specific or area-specific data tables and merges them into a single `data.table`. Optionally, it joins internal package lookup tables for industry and area descriptions.

Usage

```
get_qcew(
  period_type = "quarter",
  year_start = NULL,
  year_end = NULL,
  industry_code = NULL,
  area_code = NULL,
  add_lookups = TRUE,
  silently = FALSE
)
```

Arguments

<code>period_type</code>	Character. Either "quarter" or "year". Defaults to "quarter".
<code>year_start</code>	Numeric. The first year to retrieve data for. Defaults to the year of the date 6 months prior to the current system date.
<code>year_end</code>	Numeric. The last year to retrieve data for. Defaults to the year of the date 6 months prior to the current system date.
<code>industry_code</code>	Character. The NAICS industry code (e.g., "10", "31-33"). Constructs a URL for an Industry Data Slice. Mutually exclusive with 'area_code'.
<code>area_code</code>	Character. The QCEW area code (e.g., "US000", "32000", "C2982"). Constructs a URL for an Area Data Slice. Mutually exclusive with 'industry_code'.
<code>add_lookups</code>	Logical. If TRUE, joins the package's <code>ind_lookup</code> and <code>area_lookup</code> tables to the results to provide descriptive labels. Defaults to TRUE.
<code>silently</code>	Logical. If TRUE, suppresses status messages about the URLs being accessed. Defaults to FALSE.

Value

A combined `data.table` containing the requested QCEW data, optionally merged with lookup columns and a calculated date column. The data layout is different for quarterly or annual data files set by the 'period_type' argument.

For Quarterly files:

- area_fips - Character. Area code of row. Included 'area_lookup' data file contains mapping information.
- industry_code - Character. NAICS, Supersector, Cluster, or Total All Industries code. Numeric characters as a string to preserve examining the structure heirarchy.
- own_code - Integer. Values of 0-5 to designate ownership. See definitions at <https://www.bls.gov/cew/classifications/ownerships/ownership-titles.htm>
- agglvl_code - Integer. Two digit code identifying the level of aggregation. See definitons at <https://www.bls.gov/cew/classifications/aggregation/agg-level-titles.htm>
- size_code - Integer. Single-digit code representing the size of establishments. See definitions at <https://www.bls.gov/cew/classifications/size/size-titles.htm>
- year Integer. Four-digit calendar year for the returned data.
- qtr Integer. The calendar quarter of the data.
- disclosure_code Character. Values are either a blank string on "N". Values of N do not disclose employment or wages to maintain confidentiality.
- qtrly_estabs Integer. The number of business establishments (worksites) for the industry in the area in the quarter.
- month1_emplvl Integer. Employment in the first month of the quarter (January, April, July, or October).
- month2_emplvl Integer. Employment in the second month of the quarter (February, May, August, November).
- month3_emplvl Integer. Employment in the third month of the quarter (March, June, September, December).
- total_qtrly_wages Ingeger64. Total wages paid during the quarter.
- taxable_qtrly_wages Ingeger64. Wages subject to unemployment insurance (UI) taxes during the quarter. Note - wages subject to UI vary by state and will follow different seasonal patterns as a result.
- qtrly_contributions Integer. UI taxes (Contributions) paid by employers for this quarter. Note - UI tax policy varies by state.
- avg_wkly_wage Integer. Average weekly wage during the quarter (Total wages divided by average employment, divided by 13).
- lq_disclosure_code Character. Blank or "N". Values of "N" will suppress location quotient data for confidentiality.
- lq_qtrly_estabs Numeric. Location quotient of establishments relative to the U.S.
- lq_month1_emplvl Numeric. Location quotient of month 1 employment relative to the U.S.
- lq_month2_emplvl Numeric. Location quotient of month 2 employment relative to the U.S.
- lq_month3_emplvl Numeric. Location quotient of month 3 employment relative to the U.S.
- lq_total_qtrly_wages Numeric. Location quotient of total wages relative to the U.S.
- lq_taxable_qtrly_wages Numeric. Location quotient of taxable quarterly wages relative to the U.S.
- lq_qtrly_contributions Numeric. Location quotient of quarterly UI taxes paid relative to the U.S.

- `lq_avg_wkly_wage` Numeric. Location quotient of average weekly wages relative to the U.S.
- `oty_disclosure_code` Character. Blank or "N". Values of "N" will suppress over-the-year data for confidentiality.
- `oty_qtrly_estabs_chg` Numeric. Over-the-year change in establishments.
- `oty_qtrly_estabs_pct_chg` Numeric. Over-the-year percent change in establishments.
- `oty_month1_emplvl_chg` Numeric. Over-the-year change in month 1 employment.
- `oty_month1_emplvl_pct_chg` Numeric. Over-the-year percent change in month 1 employment.
- `oty_month2_emplvl_chg` Numeric. Over-the-year change in month 2 employment.
- `oty_month2_emplvl_pct_chg` Numeric. Over-the-year percent change in month 2 employment.
- `oty_month3_emplvl_chg` Numeric. Over-the-year change in month 3 employment.
- `oty_month3_emplvl_pct_chg` Numeric. Over-the-year percent change in month 3 employment.
- `oty_total_qtrly_wages_chg` Numeric. Over-the-year change in total wages.
- `oty_total_qtrly_wages_pct_chg` Numeric. Over-the-year percent change in total wages.
- `oty_taxable_qtrly_wages_chg` Numeric. Over-the-year change in taxable quarterly wages.
- `oty_taxable_qtrly_wages_pct_chg` Numeric. Over-the-year percent change in taxable quarterly wages.
- `oty_qtrly_contributions_chg` Numeric. Over-the-year change in quarterly UI taxes paid.
- `oty_qtrly_contributions_pct_chg` Numeric. Over-the-year percent change in quarterly UI taxes paid.
- `oty_avg_wkly_wage_chg` Numeric. Over-the-year change in average weekly wages.
- `oty_avg_wkly_wage_pct_chg` Numeric. Over-the-year percent change in average weekly wages.
- `date` Date. Calculated calendar date based on year and quarter. Reflects first day of the quarter.
- `industry_title` Character. Added based on `industry_code`
- `ind_level` Character. Description of the level of aggregation based on the `industry_code`.
- `naics_2d` Character. First two characters in the `industry_code`, useful for identifying industries.
- `sector` Character. Similar to `naics_2d`, but for industries like Manufacturing which have multiple two digit NAICS codes, this will span those groupings, for example "31-33"
- `vintage_start`. Integer. Calendar year of the earliest vintage for this `industry_code`. NAICS codes are updated every 5 years. When using this industry codes from before this date, these titles may not exist or may be incorrect.
- `vintage_end`. Integer. Calendar year of the last year this industry code was used. Years after this point should not contain this industry code. Set to 3000 for current data.
- `area_title` Character. Area description based on `area_fips` as provided by the BLS.
- `area_type` Character. Description of the type of area based on the `area_title`. More consistent naming and grouping than BLS data.
- `stfips` Character. The two-digit FIPS code of the state containing a given area. Set to "00" for multi-state regions.

- `specified_region`. Either a two-character US Postal Service abbreviation for the state containing an area or a hyphenated list of such codes for multi-state areas.

For Annual files:

- `area_fips` - Character. Area code of row. Included 'area_lookup' data file contains mapping information.
- `industry_code` - Character. NAICS, Supersector, Cluster, or Total All Industries code. Numeric characters as a string to preserve examining the structure hierarchy.
- `own_code` - Integer. Values of 0-5 to designate ownership. See definitions at <https://www.bls.gov/cew/classifications/ownerships/ownership-titles.htm>
- `agglvl_code` - Integer. Two digit code identifying the level of aggregation. See definitions at <https://www.bls.gov/cew/classifications/aggregation/agg-level-titles.htm>
- `size_code` - Integer. Single-digit code representing the size of establishments. See definitions at <https://www.bls.gov/cew/classifications/size/size-titles.htm>
- `year` Integer. Four-digit calendar year for the returned data.
- `qtr` Character. Set to "A" to represent annual data.
- `disclosure_code` Character. Values are either a blank string or "N". Values of N do not disclose employment or wages to maintain confidentiality.
- `annual_avg_estabs` Integer. The average number of business establishments (worksites) for the industry in the area for the year.
- `annual_avg_emplvl` Integer. The average monthly employment level in a given year.
- `total_annual_wages` Integer64. Total wages paid during the year.
- `taxable_annual_wages` Integer64. Wages subject to unemployment insurance (UI) taxes during the year. Note - wages subject to UI vary by state and will follow different seasonal patterns as a result.
- `annual_contributions` Integer. UI taxes (Contributions) paid by employers for this year. Note - UI tax policy varies by state.
- `annual_avg_wkly_wage` Integer. Average weekly wage during the year (Total wages divided by average employment, divided by 52).
- `avg_annual_pay` Integer. Average annual pay during the year.
- `lq_disclosure_code` Character. Blank or "N". Values of "N" will suppress location quotient data for confidentiality.
- `lq_annual_avg_estabs` Numeric. Location quotient of establishments relative to the U.S.
- `lq_annual_avg_emplvl` Numeric. Location quotient of annual employment relative to the U.S.
- `lq_total_annual_wages` Numeric. Location quotient of total wages relative to the U.S.
- `lq_taxable_annual_wages` Numeric. Location quotient of taxable annual wages relative to the U.S.
- `lq_annual_contributions` Numeric. Location quotient of annual UI taxes paid relative to the U.S.
- `lq_annual_avg_wkly_wage` Numeric. Location quotient of average weekly wages relative to the U.S.

- `lq_avg_annual_pay` Numeric. Location quotient of average annual pay relative to the U.S.
- `oty_disclosure_code` Character. Blank or "N". Values of "N" will suppress over-the-year data for confidentiality.
- `oty_annual_avg_estabs_chg` Integer. Over-the-year change in establishments.
- `oty_annual_avg_estabs_pct_chg` Numeric. Over-the-year percent change in establishments.
- `oty_annual_avg_emplvl_chg` Integer. Over-the-year change in average annual employment.
- `oty_annual_avg_emplvl_pct_chg` Numeric. Over-the-year percent change in average annual employment.
- `oty_total_annual_wages_chg` Integer. Over-the-year change in total wages.
- `oty_total_annual_wages_pct_chg` Numeric. Over-the-year percent change in total wages.
- `oty_taxable_annual_wages_chg` Integer. Over-the-year change in taxable annual wages.
- `oty_taxable_annual_wages_pct_chg` Numeric. Over-the-year percent change in taxable annual wages.
- `oty_annual_contributions_chg` Integer. Over-the-year change in annual UI taxes paid.
- `oty_annual_contributions_pct_chg` Numeric. Over-the-year percent change in annual UI taxes paid.
- `oty_annual_avg_wkly_wage_chg` Integer. Over-the-year change in average weekly wages.
- `oty_annual_avg_wkly_wage_pct_chg` Numeric. Over-the-year percent change in average weekly wages.
- `oty_avg_annual_pay_chg` Integer. Over-the-year change in average annual pay.
- `oty_avg_annual_pay_pct_chg` Numeric. Over-the-year percent change in average annual pay.
- `date` Date. Calculated calendar date based on year and quarter. Reflects first day of the quarter.
- `industry_title` Character. Added based on `industry_code`
- `ind_level` Character. Description of the level of aggregation based on the `industry_code`.
- `naics_2d` Character. First two characters in the `industry_code`, useful for identifying industries.
- `sector` Character. Similar to `naics_2d`, but for industries like Manufacturing which have multiple two digit NAICS codes, this will span those groupings, for example "31-33"
- `vintage_start`. Integer. Calendar year of the earliest vintage for this `industry_code`. NAICS codes are updated every 5 years. When using this industry codes from before this date, these titles may not exist or may be incorrect.
- `vintage_end`. Integer. Calendar year of the last year this industry code was used. Years after this point should not contain this industry code. Set to 3000 for current data.
- `area_title` Character. Area description based on `area_fips` as provided by the BLS.
- `area_type` Character. Description of the type of area based on the `area_title`. More consistent naming and grouping than BLS data.
- `stfips` Character. The two-digit FIPS code of the state containing a given area. Set to "00" for multi-state regions.
- `specified_region`. Either a two-character US Postal Service abbreviation for the state containing an area or a hyphenated list of such codes for multi-state areas.

Examples

```
# Get quarterly data for "Total, all industries" (Code 10)
# Includes industry/area descriptions and a date column by default
dt_default <- get_qcew(industry_code = "10")

# Get annual data for Nevada (Code 32000) for 2023 without lookups or messages
dt_year <- get_qcew(period_type = "year",
                    year_start = 2023,
                    year_end = 2023,
                    area_code = "32000",
                    add_lookups = FALSE,
                    silently = TRUE)
```

get_salt

*Download State Alternative Labor Market Measures (SALT) Data***Description**

This function downloads detailed alternative unemployment measures data from BLS, including U-1 through U-6 measures. The data provides a more comprehensive view of labor market conditions beyond the standard unemployment rate (U-3).

Usage

```
get_salt(
  only_states = TRUE,
  geometry = FALSE,
  suppress_warnings = TRUE,
  return_diagnostics = FALSE
)
```

Arguments

only_states	Logical. If TRUE (default), includes only state-level data. If FALSE, includes sub-state areas like New York City where available.
geometry	Logical. If TRUE, uses <code>tigris::states()</code> to download shapefiles for the states to include in the data. If FALSE (default), only returns data table.
suppress_warnings	Logical. If TRUE (default), suppress individual download warnings and diagnostic messages for cleaner output during batch processing. If FALSE, returns the data and prints warnings and messages to the console.
return_diagnostics	Logical. If TRUE, returns a <code>bls_data_collection</code> object with full diagnostics. If FALSE (default), returns just the data table.

Value

By default, returns a `data.table` with Alternative Measures of Labor Underutilization data. If `return_diagnostics = TRUE`, returns a `bls_data_collection` object containing data and comprehensive diagnostics. The function also adds derived measures and quartile comparisons across states.

Examples

```
# Download state-level SALT data
salt_data <- get_salt()

# View top 10 highest U-6 rates by state in current data
latest <- salt_data |>
  dplyr::filter(date == max(date)) |>
  dplyr::select(state, u6) |>
  dplyr::arrange(-u6)
head(latest)

# Include sub-state areas
salt_all <- get_salt(only_states = FALSE)

# Download SALT with geometry included
get_salt(geometry = TRUE)

# Get full diagnostic object if needed
data_with_diagnostics <- get_salt(return_diagnostics = TRUE)
print_bls_warnings(data_with_diagnostics)
```

has_bls_issues

Check if BLS data object has potential issues with import.

Description

Check if BLS data object has potential issues with import.

Usage

```
has_bls_issues(bls_obj)
```

Arguments

`bls_obj` A BLS data object

Value

Logical indicating if there were any import issues detected.

`ind_lookup`*NAICS Industry Titles Lookup Table (QCEW)*

Description

A data frame containing the structure of the North American Industry Classification System (NAICS) titles and codes used by the U.S. Bureau of Labor Statistics (BLS), including in the Quarterly Census of Employment and Wages (QCEW) program. This data is current as of the 2022 NAICS revision.

This table is primarily used internally by ‘BLSloadR’ functions to add human-readable industry titles or define valid industry lookups.

Usage

```
data(ind_lookup)
```

Format

A data frame with 2678 rows and 7 columns:

- `industry_code` - Character. The NAICS industry code. Items starting with 10 are aggregated values, not corresponding to a unique 6-digit NAICS.
- `industry_title` - Character. The description of the industry code provided by the BLS. This title also includes the code value for clarity.
- `ind_level` - Character. A description of the level of aggregation. Values are "Total", "Cluster", or "Supersector" for the "10" code aggregations, or else the length of the NAICS code, from 2-6 digits.
- `naics_2d` - Character. The first two digits of ‘`industry_code`’, which may be helpful to filter the results.
- `sector` - Character. Similar to ‘`naics_2d`’ except that when the industry sector spans multiple two digit codes, those codes are hyphenated (e.g. Manufacturing is NAICS 31, 32, and 33, so this displays ‘31-33’).
- `vintage_start` - Integer. The earliest year reviewed for NAICS code use. NAICS will change every 5 years, so data from before this year will have some missing values.
- `vintage_end` - Integer. The last year that a particular code is used, if applicable. Set to 3000 for current codes.

Details

The NAICS structure is hierarchical. Codes are typically longer for more detailed industries.

Source

<https://www.bls.gov/cew/classifications/industry/industry-titles.htm>

Examples

```
# Load the lookup table
data(ind_lookup)

# Find the industry title for NAICS 51 (Information)
ind_lookup[ind_lookup$industry_code == "51", ]

# Get the supersector codes
supersectors <- ind_lookup[ind_lookup$ind_level == "Supersector", ]

# Get all 3-digit NAICS codes in the Manufacturing industry
mfg_codes <- ind_lookup |>
dplyr::filter(sector == "31-33" & ind_level == "NAICS 3-digit")
```

list_ces_industries *List Available Industry Filters for CES Data*

Description

Lists all available industry categories that can be used with the ‘industry_filter’ parameter in ‘get_ces()’ function. These filters allow you to download specific industry data instead of the complete dataset.

Usage

```
list_ces_industries(show_descriptions = FALSE)
```

Arguments

show_descriptions
Logical. If TRUE, returns a data frame with filter names and descriptions. If FALSE, returns just the filter names.

Value

A character vector of industry filter names, or a data frame with names and descriptions if show_descriptions = TRUE

Examples

```
# See all available industry filters
list_ces_industries()

# See filters with descriptions
list_ces_industries(show_descriptions = TRUE)

# Use with get_ces
# manufacturing_data <- get_ces(industry_filter = "manufacturing") # All states
```

list_ces_states *List Available States for CES Data*

Description

Lists all available U.S. states and territories that can be used with the 'states' parameter in 'get_ces()' function.

Usage

```
list_ces_states()
```

Value

A character vector of available state/territory abbreviations

Examples

```
# See all available states
list_ces_states()

# Use with get_ces
# ces_data <- get_ces(states = c("MA", "NY")) # All industries for these states
```

list_national_ces_options
List Available National CES Dataset Options

Description

This function displays the available dataset filtering options for the get_national_ces() function, helping users understand what specialized datasets are available for download.

Usage

```
list_national_ces_options(show_descriptions = FALSE)
```

Arguments

```
show_descriptions
```

Logical. If TRUE, shows detailed descriptions of each dataset option. If FALSE (default), shows only the filter names.

Value

A data frame with dataset filter options and their descriptions.

Examples

```
# Show available dataset filters
list_national_ces_options()

# Show detailed descriptions
list_national_ces_options(show_descriptions = TRUE)
```

load_bls_dataset

Generic BLS Dataset Download

Description

This function generalizes a method to download all BLS data for a given time series database. These files are accessed from <https://download.bls.gov/pub/time.series/> and several datasets are available. A summary of an identified database can be generated using the 'bls_overview()' function. When multiple potential data files exist (common in large data sets), the function will prompt for an input of which file to use.

Usage

```
load_bls_dataset(
  database_code,
  return_full = FALSE,
  simplify_table = TRUE,
  suppress_warnings = FALSE,
  which_data = NULL,
  cache = check_bls_cache_env()
)
```

Arguments

- | | |
|----------------|---|
| database_code | This is the two digit character identifier for the desired database. Some Valid options are: <ul style="list-style-type: none"> • "ce" - National Current Employment Statistics Data • "sm" - State and Metro area Current Employment Statistics Data • "mp" - Major Sector Total Factor Productivity • "ci" - Employment Cost Index • "eb" - Employee Benefits Survey |
| return_full | This argument defaults to FALSE. If set to TRUE it will return a list of the elements of data retrieved from the BLS separating the data, series, and mapping values downloaded. |
| simplify_table | This parameter defaults to TRUE. When TRUE it will remove all columns from the date with "_code" in the column name, as well as a series of internal identifiers which provide general information about the series but which are not needed for performing time series analysis. This parameter also converts the |

	column "value" to numeric and generates a date column from the year and period columns in the data.
suppress_warnings	Logical. If TRUE, suppress individual download warnings during processing.
which_data	Character string or NULL. Defaults to NULL. <ul style="list-style-type: none"> • "all" - Automatically selects the data file containing ".1.All" (e.g., "bd.data.1.AllItems" or "le.data.1.AllData"). • "current" - Automatically selects the data file containing "Current" (e.g., "ce.data.0.Current"). • NULL - Default behavior. Prompts the user to select a file if multiple exist, or selects the single available file. <p>If the requested pattern is not found, the function falls back to the default behavior, prompting the user to select a file.</p>
cache	Logical. Uses USE_BLS_CACHE environment variable, or defaults to FALSE. If TRUE, will download a cached file from BLS server and update cache if BLS server indicates an updated file.

Value

This function will return either a `bls_data_collection` object (if `return_full` is FALSE or not provided) or a named list of the returned data including the `bls_data_collection` object.

Examples

```
# Import All Data
fm_import <- load_bls_dataset("fm", which_data = "all")

# Get $data element
fm_data <- fm_import$data

# Filter to a Series
# Families with Children Under 6 and No Employed Parent

u6_no_emp <- fm_data |>
  dplyr::filter(series_title == "Total families with children under 6 - with no parent employed") |>
  dplyr::select(year, value, fchld_text, fhlf_text, tdat_text)

head(u6_no_emp)

## Not run:
# Examples requiring manual intervention in the console
# Download Employer Cost Index Data
cost_index <- load_bls_dataset("ci")

# Download separated data, series, and mapping columns
benefits <- load_bls_dataset("eb", return_full = TRUE)

# Download data without removing excess columns and value conversions
```

```

productivity <- load_bls_dataset("mp", simplify_table = FALSE)

# Check for download issues
if (has_bls_issues(cost_index)) {
  print_bls_warnings(cost_index, detailed = TRUE)
}

## End(Not run)

```

```

print_bls_warnings      Print warnings for BLS data object

```

Description

Print warnings for BLS data object

Usage

```
print_bls_warnings(bls_obj, detailed = FALSE, silent = FALSE)
```

Arguments

bls_obj	A bls_data_collection object
detailed	Logical. If TRUE, shows detailed diagnostics for each file
silent	Logical. If TRUE, suppress console output

Value

Character vector of warnings (invisibly)

```

show_ces_options      Show CES Data Filtering Options

```

Description

Displays a comprehensive overview of all filtering options available for the 'get_ces()' function, including states, industries, and usage examples.

Usage

```
show_ces_options()
```

Examples

```

# See all filtering options
show_ces_options()

```

`show_national_ces_options`*Show National CES Dataset Options and Usage Examples*

Description

This function provides a comprehensive overview of the national CES dataset filtering options available in `get_national_ces()`, including examples of how to use each option.

Usage

```
show_national_ces_options()
```

Value

Prints formatted information to the console.

Examples

```
show_national_ces_options()
```

`smart_bls_download`*Check and Download BLS File with Local Caching*

Description

This function manages the downloading of files from the BLS server with a local caching layer. It uses HTTP HEAD requests to compare the server's 'Content-Length' and 'Last-Modified' headers with local file attributes. The file is only downloaded if it does not exist locally, or if the remote version is newer or a different size.

Usage

```
smart_bls_download(url, cache_dir = NULL, verbose = FALSE)
```

Arguments

<code>url</code>	A character string representing the URL of the BLS file (e.g., a '.txt' or '.gz' file from <code>download.bls.gov</code>).
<code>cache_dir</code>	A character string specifying the local directory to store cached files. May also be set with the environment variable 'BLS_CACHE_DIR'. Defaults to a persistent user data directory managed by <code>tools::R_user_dir</code> .
<code>verbose</code>	Logical. Defaults to FALSE. If TRUE, returns status messages for download.

Details

The function uses a specific set of browser-like headers to ensure compatibility with BLS server security policies. Upon a successful download, the local file's modification time is synchronized with the server's 'Last-Modified' header using `Sys.setFileTime` to ensure accurate future comparisons.

Value

A character string containing the local path to the downloaded (or cached) file.

Examples

```
## Not run:  
url <- "https://download.bls.gov/pub/time.series/ce/ce.data.0.AllCESeries"  
local_path <- smart_bls_download(url)  
data <- data.table::fread(local_path)  
  
## End(Not run)
```

Index

* datasets

- area_lookup, 3
- ind_lookup, 29

area_lookup, 3

bls_get_cache_dir, 4

bls_overview, 4

create_bls_object, 5

display_in_console, 6

display_in_popup, 6

display_in_viewer, 7

download_bls_files, 7

fread_bls, 8

get_bls_data, 8

get_bls_diagnostics, 9

get_bls_summary, 9

get_ces, 10

get_jolts, 12

get_laus, 14

get_national_ces, 16

get_oews, 19

get_oews_areas, 21

get_qcew, 22

get_salt, 27

has_bls_issues, 28

ind_lookup, 29

list_ces_industries, 11, 30

list_ces_states, 11, 31

list_national_ces_options, 31

load_bls_dataset, 32

print_bls_warnings, 34

show_ces_options, 11, 34

show_national_ces_options, 35

smart_bls_download, 35