Package: tceper (via r-universe)

October 29, 2026

Title Access the 'Open Data API' of Pernambuco Court of Accounts

Version 0.1.5

Description An R interface to the 'Open Data API' of the Tribunal de Contas do Estado de Pernambuco (TCE-PE), the Court of Accounts of the State of Pernambuco, Brazil. Provides tidy, ready-to-use functions to query public data on revenues, expenditures, commitments, procurement, contracts, agreements, public works, legal processes, personnel and reference tables for all state and municipal government entities in Pernambuco. All results are returned as tibbles with column names converted to 'snake_case' by default. Uses 'httr2' for HTTP requests and 'cli' for user-friendly messages. See <https://sistemas.tcepe.tc.br/DadosAbertos/> for the API documentation.

License MIT + file LICENSE

URL https://strategicprojects.github.io/tceper/, https://github.com/StrategicProjects/tceper

BugReports https://github.com/StrategicProjects/tceper/issues

Encoding UTF-8

Language en

Depends R (>= 4.1.0)

Imports cli (>= 3.6.0), httr2 (>= 1.0.0), jsonlite, rlang (>= 1.1.0), tibble, purrr, janitor

Suggests testthat (>= 3.0.0), knitr, rmarkdown, dplyr, stringr, ggplot2

Config/testthat/edition 3

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author André Leite [aut, cre] (ORCID: <https://orcid.org/0000-0002-4718-9766>), Marcos Wasiliew [aut] (ORCID: <https://orcid.org/0009-0004-4694-3159>), Hugo Vasconcelos [aut] (ORCID: <https://orcid.org/0000-0001-6249-0920>), Carlos Amorim [aut] (ORCID: <https://orcid.org/0000-0001-6315-8305>), Diogo Bezerra [aut] (ORCID: <https://orcid.org/0000-0002-1216-8674>), Júlia Nascimento Barreto [aut] (ORCID: <https://orcid.org/0009-0004-2851-7770>)

Maintainer André Leite <leite@castlab.org>

Repository https://cran.r-universe.dev

Date/Publication 2026-10-29 08:50:02 UTC

RemoteUrl https://github.com/cran/tceper

RemoteRef HEAD

RemoteSha c27af6eedf5bb71e9bd5e36d79fb0e8c6649e54f

Contents

tce_agreements . . . 4
tce_benefit_types . . . 5
tce_bid_details . . . 5
tce_bid_documents . . . 6
tce_bid_legal_basis . . . 7
tce_bid_modalities . . . 8
tce_bid_stages . . . 9
tce_bid_statuses . . . 9
tce_bids . . . 10
tce_budget_statuses . . . 11
tce_budgeted_revenues . . . 12
tce_cache_clear . . . 13
tce_cache_info . . . 13
tce_catalog . . . 14
tce_commitment_liquidations . . . 15
tce_commitment_payments . . . 15
tce_commitment_summary . . . 16
tce_consideranda . . . 17
tce_contract_amendments . . . 18
tce_contract_documents . . . 18
tce_contract_items . . . 19
tce_contracts . . . 20
tce_creditor_types . . . 21
tce_debts_fines . . . 22
tce_determinations . . . 22
tce_endpoint . . . 23
tce_entities . . . 24
tce_fields . . . 25

tce_funding_sources . . . 25
tce_inactivation_reasons . . . 26
tce_legal_instrument_documents . . . 27
tce_legal_instrument_items . . . 28
tce_legal_instrument_participants . . . 29
tce_legal_instrument_terms . . . 30
tce_legal_instruments . . . 31
tce_municipal_creditor_types . . . 31
tce_municipal_entities . . . 32
tce_municipal_expenditures . . . 33
tce_municipal_revenues . . . 34
tce_municipal_transfers . . . 35
tce_municipalities . . . 35
tce_object_characteristics . . . 36
tce_object_classifications . . . 37
tce_object_nature . . . 38
tce_outcomes . . . 38
tce_params . . . 39
tce_payroll_types . . . 40
tce_person_creditor_types . . . 41
tce_processes . . . 41
tce_procurement_process_budget . . . 42
tce_procurement_process_documents . . . 43
tce_procurement_process_participants . . . 44
tce_procurement_processes . . . 45
tce_public_works . . . 46
tce_public_works_audits . . . 46
tce_public_works_contractors . . . 47
tce_recommendations . . . 48
tce_reference_codes . . . 49
tce_reference_dates . . . 49
tce_reference_sources . . . 50
tce_remessa_works . . . 51
tce_remessa_works_execution . . . 52
tce_remessa_works_geometry . . . 53
tce_request . . . 54
tce_retirement_outcomes . . . 56
tce_sanctions . . . 56
tce_school_transport . . . 57
tce_servants . . . 58
tce_special_accountability . . . 58
tce_spending_limits . . . 59
tce_state_commitment_items . . . 60
tce_state_creditor_types . . . 60
tce_state_entities . . . 61
tce_state_expenditures . . . 62
tce_state_price_comparison . . . 63
tce_state_revenues . . . 64

tce_sub_units 65
tce_suppliers 65
tce_update_indices 66

Index **68**

`tce_agreements` *Agreements (Convenios)*

Description

Retrieve state and municipal agreements (*convênios*).

Usage

```
tce_agreements(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`, `CodigoMunicipio`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other procurement: `tce_bid_details()`, `tce_bid_documents()`, `tce_bid_legal_basis()`, `tce_bid_modalities()`, `tce_bid_stages()`, `tce_bid_statuses()`, `tce_bids()`, `tce_contract_amendments()`, `tce_contract_documents()`, `tce_contract_items()`, `tce_contracts()`, `tce_object_characteristics()`, `tce_object_classifications()`, `tce_object_nature()`

tce_benefit_types *Benefit types (Remessa)*

Description

Lookup table of benefit types (RemessaTCEPE system).

Usage

```
tce_benefit_types(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_participants()`, `tce_legal_instrument_terms()`, `tce_legal_instruments()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_process_partic`
`tce_procurement_processes()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_remessa_works_geometry()`, `tce_update_indices()`

tce_bid_details *Bid details with bidders*

Description

Retrieve bids with their respective bidders (*licitantes*).

Usage

```
tce_bid_details(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

---

tce_bid_documents *Bid documents*

Description

Retrieve documents associated with bids.

Usage

```
tce_bid_documents(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

---

tce_bid_legal_basis *Bid legal basis*

Description

Lookup table of legal basis types for bids.

Usage

```
tce_bid_legal_basis(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

---

tce_bid_modalities *Bid modalities*

Description

Lookup table of bid modality types.

Usage

```
tce_bid_modalities(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

tce_bid_stages *Bid stages*

Description

Lookup table of bid process stages.

Usage

```
tce_bid_stages(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`, `CodigoMunicipio`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other procurement: `tce_agreements()`, `tce_bid_details()`, `tce_bid_documents()`, `tce_bid_legal_basis()`, `tce_bid_modalities()`, `tce_bid_statuses()`, `tce_bids()`, `tce_contract_amendments()`, `tce_contract_documents(`
`tce_contract_items()`, `tce_contracts()`, `tce_object_characteristics()`, `tce_object_classifications()`, `tce_object_nature()`

tce_bid_statuses *Bid statuses*

Description

Lookup table of bid status types.

Usage

```
tce_bid_statuses(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

---

tce_bids *Bids*

Description

Retrieve public procurement bids (*licitações*).

Usage

```
tce_bids(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`, `CodigoMunicipio`.
verbose	If `TRUE`, prints the final API URL.
cache	If `TRUE` (default), returns cached results when available.
clean_names	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other procurement: `tce_agreements()`, `tce_bid_details()`, `tce_bid_documents()`, `tce_bid_legal_basis()`, `tce_bid_modalities()`, `tce_bid_stages()`, `tce_bid_statuses()`, `tce_contract_amendments()`, `tce_contract_documents()`, `tce_contract_items()`, `tce_contracts()`, `tce_object_characteristics()`, `tce_object_classifications()`, `tce_object_nature()`

`tce_budget_statuses` *Budget item statuses (Remessa)*

Description

Possible statuses for cost-estimate items/lots.

Usage

```
tce_budget_statuses(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If `TRUE`, prints the final API URL.
cache	If `TRUE` (default), returns cached results when available.
clean_names	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_partic tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry(), tce_update_indices()

---

tce_budgeted_revenues *Budgeted revenues (municipal)*

Description

Retrieve budgeted (forecast) revenue data. Municipal entities only.

Usage

```
tce_budgeted_revenues(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other revenues: tce_municipal_revenues(), tce_state_revenues()

tce_cache_clear *Clear the tceper in-memory cache*

Description

Removes cached API responses so that subsequent calls hit the API again. By default clears everything; use `pattern` to selectively clear entries for specific endpoints.

Usage

```
tce_cache_clear(pattern = NULL)
```

Arguments

`pattern`	Optional regex pattern to clear only matching keys. If `NULL` (default), clears the entire cache. For example, `"Contratos"` clears all cached contract queries.

Value

Invisibly returns the number of entries cleared.

See Also

`tce_cache_info()`

Examples

```
tce_cache_clear()              # clear everything
tce_cache_clear("Contratos")   # clear contract queries only
```

tce_cache_info *Show cache status*

Description

Returns a tibble describing all currently cached entries, including the cache key (endpoint + parameters), row count, timestamp and age. Useful for understanding what is cached and when it will expire.

Usage

```
tce_cache_info()
```

Value

A tibble with columns `key`, `rows`, `cached_at` and `age_secs`, or an empty tibble if the cache is empty. The cache TTL is controlled by `getOption("tceper.cache_ttl", 3600)`.

See Also

tce_cache_clear()

Examples

```
tce_cache_info()
```

`tce_catalog` *List available API endpoints*

Description

Returns a tibble listing all 72 endpoints available in the TCE-PE Open Data API, with their group, description and URL. Use `search` to filter by keyword.

Usage

```
tce_catalog(search = NULL)
```

Arguments

`search`	Optional character string to filter endpoints by name or description (case-insensitive). For example, `search = "licit"` returns all procurement-related endpoints.

Value

A tibble with columns `endpoint`, `group`, `title`, and `url`.

See Also

tce_params(), tce_fields(), tce_endpoint()

Examples

```
tce_catalog()
tce_catalog(search = "contrat")
tce_catalog(search = "municipio")
```

tce_commitment_liquidations

Commitment liquidations

Description

Retrieve liquidation records for budget commitments.

Usage

```
tce_commitment_liquidations(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `NumeroEmpenho`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other commitments: `tce_commitment_payments()`, `tce_commitment_summary()`, `tce_state_commitment_items()`, `tce_state_price_comparison()`

tce_commitment_payments

Commitment payments

Description

Retrieve payment records for budget commitments (municipal only).

Usage

```
tce_commitment_payments(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, NumeroEmpenho, AnoReferencia.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other commitments: tce_commitment_liquidations(), tce_commitment_summary(), tce_state_commitment_items()
tce_state_price_comparison()

---

tce_commitment_summary

Commitment summary

Description

Retrieve original values, reinforcements and cancellations for budget commitments (*empenhos*), for both municipal and state entities.

Usage

```
tce_commitment_summary(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, NumeroEmpenho, AnoReferencia.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other commitments: tce_commitment_liquidations(), tce_commitment_payments(), tce_state_commitment_items(), tce_state_price_comparison()

---

tce_consideranda *Consideranda*

Description

Retrieve *considerandos* (findings/grounds) from adjudicated processes.

Usage

```
tce_consideranda(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_debts_fines(), tce_determinations(), tce_outcomes(), tce_processes(), tce_recommendations(), tce_retirement_outcomes(), tce_special_accountability(), tce_spending_limits()

tce_contract_amendments

Contract amendments

Description

Retrieve addenda / amendments (*termos aditivos*) to contracts.

Usage

```
tce_contract_amendments(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`, `CodigoMunicipio`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other procurement: `tce_agreements()`, `tce_bid_details()`, `tce_bid_documents()`, `tce_bid_legal_basis()`, `tce_bid_modalities()`, `tce_bid_stages()`, `tce_bid_statuses()`, `tce_bids()`, `tce_contract_documents()`, `tce_contract_items()`, `tce_contracts()`, `tce_object_characteristics()`, `tce_object_classifications()`, `tce_object_nature()`

tce_contract_documents

Contract documents

Description

Retrieve documents associated with contracts.

Usage

```
tce_contract_documents(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

---

tce_contract_items *Contract line items*

Description

Retrieve the individual items that compose a contract.

Usage

```
tce_contract_items(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contracts(), tce_object_characteristics(), tce_object_classifications(), tce_object_nature()

---

tce_contracts *Contracts*

Description

Retrieve state and municipal contracts.

Usage

```
tce_contracts(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

tce_params() to discover available parameters, tce_fields() to see output columns.

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_object_characteristics(), tce_object_classificatio tce_object_nature()

Examples

```
## Not run:
# Not run: reaches the live TCE-PE API, which only accepts connections
# from Brazilian IP addresses (unreachable from CRAN / most CI).
tce_contracts(codigo_efisco_ug = "510101", verbose = TRUE)

## End(Not run)
```

`tce_creditor_types` *Creditor types*

Description

Lookup table of supplier/creditor types.

Usage

```
tce_creditor_types(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other reference: tce_entities(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipal_entities(), tce_municipalities(), tce_payroll_types(), tce_state_entities(), tce_sub_units()

tce_debts_fines *Debts and fines*

Description

Retrieve debts and fines imposed.

Usage

```
tce_debts_fines(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_consideranda(), tce_determinations(), tce_outcomes(), tce_processes(), tce_recommendations(), tce_retirement_outcomes(), tce_special_accountability(), tce_spending_limits()

tce_determinations *Determinations*

Description

Retrieve determinations from adjudicated electronic processes.

Usage

```
tce_determinations(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_consideranda(), tce_debts_fines(), tce_outcomes(), tce_processes(), tce_recommendations(), tce_retirement_outcomes(), tce_special_accountability(), tce_spending_limits()

---

tce_endpoint *Get the catalog entry for a given endpoint*

Description

Returns the full metadata for a single endpoint, including its input parameters (entrada) and output fields (saida). Useful for programmatic access to endpoint metadata.

Usage

```
tce_endpoint(endpoint)
```

Arguments

endpoint	Endpoint name (e.g., "Municipios", "LicitacaoUG", "Contratos"). Use tce_catalog() to see all available endpoints.

Value

A list with endpoint metadata, including entrada (input parameters) and saida (output fields) when available.

See Also

tce_catalog(), tce_params(), tce_fields()

Examples

```
tce_endpoint("Contratos")
```

`tce_entities` *Managed entities (all)*

Description

Retrieve all managed entities (*unidades jurisdicionadas*), state and municipal.

Usage

```
tce_entities(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `Esfera` (`"E"` state, `"M"` municipal), `Natureza`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipal_entities(), tce_municipalities(), tce_payroll_types(), tce_state_entities(), tce_sub_units()

tce_fields *List output fields for an endpoint*

Description

Returns a tibble describing the columns returned by a given endpoint. When `clean_names = TRUE` (the default), column names are converted from the API's original names to `snake_case`.

Usage

```
tce_fields(endpoint)
```

Arguments

`endpoint`	Endpoint name (e.g., `"Municipios"`, `"Contratos"`). Use `tce_catalog()` to see all available endpoints.

Value

A tibble with columns `name`, `type`, and `description`. Has a custom print method that summarises the field names at the top.

See Also

`tce_params()`, `tce_catalog()`, `tce_endpoint()`

Examples

```
tce_fields("Contratos")
tce_fields("Municipios")
```

tce_funding_sources *Funding source types*

Description

Lookup table of funding source types.

Usage

```
tce_funding_sources(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_inactivation_reasons(), tce_municipal_entities(
tce_municipalities(), tce_payroll_types(), tce_state_entities(), tce_sub_units()

---

tce_inactivation_reasons

Inactivation reason types

Description

Lookup table of inactivation reason types.

Usage

```
tce_inactivation_reasons(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_funding_sources(), tce_municipal_entities(),
tce_municipalities(), tce_payroll_types(), tce_state_entities(), tce_sub_units()

tce_legal_instrument_documents *Legal instrument documents (Remessa)*

Description

Legal instrument documents (Remessa)

Usage

```
tce_legal_instrument_documents(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoEfiscoUG.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_partic tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry(), tce_update_indices()

tce_legal_instrument_items *Legal instrument items (Remessa)*

Description

Legal instrument items (Remessa)

Usage

```
tce_legal_instrument_items(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoEfiscoUG.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_partic
tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry(), tce_update_indices()

`tce_legal_instrument_participants` *Legal instrument participants (Remessa)*

Description

Legal instrument participants (Remessa)

Usage

```
tce_legal_instrument_participants(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_benefit_types()`, `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_terms()`, `tce_legal_instruments()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_process_partic`
`tce_procurement_processes()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_remessa_works_geometry()`, `tce_update_indices()`

`tce_legal_instrument_terms` *Legal instrument terms (Remessa)*

Description

Amendment terms ("termos") of legal instruments: number, object, signature date, updated validity dates and values.

Usage

```
tce_legal_instrument_terms(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_benefit_types()`, `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_participants()`, `tce_legal_instruments()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_process_partic`
`tce_procurement_processes()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_remessa_works_geometry()`, `tce_update_indices()`

tce_legal_instruments *Legal instruments (Remessa)*

Description

Retrieve legal instruments from RemessaTCEPE.

Usage

```
tce_legal_instruments(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_benefit_types()`, `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_participants()`, `tce_legal_instrument_terms()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_process_partic`
`tce_procurement_processes()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_remessa_works_geometry()`, `tce_update_indices()`

tce_municipal_creditor_types *Municipal creditor types*

Description

Lookup table of municipal creditor types.

Usage

```
tce_municipal_creditor_types(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other expenditures: tce_municipal_expenditures(), tce_municipal_transfers(), tce_state_creditor_types(), tce_state_expenditures()

---

tce_municipal_entities

Municipal managed entities

Description

Retrieve municipal-level managed entities.

Usage

```
tce_municipal_entities(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipalities(), tce_payroll_types(), tce_state_entities(), tce_sub_units()

`tce_municipal_expenditures` *Municipal expenditures*

Description

Retrieve municipal-level expenditure data.

Usage

```
tce_municipal_expenditures(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other expenditures: tce_municipal_creditor_types(), tce_municipal_transfers(), tce_state_creditor_types() tce_state_expenditures()

`tce_municipal_revenues` *Municipal revenues*

Description

Retrieve municipal-level revenue data.

Usage

```
tce_municipal_revenues(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other revenues: `tce_budgeted_revenues()`, `tce_state_revenues()`

Examples

```
## Not run:
# Not run: reaches the live TCE-PE API, which only accepts connections
# from Brazilian IP addresses (unreachable from CRAN / most CI).
tce_municipal_revenues(id_unidade_gestora = "384")

## End(Not run)
```

tce_municipal_transfers *Municipal granted transfers*

Description

Retrieve inter-unit transfers granted within the same municipality.

Usage

```
tce_municipal_transfers(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other expenditures: `tce_municipal_creditor_types()`, `tce_municipal_expenditures()`, `tce_state_creditor_type`
`tce_state_expenditures()`

tce_municipalities *Municipalities*

Description

Retrieve the list of municipalities in Pernambuco.

Usage

```
tce_municipalities(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipal_entities(), tce_payroll_types(), tce_state_entities(), tce_sub_units()

---

tce_object_characteristics

Object characteristics

Description

Lookup table of procurement object characteristic types.

Usage

```
tce_object_characteristics(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_classifications(), tce_object_nature()

---

tce_object_classifications

Object classifications

Description

Lookup table of procurement object classification types.

Usage

```
tce_object_classifications(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_nature()

tce_object_nature *Object nature types*

Description

Lookup table of procurement object nature types.

Usage

```
tce_object_nature(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other procurement: tce_agreements(), tce_bid_details(), tce_bid_documents(), tce_bid_legal_basis(), tce_bid_modalities(), tce_bid_stages(), tce_bid_statuses(), tce_bids(), tce_contract_amendments(), tce_contract_documents(), tce_contract_items(), tce_contracts(), tce_object_characteristics(), tce_object_classifications()

tce_outcomes *Process outcomes*

Description

Retrieve outcomes of adjudicated electronic processes.

Usage

```
tce_outcomes(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_consideranda(), tce_debts_fines(), tce_determinations(), tce_processes(), tce_recommendations(), tce_retirement_outcomes(), tce_special_accountability(), tce_spending_limits()

---

tce_params *List input parameters for an endpoint*

Description

Returns a tibble describing the query parameters accepted by a given endpoint. The api_name column shows the exact name expected by the API; the r_name column shows the snake_case equivalent that you can pass to wrapper functions.

Usage

```
tce_params(endpoint)
```

Arguments

endpoint	Endpoint name (e.g., "Municipios", "Contratos"). Use tce_catalog() to see all available endpoints.

Value

A tibble with columns api_name, r_name, required, type, and description. Has a custom print method that summarises the parameter names at the top.

See Also

tce_fields(), tce_catalog(), tce_endpoint()

Examples

```
tce_params("Contratos")
tce_params("Municipios")
tce_params("LicitacaoUG")
```

`tce_payroll_types` *Payroll types*

Description

Lookup table of payroll (*folha*) types.

Usage

```
tce_payroll_types(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipal_entities(), tce_municipalities(), tce_state_entities(), tce_sub_units()

tce_person_creditor_types *Person creditor types*

Description

Lookup table of person creditor types.

Usage

```
tce_person_creditor_types(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CpfCnpj`, `Nome`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other suppliers: `tce_sanctions()`, `tce_suppliers()`

tce_processes *Legal processes*

Description

Retrieve the list of formalized physical and electronic processes.

Usage

```
tce_processes(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_consideranda(), tce_debts_fines(), tce_determinations(), tce_outcomes(), tce_recommendations(), tce_retirement_outcomes(), tce_special_accountability(), tce_spending_limits()

---

tce_procurement_process_budget

Procurement process budget (Remessa)

Description

Procurement process budget (Remessa)

Usage

```
tce_procurement_process_budget(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoEfiscoUG.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry(), tce_update_indices()

---

tce_procurement_process_documents

Procurement process documents (Remessa)

Description

Procurement process documents (Remessa)

Usage

```
tce_procurement_process_documents(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoEfiscoUG.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry(), tce_update_indices()

`tce_procurement_process_participants` *Procurement process participants (Remessa)*

Description

Procurement process participants (Remessa)

Usage

```
tce_procurement_process_participants(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_benefit_types()`, `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_participants()`, `tce_legal_instrument_terms()`, `tce_legal_instruments()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_processes()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_remessa_works_geometry()`, `tce_update_indices()`

`tce_procurement_processes` *Procurement processes (Remessa)*

Description

Retrieve procurement processes from RemessaTCEPE.

Usage

```
tce_procurement_processes(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_benefit_types()`, `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_participants()`, `tce_legal_instrument_terms()`, `tce_legal_instruments()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_process_participants()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_remessa_works_geometry`
`tce_update_indices()`

tce_public_works *Public works*

Description

Retrieve engineering works and services overseen by the Court of Accounts.

Usage

```
tce_public_works(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other public_works: tce_public_works_audits(), tce_public_works_contractors()

tce_public_works_audits *Public works audit data*

Description

Retrieve audit data on public works.

Usage

```
tce_public_works_audits(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other public_works: tce_public_works(), tce_public_works_contractors()

---

tce_public_works_contractors

Public works contractor data

Description

Retrieve data on companies contracted for public works.

Usage

```
tce_public_works_contractors(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other public_works: `tce_public_works()`, `tce_public_works_audits()`

`tce_recommendations` *Recommendations*

Description

Retrieve recommendations from adjudicated electronic processes.

Usage

```
tce_recommendations(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`, `CodigoMunicipio`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other processes: `tce_consideranda()`, `tce_debts_fines()`, `tce_determinations()`, `tce_outcomes()`, `tce_processes()`, `tce_retirement_outcomes()`, `tce_special_accountability()`, `tce_spending_limits()`

tce_reference_codes *Reference codes (Remessa)*

Description

Reference codes for cost-estimate item imports.

Usage

```
tce_reference_codes(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry tce_update_indices()

tce_reference_dates *Reference dates (Remessa)*

Description

Available dates for reference sources in cost-estimate imports.

Usage

```
tce_reference_dates(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry tce_update_indices()

---

tce_reference_sources *Reference sources (Remessa)*

Description

Lookup table of reference sources for cost-estimate imports.

Usage

```
tce_reference_sources(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry(), tce_update_indices()

---

tce_remessa_works *Public works details (Remessa)*

Description

Public works details (Remessa)

Usage

```
tce_remessa_works(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoEfiscoUG.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works_execution(), tce_remessa_works_geomet tce_update_indices()

`tce_remessa_works_execution` *Public works execution (Remessa)*

Description

Public works execution (Remessa)

Usage

```
tce_remessa_works_execution(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_geometry(), tce_update_indices()

`tce_remessa_works_geometry` *Public works geometry (Remessa)*

Description

Public works geometry (Remessa)

Usage

```
tce_remessa_works_geometry(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters. Common: `CodigoEfiscoUG`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other remessa: `tce_benefit_types()`, `tce_budget_statuses()`, `tce_legal_instrument_documents()`, `tce_legal_instrument_items()`, `tce_legal_instrument_participants()`, `tce_legal_instrument_terms()`, `tce_legal_instruments()`, `tce_procurement_process_budget()`, `tce_procurement_process_documents()`, `tce_procurement_process_participants()`, `tce_procurement_processes()`, `tce_reference_codes()`, `tce_reference_dates()`, `tce_reference_sources()`, `tce_remessa_works()`, `tce_remessa_works_execution()`, `tce_update_indices()`

`tce_request` *Perform a request to the TCE-PE Open Data API*

Description

Low-level function that sends a GET request to the TCE-PE Open Data API and returns the response content as a tibble. Higher-level wrapper functions such as `tce_contracts()`, `tce_state_expenditures()`, etc. call this function internally.

You can use `tce_request()` directly to access any API endpoint, including those that do not yet have a dedicated wrapper.

Usage

```
tce_request(
  endpoint,
  ...,
  clean_names = TRUE,
  max_tries = 3,
  progress = getOption("tceper.progress", TRUE),
  verbose = getOption("tceper.verbose", FALSE)
)
```

Arguments

`endpoint`	A character string with the API method name (e.g., `"Contratos"`, `"DespesasEstaduais"`). See `https://sistemas.tcepe.tc.br/DadosAbertos/Exemplo!listar` for the full list of available endpoints.
`...`	Named query parameters. You can use either the API names (`CamelCase`/`UPPERCASE`) or the snake_case aliases shown by `tce_params()`. `NULL` values are silently dropped.
`clean_names`	If `TRUE` (default), column names are converted to `snake_case` via `janitor::clean_names()`. Set to `FALSE` to keep the original column names returned by the API.
`max_tries`	Maximum number of retry attempts for failed requests (default: `3`). Passed to `httr2::req_retry()`.
`progress`	If `TRUE`, displays progress messages via cli. Defaults to the value of `getOption("tceper.progress")`, or `TRUE` if unset.
`verbose`	If `TRUE`, prints the final API URL, parameters, and response metadata (status, content-type, size, elapsed time). Defaults to `getOption("tceper.verbose", FALSE)`.

Details

API base URL:
`https://sistemas.tcepe.tc.br/DadosAbertos/`

Network access:
The API host only accepts connections from **Brazilian IP addresses**. Calls from outside Brazil are dropped at the network level and surface as a connection or TLS error. The offline discovery functions (`tce_catalog()`, `tce_params()`, `tce_fields()`) do not require network access and work anywhere.

URL construction:
The TCE-PE API uses Struts2-style URLs where `!` invokes a method (e.g. `Contratos!json`). The URL is built with `paste0()` to preserve the literal `!` – using `httr2::req_url_path_append()` would encode it as `%21`, causing the server to ignore all query parameters.

Response limit:
The API returns at most **100,000 records** per request. When this limit is reached, a warning is issued suggesting the use of additional filters.

Encoding:
The TCE-PE backend runs on **ISO-8859-1** (Latin-1), not UTF-8. The package handles this transparently in both directions:

- **Response body** is decoded as Latin-1 and returned as UTF-8 R strings, so accented Portuguese characters come through cleanly (`São`, `Município`, `Conceição`, etc.).
- **Query parameters** are transcoded UTF-8 -> Latin-1 before percent-encoding, so filtering by accented values works as expected (e.g. `tce_municipalities(municipio = "São José da Coroa Grande")`). You can pass any R string with accents – the conversion is automatic.

Options:

- `tceper.progress` – set to `FALSE` to suppress all progress messages.
- `tceper.verbose` – set to `TRUE` to always print request/response details.
- `tceper.cache_ttl` – cache time-to-live in seconds (default: `3600`).

Value

A tibble with the response content. Returns a zero-row tibble if no records are found.

Examples

```
## Not run:
# Not run: these examples reach the live TCE-PE API, which only accepts
# connections from Brazilian IP addresses and is therefore unreachable
# from CRAN check machines and most CI runners.

# Direct query to any endpoint
tce_request("Contratos", CodigoEfiscoUG = "510101")

# Show full request/response details
tce_request("Contratos", CodigoEfiscoUG = "510101", verbose = TRUE)

# Keep original column names
tce_request("Municipios", clean_names = FALSE)

## End(Not run)
```

tce_retirement_outcomes *Retirement, pension and reform outcomes*

Description

Retrieve outcomes from processes related to retirement, pension and reform (*Aposentadoria, Pensão e Reforma*).

Usage

```
tce_retirement_outcomes(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`, `CodigoMunicipio`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other processes: `tce_consideranda()`, `tce_debts_fines()`, `tce_determinations()`, `tce_outcomes()`, `tce_processes()`, `tce_recommendations()`, `tce_special_accountability()`, `tce_spending_limits()`

tce_sanctions *Sanctions*

Description

Retrieve sanctions applied to suppliers.

Usage

```
tce_sanctions(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CpfCnpj, Nome.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other suppliers: tce_person_creditor_types(), tce_suppliers()

---

tce_school_transport *School transport*

Description

Retrieve school transport services overseen by the Court of Accounts.

Usage

```
tce_school_transport(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

---

tce_servants *Public servants*

Description

Retrieve municipal and state public servants (SAGRES Personnel).

Usage

```
tce_servants(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, Nome, Cpf, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

---

tce_special_accountability *Special accountability outcomes*

Description

Retrieve outcomes from special accountability processes (*Tomada de Conta Especial*).

Usage

```
tce_special_accountability(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_consideranda(), tce_debts_fines(), tce_determinations(), tce_outcomes(), tce_processes(), tce_recommendations(), tce_retirement_outcomes(), tce_spending_limits()

---

tce_spending_limits *Spending limits*

Description

Retrieve spending-limit data from government account processes.

Usage

```
tce_spending_limits(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia, CodigoMunicipio.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other processes: tce_consideranda(), tce_debts_fines(), tce_determinations(), tce_outcomes(), tce_processes(), tce_recommendations(), tce_retirement_outcomes(), tce_special_accountability()

tce_state_commitment_items

State commitment line items

Description

Retrieve individual line items of state-level commitments.

Usage

```
tce_state_commitment_items(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `NumeroEmpenho`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other commitments: `tce_commitment_liquidations()`, `tce_commitment_payments()`, `tce_commitment_summary()`, `tce_state_price_comparison()`

tce_state_creditor_types

State creditor types

Description

Lookup table of state creditor types.

Usage

```
tce_state_creditor_types(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other expenditures: tce_municipal_creditor_types(), tce_municipal_expenditures(), tce_municipal_transfers tce_state_expenditures()

---

tce_state_entities *State managed entities*

Description

Retrieve state-level managed entities.

Usage

```
tce_state_entities(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipal_entities(), tce_municipalities(), tce_payroll_types(), tce_sub_units()

---

tce_state_expenditures

State expenditures

Description

Retrieve state-level expenditure data.

Usage

```
tce_state_expenditures(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CodigoEfiscoUG, AnoReferencia.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

tce_params() to discover available parameters, tce_fields() to see output columns.

Other expenditures: tce_municipal_creditor_types(), tce_municipal_expenditures(), tce_municipal_transfers tce_state_creditor_types()

Examples

```
## Not run:
# Not run: reaches the live TCE-PE API, which only accepts connections
# from Brazilian IP addresses (unreachable from CRAN / most CI).
tce_state_expenditures(id_unidade_gestora = "510101")

## End(Not run)
```

`tce_state_price_comparison` *State price comparison*

Description

Retrieve state commitment items with values for price comparison.

Usage

```
tce_state_price_comparison(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `NumeroEmpenho`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

Other commitments: `tce_commitment_liquidations()`, `tce_commitment_payments()`, `tce_commitment_summary()`, `tce_state_commitment_items()`

`tce_state_revenues` *State revenues*

Description

Retrieve state-level revenue data from the TCE-PE API.

Usage

```
tce_state_revenues(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

`...`	Named query parameters passed directly to the API. Common filters: `CodigoEfiscoUG`, `AnoReferencia`.
`verbose`	If `TRUE`, prints the final API URL.
`cache`	If `TRUE` (default), returns cached results when available.
`clean_names`	If `TRUE` (default), converts column names to `snake_case`.

Value

A tibble.

See Also

`tce_params()` to discover available parameters, `tce_fields()` to see output columns.

Other revenues: `tce_budgeted_revenues()`, `tce_municipal_revenues()`

Examples

```
## Not run:
# Not run: reaches the live TCE-PE API, which only accepts connections
# from Brazilian IP addresses (unreachable from CRAN / most CI).
tce_state_revenues(id_unidade_gestora = "510101")

## End(Not run)
```

tce_sub_units *Sub-units of managed entities*

Description

Retrieve sub-units belonging to managed entities.

Usage

```
tce_sub_units(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters. Common: CodigoEfiscoUG.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other reference: tce_creditor_types(), tce_entities(), tce_funding_sources(), tce_inactivation_reasons(), tce_municipal_entities(), tce_municipalities(), tce_payroll_types(), tce_state_entities()

tce_suppliers *Suppliers*

Description

Retrieve registered suppliers. Partial matching is supported.

Usage

```
tce_suppliers(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API. Common filters: CpfCnpj, Nome.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other suppliers: tce_person_creditor_types(), tce_sanctions()

---

tce_update_indices *Price update indices (Remessa)*

Description

Indices for updating prices in cost-estimate spreadsheets.

Usage

```
tce_update_indices(
  ...,
  verbose = getOption("tceper.verbose", FALSE),
  cache = TRUE,
  clean_names = TRUE
)
```

Arguments

...	Named query parameters passed directly to the API.
verbose	If TRUE, prints the final API URL.
cache	If TRUE (default), returns cached results when available.
clean_names	If TRUE (default), converts column names to snake_case.

Value

A tibble.

See Also

Other remessa: tce_benefit_types(), tce_budget_statuses(), tce_legal_instrument_documents(), tce_legal_instrument_items(), tce_legal_instrument_participants(), tce_legal_instrument_terms(), tce_legal_instruments(), tce_procurement_process_budget(), tce_procurement_process_documents(), tce_procurement_process_participants(), tce_procurement_processes(), tce_reference_codes(), tce_reference_dates(), tce_reference_sources(), tce_remessa_works(), tce_remessa_works_execution(), tce_remessa_works_geometry()

Index

∗ **commitments**
  tce_commitment_liquidations, 15
  tce_commitment_payments, 15
  tce_commitment_summary, 16
  tce_state_commitment_items, 60
  tce_state_price_comparison, 63
∗ **expenditures**
  tce_municipal_creditor_types, 31
  tce_municipal_expenditures, 33
  tce_municipal_transfers, 35
  tce_state_creditor_types, 60
  tce_state_expenditures, 62
∗ **personnel**
  tce_servants, 58
∗ **processes**
  tce_consideranda, 17
  tce_debts_fines, 22
  tce_determinations, 22
  tce_outcomes, 38
  tce_processes, 41
  tce_recommendations, 48
  tce_retirement_outcomes, 56
  tce_special_accountability, 58
  tce_spending_limits, 59
∗ **procurement**
  tce_agreements, 4
  tce_bid_details, 5
  tce_bid_documents, 6
  tce_bid_legal_basis, 7
  tce_bid_modalities, 8
  tce_bid_stages, 9
  tce_bid_statuses, 9
  tce_bids, 10
  tce_contract_amendments, 18
  tce_contract_documents, 18
  tce_contract_items, 19
  tce_contracts, 20
  tce_object_characteristics, 36
  tce_object_classifications, 37
  tce_object_nature, 38
∗ **public_works**
  tce_public_works, 46
  tce_public_works_audits, 46
  tce_public_works_contractors, 47
∗ **reference**
  tce_creditor_types, 21
  tce_entities, 24
  tce_funding_sources, 25
  tce_inactivation_reasons, 26
  tce_municipal_entities, 32
  tce_municipalities, 35
  tce_payroll_types, 40
  tce_state_entities, 61
  tce_sub_units, 65
∗ **remessa**
  tce_benefit_types, 5
  tce_budget_statuses, 11
  tce_legal_instrument_documents, 27
  tce_legal_instrument_items, 28
  tce_legal_instrument_participants, 29
  tce_legal_instrument_terms, 30
  tce_legal_instruments, 31
  tce_procurement_process_budget, 42
  tce_procurement_process_documents, 43
  tce_procurement_process_participants, 44
  tce_procurement_processes, 45
  tce_reference_codes, 49
  tce_reference_dates, 49
  tce_reference_sources, 50
  tce_remessa_works, 51
  tce_remessa_works_execution, 52
  tce_remessa_works_geometry, 53
  tce_update_indices, 66
∗ **revenues**
  tce_budgeted_revenues, 12

tce_municipal_revenues, 34
tce_state_revenues, 64
∗ **suppliers**
tce_person_creditor_types, 41
tce_sanctions, 56
tce_suppliers, 65

cli, *54*

httr2::req_retry(), *54*

janitor::clean_names(), *54*

tce_agreements, 4
tce_agreements(), *6–11*, *18–21*, *37, 38*
tce_benefit_types, 5
tce_benefit_types(), *12*, *27–31*, *43–45*, *49–53*, *67*
tce_bid_details, 5
tce_bid_details(), *4*, *7–11*, *18–21*, *37, 38*
tce_bid_documents, 6
tce_bid_documents(), *4*, *6*, *8–11*, *18–21*, *37, 38*
tce_bid_legal_basis, 7
tce_bid_legal_basis(), *4*, *6–11*, *18–21*, *37, 38*
tce_bid_modalities, 8
tce_bid_modalities(), *4*, *6–11*, *18–21*, *37, 38*
tce_bid_stages, 9
tce_bid_stages(), *4*, *6–8*, *10, 11*, *18–21*, *37, 38*
tce_bid_statuses, 9
tce_bid_statuses(), *4*, *6–9*, *11*, *18–21*, *37, 38*
tce_bids, 10
tce_bids(), *4*, *6–10*, *18–21*, *37, 38*
tce_budget_statuses, 11
tce_budget_statuses(), *5*, *27–31*, *43–45*, *49–53*, *67*
tce_budgeted_revenues, 12
tce_budgeted_revenues(), *34*, *64*
tce_cache_clear, 13
tce_cache_clear(), *14*
tce_cache_info, 13
tce_cache_info(), *13*
tce_catalog, 14
tce_catalog(), *23–25*, *39, 40*, *55*
tce_commitment_liquidations, 15
tce_commitment_liquidations(), *16, 17*, *60*, *63*
tce_commitment_payments, 15
tce_commitment_payments(), *15*, *17*, *60*, *63*
tce_commitment_summary, 16
tce_commitment_summary(), *15, 16*, *60*, *63*
tce_consideranda, 17
tce_consideranda(), *22, 23*, *39*, *42*, *48*, *56*, *59*
tce_contract_amendments, 18
tce_contract_amendments(), *4*, *6–11*, *19–21*, *37, 38*
tce_contract_documents, 18
tce_contract_documents(), *4*, *6–11*, *18*, *20, 21*, *37, 38*
tce_contract_items, 19
tce_contract_items(), *4*, *6–11*, *18, 19*, *21*, *37, 38*
tce_contracts, 20
tce_contracts(), *4*, *6–11*, *18–20*, *37, 38*, *54*
tce_creditor_types, 21
tce_creditor_types(), *24*, *26*, *33*, *36*, *40*, *62*, *65*
tce_debts_fines, 22
tce_debts_fines(), *17*, *23*, *39*, *42*, *48*, *56*, *59*
tce_determinations, 22
tce_determinations(), *17*, *22*, *39*, *42*, *48*, *56*, *59*
tce_endpoint, 23
tce_endpoint(), *14*, *25*, *40*
tce_entities, 24
tce_entities(), *21*, *26*, *33*, *36*, *40*, *62*, *65*
tce_fields, 25
tce_fields(), *14*, *21*, *24*, *40*, *55*, *62*, *64*
tce_funding_sources, 25
tce_funding_sources(), *21*, *24*, *26*, *33*, *36*, *40*, *62*, *65*
tce_inactivation_reasons, 26
tce_inactivation_reasons(), *21*, *24*, *26*, *33*, *36*, *40*, *62*, *65*
tce_legal_instrument_documents, 27
tce_legal_instrument_documents(), *5*, *12*, *28–31*, *43–45*, *49–53*, *67*
tce_legal_instrument_items, 28
tce_legal_instrument_items(), *5*, *12*, *27*, *29–31*, *43–45*, *49–53*, *67*
tce_legal_instrument_participants, 29
tce_legal_instrument_participants(), *5*,

12, *27, 28*, *30, 31*, *43–45*, *49–53*, *67*
tce_legal_instrument_terms, 30
tce_legal_instrument_terms(), *5*, *12*, *27–29*, *31*, *43–45*, *49–53*, *67*
tce_legal_instruments, 31
tce_legal_instruments(), *5*, *12*, *27–30*, *43–45*, *49–53*, *67*
tce_municipal_creditor_types, 31
tce_municipal_creditor_types(), *33*, *35*, *61, 62*
tce_municipal_entities, 32
tce_municipal_entities(), *21*, *24*, *26*, *36*, *40*, *62*, *65*
tce_municipal_expenditures, 33
tce_municipal_expenditures(), *32*, *35*, *61, 62*
tce_municipal_revenues, 34
tce_municipal_revenues(), *12*, *64*
tce_municipal_transfers, 35
tce_municipal_transfers(), *32, 33*, *61, 62*
tce_municipalities, 35
tce_municipalities(), *21*, *24*, *26*, *33*, *40*, *62*, *65*
tce_object_characteristics, 36
tce_object_characteristics(), *4*, *6–11*, *18–21*, *37, 38*
tce_object_classifications, 37
tce_object_classifications(), *4*, *6–11*, *18–21*, *37, 38*
tce_object_nature, 38
tce_object_nature(), *4*, *6–11*, *18–21*, *37*
tce_outcomes, 38
tce_outcomes(), *17*, *22, 23*, *42*, *48*, *56*, *59*
tce_params, 39
tce_params(), *14*, *21*, *24, 25*, *54, 55*, *62*, *64*
tce_payroll_types, 40
tce_payroll_types(), *21*, *24*, *26*, *33*, *36*, *62*, *65*
tce_person_creditor_types, 41
tce_person_creditor_types(), *57*, *66*
tce_processes, 41
tce_processes(), *17*, *22, 23*, *39*, *48*, *56*, *59*
tce_procurement_process_budget, 42
tce_procurement_process_budget(), *5*, *12*, *27–31*, *43–45*, *49–53*, *67*
tce_procurement_process_documents, 43
tce_procurement_process_documents(), *5*, *12*, *27–31*, *43–45*, *49–53*, *67*
tce_procurement_process_participants, 44
tce_procurement_process_participants(), *5*, *12*, *27–31*, *43*, *45*, *49–53*, *67*
tce_procurement_processes, 45
tce_procurement_processes(), *5*, *12*, *27–31*, *43, 44*, *49–53*, *67*
tce_public_works, 46
tce_public_works(), *47, 48*
tce_public_works_audits, 46
tce_public_works_audits(), *46*, *48*
tce_public_works_contractors, 47
tce_public_works_contractors(), *46, 47*
tce_recommendations, 48
tce_recommendations(), *17*, *22, 23*, *39*, *42*, *56*, *59*
tce_reference_codes, 49
tce_reference_codes(), *5*, *12*, *27–31*, *43–45*, *50–53*, *67*
tce_reference_dates, 49
tce_reference_dates(), *5*, *12*, *27–31*, *43–45*, *49*, *51–53*, *67*
tce_reference_sources, 50
tce_reference_sources(), *5*, *12*, *27–31*, *43–45*, *49, 50*, *52, 53*, *67*
tce_remessa_works, 51
tce_remessa_works(), *5*, *12*, *27–31*, *43–45*, *49–53*, *67*
tce_remessa_works_execution, 52
tce_remessa_works_execution(), *5*, *12*, *27–31*, *43–45*, *49–53*, *67*
tce_remessa_works_geometry, 53
tce_remessa_works_geometry(), *5*, *12*, *27–31*, *43–45*, *49–52*, *67*
tce_request, 54
tce_retirement_outcomes, 56
tce_retirement_outcomes(), *17*, *22, 23*, *39*, *42*, *48*, *59*
tce_sanctions, 56
tce_sanctions(), *41*, *66*
tce_school_transport, 57
tce_servants, 58
tce_special_accountability, 58
tce_special_accountability(), *17*, *22, 23*, *39*, *42*, *48*, *56*, *59*
tce_spending_limits, 59
tce_spending_limits(), *17*, *22, 23*, *39*, *42*, *48*, *56*, *59*

tce_state_commitment_items, 60
tce_state_commitment_items(), *15–17*, *63*
tce_state_creditor_types, 60
tce_state_creditor_types(), *32, 33*, *35*, *62*
tce_state_entities, 61
tce_state_entities(), *21*, *24*, *26*, *33*, *36*, *40*, *65*
tce_state_expenditures, 62
tce_state_expenditures(), *32, 33*, *35*, *54*, *61*
tce_state_price_comparison, 63
tce_state_price_comparison(), *15–17*, *60*
tce_state_revenues, 64
tce_state_revenues(), *12*, *34*
tce_sub_units, 65
tce_sub_units(), *21*, *24*, *26*, *33*, *36*, *40*, *62*
tce_suppliers, 65
tce_suppliers(), *41*, *57*
tce_update_indices, 66
tce_update_indices(), *5*, *12*, *27–31*, *43–45*, *49–53*
tibble, *4–13*, *15–24*, *26–66*