cbs_get_data {cbsodataR}

Get data from Statistics Netherlands (CBS)

Description

Retrieves data from a table of Statistics Netherlands. A list of available tables can be retrieved with `cbs_get_toc()`. Use the `Identifier` column of `cbs_get_toc` as `id` in `cbs_get_data` and `cbs_get_meta`.

Usage

```
cbs_get_data(
  id,
  ...,
  catalog = "CBS",
  select = NULL,
  typed = TRUE,
  add_column_labels = TRUE,
  dir = tempdir(),
  verbose = FALSE,
  base_url = getOption("cbsodataR.base_url", BASE_URL),
  include_ID = FALSE
)
```

Arguments

`id`	Identifier of table, can be found in `cbs_get_toc()`
`...`	optional filter statements, see details.
`catalog`	catalog id, can be retrieved with `cbs_get_datasets()`
`select`	`character` optional, columns to select
`typed`	Should the data automatically be converted into integer and numeric?
`add_column_labels`	Should column titles be added as a label (TRUE) which are visible in `View`
`dir`	Directory where the table should be downloaded. Defaults to temporary directory
`verbose`	Print extra messages what is happening.
`base_url`	optionally specify a different server. Useful for third party data services implementing the same protocol.
`include_ID`	Should the data include the ID column for the rows?

Details

To reduce the download time, optionaly the data can be filtered on category values: for large tables (> 100k records) this is a wise thing to do.

The filter is specified with (see examples below):

- `<column_name> = <values>` in which `<values>` is a character vector. Rows with values that are not part of the character vector are not returned. **Note that the values have to be values from the `$Key` column of the corresponding meta data. These may contain trailing spaces...**
- `<column_name> = has_substring(x)` in which x is a character vector. Rows with values that do not have a substring that is in x are not returned. Useful substrings are "JJ", "KW", "MM" for Periods (years, quarters, months) and "PV", "CR" and "GM" for Regions (provinces, corops, municipalities).
- `<column_name> = eq(<values>) | has_substring(x)`, which combines the two statements above.

By default the columns will be converted to their type (`typed=TRUE`). CBS uses multiple types of missing (unknown, surpressed, not measured, missing): users wanting all these nuances can use `typed=FALSE` which results in character columns.

Value

`data.frame` with the requested data. Note that a csv copy of the data is stored in `dir`.

Note

All data are downloaded using `cbs_download_table()`

See Also

`cbs_get_meta()`, `cbs_download_data()`

Other data retrieval: `cbs_add_date_column()`, `cbs_add_label_columns()`, `cbs_download_data()`, `cbs_extract_table_id()`, `cbs_get_data_from_link()`

Other query: `eq()`, `has_substring()`

Examples

Run examples

```
## Not run:
cbs_get_data( id      = "7196ENG"      # table id
            , Periods = "2000MM03"     # March 2000
            , CPI     = "000000"       # Category code for total
            )

# useful substrings:
## Periods: "JJ": years, "KW": quarters, "MM", months
```

```
## Regions: "NL", "PV": provinces, "GM": municipalities

cbs_get_data( id      = "7196ENG"      # table id
            , Periods = has_substring("JJ")     # all years
            , CPI     = "000000"       # Category code for total
            )

cbs_get_data( id      = "7196ENG"      # table id
            , Periods = c("2000MM03","2001MM12")     # March 2000 and Dec
2001
            , CPI     = "000000"       # Category code for total
            )

# combine either this
cbs_get_data( id      = "7196ENG"      # table id
            , Periods = has_substring("JJ") | "2000MM01" # all years and
Jan 2001
            , CPI     = "000000"       # Category code for total
            )

# or this: note the "eq" function
cbs_get_data( id      = "7196ENG"      # table id
            , Periods = eq("2000MM01") | has_substring("JJ") # Jan 2000 and
all years
            , CPI     = "000000"       # Category code for total
            )

## End(Not run)
```
