blsr

February 1, 2021

Type Package

Title Accessing and manipulating Bureau of Labor Statistics and NAICS data from R.

Version 0.1.0

Description A set of functions for remotely accessing and manipulating BLS data from R. It contains libraries of functions for data from the Current Employment Statistics (CES) survey, the Job Openings and Labor Turnover Survey (JOLTS), and the Quarterly Census of Employment and Wages (QCEW), and the NAICS supersector mapping.

License MIT

Encoding UTF-8

LazyData true

Imports dplyr,
reshape2,
blscrapeR,
lubridate

RoxygenNote 7.1.0

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

`bls_download` *Download data from the Bureau of Labor Statistics*

Description

`bls_downloads` remotely downloads data for a given set of series IDs from the Bureau of Labor Statistics (BLS).

Usage

```
bls_download(seriesid, start_year, end_year, bls_key)
```

Arguments

`seriesid`	Character vector. Vector of series IDs for BLS data. Limit of 50 strings.
`start_year`	Numeric.
`end_year`	Numeric.
`bls_key`	Character. BLS API key. Required parameter.

Details

This function acts as a wrapper for the function `bls_api` from the package `blscrapeR`. Please see documentation for `bls_api`.

Examples

```
# National series for seasonally adjusted private sector employment and earnings
series_str = ces_seriesid(adjustment = c("S"), industries = c("05000000"),
data_types = c("01", "03"))

bls_df = bls_download(seriesid = series_str, start_year = 2006, end_year = 2010,
bls_key = Sys.getenv("BLS_KEY"))
```

ces_download *Download data from the Current Employment Statistics (CES) survey*

Description

ces_download retrieves and downloads data from the Current Employment Statistics (CES) database.

Usage

```
ces_download(
  bls_key,
  start_year,
  end_year,
  adjustment,
  industries,
  data_types,
  states,
  clean = TRUE
)
```

Arguments

bls_key	BLS API key for the user. See vignette.
start_year	Year to start data download.
end_year	Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.
industries	Character vector. See vignette. List of available industries given in ces_national_codes_list or ces_state_codes_list datasets.
data_types	Character vector. Desired output. See series vignette. List of available data_types given in ces_national_codes_list or ces_state_codes_list datasets.
states	Optional character vector. See vignette. Leave blank for total US. List of states given in ces_national_codes_list or ces_state_codes_list datasets.
clean	Optional logical. Whether to clean the data into usable format or not. Defaults to TRUE.

Details

This function constructs BLS series IDs for a chosen set of parameters and then downloads the data from the CES database. This function also cleans the data and appends on identifiers for easy panel formatting.

Value

Cleaned data frame of CES data.

Examples

```
ces_df = ces_download(bls_key = Sys.getenv("BLS_KEY"), start_year = 2010, end_year = 2015,
adjustment = "U", industries = "05000000", data_types = c("01", "03", "11"), states = "1900000")
```

ces_emp *Download national employment data from Current Employment Statistics (CES) survey*

Description

ces_emp downloads pre-packaged national employment data from the CES database.

Usage

```
ces_emp(bls_key, series = "nfp", start_year, end_year, adjustment = "S")
```

Arguments

bls_key	BLS API key for the user. See vignette.
series	Character. Data series. Either non-farm payrolls ("nfp"), private payrolls ("private"), supersector payrolls ("super"), or sector payrolls ("sector"). Defaults to non-farm payrolls ("nfp").
start_year	Numeric. Year to start data download.
end_year	Numeric. Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both. Defaults to seasonally adjusted ("S").

Details

This function downloads and cleans pre-packaged employment data from the CES database. The user has four choices of employment data: non-farm payrolls, private payrolls, super sector payrolls, or sector payrolls. The employment data is at the national level.

Examples

```
ces_df = ces_emp(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

ces_seriesid *Series IDs for Current Employment Statistics (CES) data*

Description

ces_seriesid constructs series IDs for downloading data from the Current Employment Statistics (CES) database.

Usage

```
ces_seriesid(adjustment, industries, data_types, states)
```

Arguments

adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.
industries	Character vector. See vignette. List of available industries given in ces_national_codes_list or ces_state_codes_list datasets.
data_types	Character vector. Desired output. See series vignette. List of available data_types given in ces_national_codes_list or ces_state_codes_list datasets.
states	Optional character vector. See vignette. Leave blank for total US. List of states given in ces_national_codes_list or ces_state_codes_list datasets.

Details

The Bureau of Labor Statistics (BLS) stores data in the form of series IDs. The structure of series IDs varies by the underlying database (e.g., CES, JOLTS, etc.). Users must input the correct series IDs to download their desired data. This function generates CES series IDs for a given set of inputs.

Value

Vector of CES series IDs.

Examples

```
# National series for seasonally adjusted private sector employment and earnings
ces_seriesid(adjustment = c("S"), industries = c("05000000"), data_types = c("01", "03"))
```

---

clean_ces_national *Cleans the results of a query to the CES database.*

Description

clean_ces_national cleans the results of a query to the CES national database and appends additional information.

Usage

```
clean_ces_national(bls_df)
```

Arguments

bls_df	Data frame containing results of query to the CES national database.

Details

General data cleaning.

Value

Cleaned data frame from CES national database.

Examples

```
clean_ces_national(ces_df)
```

clean_jolts *Cleans the results of a query to the CES database.*

Description

clean_jolts cleans the results of a query to the JOLTS national database and appends additional information.

Usage

```
clean_jolts(bls_df)
```

Arguments

bls_df	Data frame containing results of query to the JOLTS national database.

Details

General data cleaning.

Value

Cleaned data frame from JOLTS national database.

Examples

```
clean_jolts(jolts_df)
```

clean_laus *Cleans the results of a query to the LAUS database.*

Description

clean_laus cleans the results of a query to the LAUS state database and appends additional information.

Usage

```
clean_laus(bls_df)
```

Arguments

bls_df	Data frame containing results of query to the LAUS national database.

Details

General data cleaning.

Value

Cleaned data frame from LAUS national database.

Examples

```
clean_laus(laus_df)
```

---

get_bls_metadata *Displays metadata from the Bureau of Labor Statistics*

Description

`get_bls_metadata` displays metatadata for a given Bureau of Labor Statistics (BLS) database.

Usage

```
get_bls_metadata(database)
```

Arguments

`database`	Character. Database name. Either "JOLTS", "CES", or "CESstate".

Details

This function displays metadata for a BLS database.

Value

Data frame containing BLS metadata.

Examples

```
get_bls_metadata("JOLTS")
```

---

hello *Hello, World!*

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

`jolts_download` *Download data from the Job Opening and Labor Turnover survey (JOLTS)*

Description

`jolts_download` retrieves and downloads data from the Job Opening and Labor Turnover survey (JOLTS) database.

Usage

```
jolts_download(
  bls_key,
  start_year,
  end_year,
  adjustment,
  industries,
  data_types,
  data_levels,
  states = "00",
  areas = "00000",
  sizes = "00",
  clean = TRUE
)
```

Arguments

`bls_key`	BLS API key for the user. See vignette.
`start_year`	Year to start data download.
`end_year`	Year to end data download.
`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.
`industries`	Character vector. See vignette. List of available industries given in `jolts_codes_list` dataset.
`data_types`	Character vector. Desired output. See series vignette. List of available data_types given in `jolts_codes_list` dataset.
`data_levels`	Character vector. Levels ("L") or rates ("R") or both.
`states`	Option character vector. See vignette. Leave blank for total US. List of states given in `jolts_codes_list` dataset.
`areas`	Optional character vector. See vignette. Leave blank for total US. List of areas in `jolts_codes_list` dataset.
`sizes`	Optional character vector. See vignette. Leave blank for all. List in `jolts_codes_list` dataset.
`clean`	Optional logical. Whether to clean the data into usable format or not. Defaults to TRUE.

Details

This function constructs BLS series IDs for a chosen set of parameters and then downloads the data from the JOLTS database. This function also cleans the data and appends on identifiers for easy panel formatting. Reflects the October 2020 update to JOLTS data series.

Value

Cleaned data frame of JOLTS data.

Examples

```
Add in from our other exercise.
```

`jolts_hires` *Download hiring data from the Job Opening and Labor Turnover survey (JOLTS).*

Description

`jolts_hires` downloads pre-packaged hiring data from the JOLTS database.

Usage

```
jolts_hires(bls_key, series = "nfp", start_year, end_year, adjustment)
```

Arguments

`bls_key`	BLS API key for the user. See vignette.
`series`	Character. Data series. Either non-farm hiring ("nfp"), private hiring ("private"), supersector hiring ("super"), or sector hiring ("sector"). Defaults to non-farm hiring.
`start_year`	Numeric. Year to start data download.
`end_year`	Numeric. Year to end data download.
`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged hiring data from the JOLTS database. The user has four choices for hiring data: non-farm hiring, private hiring, super sector or sector hiring. The data is formatted in terms of hiring rates similar to the quoted JOLTS series. The function reflects the October 2020 update to the JOLTS data series.

Examples

```
jolts_df = jolts_hires(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

jolts_layoffs — *Download layoffs data from the Job Opening and Labor Turnover survey (JOLTS).*

Description

jolts_seps downloads pre-packaged layoffs data from the JOLTS database.

Usage

```
jolts_layoffs(bls_key, series = "nfp", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette.
series	Character. Data series. Either non-farm layoffs ("nfp"), private layoffs ("private"), supersector layoffs ("super"), or sector layoffs ("sector"). Defaults to non-farm layoffs.
start_year	Numeric. Year to start data download.
end_year	Numeric. Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged layoffs data from the JOLTS database. The user has three choices for layoffs data: non-farm layoffs, private layoffs, or super sector hiring. The data is formatted in terms of rates similar to the quoted JOLTS series. The function reflects the October 2020 update to the JOLTS data series.

Examples

```
jolts_df = jolts_layoffs(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

jolts_openings — *Download job openings data from the Job Opening and Labor Turnover survey (JOLTS).*

Description

jolts_seps downloads pre-packaged openings data from the JOLTS database.

Usage

```
jolts_openings(bls_key, series = "nfp", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette.
series	Character. Data series. Either non-farm openings ("nfp"), private openings ("private"), supersector openings ("super"), or sector openings ("sector"). Defaults to non-farm openings.
start_year	Numeric. Year to start data download.
end_year	Numeric. Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged job openings data from the JOLTS database. The user has three choices for openings data: non-farm openings, private openings, or super sector hiring. The data is formatted in terms of levels similar to the quoted JOLTS series. The function reflects the October 2020 update to the JOLTS data series.

Examples

```
jolts_df = jolts_openings(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

---

jolts_others *Download other separations data from the Job Opening and Labor Turnover survey (JOLTS).*

Description

jolts_seps downloads pre-packaged other data from the JOLTS database.

Usage

```
jolts_others(bls_key, series = "nfp", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette.
series	Character. Data series. Either non-farm other ("nfp"), private other ("private"), supersector other ("super"). or sector other ("sector"). Defaults to non-farm other.
start_year	Numeric. Year to start data download.
end_year	Numeric. Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged other separations data from the JOLTS database. The user has three choices for other separations data: non-farm, private, or super sector. The data is formatted in terms of rates similar to the quoted JOLTS series. The function reflects the October 2020 update to the JOLTS data series.

Examples

```
jolts_df = jolts_others(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

`jolts_quits` *Download quits data from the Job Opening and Labor Turnover survey (JOLTS).*

Description

`jolts_seps` downloads pre-packaged quits data from the JOLTS database.

Usage

```
jolts_quits(bls_key, series = "nfp", start_year, end_year, adjustment)
```

Arguments

`bls_key`	BLS API key for the user. See vignette.
`series`	Character. Data series. Either non-farm quits ("nfp"), private quits ("private"), supersector quits ("super"), or sector quits ("sector"). Defaults to non-farm quits.
`start_year`	Numeric. Year to start data download.
`end_year`	Numeric. Year to end data download.
`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged quits data from the JOLTS database. The user has three choices for quits data: non-farm quits, private quits, or super sector hiring. The data is formatted in terms of rates similar to the quoted JOLTS series. The function reflects the October 2020 update to the JOLTS data series.

Examples

```
jolts_df = jolts_quits(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

`jolts_seps` *Download separations data from the Job Opening and Labor Turnover survey (JOLTS).*

Description

`jolts_seps` downloads pre-packaged separations data from the JOLTS database.

Usage

```
jolts_seps(bls_key, series = "nfp", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette.
series	Character. Data series. Either non-farm separations ("nfp"), private separations ("private"), supersector separations ("super"), or sector separations ("sector"). Defaults to non-farm separations.
start_year	Numeric. Year to start data download.
end_year	Numeric. Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged separations data from the JOLTS database. The user has three choices for separations data: non-farm separations, private separations, or super sector hiring. The data is formatted in terms of rates similar to the quoted JOLTS series. Separations is equal to the sum of layoffs, quits, and other separations. The function reflects the October 2020 update to the JOLTS data series.

Examples

```
jolts_df = jolts_seps(Sys.getenv("BLS_KEY"), "nfp", 2010, 2015, "U")
```

---

jolts_seriesid *Series IDs for Job Openings and Labor Turnover Survey (JOLTS) data*

Description

jolts_seriesid constructs series IDs for downloading data from the Job Openings and Labor Turnover Survey (JOLTS) database.

Usage

```
jolts_seriesid(
  adjustment,
  industries,
  data_types,
  data_levels,
  states,
  areas,
  sizes
)
```

Arguments

adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.
industries	Character vector. See vignette. List of available industries given in jolts_codes_list dataset.
data_types	Character vector. Desired output. See series vignette. List of available data_types given in jolts_codes_list dataset.

data_levels	Character vector. Levels ("L") or rates ("R") or both.
states	Option character vector. See vignette. Leave blank for total US. List of states given in jolts_codes_list dataset.
areas	Optional character vector. See vignette. Leave blank for total US. List of areas in jolts_codes_list dataset. #' @param sizes Optional character vector. See vignette. Leave blank for all. List in jolts_codes_list dataset.

Details

The Bureau of Labor Statistics (BLS) stores data in the form of series IDs. The structure of series IDs varies by the underlying database (e.g., CES, JOLTS, etc.). Users must input the correct series IDs to download their desired data. This function helps overcome this problem by automatically generating JOLTS series IDs for a given set of inputs. Note that only total non-farm data series IDs can be extracted for different regions of the United States. Reflects the October 2020 updates to JOLTS data series.

Value

Vector of JOLTS series IDs.

Examples

```
# National series for seasonally adjusted private sector hires and job openings
jolts_seriesid(adjustment = c("S"), industries = c("100000"), data_types = c("HI", "JO"),
data_levels = c("L"))
```

---

laus_download *Download data from the Local Area Unemployment Statistics (LAUS) database*

Description

laus_download retrieves and downloads data from the Local Area Unemployment Statistics (LAUS) database.

Usage

```
laus_download(
  bls_key,
  start_year,
  end_year,
  adjustment,
  states,
  data_types,
  clean = TRUE
)
```

Arguments

bls_key	BLS API key for the user. See vignette.
start_year	Year to start data download.
end_year	Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.
states	Character vector. See vignette and laus_codes_list object.
data_types	Character vector. See vignette and laus_codes_list object.
clean	Optional logical. Whether to clean the data into usable format or not. Defaults to TRUE.

Details

This function constructs BLS series IDs for a chosen set of parameters and then downloads the data from the LAUS database. This function also cleans the data and appends on identifiers for easy panel formatting.

Value

Cleaned data frame of LAUS data.

Examples

```
laus_download(bls_key = "inputs", start_year = 2010, end_year = 2015,
adjustment = "U", states = c("ST0100000000000", "ST0200000000000"), data_types = c("03"))
```

---

laus_emp *Download state empoyment numbers from the Local Area Unemployment Statistics (LAUS) database*

Description

laus_emp downloads state empoyment numbers from the Local Area Unemployment Statistics (LAUS) database.

Usage

```
laus_emp(bls_key, states = "all", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette. #' @param states Character vector. All states ("all"), Washington D.C. ("DC"), or list of state codes. See vignette and laus_codes_list object.
start_year	Year to start data download.
end_year	Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged state empoyment levels data from the LAUS database. The default setting is to download empoyment levels for all states in the U.S. excluding D.C. The user can shorten or lengthen the list of states but is constrained by the BLS 50 series limit.

Value

Cleaned data frame of LAUS data.

Examples

```
laus_emp(Sys.getenv("BLS_KEY"), "all", 2010, 2015, "U")
```

---

laus_labor_force *Download state labor force numbers from the Local Area Unemployment Statistics (LAUS) database*

Description

`laus_labor_force` downloads state labor force numbers from the Local Area Unemployment Statistics (LAUS) database.

Usage

```
laus_labor_force(bls_key, states = "all", start_year, end_year, adjustment)
```

Arguments

`bls_key`	BLS API key for the user. See vignette. #' @param states Character vector. All states ("all"), Washington D.C. ("DC"), or list of state codes. See vignette and `laus_codes_list` object.
`start_year`	Year to start data download.
`end_year`	Year to end data download.
`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged state labor force levels data from the LAUS database. The default setting is to download labor force levels for all states in the U.S. excluding D.C. The user can shorten or lengthen the list of states but is constrained by the BLS 50 series limit.

Value

Cleaned data frame of LAUS data.

Examples

```
laus_labor_force(Sys.getenv("BLS_KEY"), "all", 2010, 2015, "U")
```

laus_seriesid *Series IDs for Local Area Unemployment Statistics (LAUS) data*

Description

laus_seriesid constructs series IDs for downloading data from the Local Area Unemployment Statistics (LAUS) database.

Usage

```
laus_seriesid(adjustment, states, data_types)
```

Arguments

adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.
states	Character vector. See vignette and laus_codes_list object.
data_types	Character vector. See vignette and laus_codes_list object.

Details

The Bureau of Labor Statistics (BLS) stores data in the form of series IDs. The structure of series IDs varies by the underlying database (e.g., CES, JOLTS, LAUS, etc.). Users must input the correct series IDs to download their desired data. This function generates LAUS series IDs given a set of user inputs.

Value

Vector of LAUS series IDs.

Examples

```
laus_seriesid(adjustment = "U", states = c("ST0100000000000", "ST0200000000000"), data_types = c("03"))
```

laus_unemp *Download state unemployment numbers from the Local Area Unemployment Statistics (LAUS) database*

Description

laus_unemp downloads state unemployment numbers from the Local Area Unemployment Statistics (LAUS) database.

Usage

```
laus_unemp(bls_key, states = "all", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette. #' @param states Character vector. All states ("all"), Washington D.C. ("DC"), or list of state codes. See vignette and laus_codes_list object.
start_year	Year to start data download.
end_year	Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged state unemployment levels data from the LAUS database. The default setting is to download unemployment levels for all states in the U.S. excluding D.C. The user can shorten or lengthen the list of states but is constrained by the BLS 50 series limit.

Value

Cleaned data frame of LAUS data.

Examples

```
laus_unemp(Sys.getenv("BLS_KEY"), "all", 2010, 2015, "U")
```

---

laus_urate *Download state unemployment rates from the Local Area Unemployment Statistics (LAUS) database*

Description

laus_urate downloads state unemployment rates from the Local Area Unemployment Statistics (LAUS) database.

Usage

```
laus_urate(bls_key, states = "all", start_year, end_year, adjustment)
```

Arguments

bls_key	BLS API key for the user. See vignette. #' @param states Character vector. All states ("all"), Washington D.C. ("DC"), or list of state codes. See vignette and laus_codes_list object.
start_year	Year to start data download.
end_year	Year to end data download.
adjustment	Character vector. Seasonal adjustment ("S") or not ("U") or both.

Details

This function downloads and cleans pre-packaged state unemployment rate data from the LAUS database. The default setting is to download unemployment rates for all states in the U.S. excluding D.C. The user can shorten or lengthen the list of states but is constrained by the BLS 50 series limit.

Value

Cleaned data frame of LAUS data.

Examples

```
laus_urate(Sys.getenv("BLS_KEY"), "all", 2010, 2015, "U")
```

`okay_ces_seriesid` *Check whether CES series ID is valid.*

Description

`okay_ces_seriesid` checks whether the inputs yield a valid CES series ID string.

Usage

```
okay_ces_seriesid(adjustment, industries, data_types, states)
```

Arguments

`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.
`industries`	Character vector. See vignette. List of available industries given in `ces_national_codes_list` or `ces_state_codes_list` datasets.
`data_types`	Character vector. Desired output. See series vignette. List of available data_types given in `ces_national_codes_list` or `ces_state_codes_list` datasets.
`states`	Optional character vector. See vignette. Leave blank for total US. List of states given in `ces_national_codes_list` or `ces_state_codes_list` datasets.

Details

General error checking for CES data downloads.

Value

TRUE if the series IDs are valid. FALSE otherwise.

Examples

```
okay_ces_seriesid("S", "0000000", "01")
```

`okay_jolts_seriesid` *Check whether JOLTS series ID is valid.*

Description

`okay_jolts_seriesid` checks whether the inputs yield a valid JOLTS series ID string.

Usage

```
okay_jolts_seriesid(
  adjustment,
  industries,
  data_types,
  data_levels,
  states,
  areas,
  sizes
)
```

Arguments

`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.
`industries`	Character vector. See vignette. List of available industries given in `jolts_codes_list` dataset.
`data_types`	Character vector. Desired output. See series vignette. List of available data_types given in `jolts_codes_list` dataset.
`data_levels`	Character vector. Levels ("L") or rates ("R") or both.
`states`	Option character vector. See vignette. Leave blank for total US. List of states given in `jolts_codes_list` dataset.
`areas`	Optional character vector. See vignette. Leave blank for total US. List of areas in `jolts_codes_list` dataset.
`sizes`	Optional character vector. See vignette. Leave blank for all. List in `jolts_codes_list` dataset.

Details

General error checking for JOLTS data downloads. Reflects the October 2020 update to JOLTS data series.

Value

TRUE if the series IDs are valid. FALSE otherwise.

Examples

```
NA
```

okay_laus_seriesid *Check whether LAUS series ID is valid.*

Description

`okay_laus_seriesid` checks whether the given inputs yield a valid LAUS series ID string.

Usage

```
okay_laus_seriesid(adjustment, states, data_types)
```

Arguments

`adjustment`	Character vector. Seasonal adjustment ("S") or not ("U") or both.
`states`	Character vector. See vignette and `laus_codes_list` object.
`data_types`	Character vector. See vignette and `laus_codes_list` object.

Details

General error checking for LAUS data downloads.

Value

TRUE if the series IDs are valid. FALSE otherwise.

Examples

```
okay_laus_seriesid(adjustment = "U", states = c("ST0100000000000"), data_types = c("03"))
```

okay_series_input *Checks series ID inputs.*

Description

`okay_series_input` returns whether a series ID input is valid.

Usage

```
okay_series_input(inputs, valid_inputs)
```

Arguments

`inputs`	Character vector. List of inputs to series ID string.
`valid_inputs`	Character vector. List of valid inputs to series ID string.

Value

TRUE if the series inputs are valid. FALSE otherwise.

Index