Package ‘stortingscrape’

January 18, 2024

Type Package

Title Access Data from the Norwegian Parliament API

Version 0.3.0

Description Functions for retrieving general and specific data from the Norwegian Parliament, through the Norwegian Parliament API at <https://data.stortinget.no>.

URL https://github.com/martigso/stortingscrape,
https://martigso.github.io/stortingscrape/

BugReports https://github.com/martigso/stortingscrape/issues

Depends R (>= 4.2.0)

Encoding UTF-8

Imports rvest, httr2, parallel, stringr

Suggests imager, rmarkdown, knitr, pscl

License GPL (>= 3)

VignetteBuilder knitr

RoxygenNote 7.2.3

LazyData true

NeedsCompilation no

Author Martin Søyland [aut, cre]

Maintainer Martin Søyland <martin.soyland@stv.uio.no>

Repository CRAN

Date/Publication 2024-01-18 12:30:02 UTC

R topics documented:
cases ................................................................. 3
covid_relief ....................................................... 3
covid_relief_result .............................................. 4
get_all_committees .............................................. 5
get_all_parties .................................................... 6
get_case ................................................................. 6
get_counties .......................................................... 9
get_decision_votes .................................................... 10
get_hearing_input ..................................................... 11
get_hearing_program ................................................ 12
get_meeting_agenda .................................................. 13
get_mp ................................................................. 14
get_mp_bio ............................................................. 15
get_mp_pic ............................................................. 18
get_parlperiods ....................................................... 19
get_parlperiod_mps ................................................... 20
get_parlperiod_presidency .......................................... 21
get_parlsessions ..................................................... 22
get_proceedings ...................................................... 23
get_proposal_votes ................................................... 24
get_publication ...................................................... 25
get_question ........................................................ 26
get_question_hour .................................................... 28
get_result_vote ....................................................... 30
get_session_cases .................................................... 31
get_session_committees ............................................. 33
get_session_decisions .............................................. 34
get_session_delegations ............................................ 35
get_session_hearings ............................................... 36
get_session_meetings ............................................... 37
get_session_mp_speech_activity .................................. 38
get_session_parties .................................................. 40
get_session_publications .......................................... 41
get_session_questions .............................................. 42
get_topics ........................................................... 43
get_vote ............................................................. 44
get_written_hearing_input ........................................ 46
interp0203 ........................................................... 47
mps4549 ............................................................. 48
parl_periods .......................................................... 49
parl_sessions ........................................................ 49
read_obt ............................................................. 50
vote ................................................................. 50
vote_result .......................................................... 52

Index 53
cases  Storting cases in the 2019-2020 session

Description
A dataset containing all cases of the 2019-2020 parliamentary session in Stortinget

Usage
cases

Format
A list with four elements

$root  main data on the MP
$topics  named list by case id
$proposers  named list by case id
$spokespersons  named list by case id

Further description: get_session_cases

Source

covid_relief  Vote id 85196

Description
A dataset containing all vote information on case id 85196

Usage
covid_relief

Format
A data frame with 22 columns and 71 rows

response_date  Date of data retrieval
version  Data version from the API
case_id  Case id up for vote
alternative_vote  Whether vote is an alternative vote
n_for Number of votes for
n_absent Number of MPs absent
n_against Number of votes against
treatment_order Order of treated votes
agenda_case_number Case number on the agenda of the meeting
free_vote Logical indication of whether the vote is related to the case as a whole
comment Vote comment
meeting_map_number Number on the meeting map
personal_vote Logical indication of whether vote was recorded as roll call or not
president_id Id of president holding president chair at the time of voting
president_party_id Party of the sitting president
adopted Logical indication of whether the proposal voted on was adopted
vote_id Id of vote
vote_method Voting method
vote_result_type Result type (enstemmig_vedtatt = unanimously adopted)
vote_result_type_text See vote_result_type
vote_topic Description of the proposal voted upon
vote_datetime Date and time of vote

Source
https://data.stortinget.no/eksport/voteringer?sakid=85196

---

covid_relief_result Vote id 85196 results

Description
A dataset containing vote matrix on vote id 17689

Usage
covid_relief_result

Format
A data frame with 8 columns and 169 rows

response_date Date of data retrieval
version Data version from the API
vote_id Id of vote
mp_id MP id
get_all_committees

- **party_id**  Party id
- **vote**  Vote: for, mot (against), ikke_tilstede (absent)
- **permanent_sub_for**  Id of the MP originally holding the seat, if the substitute is
- **sub_for**  Id of the MP originally holding the seat

**Source**

https://data.stortinget.no/eksport/voteringsresultat?voteringid=17689

---

get_all_committees  Parliamentary committees over all sessions

**Description**

Imports data on all committee names and ids over all sessions in the data.stortinget.no API.

**Usage**

get_all_committees()

**Value**

A data frame of committees, with the variables:

- **response_date**  Date of data retrieval
- **version**  Data version from the API
- **id**  Id of the committee
- **name**  Name of the committee
- **sessionid**  Id of session (empty)

**See Also**

get_session_committees

**Examples**

```r
## Not run:
coms <- get_all_committees()
head(coms)

## End(Not run)
```
get_all_parties  
*All parliamentary parties*

**Description**

A function for retrieving all Norwegian parliamentary parties in the data.stortinget.no API.

**Usage**

`get_all_parties()`

**Value**

A data.frame of all parties, with the variables:

- `response_date`: Date of data retrieval
- `version`: Data version from the API
- `id`: Id of the party
- `name`: Name of the party
- `represented_party`: Whether the party is represented at the time of download
- `sessionid`: Id of session (empty)
- `period_id`: Id of parliamentary period (empty)

**See Also**

`get_session_parties`

**Examples**

```r
## Not run:
parties <- get_all_parties()
head(parties)

## End(Not run)
```

get_case  
*Retreive a parliamentary case*

**Description**

A function for retrieving single parliamentary case by id.
get_case

Usage

get_case(caseid = NA, good_manners = 0)

Arguments

caseid: Character string indicating the id of the case to request

good_manners: Integer. Seconds delay between calls when making multiple calls to the same function

Value

A list with seven data frame elements:

1. $root (main data on the case)

   response_date: Date of data retrieval
   version: Data version from the API
   document_group: Case document group type
   finalized: Whether the case finalized
   reference: Relevant publication references
   id: Case id
   req_text: Recommendation (proposal) text
   committee_id: Id of committee handling the case
   title_short: Case short title
   decision_short: Case decision_short
   parenthesis_text: Case parenthesis text
   case_number: Case number
   session_id: Session id
   proceedings_id: Type of proceeding id
   proceedings_name: Type of proceeding name
   status: Status for case
   title: Case title (long)
   type: Case type
   decision_text: Decision text

2. $topic (the topics related to the case)

   is_main_topic: Is this (row) the main topic?
   main_topic_id: Id for main topic
   id: Topic id
   navn: Topic name

3. $publication_references (references for publications on the case)
get_case

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>export_id</td>
<td>Id for export of publication (used in ?get_publication)</td>
</tr>
<tr>
<td>link_text</td>
<td>Publication title</td>
</tr>
<tr>
<td>link_url</td>
<td>URL to publication</td>
</tr>
<tr>
<td>type</td>
<td>Publication type</td>
</tr>
<tr>
<td>subtype</td>
<td>Publication subtype (chamber)</td>
</tr>
</tbody>
</table>

4. **$proposers** (MPs behind case proposal, when relevant)

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mp_id</td>
<td>MPs id</td>
</tr>
<tr>
<td>party_id</td>
<td>Party id of MPs</td>
</tr>
<tr>
<td>sub_mp</td>
<td>Whether MPs are substitutes</td>
</tr>
</tbody>
</table>

5. **$proceeding_steps** (case proceeding steps)

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>step_name</td>
<td>Name of steps</td>
</tr>
<tr>
<td>step_number</td>
<td>Step order for case</td>
</tr>
<tr>
<td>outdated</td>
<td>Whether the step type is outdated</td>
</tr>
</tbody>
</table>

6. **$spokespersons** (all MPs that are spokespersons for the case)

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mp_id</td>
<td>MPs id</td>
</tr>
<tr>
<td>party_id</td>
<td>Party id of MPs</td>
</tr>
<tr>
<td>sub_mp</td>
<td>Whether MPs are substitutes</td>
</tr>
</tbody>
</table>

7. **$keywords** (all keywords associated with the case)

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>keyword</td>
<td>Keywords for the case</td>
</tr>
</tbody>
</table>

See Also

get_session_cases

Examples

```r
## Not run:

# Get one case
case <- get_case("30233")
case
```
# Get multiple cases

```r
cases <- lapply(c("30233", "30362", "30234", "30236"), get_case, good_manners = 2)
cases_root <- lapply(cases, function(x) x$root)
cases_root <- do.call(rbind, cases_root)
cases_root

cases_keywords <- lapply(1:nrow(cases_root), function(x){
  tmp <- cases[[x]]$keywords
  tmp$case_id <- cases_root$id[x]
  return(tmp)
})
cases_keywords <- do.call(rbind, cases_keywords)
cases_keywords
```

## End(Not run)

---

**get_counties**  
*Get list of electoral districts*

**Description**

A function for retrieving current and/or historical electoral districts (counties) for the Norwegian parliament.

**Usage**

```r
get_counties(historical = FALSE)
```

**Arguments**

- `historical` Logical. Whether or not to include historical counties.

**Value**

A data frame with the following variables:

- `response_date` Date of data retrieval
- `version` Data version from the API
- `historical_county` Whether the county is historical (no longer exists)
- `id` Id of the county
- `name` Name of the county

**Examples**

```r
## Not run:
```
# Request one MP by id
get_counties()

# With historical counties
get_counties(historical = TRUE)

## End(Not run)

get_decision_votes  
Retreive vote decision for a specified vote

Description

A function for retrieving vote decisions from a specific vote. Vote data are only available from the 2011-2012 session

Usage

get_decision_votes(voteid = NA, good_manners = 0)

Arguments

voteid  Character string indicating the id of the vote to request all votes from

good_manners  Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- `response_date`: Date of data retrieval
- `version`: Data version from the API
- `vote_id`: Id of the vote
- `decision_code`: General code for decision
- `decision_comment`: Comments for the decision
- `decision_number`: Decision number
- `decision_reference`: Reference for the decision
- `decision_text`: Full text of the decision

See Also

get_session_decisions get_proposal_votes get_vote get_result_vote
get_hearing_input

Examples

```r
## Not run:
decision <- get_decision_votes(123)
decision

## End(Not run)
```

get_hearing_input Retrieve the hearing input for a specified hearing

Description

A function for retrieving the hearing input for a specified hearing.

Usage

```r
get_hearing_input(hearingid = NA, good_manners = 0)
```

Arguments

- `hearingid`: Character string indicating the id of the hearing to retrieve.
- `good_manners`: Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame the following variables:

- `response_date`: Date of data retrieval
- `version`: Data version from the API
- `hearing_id`: Id of the hearing
- `hearing_type`: Type of hearing
- `committee_id`: Id of committee responsible for the hearing
- `hearing_input_date`: Date of receiving input
- `hearing_input_id`: Hearing input id
- `hearing_input_organization`: Organization giving input
- `hearing_input_text`: Full text of the hearing input
- `hearing_input_title`: Title of the hearing input

See Also

- `get_session Hearings`
- `get_hearing Program`
- `get_written_hearing_input`
get_hearing_program

Examples

```r
## Not run:
get_hearing_input(hearingid = 10004166)
## End(Not run)
```

get_hearing_program  Retrieve the hearing program for a specified hearing

Description

A function for retrieving the hearing program for a specified hearing. The earlier periods (around 2005 and back) are less rich with data on some variables

Usage

`get_hearing_program(hearingid = NA, good_manners = 0)`

Arguments

- `hearingid` Character string indicating the id of the hearing to retrieve.
- `good_manners` Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- `response_date` Date of data retrieval
- `version` Data version from the API
- `hearing_id` Id of the hearing
- `hearing_type` Type of hearing
- `committee_id` Id of committee responsible for the hearing
- `hearing_program_date` Date hearing program
- `hearing_program_footnote` Footnote for hearing program
- `order_number` Order number for the hearing program element
- `text` Description of the hearing participant
- `time_indication` Time stamp for participant hearing input
- `date` Date of participant input in hearing

See Also

`get_session_hearings` `get_hearing_input` `get_written_hearing_input`
get_meeting_agenda

Retreive agenda for a specified meeting

description
A function for retrieving the agenda for a specific meeting.

usage
get_meeting_agenda(meetingid = NA, good_manners = 0)

arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>meetingid</td>
<td>Character string indicating the id of the meeting to request all votes from</td>
</tr>
<tr>
<td>good_manners</td>
<td>Integer. Seconds delay between calls when making multiple calls to the same</td>
</tr>
</tbody>
</table>

value
A data.frame with the following variables:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>response_date</td>
<td>Date of data retrieval</td>
</tr>
<tr>
<td>version</td>
<td>Data version from the API</td>
</tr>
<tr>
<td>agenda_number</td>
<td>The agenda number for the session</td>
</tr>
<tr>
<td>meeting_date</td>
<td>Date of the meeting</td>
</tr>
<tr>
<td>meeting_id</td>
<td>Meeting id</td>
</tr>
<tr>
<td>meeting_place</td>
<td>Where the meeting was held</td>
</tr>
<tr>
<td>agenda_case_reference</td>
<td>Reference for the case on the agenda</td>
</tr>
<tr>
<td>agenda_case_number</td>
<td>Case number</td>
</tr>
<tr>
<td>agenda_case_text</td>
<td>Case description</td>
</tr>
<tr>
<td>agenda_case_type</td>
<td>Case type</td>
</tr>
<tr>
<td>footnote</td>
<td>Footnote for the case</td>
</tr>
<tr>
<td>proposition_id</td>
<td>If relevant, belonging proposition id</td>
</tr>
<tr>
<td>committee_id</td>
<td>If relevant, id of the responsible committee</td>
</tr>
<tr>
<td>loose_proposals</td>
<td>Whether there are loose proposals to the case</td>
</tr>
<tr>
<td>case_id</td>
<td>Id of the case</td>
</tr>
</tbody>
</table>
get_mp

**Extract information on specific MPs**

**Description**

A function for retrieving information on Norwegian MPs from the parliament API

**Usage**

```r
going get_mp(mpid = NA, good_manners = 0)
```

**Arguments**

- `mpid` Character string indicating the id of the MP to retrieve.
- `good_manners` Integer. Seconds delay between calls when making multiple calls to the same function

**Value**

A data.frame with the following variables:

- `response_date` Date of data retrieval
- `version` Data version from the API
- `death` MP date of death, if applicable
- `last_name` MP last name
- `birth` MP date of birth
- `first_name` MP first name
- `id` MP id
- `gender` MP gender

**See Also**

`get_session_meetings` `get_case` `get_question` `get_question_hour`

**Examples**

```r
## Not run:
meetings0910 <- get_session_meetings("2009-2010")
meeting_agenda <- get_meeting_agenda(meetings0910$meeting_id[161])
meeting_agenda
## End(Not run)
```
get_mp_bio

See Also
get_mp_bio get_parlperiod_mps get_mp_pic get_session_mp_speech_activity

Examples

```r
## Not run:
# Request one MP by id
get_mp("AAMH")

# Request several MPs by id
ids <- c("AAMH", "AMSK", "MAAA")
mps <- lapply(ids, get_mp, good_manners = 2)
mps <- do.call(rbind, mps)
## End(Not run)
```

---

get_mp_bio 
Extract biography of specific MPs

Description

A function for retrieving biography of Norwegian MPs from the parliament API

Usage

`get_mp_bio(mpid = NA, good_manners = 0)`

Arguments

- `mpid` Character string indicating the id of the MP to retrieve.
- `good_manners` Integer. Seconds delay between calls when making multiple calls to the same function

Value

A list with ten data frames:

1. `$root` (main data on the MP)

   - `response_date` Date of data retrieval
   - `version` Data version from the API
   - `id` Id of the MP
2. **Literature** (all literature the MP contributed to)

- **year**: Year of entry publication
- **description**: Description of the publication
- **last_name**: MP’s last name
- **more_years**: Not described in the API
- **publisher**: Publisher
- **first_name**: First name of the MP
- **place**: Place of publication
- **title**: Title of the publication
- **type**: MP’s role in publication (author etc)

3. **Leave of absence** (times the MP was on leave)

- **from_date**: Start date of leave
- **reason**: Reason for leave
- **to_date**: End of leave
- **type**: Type of leave
- **sub_last_name**: Substitute MP last name (id not recorded)
- **sub_first_name**: Substitute MP first name

4. **Personalia** (the MP’s personalia)

- **seniority_aar**: Number of years in parliament
- **seniority_dager**: Number of extra days (addition to years)
- **county_of_birth**: Birth county of the MP
- **municipality_of_birth**: Birth municipality of the MP
- **eulogy_date**: Eulogy date of the MP, when applicable

5. **Father** (the MP’s father personalia)

- **death_year**: Father’s year of death
- **last_name**: Father’s last name
- **birth_year**: Father’s year of birth
- **first_name**: Father’s first name
- **profession**: Father’s profession

6. **Mother** (the MP’s mother personalia)
7. **$parl_periods** (parliamentary periods the MP has held a seat)

- **from_date**: Date MP held seat from
- **county**: County the MP represented
- **party_id**: Party id for the MP’s party
- **rep_number**: Representative number (within the whole parliament)
- **parl_period_id**: Id of the parliamentary period
- **to_date**: Date MP held a seat to
- **type**: Type of representation

8. **$parl_positions** (parliamentary positions held by the MP)

- **from_year**: Year MP held position from
- **from_date**: Date MP held position from
- **committee_id**: Id of the position (in committee, cabinet, delegation, etc)
- **committee_name**: Position name
- **committee_type**: Position type
- **sort**: **Not described in the API**
- **parl_period_id**: Parliamentary period the position was held (cabinet data missing)
- **to_year**: Year MP held position to
- **to_date**: Date MP held position to

9. **$vocation** (vocation and education of the MP outside of parliament)

- **several_periods_text**: Text description if the vocation was held for several periods
- **from_year**: Year MP held vocation from
- **from_year_unknown**: Logical indication for whether the start year is unknown
- **note**: Note for vocation
- **name**: Name of vocation
- **to_year**: Year MP held vocation to
- **to_year_unknown**: Logical indication for whether the end year is unknown
- **type**: Vocation type (10 = education, 20 = work)

10. **$other_positions** (other positions held outside parliament)
See Also

get_mp get_parlperiod_mps get_mp_pic get_session_mp_speech_activity

Examples

## Not run:

# Request one MP by id
get_mp_bio("AAMH")

## End(Not run)

---

**get_mp_pic**

*Retrieve picture of specific MPs*

Description

A function for retrieving Norwegian MP pictures by id.

Usage

```r
get_mp_pic(mpid = NA, size = "middels",
            destfile = NA, show_plot = FALSE,
            good_manners = 0)
```
get_parlperiods

Get Parliamentary Periods

Description

A function for retrieving dates of the parliamentary periods after WWII

Usage

get_parlperiods()

Value

A data.frame with the following variables:

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mpid</td>
<td>Character string indicating the id of the MP to retrieve.</td>
</tr>
<tr>
<td>size</td>
<td>Character string size of the picture. Accepts values &quot;lite&quot; (small), &quot;middels&quot; (medium – default), and &quot;stort&quot; (big).</td>
</tr>
<tr>
<td>destfile</td>
<td>Character string specifying where to save the picture</td>
</tr>
<tr>
<td>show_plot</td>
<td>Logical. FALSE (default) if no plot should be produced and TRUE if plot should be produced. Requires the &quot;imager&quot; package.</td>
</tr>
<tr>
<td>good_manners</td>
<td>Integer. Seconds delay between calls when making multiple calls to the same function</td>
</tr>
</tbody>
</table>

Value

Picture of the requested MP in the preferred size.

See Also

get_mp get_parlperiod_mps get_mp_bio

Examples

```r
## Not run:
# Request one MP by id
get_mp_pic(mpid = "AAMH", destfile = "~/Pictures/AAMH.jpeg", show_plot = TRUE, size = "stort")

# With good manners for multiple calls
lapply(c("AAMH", "CIH", "TKF"), function(x){
  get_mp_pic(mpid = x, destfile = paste0("~/Pictures/", x),
    show_plot = TRUE, size = "stort", good_manners = 2)
})
## End(Not run)
```
get_parlperiod_mps

response_date  Date of data retrieval
version  Data version from the API
from  Date session started
id  Id of for session (used for other functions)
to  Date session ended
years  From year to year in full format

See Also
get_parlsessions

Examples

## Not run:

parlper <- get_parlperiods()
parlper

## End(Not run)

get_parlperiod_mps  Get list of MPs in a given parliamentary period

Description

A function for retrieving Norwegian MPs for a given parliamentary period from the parliament API

Usage

get_parlperiod_mps(periodid = NA, substitute = FALSE, good_manners = 0)

Arguments

periodid  Character string indicating the id of the parliamentary period to retrieve.
substitute  Logical. Whether or not to include substitute MPs.
good_manners  Integer. Seconds delay between calls when making multiple calls to the same function
get_parlperiod_presidency

Value

A data.frame with the following variables:

- **response_date**: Date of data retrieval
- **version**: Data version from the API
- **death**: Date of death
- **lastname**: MP last name
- **birth**: Date of birth
- **firstname**: MP first name
- **mp_id**: MP id
- **gender**: MP gender
- **county_id**: Id of county MP represented
- **party_id**: Id of party MP represented
- **substitute_mp**: Logical for whether MP is a substitute
- **period_id**: Id of period represented in

See Also

get_mp_bio get_mp get_mp_pic get_session_mp_speech_activity

Examples

```r
## Not run:
# Request one MP by id
get_parlperiod_mps("2005-2009")

# Request MPs from several periods by id
mps <- lapply(ids, get_parlperiod_mps, good_manners = 2)
mps <- do.call(rbind, mps)

## End(Not run)
```

get_parlperiod_presidency

*Get list of presidency in a given parliamentary period*

Description

A function for retrieving the presidency for a given parliamentary period from the parliament API.
Usage

get_parlperiod_presidency(periodid = NA, good_manners = 0)

Arguments

periodid  Character string indicating the id of the parliamentary period to retrieve.

good_manners  Integer. Seconds delay between calls when making multiple calls to the same
function

Value

A data.frame with the following variables:

- **response_date**: Date of data retrieval
- **version**: Data version from the API
- **last_name**: Last name of presidency member
- **first_name**: First name of presidency member
- **from_date**: Presidency member from date
- **party_id**: Party affiliation of presidency member
- **person_id**: Id of the presidency member
- **to_date**: Presidency member to date
- **position**: Presidency position

See Also

get_mp get_mp_bio

Examples

```r
## Not run:

# Request one MP by id
get_parlperiod_presidency("2005-2009")
```

```r
## End(Not run)
```

get_parlsessions  Get Parliamentary Sessions

Description

A function for retrieving dates of the parliamentary sessions after WWII
Description

A function for retrieving all proceedings in Stortinget, both current and historical.

Usage

get_proceedings()

Value

A list with three dataframes:
1. $root (only download meta data)

   - **response_date**  Date of data retrieval
   - **version**  Data version from the API

2. $proceedings (description of main proceeding categories)

   - **id**  Id of proceeding
   - **name**  Name of proceeding

3. $proceedings_steps (description of proceeding steps within each main category)

   - **id**  Id of proceeding step
   - **name**  Name of proceeding step
   - **step_number**  Order of proceeding steps
   - **outdated**  Whether the step is outdated
   - **main_id**  Id for proceeding type the step belongs to

Examples

   ```r
   ## Not run:
   get_proceedings()
   
   ## End(Not run)
   ```

---

**get_proposal_votes**

Retreive all votes for a specified vote proposal

**Description**

A function for retrieving all votes from a specific vote proposal. Vote data are only available from the 2011-2012 session

**Usage**

   `get_proposal_votes(voteid = NA, good_manners = 0)`
get_publication

Arguments

voteid  Character string indicating the id of the vote to request all votes from

good_manners  Integer. Seconds delay between calls when making multiple calls to the same function

Value

A list with two elements:

1. \$proposal_vote (main data on the vote proposal)
   
   \begin{itemize}
   \item \textbf{response_date}  Date of data retrieval
   \item \textbf{version}  Data version from the API
   \item \textbf{vote_id}  Id of the vote
   \end{itemize}

2. \$proposal_by_parties\{\textbf{proposal_id}\} (what parties (id) stood behind proposal(s))

See Also

get_vote get_decision_votes get_result_vote

Examples

```
## Not run:

prop <- get_proposal_votes(7523)
prop

for(i in 1:length(prop$proposal_by_parties)){
    prop$proposal_vote$parties[i] <- paste0(prop$proposal_by_parties[[i]],
                                            collapse = "", ""
}

## End(Not run)
```

---

get_publication  

Retrieve a specific publication

Description

A function for retrieving a specific publication. Because these are formatted very differently in the API, the returning object is in a raw html_document format, best manipulated with html packages such as rvest.
get_question

Usage

get_publication(publicationid = NA, good_manners = 0)

Arguments

  publicationid  Character string indicating the id of the publication to request all votes from
  good_manners   Integer. Seconds delay between calls when making multiple calls to the same function

Value

  A raw html_document

See Also

  get_question get_question_hour get_session_publications

Examples

## Not run:
pub <- get_publication("refs-201819-03-06")
/pub |> html_elements("replikk")[1] |> html_text()

## End(Not run)

get_question

Parliamentary question

Description

  A function for retrieving single parliamentary questions by id. For retrieving the whole debate over a question, the get_publication function has to be used

Usage

  get_question(questionid = NA, good_manners = 0)

Arguments

  questionid  Character string indicating the id of the session to request interpellations from
  good_manners  Integer. Seconds delay between calls when making multiple calls to the same function

Value

  A data.frame with the following variables:
### get_question

**response_date** Date of data retrieval  
**version** Data version from the API  
**justification** Justification for question  
**answ_by_id** Id for answering minister  
**answ_by_minister_id** Id for department of answering minister  
**answ_by_minister_title** Title for department of answering minister  
**answ_date** Date question was asked  
**answ_on_belhalf_of** Id of minister answered on behalf of, when relevant  
**answ_on_belhalf_of_minister_id** Id of department answered on behalf of, when relevant  
**answ_on_belhalf_of_minister_title** Title of department answered on behalf of, when relevant  
**agenda_number** Agenda number in meeting  
**moved_to** Date moved to  
**id** Question id  
**correct_person_id** Not documented in API  
**correct_person_minister_id** Not documented in API  
**correct_person_minister_title** Not documented in API  
**sendt_date** Date question was sent  
**session_id** Session id  
**question_text** Full question text  
**question_from_id** Id of MP asking the question  
**question_number** Question number  
**question_to_id** Id of minister the question was asked to  
**question_to_minister_id** Department id of minister the question was asked to  
**question_to_minister_title** Department title of minister the question was asked to  
**answer_text** Answer text (often empty)  
**title** Question title  
**type** Question type

### See Also

* get_question_hour  
* get_publication

### Examples

```r
## Not run:  
# An example of a possible workflow

## Retreive sessions
sessions <- get_parlsessions()

## Retreive all interpellations for a particular session
qsesh <- get_session_questions(sessions$id[9], q_type = "interpellasjoner")

## Retreve detailed information on all interpellations in that session
library(pbmclapply)  # for progress bar. never use paralell on scraping
int1213 <- pbmclapply(qsesh$id, function(x){  
get_question(x, good_manners = 2)
})```

---

**Note:** The code examples are placeholders and intended for illustrative purposes only. Actual implementation may vary depending on the specific requirements and context.
get_question_hour

Retrieve question hour details for a specified meeting

Description
A function for retrieving detailed overview of the question hour for a specific meeting

Usage
get_question_hour(meetingid = NA, good_manners = 0)

Arguments
- **meetingid**: Character string indicating the id of the meeting to request all votes from
- **good_manners**: Integer. Seconds delay between calls when making multiple calls to the same function

Value
A list with ten data frames:

1. **$root** (download meta data)
   - **response_date**: Date of data retrieval
   - **version**: Data version from the API
   - **meetingid**: The called meeting id

2. **$question_hour_ministers** (id of ministers in parliament during question hour/time)
   - **id**: Id of ministers attending

3. **$question_time**
   - **question_justification**: Justification for question
   - **answer_by_id**: Id for answering minister
   - **answer_by_minister_id**: Id for department of answering minister
answer_by_minister_title  Title for department of answering minister
answer_date     Date of receiving answer
answer_on_behalf_of_id  Id of minister answered on behalf of, when relevant
answer_on_behalf_of_minister_id  Id of department answered on behalf of, when relevant
answer_on_behalf_of_minister_title  Title of department answered on behalf of, when relevant
agenda_case_number  Case number on agenda
date     Date question hour was held
moved_to     Date moved to
asked_by_other_id  Id for minister asking on behalf of another, when relevant
question_id  Question id
correct_person  Not documented in API
correct_person_minister_id  Not documented in API
correct_person_minister_title  Not documented in API
sent_date     Date question was sent
session_id     Session id
question_text  Full question text
question_from_id  Id of MP asking the question
question_number  Question number
question_to_id  Id of minister the question was asked to
question_to_minister_id  Department id of minister the question was asked to
question_to_minister_title  Department title of minister the question was asked to
status     Question status
answer     Answer text (often empty)
title     Question title
type     Question type

4. $publication_reference

   export_id  Export id for publication (see get_publication())
   link_text  Description text for publication
   link_url  URL for publication
   type  Type of publication
   sub_type  Subtype for publication (location)

See Also

get_question get_session_questions get_publication

Examples

## Not run:
get_question_hour(10232)

## End(Not run)
get_result_voteretrieve vote results on MP level for a specified vote

Description

A function for retrieving vote results from a specific vote on MP level. Vote data are only available from the 2011-2012 session. Needs some preprocessing for use with rollcall packages, such as ideal.

Usage

get_result_vote(voteid = NA, good_manners = 0)

Arguments

voteid Character string indicating the id of the vote to request all votes from

good_manners Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- response_date Date of data retrieval
- version Data version from the API
- vote_id Id of vote
- mp_id MP id
- party_id Party id
- vote Vote: for, mot (against), ikke_tilstede (absent)
- permanent_sub_for Id of the MP originally holding the seat, if the substitute is permanent
- sub_for Id of the MP originally holding the seat

See Also

get_decision_votes get_proposal_votes get_vote get_mp_bio

Examples

## Not run:

v <- get_result_vote(12345)
table(v$vote)
get_session_cases

Cases in specified session

Description
A function for retrieving all cases treated in a specified parliamentary session.

Usage
get_session_cases(sessionid = NA, good_manners = 0, cores = 1)

Arguments

- **sessionid** Character string indicating the id of the parliamentary session to retrieve.
- **good_manners** Integer. Seconds delay between calls when making multiple calls to the same function
- **cores** Integer. Number of cores (1 by default) to use in structuring the data. More than 1 will not work on windows

Value
A data.frame with the following variables:

- **response_date** Date of data retrieval
- **version** Data version from the API
- **treated_session_id** Session the case was treated in
- **document_group** Document group the case belongs to
- **reference** Document reference
- **id** Case id
- **com_req_id** Committee recommendation id
- **com_req_code** Committee recommendation code
- **title_short** Short title of case
- **caseFiled_id** Id of filed case
- **last_update_date** Date of last update on case
- **status** Status of the case
- **title** Full title of the case
get_session_cases

<table>
<thead>
<tr>
<th>type</th>
<th>Session id of the case</th>
</tr>
</thead>
<tbody>
<tr>
<td>session_id</td>
<td>Responsible committee id</td>
</tr>
</tbody>
</table>

2. **Topics** (named list by case id)

| is_main_topic | Logical indication whether the topic is the main topic |
| main_topic_id | Id of the main topic for the case |
| id | Topic id |
| name | Topic name |

3. **Proposers** (named list by case id)

| rep_id | Proposing MP id |
| county_id | County id of proposing MP |
| party_id | Party id of proposing MP |
| rep_sub | Logical indicator for whether MP is a substitute |

4. **Spokespersons** (data frame by case id)

| case_id | Case id |
| rep_id | Spokesperson(s) MP id for the case |
| county_id | County id of spokesperson MP |
| party_id | Party id of spokesperson MP |
| rep_sub | Logical indicator for whether MP is a substitute |

See Also

get_case get_vote

Examples

```
## Not run:
s0506 <- get_session_cases("2005-2006")
head(s0506)

## End(Not run)
```
get_session_committees

Parliamentary committees in specified session

Description
A function for retrieving Norwegian parliamentary committees for a specified parliamentary session

Usage
get_session_committees(sessionid = NA, good_manners = 0)

Arguments
sessionid Character string indicating the id of the parliamentary session to retrieve.
good_manners Integer. Seconds delay between calls when making multiple calls to the same function

Value
A data.frame with the following variables:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>response_date</td>
<td>Date of data retrieval</td>
</tr>
<tr>
<td>version</td>
<td>Data version from the API</td>
</tr>
<tr>
<td>id</td>
<td>Committee id</td>
</tr>
<tr>
<td>name</td>
<td>Committee name</td>
</tr>
<tr>
<td>session_id</td>
<td>Session id</td>
</tr>
</tbody>
</table>

See Also
get_all_committees get_mp_bio get_mp

Examples

## Not run:

```r
coms <- get_session_committees("2001-2002")
coms

## End(Not run)
```
get_session_decisions  Retrieve all decisions for a specified session

Description

A function for retrieving all decisions from a specific parliamentary session.

Usage

get_session_decisions(sessionid = NA, good_manners = 0)

Arguments

sessionid  Character string indicating the id of the session to request all votes from

good_manners  Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>response_date</td>
<td>Date of data retrieval</td>
</tr>
<tr>
<td>version</td>
<td>Data version from the API</td>
</tr>
<tr>
<td>session_id</td>
<td>Session id</td>
</tr>
<tr>
<td>decision_id</td>
<td>Decision id</td>
</tr>
<tr>
<td>case_id</td>
<td>Case id</td>
</tr>
<tr>
<td>case_link_url</td>
<td>URL for case to front end web page</td>
</tr>
<tr>
<td>decision_date</td>
<td>Decision date</td>
</tr>
<tr>
<td>decision_link_url</td>
<td>URL for decision to front end web page</td>
</tr>
<tr>
<td>decision_number</td>
<td>Decision number within session</td>
</tr>
<tr>
<td>decision_text</td>
<td>Decision text</td>
</tr>
<tr>
<td>decision_title</td>
<td>Decision title</td>
</tr>
<tr>
<td>decision_type_id</td>
<td>Decision type id</td>
</tr>
<tr>
<td>decision_type_name</td>
<td>Decision type name</td>
</tr>
</tbody>
</table>

See Also

get_decision_votes

Examples

```R
## Not run:

desci <- get_session_decisions("2004-2005")
head(desci)
```
get_session_delegations

Parliamentary delegations in specified session

Description
A function for retrieving delegations for a specified parliamentary session.

Usage
get_session_delegations(sessionid = NA, good_manners = 0)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sessionid</td>
<td>Character string indicating the id of the parliamentary session to retrieve.</td>
</tr>
<tr>
<td>good_manners</td>
<td>Integer. Seconds delay between calls when making multiple calls to the same function</td>
</tr>
</tbody>
</table>

Value
A data.frame with the following variables:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>response_date</td>
<td>Date of data retrieval</td>
</tr>
<tr>
<td>version</td>
<td>Data version from the API</td>
</tr>
<tr>
<td>id</td>
<td>Delegation id</td>
</tr>
<tr>
<td>name</td>
<td>Name of delegation</td>
</tr>
<tr>
<td>session_id</td>
<td>Session id</td>
</tr>
</tbody>
</table>

See Also
get_session_committees get_all_committees

Examples

## Not run:

deleg <- get_session_delegations("2015-2016")
deleg

## End(Not run)
get_session_hearings  Retrieve hearings in specified session

Description

A function for retrieving all hearings in a specified parliamentary session.

Usage

get_session_hearings(sessionid = NA, good_manners = 0, cores = 1)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sessionid</td>
<td>Character string indicating the id of the parliamentary session to retrieve.</td>
</tr>
<tr>
<td>good_manners</td>
<td>Integer. Seconds delay between calls when making multiple calls to the same</td>
</tr>
<tr>
<td></td>
<td>function</td>
</tr>
<tr>
<td>cores</td>
<td>Integer...</td>
</tr>
</tbody>
</table>

Value

A list with four elements:

1. $root (hearing meta data)
   - response_date  Date of data retrieval
   - version        Data version from the API
   - session_id     Session id

2. $hearing (main data on the hearing)
   - deadline_date  Deadline date for hearing
   - status         Data version from the API
   - hearing_id     Hearing id
   - input_deadline Deadline date for input
   - written        Logical indication of whether the input was written
   - application_deadline Deadline date for application to hearing
   - start_date     Start date for hearing
   - status_pub     Publication status for hearing
   - status_info_text Status information text
   - type           Type of hearing
   - committee_id   Committee id for committee responsible for hearing
3. **$hearing_case_info** (named list by hearing id with information on the case(s) belonging to the hearing)

<table>
<thead>
<tr>
<th>hearing_id</th>
<th>Hearing id</th>
</tr>
</thead>
<tbody>
<tr>
<td>case_reference</td>
<td>Text reference for case</td>
</tr>
<tr>
<td>case_id</td>
<td>Case id</td>
</tr>
<tr>
<td>case_short_title</td>
<td>Short title for case</td>
</tr>
<tr>
<td>case_publication</td>
<td>URL for front end web-page publication</td>
</tr>
<tr>
<td>case_title</td>
<td>Full title for case</td>
</tr>
</tbody>
</table>

4. **$hearing_date** (named list by hearing id with date(s) the hearing was held)

<table>
<thead>
<tr>
<th>hearing_id</th>
<th>Hearing id</th>
</tr>
</thead>
<tbody>
<tr>
<td>date</td>
<td>Date of hearing</td>
</tr>
<tr>
<td>place</td>
<td>Where the hearing was held</td>
</tr>
</tbody>
</table>

See Also

get_hearing_input get_hearing_program get_written_hearing_input

Examples

```r
## Not run:

hear <- get_session_hearings("2010-2011")
head(hear$hearing)

## End(Not run)
```

---

**get_session_meetings**  
Retrieve all meetings for a specified parliamentary session

**Description**

A function for retrieving meetings from a specific parliamentary session

**Usage**

```r
get_session_meetings(sessionid = NA, good_manners = 0)
```
get_session_mp_speech_activity

Arguments

- **sessionid**: Character string indicating the id of the session to request all votes from
- **good_manners**: Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- **response_date**: Date of data retrieval
- **version**: Data version from the API
- **session_id**: Session id
- **agenda_number**: Agenda number within the session
- **footnote**: Footnotes for the meeting
- **meeting_id**: Meeting id
- **no_meeting_text**: Description of why there was no meeting, if relevant
- **evening_meeting**: Whether the meeting was an evening meeting or not
- **note**: Note for the meeting
- **meeting_date**: Date the meeting took place
- **meeting_order**: Indicator for meeting order
- **meeting_place**: Where the meeting took place
- **transcript_id**: Id for transcript (usually empty)
- **additional_agenda**: Logical indicator for whether there was additional agenda to the meeting

See Also

- get_meeting_agenda
- get_question_hour

Examples

```r
## Not run:

meet <- get_session_meetings("2013-2014")
head(meet)

## End(Not run)
```

---

**get_session_mp_speech_activity**

*Retrieve all speech activity from one MP for a given session*
get_session_mp_speech_activity

Description

A function for retrieving all speech activity from an MP during a specific parliamentary session. Only available from the 2011-2012 session and onwards.

Usage

get_session_mp_speech_activity(sessionid = NA, mp_id = NA, good_manners = 0)

Arguments

sessionId Character string indicating the session to retrieve speeches from.
mp_id Character string for the MP to retrieve all speeches of in a given session.
good_manners Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- response_date Date of data retrieval
- version Data version from the API
- session_id Session id
- agenda_case_number Number indicating the agenda number for the case
- meeting_id Meeting id
- speech_start_time Start time of speech
- speech_type Type of speech
- speech_length_secs Length of speech in seconds

See Also

get_mp get_mp_bio get_publication

Examples

## Not run:

activ <- get_session_mp_speech_activity("2012-2013", "ALYS")
head(activ)

## End(Not run)
get_session_parties

Parliamentary parties in specified session

Description

A function for retrieving Norwegian parliamentary parties for a specified parliamentary session.

Usage

get_session_parties(sessionid = NA, good_manners = 0)

Arguments

- sessionid: Character string indicating the id of the parliamentary session to retrieve.
- good_manners: Integer. Seconds delay between calls when making multiple calls to the same function.

Value

A data.frame with the following variables:

- response_date: Date of data retrieval
- version: Data version from the API
- id: Party id
- name: Party name
- represented_party: Logical indication of whether party was represented
- session_id: Session id
- period_id: Parliamentary period id

See Also

- get_all_parties

Examples

```r
# Not run:
parties <- get_session_parties("2003-2004")
parties

# End(Not run)
```
get_session_publications

Retrieve publications of a type in a parliamentary session

Description

A function for retrieving one of several publication types within a parliamentary session.

Usage

get_session_publications(sessionid = NA, type = "referat", good_manners = 0)

Arguments

sessionid  Character string indicating the id of the hearing to retrieve.

 type  Character specifying type of publication to download. Available types are "referat" (minutes), "innstilling" (proposition), "innberetning" (report), "lovvedtak" (law decision), "lovanmerkning" (law note), "dok8" (MP proposal) "dok12" (Constitutional proposal), and "dokumentserie" (document series). Defaults to "referat".

good_manners  Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- response_date  Date of data retrieval
- version  Data version from the API
- session_id  Session id
- publication_date  Date of publication
- publication_id  Id of publication
- publication_format  Publication format (XML)
- publication_available_date  When the publication was made available
- publication_title  Publication title
- publication_type  Publication type

See Also

get_publication

Examples

## Not run:
get_session_questions  Parliamentary questions in a session

Description
A function for retrieving all questions within a parliamentary session.

Usage
get_session_questions(sessionid = NA, q_type = NA, status = NA, good_manners = 0)

Arguments
- **sessionid**: Character string indicating the id of the session to request interpellations from
- **q_type**: Character string indicating type of question to retrieve. Options are "interpellasjoner" (interpellations), "sporretimesporsmal" (oral questions), or "skriftligesporsmal" (written questions).
- **status**: Character string question status extraction. Possible values are NA (extract all questions), "til_behandling" (pending questions), "trukket" (withdrawn questions), "bortfalt" (lapsed questions), or "alle" (all questions)
- **good_manners**: Integer. Seconds delay between calls when making multiple calls to the same function

Value
A data.frame with the following variables:

- **response_date**: Date of data retrieval
- **version**: Data version from the API
- **answ_by_id**: Id of minister answering question
- **answ_by_minister_id**: Department id of answering minister
- **answ_by_minister_title**: Department title of answering minister
- **answ_date**: Date answer was given
- **answ_on_belhalf_of**: Answer given on behalf of
- **answ_on_belhalf_of_minister_id**: Department id of minister given answer on behalf of
- **answ_on_belhalf_of_minister_title**: Department title of minister given answer on behalf of
- **topic_ids**: Id of relevant topics for question
- **moved_to**: Question moved to
### get_topics

Get list of topics and sub-topics for the Norwegian parliament

<table>
<thead>
<tr>
<th>asked_by_other_id</th>
<th>MP id, if question was not asked by the questioning MP id</th>
</tr>
</thead>
<tbody>
<tr>
<td>correct_person</td>
<td>Question id</td>
</tr>
<tr>
<td>correct_person_minister_id</td>
<td>Not documented in API</td>
</tr>
<tr>
<td>correct_person_minister_title</td>
<td>Not documented in API</td>
</tr>
<tr>
<td>sendt_date</td>
<td>Date the question was sent</td>
</tr>
<tr>
<td>session_id</td>
<td>Session id</td>
</tr>
<tr>
<td>question_from_id</td>
<td>Question from MP id</td>
</tr>
<tr>
<td>question_number</td>
<td>Question number within session</td>
</tr>
<tr>
<td>question_to_id</td>
<td>Question directed to minister id</td>
</tr>
<tr>
<td>question_to_minister_id</td>
<td>Question directed to minister department id</td>
</tr>
<tr>
<td>question_to_minister_title</td>
<td>Question directed to minister department title</td>
</tr>
<tr>
<td>status</td>
<td>Question status</td>
</tr>
<tr>
<td>title</td>
<td>Question title</td>
</tr>
<tr>
<td>type</td>
<td>Question type</td>
</tr>
</tbody>
</table>

**See Also**

get_question_hour, get_question

### Examples

```r
## Not run:
interp <- get_session_questions(sessionid = "2013-2014",
                                 q_type = "interpellasjoner",
                                 status = "trukket")
interp
## End(Not run)
```

### Description

A function for retrieving topic keys used to label various data from the Norwegian parliament.

### Usage

```
get_topics(keep_sub_topics = TRUE)
```

### Arguments

- **keep_sub_topics**
  
  Logical. Whether to keep sub-topics (default) for all main topics or not.
Value

A list with two elements:

1. `$topics` (All topics)

   - `response_date`: Date of data retrieval
   - `version`: Data version from the API
   - `is_main_topic`: Logical indicator for whether the topic is a main topic
   - `main_topic_id`: Id of main topic
   - `id`: Id of topic
   - `name`: Name of topic

2. `$main_topics` (exclusively main topics, if `keep_sub_topics = TRUE`)

   - `response_date`: Date of data retrieval
   - `version`: Data version from the API
   - `is_main_topic`: Logical indicator for whether the topic is a main topic
   - `main_topic_id`: Id of main topic
   - `id`: Id of topic
   - `name`: Name of topic

Examples

```r
# Request the data
tops <- get_topics()

# Look at the first main topic
tops$main_topics[1,]

# Extract all sub-topics for the first main topic
tops$topics[which(tops$topics$main_topic_id == 5),]
```

---

### get_vote

Retrive votes for a specific case

**Description**

A function for retrieving all votes from a case. Vote data are only available from the 2011-2012 session

**Usage**

```r
get_vote(caseid = NA, good_manners = 0)
```
get_vote

Arguments

  caseid  Character string indicating the id of the case to request all votes from
  good_manners  Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame with the following variables:

- response_date: Date of data retrieval
- version: Data version from the API
- case_id: Case id up for vote
- alternative_vote: Whether vote is an alternative vote
- n_for: Number of votes for
- n_absent: Number of MPs absent
- n_against: Number of votes against
- treatment_order: Order of treated votes
- agenda_case_number: Case number on the agenda of the meeting
- free_vote: Logical indication of whether the vote is related to the case as a whole
- comment: Vote comment
- meeting_map_number: Number on the meeting map
- personal_vote: Logical indication of whether vote was recorded as roll call or not
- president_id: Id of president holding president chair at the time of voting
- president_party_id: Party of the sitting president
- adopted: Logical indication of whether the proposal voted on was adopted
- vote_id: Id of vote
- vote_method: Voting method
- vote_result_type: Result type (enstemmig_vedtatt = unanimously adopted)
- vote_result_type_text: See vote_result_type
- vote_topic: Description of the proposal voted upon
- vote_datetime: Date and time of vote

See Also

general_votes get_proposal_votes get_vote get_session_cases get_case

Examples

```r
## Not run:

get_vote(63033)

## End(Not run)
```
get_written_hearing_input

Retrieve written input for a specified hearing

Description

A function for retrieving written input for a specified hearing

Usage

get_written_hearing_input(hearingid = NA, good_manners = 0)

Arguments

hearingid Character string indicating the id of the hearing to retrieve.
good_manners Integer. Seconds delay between calls when making multiple calls to the same function

Value

A data.frame the following variables:

response_date Date of data retrieval
version Data version from the API
hearing_id Id of the hearing
hearing_type Type of hearing
committee_id Id of committee responsible for the hearing
hearing_input_date Date of receiving input
hearing_input_id Hearing input id
hearing_input_organization Organization giving input
hearing_input_text Full text of the hearing input
hearing_input_title Title of the hearing input

See Also

get_hearing_input get_hearing_program get_session_hearings

Examples

## Not run:

get_written_hearing_input(hearingid = 10004243)

## End(Not run)
Description

A dataset containing all interpellations in the 2002-2003 parliamentary session in Stortinget

Usage

interp0203

Format

A data frame with 26 columns and 22 rows

- **response_date** - Date of data retrieval
- **version** - Data version from the API
- **answ_by_id** - Id of minister answering question
- **answ_by_minister_id** - Department id of answering minister
- **answ_by_minister_title** - Department title of answering minister
- **answ_date** - Date answer was given
- **answ_on_belhalf_of** - Answer given on behalf of
- **answ_on_belhalf_of_minister_id** - Department id of minister given answer on behalf of
- **answ_on_belhalf_of_minister_title** - Department title of minister given answer on behalf of
- **topic_ids** - Id of relevant topics for question
- **moved_to** - Question moved to
- **asked_by_other_id** - MP id, if question was not asked by the questioning MP
- **id** - Question id
- **correct_person** - Not documented in API
- **correct_person_minister_id** - Not documented in API
- **correct_person_minister_title** - Not documented in API
- **sendt_date** - Date the question was sent
- **session_id** - Session id
- **question_from_id** - Question from MP id
- **question_number** - Question number within session
- **question_to_id** - Question directed to minister id
- **question_to_minister_id** - Question directed to minister department id
- **question_to_minister_title** - Question directed to minister department title
- **type** - Question type
- **title** - Question title
- **status** - Question status
Source


mps4549

Members of parliament from the 1945-1949

Description

A dataset containing all MPs during the 1945-1949 parliamentary period in Stortinget

Usage

mps4549

Format

A data frame with 12 columns and 150 rows

response_date  Date of data retrieval
version  Data version from the API
date  Date of death
lastname  MP last name
birth  Date of birth
firstname  MP first name
mp_id  MP id
gender  MP gender
county_id  Id of county MP represented
party_id  Id of party MP represented
substitute_mp  Logical for whether MP is a substitute
period_id  Id of period represented in

Source

https://data.stortinget.no/eksport/representanter?stortingsperiodeid=1945-49
**parl_periods**  
*Parliamentary periods*

**Description**
A dataset containing all parliamentary periods in *Stortinget*

**Usage**
parl_periods

**Format**
A data frame with 12 columns and 150 rows

- **response_date** Date of data retrieval
- **version** Data version from the API
- **from** Date session started
- **id** Id of for session (used for other functions)
- **to** Date session ended
- **years** From year to year in full format

**Source**
[https://data.stortinget.no/eksport/stortingsperioder](https://data.stortinget.no/eksport/stortingsperioder)

---

**parl_sessions**  
*Parliamentary sessions*

**Description**
A dataset containing all parliamentary sessions in *Stortinget*

**Usage**
parl_sessions

**Format**
A data frame with 6 columns and 36 rows

- **response_date** Date of data retrieval
- **version** Data version from the API
- **from** Date session started
- **id** Id of for session (used for other functions)
- **to** Date session ended
- **years** From year to year in full format
read_obt  

Read Oslo-Bergen-Tagger processed files into R

Description

A function reading OBT-tagged files

Usage

read_obt(file = NA)

Arguments

file  character. Path to OBT-tagged file

Value

A data frame with the following variables:

- `sentence`  Sentence number
- `index`  Token number in sentence
- `token`  Raw token, as read by OBT originally
- `lwr`  Lowercase raw token
- `lemma`  Lemmatized token
- `pos`  Part of Speech
- `morph`  Morphological tags

Examples

```r
## Not run:
sample_text <- read_obt("./inst/extdata/obt_sample.txt")
head(sample_text)
## End(Not run)
```

vote  

Meta data on votes of case id 78686

Description

A dataset containing vote information on case id 78686 in Stortinget
Usage

data

Format

A list with three elements (votes)

response_date  Date of data retrieval
version  Data version from the API
case_id  Case id up for vote
alternative_vote  Whether vote is an alternative vote
n_for  Number of votes for
n_absent  Number of MPs absent
n_against  Number of votes against
treatment_order  Order of treated votes
agenda_case_number  Case number on the agenda of the meeting
free_vote  Logical indication of whether the vote is related to the case as a whole
comment  Vote comment
meeting_map_number  Number on the meeting map
personal_vote  Logical indication of whether vote was recorded as roll call or not
president_id  Id of president holding president chair at the time of voting
president_party_id  Party of the sitting president
adopted  Logical indication of whether the proposal voted on was adopted
vote_id  Id of vote
vote_method  Voting method
vote_result_type  Result type (enstemmig_vedtatt = unanimously adopted)
vote_result_type_text  See vote_result_type
vote_topic  Description of the proposal voted upon
vote_datetime  Date and time of vote

Source
https://data.stortinget.no/eksport/voteringsresultat?voteringid=15404
Roll call vote results for vote ids 15404, 15405, and 15406

Description
A dataset containing all personal votes for votes 15404, 15405, and 15406 in Stortinget

Usage
vote_result

Format
A list with one vote per element

- **response_date** Date of data retrieval
- **version** Data version from the API
- **vote_id** Id of vote
- **mp_id** MP id
- **party_id** Party id
- **vote** Vote: for, mot (against), ikke_tilstede (absent)
- **permanent_sub_for** Id of the MP originally holding the seat, if the substitute is permanent
- **sub_for** Id of the MP originally holding the seat

Source
Index

* datasets
  cases, 3
  covid_relief, 3
  covid_relief_result, 4
  interp0203, 47
  mps4549, 48
  parl_periods, 49
  parl_sessions, 49
  vote, 50
  vote_result, 52

  cases, 3
  covid_relief, 3
  covid_relief_result, 4
  get_all_committees, 5, 33, 35
  get_all_parties, 6, 40
  get_case, 6, 14, 32, 45
  get_counties, 9
  get_decision_votes, 10, 25, 30, 34, 45
  get_hearing_input, 11, 12, 37, 46
  get_hearing_program, 11, 12, 37, 46
  get_meeting_agenda, 13, 38
  get_mp, 14, 18, 19, 21, 22, 33, 39
  get_mp_bio, 15, 15, 19, 21, 22, 30, 33, 39
  get_mp_pic, 15, 18, 18, 21
  get_parlperiod_mps, 15, 18, 19, 20
  get_parlperiod_presidency, 21
  get_parlperiods, 19, 23
  get_parlsessions, 20, 22
  get_proceedings, 23
  get_proposal_votes, 10, 24, 30, 45
  get_publication, 25, 26, 27, 29, 39, 41
  get_question, 14, 26, 29, 43
  get_question_hour, 14, 26, 27, 28, 38, 43
  get_result_vote, 10, 25, 30
  get_session_cases, 3, 8, 31, 45
  get_session_committees, 5, 33, 35
  get_session_decisions, 10, 34
  get_session_delegations, 35
  get_session_hearings, 11, 12, 36, 46
  get_session_meetings, 14, 37
  get_session_mp_speech_activity, 15, 18, 21, 38
  get_session_parties, 6, 40
  get_session_publications, 26, 41
  get_session_questions, 29, 42
  get_topics, 43
  get_vote, 10, 25, 30, 32, 44, 45
  get_written_hearing_input, 11, 12, 37, 46

  ideal, 30
  interp0203, 47

  mps4549, 48

  parl_periods, 49
  parl_sessions, 49

  read_obt, 50
  rvest, 25

  vote, 50
  vote_result, 52