

Package ‘legco’

October 16, 2021

Type Package

Title R Bindings for the Hong Kong Legislative Council API

Version 0.1.4

Author Elgar Teo

Maintainer Elgar Teo <elgarteo@connect.hku.hk>

URL <https://github.com/elgarteo/legco>

BugReports <https://github.com/elgarteo/legco/issues>

Description Fetching data from Hong Kong Legislative Council's open data API in R.

Functions correspond to the data endpoints of the API.

Documentations of supported API databases:

<<https://www.legco.gov.hk/odata/english/billsdb.html>>,

<<https://www.legco.gov.hk/odata/english/hansard-db.html>>,

<<https://www.legco.gov.hk/odata/english/attendance-db.html>>,

<<https://www.legco.gov.hk/odata/english/schedule-db.html>> and

<<https://www.legco.gov.hk/odata/english/vrdb.html>>.

Imports dplyr, httr, jsonlite, utils

License MIT + file LICENSE

Encoding UTF-8

Depends R(>= 3.5.0)

LazyData true

RoxygenNote 7.1.2

Suggests curl, httpptest, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2021-10-16 14:10:06 UTC

R topics documented:

addresses	2
all_bills	5
attendance	8
bills	9
committee	12
comp-fun	13
hansard	14
hansard-db	16
legco_api	17
legco_section_type	18
meeting	19
meeting_committee	21
meeting_schedule-db	23
member	24
membership	25
member_term	27
motions	28
petitions	31
questions	33
rundown	35
search_committee	37
search_member	39
search_question	40
search_voting_record	42
session	43
speakers	45
statements	47
subjects	49
summoning_bells	51
term	53
voting_record	55
voting_results	57
Index	60

addresses

Addresses Delivered in LegCo

Description

Fetch addresses made by members or government officials while presenting papers to the Council.

Usage

```
addresses(
  hansard_id = NULL,
  rundown_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_addresses(
  hansard_id = NULL,
  rundown_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.
<code>floor</code>	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.

n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Addresses* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch addresses given during the Council meeting on February 20, 2019
x <- addresses(hansard_id = 2714)
```

`all_bills`*All Bills in LegCo*

Description

Fetch the information of bills presented in LegCo since 1906.

Usage

```
all_bills(  
  id = NULL,  
  ordinance = NULL,  
  title = NULL,  
  proposer = NULL,  
  gazette_from = "all",  
  gazette_to = "all",  
  first_from = "all",  
  first_to = "all",  
  second_from = "all",  
  second_to = "all",  
  third_from = "all",  
  third_to = "all",  
  n = 10000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

```
legco_all_bills(  
  id = NULL,  
  ordinance = NULL,  
  title = NULL,  
  proposer = NULL,  
  gazette_from = "all",  
  gazette_to = "all",  
  first_from = "all",  
  first_to = "all",  
  second_from = "all",  
  second_to = "all",  
  third_from = "all",  
  third_to = "all",  
  n = 10000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

Arguments

id	the id of a bill, or a vector of ids. Defaults to NULL.
ordinance	the ordinance associated with the bill. Takes the full English name of the ordinance, e.g. "Buildings Ordinance". Defaults to NULL.
title	the title of the bill. Takes the full English name of the bill, e.g. "National Anthem Bill". Defaults to NULL.
proposer	the proposer of the bill. Takes the full English title of the proposer, e.g. "Secretary for Security". Defaults to NULL.
gazette_from	only fetch bills gazetted on or after this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not been gazetted before being tabled in LegCo and overrides the gazette_to parameter. If NULL, returns bills that have not been gazetted before being tabled in LegCo (mainly older bills as newer ones required to be gazetted before tabling as required by the Rules of Procedure) and overrides the gazette_to parameter. Defaults to "all".
gazette_to	only fetch bills gazetted on or before this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not been gazetted before being tabled in LegCo and overrides the gazette_from parameter. If NULL, returns bills that have not been gazetted before being tabled in LegCo (mainly older bills as newer ones required to be gazetted before tabling as required by the Rules of Procedure) and overrides the gazette_from parameter. Defaults to "all".
first_from	only fetch bills that have gone through first reading on or after this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not gone through first reading and overrides the first_to parameter. If NULL, returns bills that have not gone through first reading and overrides the first_to parameter. Defaults to "all".
first_to	only fetch bills that have gone through first reading on or before this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not gone through first reading and overrides the first_from parameter. If NULL, returns bills that have not gone through first reading and overrides the first_from parameter. Defaults to "all".
second_from	only fetch bills that have gone through second reading on or after this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not gone through second reading and overrides the second_to parameter. If NULL, returns bills that have not gone through second reading and overrides the second_to parameter. Defaults to "all".

second_to	only fetch bills that have gone through second reading on or before this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not gone through second reading and overrides the second_from parameter. If NULL, returns bills that have not gone through second reading and overrides the second_from parameter. Defaults to "all".
third_from	only fetch bills that have gone through third reading on or after this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not gone through third reading and overrides the third_to parameter. If NULL, returns bills that have not gone through third reading and overrides the third_to parameter. Defaults to "all".
third_to	only fetch bills that have gone through third reading on or before this date. Accepts "all", NULL or character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). If "all", returns all bills that have and have not gone through third reading and overrides the third_from parameter. If NULL, returns bills that have not gone through third reading and overrides the third_from parameter. Defaults to "all".
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Vbills* data endpoint of the Bills Database.

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/billsdb.html>

Examples

```
# Fetch bills that passed third reading on February 20, 2019
x <- all_bills(third_from = "2019-02-20")
```

attendance	<i>Attendance of LegCo Members</i>
------------	------------------------------------

Description

Fetch attendance record of LegCo committee meetings

Usage

```
attendance(
  committee_id = NULL,
  meet_id = NULL,
  member_id = NULL,
  attn = "all",
  from = "1900-01-01",
  to = Sys.Date(),
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_attendance(
  committee_id = NULL,
  meet_id = NULL,
  member_id = NULL,
  attn = "all",
  from = "1900-01-01",
  to = Sys.Date(),
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

committee_id	the id of a committee, or a vector of ids. If NULL, returns results of all committees. Defaults to NULL.
meet_id	the id of a meeting, or a vector of ids. If NULL, returns all meetings. Useful for matching meeting with records from the Attendance Database. Defaults to NULL.
member_id	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
attn	the attendance. "p" returns members who were present. "a" returns members who were absent. "all" returns all members. Defaults to "all".

from	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01".
to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tattendance* data endpoint of the Meeting Attendance Database.

See Also

LegCo API documentation for the Attendance database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch members who were absent from the Subcommittee on Issues
# Relating to Bazaars meetings on October 13, 2017
x <- attendance(committee_id = 2704, from = "2017-10-13", to = "2017-10-13", attn = "a")
```

bills

Bills in LegCo

Description

Fetch bills presented in LegCo council meetings.

Usage

```
bills(
  rundown_id = NULL,
  hansard_id = NULL,
  section_code = NULL,
  lang = "en",
```

```

    from = "1900-01-01",
    to = Sys.Date(),
    floor = FALSE,
    n = 1000,
    extra_param = NULL,
    count = FALSE,
    verbose = TRUE
)

legco_bills(
  rundown_id = NULL,
  hansard_id = NULL,
  section_code = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

```

Arguments

<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>section_code</code>	the section code, or a vector of codes. If NULL, returns results of sections. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.
<code>floor</code>	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
<code>n</code>	the number of record to fetch. Defaults to 1000.

extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Bills* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch bills debated during the Council meeting on February 20, 2019
x <- bills(hansard_id = 2714)
```

 committee

Committees of LegCo

Description

Fetch basic information of LegCo committees.

Usage

```
committee(
  committee_id = NULL,
  code = NULL,
  term_id = NULL,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_committee(
  committee_id = NULL,
  code = NULL,
  term_id = NULL,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

committee_id	the id of a committee, or a vector of ids. If NULL, returns results of all committees. Defaults to NULL.
code	the code of a committee, or a vector of ids. If NULL, returns all committees. Defaults to NULL.
term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tcommittee* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- `term`: LegCo terms
- `session`: LegCo sessions
- `committee`: LegCo committees
- `membership`: Membership of LegCo committees
- `member`: LegCo members
- `member_term`: Terms served by LegCo members
- `meeting`: Meetings of LegCo committees
- `meeting_committee`: Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch all LegCo committees from the sixth term
x <- committee(term_id = 5)
```

comp-fun

Complementary Functions

Description

This set of complementary functions are wrappers around the database functions that facilitate the use of the LegCo API or return data from the API in a more usable form. All complementary functions have a prefix of `search_`.

Functions

Complementary Functions:

- `search_committee`: Search LegCo committees
- `search_member`: Search LegCo members
- `search_voting_record`: Search Voting Record in LegCo meetings
- `search_question`: Search full text of question put to the government by LegCo members

Notes

The complementary functions work by calling a number of database functions and combining their output, meaning that a function call usually involves multiple API calls. Use with caution to prevent reaching the API's rate limit too quickly.

hansard

Hansard of LegCo

Description

Fetch metadata and URLs of the hansard files of LegCo council meetings.

Usage

```
hansard(
  hansard_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_hansard(
  hansard_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.

floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Hansard* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch metadata of hansard files of all LegCo meetings
# conducted between February 20 to March 20, 2019
x <- hansard(from = "2019-02-20", to = "2019-03-20")
```

hansard-db

*Hansard Database***Description**

This database contains information on matters discussed in Council meeting since the fifth term of LegCo. Hinting by its name, the database is built upon the PDF hansard files. Relying on the bookmarks and section codes of the hansard files, the data endpoints retrieve data from the files directly.

Arguments

hansard_id	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
speaker_id	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
rundown_id	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
bookmark_name	the bookmark name of the rundown, or a vector of names. Defaults to NULL.
section_code	the section code, or a vector of codes. If NULL, returns results of sections. Defaults to NULL.
lang	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
from	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01".
to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
result	the voting result. "passed" returns motions that have been passed. "vetoed" returns motions that have been vetoed. "all" returns all motions that has been voted in LegCo. Defaults to all.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
n	the number of record to fetch. Defaults to 1000.
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

legco_api

Generic LegCo API Function

Description

A generic function to access LegCo APIs.

Usage

```
legco_api(db, query, n = 1000, count = FALSE, verbose = TRUE)
```

Arguments

- | | |
|-------|---|
| db | the database you wish to access. "hansard" for the hansard database. "attn" for the attendance database. "bill" for the bills database. "schedule" for the schedule database. Or the path name for databases not listed here (i.e. the string between the domain name and data endpoints, e.g. "OpenData/HansardDB"). |
| query | the query for retrieving data. Should include the data endpoint and parameters if any. |
| n | the number of record to fetch. Defaults to 1000. |

count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Examples

```
# Fetch data from the "bills" endpoint of the hansard database
x <- legco_api("hansard", "Bills")
```

legco_section_type	<i>Section Types of LegCo Meetings</i>
--------------------	--

Description

A data frame containing possible types of sections in LegCo meetings. For use to look up the meaning of a section code returned by the API.

Usage

```
legco_section_type
```

Format

A dataframe with 51 rows and 3 variables:

- `SectionCode`: Unique code(s) for each section type
- `NameEng`: English name of the section
- `NameChi`: Chinese name of the section

Details

This data frame is a formatted offline copy of the *Sections* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members

- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Look up the meaning of section code "mbm"
x <- legco_section_type[legco_section_type$SectionCode == "mbm", ]
```

meeting

Meetings of LegCo Committees

Description

Fetch basic information of LegCo committee meetings.

Usage

```
meeting(
  slot_id = NULL,
  meet_id = NULL,
  from = "1900-01-01",
  to = Sys.Date(),
  type = "all",
  term_id = NULL,
  n = 10000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

legco_meeting(
  slot_id = NULL,
  meet_id = NULL,
  from = "1900-01-01",
```

```

to = Sys.Date(),
type = "all",
term_id = NULL,
n = 10000,
extra_param = NULL,
count = FALSE,
verbose = TRUE
)

```

Arguments

slot_id	the id of a meeting slot, or a vector of ids. If NULL, returns all meetings. Defaults to NULL.
meet_id	the id of a meeting, or a vector of ids. If NULL, returns all meetings. Useful for matching meeting with records from the Attendance Database. Defaults to NULL.
from	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01".
to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
type	the type of meeting. If "open", returns open meetings. If "closed", returns closed meetings. If "all", returns all meetings. Defaults to "all".
term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tmeeting* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees

- **membership**: Membership of LegCo committees
- **member**: LegCo members
- **member_term**: Terms served by LegCo members
- **meeting**: Meetings of LegCo committees
- **meeting_committee**: Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch all meeting conducted on March 20, 2019
x <- meeting(from = "2019-03-20", to = "2019-03-20")
```

meeting_committee	<i>Committee of Meetings in LegCo</i>
-------------------	---------------------------------------

Description

Fetch all the meeting slots of a given committee(s), or the committee of a given meeting slot(s).

Usage

```
meeting_committee(  
  slot_id = NULL,  
  meet_id = NULL,  
  committee_id = NULL,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)  
  
legco_meeting_committee(  
  slot_id = NULL,  
  meet_id = NULL,  
  committee_id = NULL,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

Arguments

slot_id	the id of a meeting slot, or a vector of ids. If NULL, returns all meetings. Defaults to NULL.
meet_id	the id of a meeting, or a vector of ids. If NULL, returns all meetings. Useful for matching meeting with records from the Attendance Database. Defaults to NULL.
committee_id	the id of a committee, or a vector of ids. If NULL, returns results of all committees. Defaults to NULL.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tmeeting_committee* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees
- [membership](#): Membership of LegCo committees
- [member](#): LegCo members
- [member_term](#): Terms served by LegCo members
- [meeting](#): Meetings of LegCo committees
- [meeting_committee](#): Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch all meetings of Subcommittee on Issues Relating to Bazaars
x <- meeting_committee(committee_id = 2704)
```

meeting_schedule-db *Meeting Schedule Database*

Description

This database contains the information of LegCo committees, namely the name of all LegCo committees established since the fifth term and their corresponding member lists and meeting schedule.

Arguments

member_id	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
session_id	the id of a session, or a vector of ids. If NULL, returns result of all sessions. Defaults to NULL.
slot_id	the id of a meeting slot, or a vector of ids. If NULL, returns all meetings. Defaults to NULL.
meet_id	the id of a meeting, or a vector of ids. If NULL, returns all meetings. Useful for matching meeting with records from the Attendance Database. Defaults to NULL.
committee_id	the id of a committee, or a vector of ids. If NULL, returns results of all committees. Defaults to NULL.
code	the code of a committee, or a vector of ids. If NULL, returns all committees. Defaults to NULL.
type	the type of meeting. If "open", returns open meetings. If "closed", returns closed meetings. If "all", returns all meetings. Defaults to "all".
post	the post of a member in the committee, or a vector of posts. "President", "Chairman", "Deputy Chairman", "Member" and "Convenor" return members of the respective post. If NULL, returns members of all posts. Defaults to NULL.
date	only fetch the result in which the specified date falls within. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to NULL.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
n	the number of record to fetch. Defaults to 1000.
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Functions

Functions of the Meeting Schedule Database:

- `term`: LegCo terms
- `session`: LegCo sessions
- `committee`: LegCo committees
- `membership`: Membership of LegCo committees
- `member`: LegCo members
- `member_term`: Terms served by LegCo members
- `meeting`: Meetings of LegCo committees
- `meeting_committee`: Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

member

Members of LegCo

Description

Fetch the basic information of LegCo members.

Usage

```
member(member_id = NULL, extra_param = NULL, count = FALSE, verbose = TRUE)
```

```
legco_member(
  member_id = NULL,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

<code>member_id</code>	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
<code>extra_param</code>	additional query parameters defined in LegCo API. Must begin with "&".
<code>count</code>	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
<code>verbose</code>	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tmember* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees
- [membership](#): Membership of LegCo committees
- [member](#): LegCo members
- [member_term](#): Terms served by LegCo members
- [meeting](#): Meetings of LegCo committees
- [meeting_committee](#): Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch full list of members
x <- member()
```

membership

Membership of LegCo Committees

Description

Fetch the members of LegCo committees.

Usage

```
membership(  
  member_id = NULL,  
  committee_id = NULL,  
  term_id = NULL,  
  post = NULL,  
  n = 10000,  
  extra_param = NULL,  
  count = FALSE,
```

```

    verbose = TRUE
  )

legco_membership(
  member_id = NULL,
  committee_id = NULL,
  term_id = NULL,
  post = NULL,
  n = 10000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

```

Arguments

<code>member_id</code>	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
<code>committee_id</code>	the id of a committee, or a vector of ids. If NULL, returns results of all committees. Defaults to NULL.
<code>term_id</code>	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
<code>post</code>	the post of a member in the committee, or a vector of posts. "President", "Chairman", "Deputy Chairman", "Member" and "Convenor" return members of the respective post. If NULL, returns members of all posts. Defaults to NULL.
<code>n</code>	the number of record to fetch. Defaults to 1000.
<code>extra_param</code>	additional query parameters defined in LegCo API. Must begin with "&".
<code>count</code>	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
<code>verbose</code>	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tmembership* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees
- [membership](#): Membership of LegCo committees
- [member](#): LegCo members
- [member_term](#): Terms served by LegCo members

- [meeting](#): Meetings of LegCo committees
- [meeting_committee](#): Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch members of the Subcommittee on Issues Relating to Bazaars
x <- membership(committee_id = 2704)
```

member_term	<i>Terms of Members of LegCo</i>
-------------	----------------------------------

Description

Fetch the serving terms in Term ID of LegCo members

Usage

```
member_term(
  member_id = NULL,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

legco_member_term(
  member_id = NULL,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

member_id	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tmember_term* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees
- [membership](#): Membership of LegCo committees
- [member](#): LegCo members
- [member_term](#): Terms served by LegCo members
- [meeting](#): Meetings of LegCo committees
- [meeting_committee](#): Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
#Fetches the term served by Hon Chan Kin-por and Kwong Chun-yu
x <- member_term(member_id = c(273, 924))
```

motions

Motions in LegCo

Description

Fetch motions presented in LegCo council meetings.

Usage

```
motions(  
  rundown_id = NULL,  
  hansard_id = NULL,  
  section_code = NULL,  
  lang = "en",  
  from = "1900-01-01",  
  to = Sys.Date(),  
  floor = FALSE,
```

```

    n = 1000,
    extra_param = NULL,
    count = FALSE,
    verbose = TRUE
)

legco_motions(
  rundown_id = NULL,
  hansard_id = NULL,
  section_code = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

```

Arguments

rundown_id	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
hansard_id	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
section_code	the section code, or a vector of codes. If NULL, returns results of sections. Defaults to NULL.
lang	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
from	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01".
to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.

verbose logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Motions* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch motions discussed during the Council meeting on February 20, 2019
x <- motions(hansard_id = 2714)
```

 petitions

Petitions in LegCo

Description

Fetch petitions presented in LegCo council meetings.

Usage

```

petitions(
  rundown_id = NULL,
  hansard_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

```

```

legco_petitions(
  rundown_id = NULL,
  hansard_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

```

Arguments

<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".

to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Petitions* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch petition laid during the Council meeting on January 19, 2018
x <- petitions(from = "2018-01-10")
```

questions

Questions by LegCo Members

Description

Fetch questions put to the government by LegCo members in council meetings.

Usage

```
questions(
  rundown_id = NULL,
  speaker_id = NULL,
  type = "all",
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_questions(
  rundown_id = NULL,
  speaker_id = NULL,
  type = "all",
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

rundown_id	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
speaker_id	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
type	the type of question. "oral" returns oral questions. "written" returns written questions. "all" returns all questions. Defaults to "all".
lang	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
from	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01".
to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Questions* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions

- `petitions`: Petitions
- `addresses`: Addresses made by members or government officials when presenting papers to the Council
- `statements`: Statements made by government officials
- `voting_results`: Results of votes in council meetings
- `summoning_bells`: Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch questions being put on the government during the Council meeting on February 20, 2019
x <- questions(from = "2019-02-20", to = "2019-02-20")
```

rundown

Rundown from LegCo Hansards

Description

Fetch the rundown from hansard files of LegCo council meetings.

Usage

```
rundown(
  rundown_id = NULL,
  hansard_id = NULL,
  speaker_id = NULL,
  lang = "en",
  bookmark_name = NULL,
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

legco_rundown(
  rundown_id = NULL,
  hansard_id = NULL,
```

```

    speaker_id = NULL,
    lang = "en",
    bookmark_name = NULL,
    from = "1900-01-01",
    to = Sys.Date(),
    floor = FALSE,
    n = 1000,
    extra_param = NULL,
    count = FALSE,
    verbose = TRUE
)

```

Arguments

<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>speaker_id</code>	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>bookmark_name</code>	the bookmark name of the rundown, or a vector of names. Defaults to NULL.
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.
<code>floor</code>	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
<code>n</code>	the number of record to fetch. Defaults to 1000.
<code>extra_param</code>	additional query parameters defined in LegCo API. Must begin with "&".
<code>count</code>	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
<code>verbose</code>	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Rundown* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- `hansard`: Hansard files
- `legco_section_type`: Section code
- `subjects`: Subjects `speakers`: Speakers in the council, including members, government officials and secretariat staff
- `rundown`: Rundown (Paragraphs in hansard)
- `questions`: Questions raised by members
- `bills`: Bills
- `motions`: Motions
- `petitions`: Petitions
- `addresses`: Addresses made by members or government officials when presenting papers to the Council
- `statements`: Statements made by government officials
- `voting_results`: Results of votes in council meetings
- `summoning_bells`: Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch records of proceedings of the second reading of
# Smoking (Public Health) (Amendment) Bill 2019
x <- rundown(rundown_id = 899628:899649)
```

search_committee

Search Committee

Description

Search committee by full or partial name of committee.

Usage

```
search_committee(search_string, term_id = NULL, exact = TRUE, verbose = TRUE)

legco_search_committee(
  search_string,
  term_id = NULL,
  exact = TRUE,
  verbose = TRUE
)
```

Arguments

search_string	search string of committee name. Accepts Chinese or English full or partial name. Defaults to NULL.
term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
exact	logical: Whether to look for exact match of the search term. Defaults to TRUE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This is a complementary function to [voting_record\(\)](#).

Functions

Complementary Functions:

- [search_committee](#): Search LegCo committees
- [search_member](#): Search LegCo members
- [search_voting_record](#): Search Voting Record in LegCo meetings
- [search_question](#): Search full text of question put to the government by LegCo members

Notes

The complementary functions work by calling a number of database functions and combining their output, meaning that a function call usually involves multiple API calls. Use with caution to prevent reaching the API's rate limit too quickly.

search_member	<i>Search LegCo Member</i>
---------------	----------------------------

Description

Search LegCo member by SpeakerID, MemberID or/and full or partial English or Chinese name.

Usage

```
search_member(
  search_string = NULL,
  speaker_id = NULL,
  member_id = NULL,
  exact = TRUE,
  verbose = TRUE
)
```

```
legco_search_member(
  search_string = NULL,
  speaker_id = NULL,
  member_id = NULL,
  exact = TRUE,
  verbose = TRUE
)
```

Arguments

search_string	search string of member's name. Accepts Chinese or English full or partial name. Defaults to NULL.
speaker_id	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
member_id	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
exact	logical: Whether to look for exact match of the search term. Defaults to TRUE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This is a complementary function to [speakers\(\)](#) and [member\(\)](#).

Functions

Complementary Functions:

- [search_committee](#): Search LegCo committees

- [search_member](#): Search LegCo members
- [search_voting_record](#): Search Voting Record in LegCo meetings
- [search_question](#): Search full text of question put to the government by LegCo members

Notes

The complementary functions work by calling a number of database functions and combining their output, meaning that a function call usually involves multiple API calls. Use with caution to prevent reaching the API's rate limit too quickly.

search_question	<i>Search Question in LegCo</i>
-----------------	---------------------------------

Description

Search full text of questions put to the government by LegCo member(s).

Usage

```
search_question(
  speaker_id = NULL,
  member_id = NULL,
  rundown_id = NULL,
  type = "all",
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 50,
  verbose = TRUE
)
```

```
legco_search_question(
  speaker_id = NULL,
  member_id = NULL,
  rundown_id = NULL,
  type = "all",
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 50,
  verbose = TRUE
)
```


Arguments

speaker_id	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
member_id	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
rundown_id	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
type	the type of meeting. If "open", returns open meetings. If "closed", returns closed meetings. If "all", returns all meetings. Defaults to "all".
lang	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
from	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01".
to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
n	the number of record to fetch. Defaults to 50.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This is a complementary function to [questions\(\)](#).

Functions

Complementary Functions:

- [search_committee](#): Search LegCo committees
- [search_member](#): Search LegCo members
- [search_voting_record](#): Search Voting Record in LegCo meetings
- [search_question](#): Search full text of question put to the government by LegCo members

Notes

The complementary functions work by calling a number of database functions and combining their output, meaning that a function call usually involves multiple API calls. Use with caution to prevent reaching the API's rate limit too quickly.

search_voting_record *Search Voting Record*

Description

Search voting record by conducted committee or date or meeting slot or LegCo member(s). Note that only voting records from the Council, House Committee, Finance Committee and its subcommittees are available.

Usage

```
search_voting_record(
  speaker_id = NULL,
  member_id = NULL,
  committee_id = NULL,
  slot_id = NULL,
  from = "1900-01-01 00:00:00",
  to = Sys.time(),
  verbose = TRUE,
  ...
)
```

```
legco_search_voting_record(
  speaker_id = NULL,
  member_id = NULL,
  committee_id = NULL,
  slot_id = NULL,
  from = "1900-01-01 00:00:00",
  to = Sys.time(),
  verbose = TRUE,
  ...
)
```

Arguments

speaker_id	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
member_id	the id of a LegCo member, or a vector of ids. If NULL, returns results of all LegCo members. Defaults to NULL.
committee_id	the id of a committee, or a vector of ids. If NULL, returns results of all committees. Defaults to NULL.
slot_id	the id of a meeting slot, or a vector of ids. If NULL, returns all meetings. Defaults to NULL.
from	only fetch results of meetings on or after this date and time. Accepts character values in "YYYY-MM-DD" or "YYYY-MM-DD HH:MM:SS" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01 00:00:00".

to	only fetch results of meetings on or before this date and time. Accepts character values in "YYYY-MM-DD" or "YYYY-MM-DD HH:MM:SS" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system time.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.
...	optional arguments passed to voting_record() .

Details

This is a complementary function to [voting_record\(\)](#).

Functions

Complementary Functions:

- [search_committee](#): Search LegCo committees
- [search_member](#): Search LegCo members
- [search_voting_record](#): Search Voting Record in LegCo meetings
- [search_question](#): Search full text of question put to the government by LegCo members

Notes

The complementary functions work by calling a number of database functions and combining their output, meaning that a function call usually involves multiple API calls. Use with caution to prevent reaching the API's rate limit too quickly.

session	<i>Session of LegCo</i>
---------	-------------------------

Description

Fetch the basic information of LegCo sessions.

Usage

```
session(
  session_id = NULL,
  term_id = NULL,
  date = NULL,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)

legco_session(
```

```

    session_id = NULL,
    term_id = NULL,
    date = NULL,
    extra_param = NULL,
    count = FALSE,
    verbose = TRUE
)

```

Arguments

session_id	the id of a session, or a vector of ids. If NULL, returns result of all sessions. Defaults to NULL.
term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
date	only fetch the result in which the specified date falls within. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to NULL.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tsession* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees
- [membership](#): Membership of LegCo committees
- [member](#): LegCo members
- [member_term](#): Terms served by LegCo members
- [meeting](#): Meetings of LegCo committees
- [meeting_committee](#): Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch all LegCo sessions of the fifth term
x <- session(term_id = 4)
```

speakers

*Speakers at LegCo***Description**

Fetch the basic information of speakers in LegCo council meetings, including LegCo members, government officials and Secretariat staff.

Usage

```
speakers(
  speaker_id = NULL,
  type = "all",
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_speakers(
  speaker_id = NULL,
  type = "all",
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

speaker_id	the id of a speaker at the Legislative Council, or a vector of ids. If NULL, returns all speakers. Defaults to NULL.
type	the position of a speaker. "all" returns all speakers. "PO" returns public officers. "LC" returns key appointment holders and staff at LegCo, such as President, Chairman and clerk. "MB" returns LegCo members. Default to "all".
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Speakers* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- `hansard`: Hansard files
- `legco_section_type`: Section code
- `subjects`: Subjects `speakers`: Speakers in the council, including members, government officials and secretariat staff
- `rundown`: Rundown (Paragraphs in hansard)
- `questions`: Questions raised by members
- `bills`: Bills
- `motions`: Motions
- `petitions`: Petitions
- `addresses`: Addresses made by members or government officials when presenting papers to the Council
- `statements`: Statements made by government officials
- `voting_results`: Results of votes in council meetings
- `summoning_bells`: Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch a list of all speakers in LegCo council meetings
x <- speakers()
# Fetch a list of all speakers who are LegCo members
x <- speakers(type = "MB")
# Look up the details of a speaker with the id 6
x <- speakers(speaker_id = 6)
```

statements

Statements by Public Officers

Description

Fetch statements made by Public Officers in LegCo council meetings.

Usage

```
statements(
  hansard_id = NULL,
  rundown_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_statements(
  hansard_id = NULL,
  rundown_id = NULL,
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

Arguments

<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".

to	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system date.
floor	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Statements* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch statements made by government officials during the Council meeting on April 22, 2015
x <- statements(from = "2015-04-22")
```

subjects	<i>Subjects from LegCo Hansard</i>
----------	------------------------------------

Description

Fetch the subjects from hansard files of LegCo council meetings.

Usage

```
subjects(  
  hansard_id = NULL,  
  rundown_id = NULL,  
  section_code = NULL,  
  lang = "en",  
  from = "1900-01-01",  
  to = Sys.Date(),  
  floor = FALSE,  
  n = 1000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

```
legco_subjects(  
  hansard_id = NULL,  
  rundown_id = NULL,  
  section_code = NULL,  
  lang = "en",  
  from = "1900-01-01",  
  to = Sys.Date(),  
  floor = FALSE,  
  n = 1000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

Arguments

<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>section_code</code>	the section code, or a vector of codes. If NULL, returns results of sections. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.
<code>floor</code>	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
<code>n</code>	the number of record to fetch. Defaults to 1000.
<code>extra_param</code>	additional query parameters defined in LegCo API. Must begin with "&".
<code>count</code>	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
<code>verbose</code>	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Subjects* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- `hansard`: Hansard files
- `legco_section_type`: Section code
- `subjects`: Subjects `speakers`: Speakers in the council, including members, government officials and secretariat staff
- `rundown`: Rundown (Paragraphs in hansard)
- `questions`: Questions raised by members
- `bills`: Bills
- `motions`: Motions

- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch second reading of bills during the Council meeting on February 20, 2019
x <- subjects(hansard_id = 2714, section_code = "b2r")
```

summoning_bells	<i>Summoning Bells in LegCo</i>
-----------------	---------------------------------

Description

Fetch instances of summoning bell being rung in LegCo council meetings

Usage

```
summoning_bells(  
  hansard_id = NULL,  
  rundown_id = NULL,  
  section_code = NULL,  
  lang = "en",  
  from = "1900-01-01",  
  to = Sys.Date(),  
  floor = FALSE,  
  n = 1000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

```
legco_summoning_bells(  
  hansard_id = NULL,  
  rundown_id = NULL,  
  section_code = NULL,
```

```

lang = "en",
from = "1900-01-01",
to = Sys.Date(),
floor = FALSE,
n = 1000,
extra_param = NULL,
count = FALSE,
verbose = TRUE
)

```

Arguments

<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>section_code</code>	the section code, or a vector of codes. If NULL, returns results of sections. Defaults to NULL.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.
<code>floor</code>	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
<code>n</code>	the number of record to fetch. Defaults to 1000.
<code>extra_param</code>	additional query parameters defined in LegCo API. Must begin with "&".
<code>count</code>	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
<code>verbose</code>	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *SummoningBells* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- **hansard**: Hansard files
- **legco_section_type**: Section code
- **subjects**: Subjects **speakers**: Speakers in the council, including members, government officials and secretariat staff
- **rundown**: Rundown (Paragraphs in hansard)
- **questions**: Questions raised by members
- **bills**: Bills
- **motions**: Motions
- **petitions**: Petitions
- **addresses**: Addresses made by members or government officials when presenting papers to the Council
- **statements**: Statements made by government officials
- **voting_results**: Results of votes in council meetings
- **summoning_bells**: Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch instances of summoning bell during the Council meeting on January 19, 2018
x <- summoning_bells(hansard_id = 2714)
```

term

Term of LegCo

Description

Fetch the basic information of LegCo terms.

Usage

```
term(term_id = NULL, date = NULL, extra_param = NULL, verbose = TRUE)
```

```
legco_term(term_id = NULL, date = NULL, extra_param = NULL, verbose = TRUE)
```

Arguments

term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
date	only fetch the result in which the specified date falls within. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to NULL.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *Tterm* data endpoint of the Meeting Schedule Database.

Functions

Functions of the Meeting Schedule Database:

- [term](#): LegCo terms
- [session](#): LegCo sessions
- [committee](#): LegCo committees
- [membership](#): Membership of LegCo committees
- [member](#): LegCo members
- [member_term](#): Terms served by LegCo members
- [meeting](#): Meetings of LegCo committees
- [meeting_committee](#): Committees of LegCo meetings

See Also

LegCo API documentation for the Meeting Schedule database: <https://www.legco.gov.hk/odata/english/attendance-db.html>

Examples

```
# Fetch all LegCo terms
x <- term()
```

voting_record	<i>Voting Record in LegCo Meetings</i>
---------------	--

Description

Fetch voting records of LegCo council, the Finance Committee and its subcommittees and the House Committee meetings.

Usage

```
voting_record(  
  committee = NULL,  
  term_id = NULL,  
  result = "all",  
  vote = "all",  
  name_ch = NULL,  
  name_en = NULL,  
  separate_mechanism = NULL,  
  mover_type = "all",  
  from = "1900-01-01 00:00:00",  
  to = Sys.time(),  
  n = 10000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

```
legco_voting_record(  
  committee = NULL,  
  term_id = NULL,  
  result = "all",  
  vote = "all",  
  name_ch = NULL,  
  name_en = NULL,  
  separate_mechanism = NULL,  
  mover_type = "all",  
  from = "1900-01-01 00:00:00",  
  to = Sys.time(),  
  n = 10000,  
  extra_param = NULL,  
  count = FALSE,  
  verbose = TRUE  
)
```

Arguments

`committee` the name of the committee or subcommittee. Defaults to NULL.

term_id	the id of a term, or a vector of ids. If NULL, returns results of all terms. Defaults to NULL.
result	the voting result. If "passed", returns motions that have been passed. If "vetoed", returns motions that have been vetoed. If "all", returns all motions that has been voted in LegCo. Defaults to all.
vote	the vote cast. If "yes", returns only members who cast affirmative votes. If "no", returns only members who cast negative votes. If "abstain", returns only members who abstained from voting. If "absent", returns only members who were absent. If "present", returns only members who were present and did not vote (e.g. President). If "all", returns all votes. Defaults to "all".
name_ch	the name of a LegCo member in Traditional Chinese, or a vector of names. If NULL, returns voting records of all members. Defaults to NULL.
name_en	the name of a LegCo member in English, or a vector of names. If the member has an English name, the English name should go first followed by the surname in capital letters, e.g. "Peter CHAN". If the member does not have an English name, the surname in capital letters should go first followed by the translated first name with a hyphen separating the different syllable, e.g. "CHAN Tai-man". Check the names of the members with member() . If NULL, returns voting records of all members. Defaults to NULL.
separate_mechanism	only fetch votes that were counted with the vote separate mechanism, i.e. requiring majority in both geographical and functional constituencies to pass. If NULL, returns all votes regardless of the vote counting mechanism used. Defaults to NULL.
mover_type	the type of motion being put on vote. If "PO", returns votes on government motions only. If "MB", returns votes on members' motions only. If "all", returns votes on all motions. Defaults to "all".
from	only fetch results of meetings on or after this date and time. Accepts character values in "YYYY-MM-DD" or "YYYY-MM-DD HH:MM:SS" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to "1900-01-01 00:00:00".
to	only fetch results of meetings on or before this date and time. Accepts character values in "YYYY-MM-DD" or "YYYY-MM-DD HH:MM:SS" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with as.Date(). Defaults to the current system time.
n	the number of record to fetch. Defaults to 1000.
extra_param	additional query parameters defined in LegCo API. Must begin with "&".
count	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
verbose	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *vVotingResult* data endpoint of the Voting Result Database.

See Also

LegCo API documentation for the Voting Record database: <https://www.legco.gov.hk/odata/english/vrdb.html>

Examples

```
# Fetch how members voted the motion on
# Abolishing the MPF Offsetting Mechanism on November 11, 2016
x <- voting_record(committee = "Council Meeting",
                  from = "2016-11-16 13:51:53",
                  to = "2016-11-16 13:51:53")
```

voting_results

Voting Results in LegCo Meetings

Description

Fetch result of votes made in LegCo council meetings.

Usage

```
voting_results(
  hansard_id = NULL,
  rundown_id = NULL,
  section_code = NULL,
  result = "all",
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
  floor = FALSE,
  n = 1000,
  extra_param = NULL,
  count = FALSE,
  verbose = TRUE
)
```

```
legco_voting_results(
  hansard_id = NULL,
  rundown_id = NULL,
  section_code = NULL,
  result = "all",
  lang = "en",
  from = "1900-01-01",
  to = Sys.Date(),
```

```

    floor = FALSE,
    n = 1000,
    extra_param = NULL,
    count = FALSE,
    verbose = TRUE
)

```

Arguments

<code>hansard_id</code>	the id of a hansard file, or a vector of ids. If NULL, returns results of all hansard files. Defaults to NULL.
<code>rundown_id</code>	the id of a rundown, or a vector of ids. If NULL, returns results of all rundowns. Defaults to NULL.
<code>section_code</code>	the section code, or a vector of codes. If NULL, returns results of sections. Defaults to NULL.
<code>result</code>	the voting result. "passed" returns motions that have been passed. "vetoed" returns motions that have been vetoed. "all" returns all motions that has been voted in LegCo. Defaults to all.
<code>lang</code>	the language of hansard files to search from. "en" returns the English version. "zh" returns the Traditional Chinese version. Defaults to "en".
<code>from</code>	only fetch results of meetings on or after this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to "1900-01-01".
<code>to</code>	only fetch results of meetings on or before this date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else that can be coerced to a date with <code>as.Date()</code> . Defaults to the current system date.
<code>floor</code>	logical: whether to fetch results from the floor version of the hansard files? The floor version is the first presented version of hansard file in the original language delivered by the speakers in LegCo. If TRUE, the language option is ignored. Defaults to FALSE.
<code>n</code>	the number of record to fetch. Defaults to 1000.
<code>extra_param</code>	additional query parameters defined in LegCo API. Must begin with "&".
<code>count</code>	logical: Whether to return only the total count of records that matches the parameter(s) instead of the result. Defaults to FALSE.
<code>verbose</code>	logical: Whether to display progress messages when fetching data? Defaults to TRUE.

Details

This function corresponds to the *VotingResults* data endpoint of the Hansard Database.

Functions

Functions of the Hansard database:

- [hansard](#): Hansard files
- [legco_section_type](#): Section code
- [subjects](#): Subjects [speakers](#): Speakers in the council, including members, government officials and secretariat staff
- [rundown](#): Rundown (Paragraphs in hansard)
- [questions](#): Questions raised by members
- [bills](#): Bills
- [motions](#): Motions
- [petitions](#): Petitions
- [addresses](#): Addresses made by members or government officials when presenting papers to the Council
- [statements](#): Statements made by government officials
- [voting_results](#): Results of votes in council meetings
- [summoning_bells](#): Instances of summoning bells being rung

See Also

LegCo API documentation for the Hansard database: <https://www.legco.gov.hk/odata/english/hansard-db.html>

Examples

```
# Fetch results of votes conducted during the Council meeting on January 19, 2018
x <- voting_results(hansard_id = 2714)
```

Index

* datasets

- legco_section_type, 18
- addresses, 2, 4, 11, 15, 17, 19, 30, 32, 35, 37, 46, 48, 51, 53, 59
- all_bills, 5
- attendance, 8
- bills, 4, 9, 11, 15, 17, 19, 30, 32, 34, 37, 46, 48, 50, 53, 59
- committee, 12, 13, 20, 22, 24–26, 28, 44, 54
- comp-fun, 13
- hansard, 4, 11, 14, 15, 17, 18, 30, 32, 34, 37, 46, 48, 50, 53, 59
- hansard-db, 16
- legco_addresses (addresses), 2
- legco_all_bills (all_bills), 5
- legco_api, 17
- legco_attendance (attendance), 8
- legco_bills (bills), 9
- legco_committee (committee), 12
- legco_hansard (hansard), 14
- legco_meeting (meeting), 19
- legco_meeting_committee (meeting_committee), 21
- legco_member (member), 24
- legco_member_term (member_term), 27
- legco_membership (membership), 25
- legco_motions (motions), 28
- legco_petitions (petitions), 31
- legco_questions (questions), 33
- legco_rundown (rundown), 35
- legco_search_committee (search_committee), 37
- legco_search_member (search_member), 39
- legco_search_question (search_question), 40
- legco_search_voting_record (search_voting_record), 42
- legco_section_type, 4, 11, 15, 17, 18, 18, 30, 32, 34, 37, 46, 48, 50, 53, 59
- legco_session (session), 43
- legco_speakers (speakers), 45
- legco_statements (statements), 47
- legco_subjects (subjects), 49
- legco_summoning_bells (summoning_bells), 51
- legco_term (term), 53
- legco_voting_record (voting_record), 55
- legco_voting_results (voting_results), 57
- meeting, 13, 19, 21, 22, 24, 25, 27, 28, 44, 54
- meeting_committee, 13, 21, 21, 22, 24, 25, 27, 28, 44, 54
- meeting_schedule-db, 23
- member, 13, 21, 22, 24, 24, 25, 26, 28, 39, 44, 54, 56
- member_term, 13, 21, 22, 24–26, 27, 28, 44, 54
- membership, 13, 21, 22, 24, 25, 25, 26, 28, 44, 54
- motions, 4, 11, 15, 17, 19, 28, 30, 32, 34, 37, 46, 48, 50, 53, 59
- petitions, 4, 11, 15, 17, 19, 30, 31, 32, 35, 37, 46, 48, 51, 53, 59
- questions, 4, 11, 15, 17, 18, 30, 32, 33, 34, 37, 41, 46, 48, 50, 53, 59
- rundown, 4, 11, 15, 17, 18, 30, 32, 34, 35, 37, 46, 48, 50, 53, 59
- search_committee, 13, 37, 38, 39, 41, 43
- search_member, 13, 38, 39, 40, 41, 43
- search_question, 13, 38, 40, 40, 41, 43
- search_voting_record, 13, 38, 40, 41, 42, 43
- session, 13, 20, 22, 24–26, 28, 43, 44, 54

speakers, *4, 11, 15, 17, 18, 30, 32, 34, 37, 39, 45, 46, 48, 50, 53, 59*

statements, *4, 11, 15, 17, 19, 30, 32, 35, 37, 46, 47, 48, 51, 53, 59*

subjects, *4, 11, 15, 17, 18, 30, 32, 34, 37, 46, 48, 49, 50, 53, 59*

summoning_bells, *4, 11, 15, 17, 19, 30, 32, 35, 37, 46, 48, 51, 51, 53, 59*

term, *13, 20, 22, 24–26, 28, 44, 53, 54*

voting_record, *38, 43, 55*

voting_results, *4, 11, 15, 17, 19, 30, 32, 35, 37, 46, 48, 51, 53, 57, 59*