

# Package ‘congressbr’

December 12, 2019

**Type** Package

**Title** Downloads, Unpacks and Tidies Legislative Data from the  
Brazilian Federal Senate and Chamber of Deputies

**Version** 0.2.2

**Description** Downloads and tidies data from the Brazilian Federal Senate and  
Chamber of Deputies Application Programming Interfaces available at [http://  
legis.senado.gov.br/dadosabertos/](http://legis.senado.gov.br/dadosabertos/) and <https://dadosabertos.camara.leg.br/>  
respectively.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.0)

**Imports** dplyr (>= 0.7.0), purrr (>= 0.2.2), xml2 (>= 1.0.0), httr(>=  
1.2.1), lubridate (>= 1.6.0), magrittr (>= 1.5), stringi (>=  
1.1.2), tidyr (>= 0.6.0), stringr (>= 1.1.0), tibble (>= 1.2),  
pscl (>= 1.4.9), glue (>= 0.0.1), progress (>= 0.0.1), janitor  
(>= 1.1.1)

**RoxygenNote** 6.1.99.9001

**Suggests** devtools, ggplot2, httpptest, knitr, rmarkdown, covr

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Robert McDonnell [aut, cre],  
Guilherme Jardim Duarte [aut],  
Danilo Freire [aut],  
Julio Trecenti [aut]

**Maintainer** Robert McDonnell <mcdonnell.robert5@gmail.com>

**Repository** CRAN

**Date/Publication** 2019-12-12 11:20:02 UTC

**R topics documented:**

cham_bills . . . . .	3
cham_bill_info . . . . .	4
cham_bill_info_id . . . . .	5
cham_legislator_list . . . . .	6
cham_plenary_bills . . . . .	6
cham_typeauthors_bills . . . . .	7
cham_votes . . . . .	8
cham_votes_year . . . . .	9
commissions . . . . .	9
congressbr . . . . .	10
sen_agenda . . . . .	10
sen_bills . . . . .	11
sen_bills_current . . . . .	12
sen_bills_limits . . . . .	13
sen_bills_list . . . . .	14
sen_bills_locations . . . . .	14
sen_bills_passage . . . . .	15
sen_bills_passing . . . . .	16
sen_bills_situations . . . . .	17
sen_bills_status . . . . .	17
sen_bills_topics . . . . .	18
sen_bills_types . . . . .	19
sen_bills_updates . . . . .	19
sen_bills_update_types . . . . .	20
sen_bill_search . . . . .	21
sen_bill_sponsors . . . . .	23
sen_budget . . . . .	24
sen_coalitions . . . . .	24
sen_coalition_info . . . . .	25
sen_commissions . . . . .	26
sen_commissions_senators . . . . .	27
sen_commissions_type . . . . .	28
sen_commission_positions . . . . .	29
sen_nominal_votes . . . . .	30
sen_parties . . . . .	30
sen_plenary_agenda . . . . .	31
sen_plenary_leaderships . . . . .	32
sen_plenary_result . . . . .	32
sen_plenary_sessions . . . . .	33
sen_senator . . . . .	34
sen_senator_bills . . . . .	35
sen_senator_commissions . . . . .	36
sen_senator_details . . . . .	36
sen_senator_legis . . . . .	37
sen_senator_list . . . . .	38
sen_senator_mandates . . . . .	39

sen_senator_suplentes . . . . .	40
sen_senator_votes . . . . .	41
sen_sponsor_types . . . . .	42
sen_statement_list . . . . .	42
sen_votes . . . . .	43
sen_votes_year . . . . .	44
sen_votes_year_file . . . . .	45
statesBR . . . . .	45
UF . . . . .	46
vote_to_rollcall . . . . .	46

**Index** **48**

cham\_bills *Downloads and tidies data for lists of bills in Brazilian Chamber of Deputies*

**Description**

Downloads and tidies data for lists of bills in Brazilian Chamber of Deputies

**Usage**

```
cham_bills(
  type = "",
  number = "",
  year = "",
  initial_date = "",
  end_date = "",
  part_name_author = "",
  id_type_author = "",
  abbreviation_party_author = "",
  abbreviation_st_author = "",
  gender_author = "",
  cod_state = "",
  cod_branch_state = "",
  still = "",
  ascii = T
)
```

**Arguments**

- type            character. The type of the bill. For example, "PL".
- number        integer. The number of the bill
- year           integer. The year of the bill.
- initial\_date   (character) start date of the period requested. This parameter must be in the format YYYY-MM-DD (Year-Month-Day).

end\_date (character) final date for period requested. Format YYYYMMDD.  
 part\_name\_author (character). Filter bills by a part of the author's name  
 id\_type\_author (character). Filter bills by type of a branch\_author for the bill.  
 abbreviation\_party\_author (character). Filter bills by the party of the author  
 abbreviation\_st\_author (character). Filter bills by the state of the author  
 gender\_author (character). If "M", bills proposed by male authors are filtered. If "F", female authors. If empty, every bill is included.  
 cod\_state (integer). Filter bills by a part of the author's name  
 cod\_branch\_state (integer). Filter by the id of the branch regarding the bill.  
 still (integer). If 1, only bills that are still being discussed are included. If 2, only ended bills. If empty, every bill is included.  
 ascii logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
cham_bills(type = "PL", year = 2011, number = 2718, initial_date="2011-16-11")
```

---

cham_bill_info	<i>Downloads details of a specific bill by providing type, number and year</i>
----------------	--

---

**Description**

Downloads details of a specific bill by providing type, number and year

**Usage**

```
cham_bill_info(type, number, year, ascii = T)
```

**Arguments**

type	character. The type of the bill. For example, "PL"
number	integer. The number of the bill
year	integer. The year of the bill.
ascii	logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
cham_bill_info(type = "PL", number = "3962", year = "2008")
```

---

cham\_bill\_info\_id      *Downloads details of a specific bill by providing id of a bill*

---

**Description**

Downloads details of a specific bill by providing id of a bill

**Usage**

```
cham_bill_info_id(bill_id, ascii = T)
```

**Arguments**

bill_id	integer. The id number of the bill.
ascii	logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Note**

Requesting data from a long period of time with `details = TRUE` will return a large object in terms of memory. It will also be rather unwieldy, with many columns.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
cham_bill_info_id(14784)
```

---

cham\_legislator\_list *Downloads and tidies information on the legislators in the Chamber of Deputies*

---

**Description**

Downloads and tidies information on the legislators in the Federal Senate.

**Usage**

```
cham_legislator_list()
```

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
all <- cham_legislator_list()
```

---

cham\_plenary\_bills *This function lists every bill voted on in plenary.*

---

**Description**

This function lists every bill voted on in plenary.

**Usage**

```
cham_plenary_bills(year, type = "", ascii = T)
```

**Arguments**

`year` (integer) start year of the period requested.  
`type` character. The type of the bill. For example, "PL"  
`ascii` logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Note**

Requesting data from a long period of time with `details = TRUE` will return a large object in terms of memory. It will also be rather unwieldy, with many columns.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
cham_plenary_bills(year = 2008)
```

---

cham\_typeauthors\_bills

*Downloads types of authors for bills*

---

**Description**

Downloads types of authors for bills

**Usage**

```
cham_typeauthors_bills(ascii = T)
```

**Arguments**

`ascii` logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Note**

Requesting data from a long period of time with `details = TRUE` will return a large object in terms of memory. It will also be rather unwieldy, with many columns.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
cham_typeauthors_bills()
```

---

cham_votes	<i>Downloads votes of a specific bill by providing type, number and year</i>
------------	--

---

### Description

Downloads votes of a specific bill by providing type, number and year. A bill can have more than one roll call, and the API does not provide an id to identify them So we provide one (rollcall\_id).

### Usage

```
cham_votes(type, number, year, ascii = TRUE)
```

### Arguments

type	character. The type of the bill. For example, "PL" for law proposal ("projeto de lei"), "PEC" for constitutional amendments ("projeto de emenda constitucional"), "PDC" for legislative decree ("decreto legislativo"), and "PLP" for supplementary laws ("projeto de lei complementar).
number	integer. The number of the bill.
year	integer. The year of the bill.
ascii	logical. If TRUE, certain strings are converted to ascii format.

### Value

A tibble, of classes tibble, tbl\_df, tbl and data.frame.

### Note

Requesting data from a long period of time with details = TRUE will return a large object in terms of memory. It will also be rather unwieldy, with many columns.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

### Examples

```
cham_votes(type = "PL", number = "1992", year = "2007")
```



---

cham_votes_year	<i>Returns voting information from the Chamber floor for the year requested</i>
-----------------	---

---

**Description**

Returns voting information from the Chamber floor for the year requested.

**Usage**

```
cham_votes_year(year)
```

**Arguments**

year                    character or integer. Format YYYY

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
cham_votes_year("2013")
```

---

commissions	<i>A dataframe of commissions in the Brazilian Federal Senate.</i>
-------------	--

---

**Description**

- Commission character. Full name of the commission.
- Name character. Abbreviation. For use as an argument to various congressbr functions (such as sen\_senator\_comms()).

**Usage**

```
commissions
```

**Format**

A data frame with 21 rows and 2 variables

---

congressbr	<i>congressbr.</i>
------------	--------------------

---

### Description

A package for obtaining data from the APIs of the Brazilian Chamber of Deputies and the Federal Senate, respectively.

---

sen_agenda	<i>Downloads and tidies data on the agenda in the Federal Senate</i>
------------	--

---

### Description

Downloads and tidies data on the agenda in the Federal Senate.

### Usage

```
sen_agenda(
  initial_date = NULL,
  end_date = NULL,
  house = NULL,
  supervisory = NULL,
  legislator = NULL,
  details = FALSE,
  ascii = TRUE
)
```

### Arguments

<code>initial_date</code>	(character) start date of the period requested. This parameter must be in the format YYYYMMDD (Year-Month-Day). A value for this parameter is necessary, all others are optional.
<code>end_date</code>	(character) final date for period requested. Format YYYYMMDD.
<code>house</code>	(character). The acronym for the legislative house for which results are requested. Options are SF ( <i>Senado Federal</i> , Federal Senate), CN ( <i>Congresso Nacional</i> , National Congress - joint meeting of the Senate and Chamber), and CA Camara dos Deputados, Chamber of Deputies.
<code>supervisory</code>	character. Name of the commission or supervisory body. A data frame of these can be seen with <code>data("commissions")</code> .
<code>legislator</code>	integer. The numeric code given to each senator. A dataframe with these values is returned from the <code>sen_senator_list()</code> function.
<code>details</code>	(logical). If details is equal to TRUE, the data returned is an expanded dataset with additional details. This is not recommended unless necessary.
<code>ascii</code>	logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Note**

Requesting data from a long period of time with `details = TRUE` will return a large object in terms of memory. It will also be rather unwieldy, with many columns.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_agenda(initial_date = "20161105", end_date = "20161125")
```

---

sen_bills	<i>Downloads and tidies information on the legislation in the Federal Senate</i>
-----------	--

---

**Description**

Downloads and tidies information on the legislation in the Federal Senate.

**Usage**

```
sen_bills(  
  bill_id = NULL,  
  type = NULL,  
  number = NULL,  
  year = NULL,  
  ascii = TRUE  
)
```

**Arguments**

bill_id	integer. This number is the id given to each bill in the Senate database. For example, running <code>sen_bills_current()</code> will return a dataframe with the variable <code>bill_id</code> in the first column. These numbers can be used as this id. If id is not NULL (the default), all other parameters will be set to NULL, as 'bill_id' cannot be used in conjunction with the other parameters. If any one of the other parameters are used (without using <code>bill_id</code> ), all three need to be used.
type	character. The abbreviation of the vote type you're looking for. A full list of these can be obtained with the <code>sen_bill_list()</code> function. Other types can be seen with <code>sen_bills_subtypes()</code> .

number . Bill number. Simple integer, like 2, which will be bill number 00002.  
 year integer. Four-digit year, such as 2013.  
 ascii logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
pls_5_2010 <- sen_bills(type = "PLS", number = 5, year = 2010)

# Get info on the first bill in the dataframe returned
# by sen_bills_current(), which has an id of 25:
sen25 <- sen_bills(bill_id = 25)
```

---

sen_bills_current	<i>Downloads and tidies information on legislation from the current legislature of the Federal Senate</i>
-------------------	---

---

**Description**

Downloads and tidies information on legislation from the current legislature of the Federal Senate.

**Usage**

```
sen_bills_current(
  year = NULL,
  date = NULL,
  time = NULL,
  number = NULL,
  type = NULL,
  passing = NULL
)
```

**Arguments**

year Format YYYY. Returns legislation from this year.  
 date date requested, format YYYYMMDD. Returns information on legislation on or after this time.  
 time Format hour-minute-second (HHmmSS). Returns information on legislation on or after this time.  
 number bill number.

type	type of legislation.
passing	is the bill currently under consideration in the Senate? Possible values are NULL (the default), "Yes" or "No". NULL returns the information obtained from "Yes" and "No".

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
all <- sen_bills_current()
plc_2015 <- sen_bills_current(year = 2015, type = "PLS")
```

---

sen_bills_limits	<i>Downloads and tidies information on the types of deadline and time limits for legislation in the Federal Senate</i>
------------------	--

---

**Description**

Downloads and tidies information on the types of deadline and time limits for legislation in the Federal Senate.

**Usage**

```
sen_bills_limits(ascii = TRUE)
```

**Arguments**

`ascii` logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
lims <- sen_bills_limits()
```

---

sen_bills_list	<i>Downloads and tidies information on the types of acts that can be formally made in the Federal Senate</i>
----------------	--

---

**Description**

Downloads and tidies information on the types of acts that can be formally made in the Federal Senate.

**Usage**

```
sen_bills_list(active = NULL, ascii = TRUE)
```

**Arguments**

active	.Possible values are TRUE, FALSE, or can be left as NULL, in which case both active and inactive bill types are returned.
ascii	logical. TRUE by default, removes latin-1 characters from returned object.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
bills <- sen_bills_list(active = TRUE)
```

---

sen_bills_locations	<i>Downloads and tidies information on the possible locations a piece of legislation can currently be passing through the Federal Senate</i>
---------------------	--

---

**Description**

Downloads and tidies information on the possible locations a piece of legislation can currently be passing through the Federal Senate.

**Usage**

```
sen_bills_locations(active = NULL, ascii = TRUE)
```

**Arguments**

`active` character. Options are "Yes", "No" or NULL, the default. "Yes" returns active subtypes, "No" returns inactive subtypes, while the default returns both.

`ascii` logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_bills_locations()
```

---

<code>sen_bills_passage</code>	<i>Downloads and tidies information on the possible locations a piece of legislation can currently be passing through</i>
--------------------------------	---

---

**Description**

Downloads and tidies information on the possible locations a piece of legislation can currently be passing through.

**Usage**

```
sen_bills_passage(bill_id = NULL, ascii = TRUE)
```

**Arguments**

`bill_id` integer. This number is the id given to each bill in the Senate database. For example, running `sen_bills_current()` will return a dataframe with the variable `bill_id` in the first column. These numbers can be used as this id.

`ascii` logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_bills_passage(bill_id = 9123)
```

---

sen_bills_passing	<i>Downloads and tidies information on legislation that is under consideration in the Federal Senate</i>
-------------------	--

---

**Description**

Downloads and tidies information on legislation that is under consideration in the Federal Senate.

**Usage**

```
sen_bills_passing(
  year = NULL,
  number = NULL,
  type = NULL,
  date = NULL,
  time = NULL
)
```

**Arguments**

year	Format YYYY. Returns legislation from this year.
number	bill number.
type	type of legislation.
date	date requested, format YYYYMMDD. Returns information on legislation on or after this time.
time	Format hour-minute-second (HHmmSS). Returns information on legislation on or after this time.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_bills_passing()
# All MPVs (provisional presidential decrees) in 2001:
mpvs <- sen_bills_passing(year = "2001", type = "MPV")
# Bills from a certain date:
march01_2017 <- sen_bills_passing(date = 20170301)
```



---

sen\_bills\_situations *Downloads and tidies information on the possible situations a bill can be in*

---

**Description**

Downloads and tidies information on the possible situations a bill can be in.

**Usage**

```
sen_bills_situations(ascii = TRUE)
```

**Arguments**

ascii            logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_bills_situations()
```

---

sen\_bills\_status *Downloads and tidies information on legislation from the current legislature of the Federal Senate*

---

**Description**

Downloads and tidies information on legislation from the current legislature of the Federal Senate.

**Usage**

```
sen_bills_status(bill_id = NULL, ascii = TRUE)
```

**Arguments**

bill\_id            integer. This number is the id given to each bill in the Senate database. For example, running sen\_bills\_current() will return a dataframe with the variable bill\_id in the first column. These numbers can be used as this id.

ascii            logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_bills_status(bill_id = 80406)
```

---

sen_bills_topics	<i>Downloads and tidies information on the topics of legislation in the Federal Senate</i>
------------------	--

---

**Description**

Downloads and tidies information on the topics of legislation in the Federal Senate.

**Usage**

```
sen_bills_topics(active = NULL, ascii = TRUE)
```

**Arguments**

active	character. Options are "Yes", "No" or NULL, the default. "Yes" returns active subtypes, "No" returns inactive subtypes, while the default returns both.
ascii	logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
top <- sen_bills_topics()
```

---

sen_bills_types	<i>Downloads and tidies information on the types of legislation in the Federal Senate</i>
-----------------	---

---

**Description**

Downloads and tidies information on the types of legislation in the Federal Senate.

**Usage**

```
sen_bills_types(active = NULL, ascii = TRUE)
```

**Arguments**

active	character. Options are "Yes", "No" or NULL, the default. "Yes" returns active types, "No" returns inactive types, while the default returns both.
ascii	logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
legis <- sen_bills_types()
```

---

sen_bills_updates	<i>Downloads and tidies information on bills that have been recently updated in the Federal Senate</i>
-------------------	--

---

**Description**

Downloads and tidies information on bills that have been recently updated in the Federal Senate.

**Usage**

```
sen_bills_updates(  
  update = NULL,  
  year = NULL,  
  number = NULL,  
  type = NULL,  
  days = 5,  
  ascii = TRUE  
)
```

**Arguments**

update	character. This is the type of update that can be applied to a bill. For a dataframe of these, use the <code>sen_bills_update_types()</code> function, and the variable <code>update_name</code> that is returned.
year	character. Year of the bill, if a specific bill is requested. Format YYYY.
number	bill number.
type	type of legislation.
days	integer. The number of days to consider when requesting information on recent updates. The maximum is 30 and the default is 5.
ascii	logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
# Bills from 2014 that have had a "despacho" update in the last 15 days, if
# they exist:

desp_2014 <- sen_bills_updates(update = "Despacho", year = 2014, days = 15)

# PLS bills that have been updated in the last 10 days, if they exist:
pls <- sen_bills_updates(type = "PLS", days = 10)
```

---

sen\_bills\_update\_types

*Downloads and tidies information on the types of updates that can be applied to bills in the Federal Senate*

---

**Description**

Downloads and tidies information on the types of updates that can be applied to bills in the Federal Senate.

**Usage**

```
sen_bills_update_types()
```

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`, with variables:

- `update_name`: the name in the database for the type of update.
- `update_effects`: the type of item that is affected by the update. "MovimentacaoMateria" refers to the passage of the bill; "DetalheMateria" refers to the details of the bill itself; "RelatoriaMateria" refers to reports on the bill; "VotacaoMateria" refers to votes on the bill.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_bills_update_types()
```

---

sen\_bill\_search

*Search for data on legislation in the Brazilian Federal Senate*

---

**Description**

Search for data on legislation in the Brazilian Federal Senate.

**Usage**

```
sen_bill_search(  
  year = "",  
  year_law = "",  
  topic_id = "",  
  situation_id = "",  
  date_presented_init = "",  
  date_presented_end = "",  
  date_situation_init = "",  
  date_situation_end = "",  
  complementary = "",  
  present = "",  
  rapporteur = "",  
  author = "",  
  number = "",  
  type = "",  
  law_number = "",  
  party_abbr_author = "",  
  author_state = "",  
  in_passage = "",  
  ascii = TRUE  
)
```

**Arguments**

year	integer. Four-digit year, such as 2013.
year_law	integer. Year of introduction of the law, such as 2013.
topic_id	character. For a data frame of topic ids and their meanings, use <code>sen_bills_topics()</code> .
situation_id	character.
date_presented_init	character. Date when the bill that you're searching for was first presented. In the format YYYYMMDD.
date_presented_end	character. See above.
date_situation_init	character. See above.
date_situation_end	character. See above.
complementary	character. Either blank, "Yes", or "No".
present	character. Either blank, "Yes", or "No"; an indicator for whether the bill is current or not.
rapporteur	character. Name of the rapporteur of the bill, if known.
author	character. Author/sponsor of the bill. For a list, use <code>sen_bill_sponsors()</code> .
number	character. Bill number, if known.
type	character. Bill type. For a data frame of possible bill types, run <code>sen_bills_types()</code> .
law_number	character. Number of the law resulting from the bill, if known.
party_abbr_author	character. The short text code for the party of the author of the bill. For a list of the parties, use <code>sen_parties()</code> .
author_state	character. The state of the senator. For a full list, see <code>UF()</code> .
in_passage	character. Either blank, "Yes", or "No".
ascii	logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
# search for legislation from 2014:
two14 <- sen_bill_search(year = 2014)
```

---

sen_bill_sponsors	<i>Downloads and tidies data on the bill sponsors in the Federal Senate</i>
-------------------	---

---

### Description

Downloads and tidies data on the bill sponsors in the Federal Senate.

### Usage

```
sen_bill_sponsors(ascii = TRUE)
```

### Arguments

`ascii` (logical). If TRUE, bill sponsor names are converted to ascii format, stripping the latin characters from the names.

### Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`, with variables:

- `sponsor_name`: identity of the bill sponsor.
- `sponsor_code`: code for senator bill sponsors.
- `sponsor_title`: title of bill sponsor.
- `sponsor_party`: party of bill sponsor.
- `sponsor_state`: state of bill sponsor.
- `quantity`: quantity of bills sponsored by this individual or entity.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

### Examples

```
bills <- sen_bill_sponsors()
```

---

`sen_budget`*Downloads and tidies budget information from the Federal Senate*

---

**Description**

Downloads and tidies budget information from the Federal Senate.

**Usage**

```
sen_budget(ascii = TRUE)
```

**Arguments**

`ascii` logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
bud <- sen_budget()
```

---

`sen_coalitions`*Downloads and tidies data on the coalitions in the Federal Senate*

---

**Description**

Downloads and tidies data on the coalitions in the Federal Senate.

**Usage**

```
sen_coalitions(ascii = TRUE)
```

**Arguments**

`ascii` logical. If TRUE, names are converted to ascii format.



**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`, with variables:

- `bloc_code`: unique code given to each coalition.
- `bloc_name`: name of the coalition.
- `bloc_label`: additional label for the coalition.
- `date_created`: `POSIXct`, date the coalition was created.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
coalitions <- sen_coalitions()
```

---

`sen_coalition_info`      *Downloads and tidies data on specific coalitions in the Federal Senate*

---

**Description**

Downloads and tidies data on specific coalitions in the Federal Senate.

**Usage**

```
sen_coalition_info(code = NULL, ascii = TRUE)
```

**Arguments**

<code>code</code>	integer. Code of the coalition. If not known (the most likely case), these codes can be obtained from the <code>sen_coalitions()</code> function.
<code>ascii</code>	logical. If <code>TRUE</code> , names are converted to <code>ascii</code> format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`, with variables:

- `bloc_code`: unique code given to each coalition.
- `house`: legislative house for the coalition.
- `bloc_name`: name of the coalition.
- `bloc_label`: additional label for the coalition.
- `date_created`: `POSIXct`, date the coalition was created.
- `member_code`: party code.
- `member_date_joined`: `POSIXct`, date when the party first joined the coalition.
- `member_abbr`: party acronym.
- `member_name`: party name.
- `member_date_created`: `POSIXct`, date when the coalition was created.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

---

sen\_commissions

*Information on commissions in the Federal Senate*

---

**Description**

Information on commissions in the Federal Senate.

**Usage**

```
sen_commissions(active = c("Yes", "No"), ascii = TRUE)
```

**Arguments**

**active** character. Options are "Yes" or "No". If "Yes", returns only active commissions, otherwise all commissions.

**ascii** logical. If TRUE, certain strings are converted to ascii format.

**Details**

Returns a data frame with the following variables:

- **commission\_id**: unique code for each commission.
- **commission\_abbr**: Abbreviated name of the commission.
- **commission\_name**: Commission name.
- **commission\_purpose**: Objective of the commission.
- **commission\_initial\_date**: Commission starting date.
- **commission\_public**: Whether the commission is public or not.
- **commission\_house**: Legislative house where the commission is based.
- **commission\_type**: Permanent, temporary, or a parliamentary inquiry.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
comms <- sen_commissions()
```

---

`sen_commissions_senators`

*Information on the senators who serve on a certain commission in the Federal Senate*

---

## Description

Information on the senators who serve on a certain commission in the Federal Senate.

## Usage

```
sen_commissions_senators(code = NULL, ascii = TRUE)
```

## Arguments

<code>code</code>	character. Character code (abbreviation) of the commission requested. A list of these may be obtained with <code>sen_commissions()</code> , although not all of the abbreviations in this data frame will return information.
<code>ascii</code>	logical. If TRUE, certain strings are converted to ascii format.

## Details

Returns a data frame with the following variables:

- `commission`: name of the commission.
- `commission_abbr`: abbreviated name of the commission.
- `senator_id`: unique code for each senator.
- `senator_name`
- `senator_party`
- `senator_state`

## Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

## Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

## Examples

```
# get info on the senators who serve on the CCJ commission:  
ccj <- sen_commissions_senators(code = "CCJ")
```

---

sen\_commissions\_type *Information on commissions in the Federal Senate, by commission type*

---

### Description

Information on commissions in the Federal Senate, by commission type.

### Usage

```
sen_commissions_type(type = c("permanent", "cpi", "temporary"), ascii = TRUE)
```

### Arguments

`type` character. Options are permanent, cpi and temporary. See details.  
`ascii` logical. If TRUE, certain strings are converted to ascii format.

### Details

Returns a data frame with the following variables:

- `commission_id`: unique code for each commission.
- `commission_name`: name of the commission.
- `commission_type`: permanent commission, temporary, or a parliamentary inquiry *Comissão Parlamentar de Inquérito*.
- `commission_house`: Legislative house where the commission is based.
- `active`: Whether the commission is active or not.

### Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

### Examples

```
cpi <- sen_commissions_type(type = "permanent")
```

---

`sen_commission_positions`

*Information on positions (jobs) that legislators may occupy in commissions in the Federal Senate*

---

## Description

Information on positions (jobs) that legislators may occupy in commissions in the Federal Senate.

## Usage

```
sen_commission_positions(active = c("Both", "Yes", "No"), ascii = TRUE)
```

## Arguments

<code>active</code>	character. If "No", returns all positions, otherwise returns positions that are presently active.
<code>ascii</code>	logical. If TRUE, certain strings are converted to ascii format.

## Details

Returns a data frame with the following variables:

- `comm_position_id`: unique code for each position.
- `comm_position`: name of the position.
- `comm_position_active`: whether the position is currently active or not.

## Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

## Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

## Examples

```
jobs <- sen_commission_positions(active = "No")
```

---

sen_nominal_votes	<i>Nominal votes in the Brazilian Federal Senate</i>
-------------------	--

---

**Description**

This is a dataset of the nominal votes in the Brazilian Federal Senate, from all those available on the API from 1991 onwards.

**Usage**

```
sen_nominal_votes
```

**Format**

A data frame with 60691 rows and 8 variables

**Value**

- `vote_date`: POSIXct, date the vote took place.
- `bill_id`: id of the bill in the Senate API database.
- `bill`: bill type, year and number.
- `legislature`: legislature number.
- `senator_id`: unique id of the senator.
- `senator_name`: the senator's name.
- `senator_vote`: numeric vote cast. 1 = "yes"; 0 = "no", NA = other.
- `senator_party`: political party the senator was in when the vote took place.
- `senator_state`: state the senator represented when the vote took place.

**Note**

These data can easily be grouped by legislature if so desired, using the legislature variable.

---

sen_parties	<i>Downloads and tidies information on the political parties in the Federal Senate</i>
-------------	--

---

**Description**

Downloads and tidies information on the political parties in the Federal Senate.

**Usage**

```
sen_parties(ascii = TRUE)
```

**Arguments**

`ascii` logical. TRUE by default, removes latin-1 characters from returned object.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
party <- sen_parties()
```

---

<code>sen_plenary_agenda</code>	<i>Returns results from the plenary in the Federal Senate for a specified date</i>
---------------------------------	--

---

**Description**

Returns results from the plenary in the Federal Senate for a specified date.

**Usage**

```
sen_plenary_agenda(period = c("month", "day"), date = NULL, ascii = TRUE)
```

**Arguments**

`period` character. If "month" is selected, all information available from the date specified with `date` to the end of the month is returned. Otherwise, information returned from the day of `date` only is returned.

`date` character. Format YYYYMMDD.

`ascii` logical. If TRUE, the default, strips Latin characters from the results.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
# get info from the second half of March 2014:
sessions <- sen_plenary_agenda(period = "month", date = "20140315")
# or from the first of April 2016:
sessions <- sen_plenary_agenda(period = "day", date = "20160401")
```

---

sen\_plenary\_leaderships

*Returns information on leaderships in the Federal Senate*

---

**Description**

Returns information on leaderships in the Federal Senate.

**Usage**

```
sen_plenary_leaderships(parties = TRUE, ascii = TRUE)
```

**Arguments**

**parties** logical. Returns information on party leadership in the Senate when TRUE, otherwise returns information on coalitions.

**ascii** logical. If TRUE, the default, strips Latin characters from the results.

**Value**

A tibble, of classes tbl\_df, tbl and data.frame.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
party_leaders <- sen_plenary_leaderships()
```

---

sen\_plenary\_result

*Returns results from the plenary in the Federal Senate for a specified date*

---

**Description**

Returns results from the plenary in the Federal Senate for a specified date.

**Usage**

```
sen_plenary_result(date = NULL, ascii = TRUE)
```

**Arguments**

**date** character. Format YYYYMMDD.

**ascii** logical. If TRUE, the default, strips Latin characters from the results.



**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
# get info for the 3rd of March 2016:  
x <- sen_plenary_result(date = "20160303")  
# Earlier periods may not have information:  
  
sen_plenary_result(date = "20110405")
```

---

`sen_plenary_sessions` *Returns the types of sessions in the Federal Senate*

---

**Description**

Returns the types of sessions in the Federal Senate.

**Usage**

```
sen_plenary_sessions(active = c("Yes", "No"), ascii = TRUE)
```

**Arguments**

<code>active</code>	. Default "Yes", which returns session types which are still used. Otherwise, returns all types of sessions.
<code>ascii</code>	logical. If TRUE, the default, strips Latin characters from the results.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sessions <- sen_plenary_sessions()
```

---

sen_senator	<i>Downloads and tidies personal information on senators from the Federal Senate</i>
-------------	--

---

### Description

Downloads and tidies personal information on senators from the Federal Senate.

### Usage

```
sen_senator(  
  id = NULL,  
  affiliations = TRUE,  
  mandates = TRUE,  
  absences = FALSE,  
  ascii = TRUE  
)
```

### Arguments

id	integer. Unique id for a senator. A dataframe of these is available from <code>sen_senator_list()</code> .
affiliations	logical. If TRUE, the default, returns information on party affiliation.
mandates	logical. If TRUE, the default, returns information on terms served by the senator.
absences	logical. If TRUE, returns information on leaves of absence taken by the senator.
ascii	logical. If TRUE, certain strings are converted to ascii format.

### Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

### Note

Setting `affiliations`, `mandates` and particularly `absences` to TRUE will result in a rather bloated data frame.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

## Examples

```
benedito <- sen_senator(id = 3823)
aecio <- sen_senator(id = 391, absences = TRUE)
juc <- sen_senator(73)
```

---

sen_senator_bills	<i>Downloads and tidies information on bills that certain senators have sponsored/authored in the Federal Senate</i>
-------------------	--

---

## Description

Downloads and tidies information on bills that certain senators have sponsored/authored in the Federal Senate.

## Usage

```
sen_senator_bills(id = 0, ascii = TRUE)
```

## Arguments

id	integer. This number represents the id of the senator you wish to get information on. These ids can be extracted from the API using the <code>sen_senator_list()</code> function, where they will appear as the first column in the data frame returned, under the name 'id'.
ascii	logical. If TRUE, certain strings are converted to ascii format.

## Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

## Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire. `Ataides <- sen_senator_bills(id = 5164)`

## See Also

`sen_senator_list()`

---

`sen_senator_commissions`

*Downloads and tidies information on the commissions on which senators have served or are serving in the Federal Senate*

---

**Description**

Downloads and tidies information on the commissions on which senators have served or are serving in the Federal Senate.

**Usage**

```
sen_senator_commissions(id = 0, ascii = TRUE)
```

**Arguments**

`id` integer. This number represents the id of the senator you wish to get information on. These ids can be extracted from the API using the `sen_senator_list()` function, where they will appear as the first column in the data frame returned, under the name 'id'.

`ascii` logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire. `Armando <- sen_senator_commissions(id = 715)`

**See Also**

```
sen_senator_list()
```

---

`sen_senator_details`

*Downloads and tidies personal information on the senators in the Federal Senate*

---

**Description**

Downloads and tidies personal information on the senators in the Federal Senate.

**Usage**

```
sen_senator_details(id = 0, ascii = TRUE)
```

**Arguments**

`id` integer. This number represents the id of the senator you wish to get information on. These ids can be extracted from the API using the `sen_senator_list()` function, where they will appear as the first column in the data frame returned, under the name 'id'.

`ascii` logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire. `Acir_G <- sen_senator_details(id = 4981)`

**See Also**

`sen_senator_list()`

---

<code>sen_senator_legis</code>	<i>Downloads and tidies information on the senators in the Federal Senate</i>
--------------------------------	---

---

**Description**

Downloads and tidies information on the senators in the Federal Senate.

**Usage**

```
sen_senator_legis(
  start = NULL,
  end = NULL,
  state = NULL,
  status = NULL,
  ascii = TRUE
)
```

**Arguments**

`start` two-digit integer representing the first legislature of the time period requested.

`end` two-digit integer representing the final legislature of the time period requested.

`state` character. Two-letter abbreviation of Brazilian state. A list of these is available with the function `UF()`.

`status` character, either "T" or "S", representing *titular* or *suplente* (stand-in senator), respectively.

`ascii` logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
all <- sen_senator_legis(start = 50)
```

---

sen_senator_list	<i>Downloads and tidies information on the senators in the Federal Senate</i>
------------------	---

---

**Description**

Downloads and tidies information on the senators in the Federal Senate.

**Usage**

```
sen_senator_list(present = TRUE, state = NULL, status = NULL, ascii = TRUE)
```

**Arguments**

<code>present</code>	logical. If TRUE, downloads data on the legislature currently sitting in the Federal Senate, otherwise returns information on senators who are currently absent.
<code>state</code>	character. Two-letter abbreviation of Brazilian state. A list of these is available with the function <code>UF()</code> .
<code>status</code>	character, either "T" or "S", representing <i>titular</i> or <i>suplente</i> (stand-in senator), respectively.
<code>ascii</code>	logical. If TRUE, strips Latin characters from strings.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
all <- sen_senator_list()

# Who represents Rio de Janeiro?

rj <- sen_senator_list(state = "RJ")
```

---

sen\_senator\_mandates *Downloads and tidies information on senators' mandates in the Federal Senate*

---

**Description**

Downloads and tidies information on senators' mandates in the Federal Senate.

**Usage**

```
sen_senator_mandates(id = 0, ascii = TRUE)
```

**Arguments**

id	integer. This number represents the id of the senator you wish to get information on. These ids can be extracted from the API using the <code>sen_senator_list()</code> function, where they will appear as the first column in the data frame returned, under the name 'id'.
ascii	logical. If TRUE, certain strings are converted to ascii format.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**See Also**

```
sen_senator_list()
```

**Examples**

```
terms <- sen_senator_mandates(id = 4763)
terms <- sen_senator_mandates(id = 3398)
```

---

sen\_senator\_suplentes *Downloads and tidies information on titular senators and their suplentes in the Federal Senate*

---

### Description

Downloads and tidies information on titular senators and their *suplentes* in the Federal Senate.

### Usage

```
sen_senator_suplentes(id = 0, ascii = TRUE)
```

### Arguments

**id** integer. This number represents the id of the senator you wish to get information on. These ids can be extracted from the API using the `sen_senator_list()` function, where they will appear as the first column in the data frame returned, under the name 'id'.

**ascii** logical. If TRUE, certain strings are converted to ascii format.

### Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

### See Also

```
sen_senator_list()
```

### Examples

```
# A titular senator, José Serra:
Serra <- sen_senator_suplentes(id = 90)

# Or one of his suplentes:
suplente <- sen_senator_suplentes(id = 878)
```



---

sen_senator_votes	<i>Downloads and tidies information on senators' votes in the Federal Senate</i>
-------------------	--

---

### Description

Downloads and tidies information on senators' votes in the Federal Senate.

### Usage

```
sen_senator_votes(id = 0, ascii = TRUE)
```

### Arguments

<code>id</code>	integer. This number represents the id of the senator you wish to get information on. These ids can be extracted from the API using the <code>sen_senator_list()</code> function, where they will appear as the first column in the data frame returned, under the name 'id'.
<code>ascii</code>	logical. If TRUE, certain strings are converted to ascii format.

### Value

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

### See Also

```
sen_senator_list()
```

### Examples

```
ant <- sen_senator_votes(id = 5529)

# some have never voted, as they are suplentes:
sen_senator_votes(898)
```

---

sen_sponsor_types	<i>Downloads and tidies data on the types of bill sponsors in the Federal Senate</i>
-------------------	--

---

**Description**

Downloads and tidies data on the types of bill sponsors in the Federal Senate.

**Usage**

```
sen_sponsor_types(ascii = TRUE)
```

**Arguments**

`ascii` (logical). If TRUE, bill sponsor names are converted to ascii format, stripping the latin characters from the names.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`, with variables:

- `sponsor_abbr`: Abbreviation of sponsor type name.
- `sponsor_name`: Sponsor type name.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
types <- sen_sponsor_types()
```

---

sen_statement_list	<i>Downloads and tidies information on the types of declarations senators can make in the Federal Senate</i>
--------------------	--

---

**Description**

Downloads and tidies information on the types of declarations senators can make in the Federal Senate.

**Usage**

```
sen_statement_list(ascii = TRUE, print = TRUE)
```

**Arguments**

`ascii` logical. TRUE by default, removes latin-1 characters from returned object.  
`print` logical. If TRUE (the default), prints the dataframe to the console.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
st <- sen_statement_list()
```

---

<code>sen_votes</code>	<i>Returns voting information from the Senate floor for the date requested</i>
------------------------	--

---

**Description**

Returns voting information from the Senate floor for the date requested.

**Usage**

```
sen_votes(date = NULL, end_date = NULL, binary = TRUE, ascii = TRUE)
```

**Arguments**

`date` character or integer. Format YYYYMMDD.  
`end_date` character or integer. Format YYYYMMDD. If `end_date` is supplied, information on all votes between the initial date (`date`) and `end_date` is returned. See the notes below.  
`binary` logical. If TRUE, the default, transforms votes into 1 for "yes", 0, for "no" and NA for everything else. If FALSE, returns a character vector of vote decisions.  
`ascii` logical. If TRUE, the default, strips Latin characters from the results.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Note**

The maximum period allowed by the API is currently 60 days. If the period requested is larger than this, an error message is returned. For a full set of votes for each legislature, see `data("senate_nominal_votes")`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
## Not run:
sen_votes(date = "20130516")

# Some votes are secret:
ssshhh <- sen_votes("20160301", binary = FALSE)

# All votes between two periods (might take a little while):
longer <- sen_votes(date = "20160301", end_date = "20160415")

## End(Not run)
```

---

sen_votes_year	<i>Returns voting information from the Senate floor for the year requested using standard method</i>
----------------	--

---

**Description**

Returns voting information from the Senate floor for the year requested.

**Usage**

```
sen_votes_year(year, ascii = T, binary = T)
```

**Arguments**

year	character or integer. Format YYYY
ascii	logical. If TRUE, the default, strips Latin characters from the results.
binary	logical. If TRUE, the default, transforms votes into 1 for "yes", 0, for "no" and NA for everything else. If FALSE, returns a character vector of vote decisions.

**Value**

A tibble, of classes `tbl_df`, `tbl` and `data.frame`.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
sen_votes_year("2013")
```

---

sen_votes_year_file	Returns voting information from the Senate floor for the year requested using xml file method
---------------------	---

---

### Description

Returns voting information from the Senate floor for the year requested (Only available from 1991 through 2017).

### Usage

```
sen_votes_year_file(year = NULL, binary = TRUE, ascii = TRUE)
```

### Arguments

year	character or integer. Format YYYY
binary	logical. If TRUE, the default, transforms votes into 1 for "yes", 0, for "no" and NA for everything else. If FALSE, returns a character vector of vote decisions.
ascii	logical. If TRUE, the default, strips Latin characters from the results.

### Value

A tibble, of classes tbl\_df, tbl and data.frame.

### Author(s)

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

### Examples

```
sen_votes_year_file("2013")
```

---

statesBR	A dataframe of Brazilian states, by their sigla (acronym) and full name.
----------	--

---

### Description

- UF character. Acronyms of Brazilian states.
- Name character. Full names of Brazilian states.

**Usage**

```
statesBR
```

**Format**

A data frame with 27 rows and 2 variables

---

UF	<i>A list of state abbreviations for use with the functions of congressbr.</i>
----	--

---

**Description**

This function prints out a character vector of Brazilian state abbreviations to the console.

**Usage**

```
UF()
```

**Note**

This is most useful if you already have an idea of what state you want and its abbreviation. If not, to see the full dataset, use `data("statesBR")`, which has the abbreviation, "UF" (*Unidade Federal*) and the full name of the states.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
UF()
```

---

vote_to_rollcall	<i>Create rollcall or matrix format data structures from voting records</i>
------------------	---

---

**Description**

Returns votes from the Senate or Chamber of Deputies in a format that can be easily used with other R packages (wnominate, pscl, MCMCpack) or with other modelling languages such as JAGS.

**Usage**

```
vote_to_rollcall(votes = NULL, legislators = NULL, bills = NULL, ideal = TRUE)
```

**Arguments**

votes	integer. The recorded nominal votes. Must be in the format 1, 0, where 1 indicates a 'Yes' vote, 0 a 'No' vote. Missing values must be NA. All of the votes, legislators and bills parameters must be vectors.
legislators	This may be a vector of legislator names or id numbers.
bills	A vector of bill numbers or ids.
ideal	If TRUE, the default, returns a rollcall object for use with pscl and/or wnominate. If FALSE, returns a matrix suitable for use with MCMCpack or the JAGS modelling language.

**Author(s)**

Robert Myles McDonnell, Guilherme Jardim Duarte & Danilo Freire.

**Examples**

```
# get votes:
data(sen_nominal_votes)
votes <- sen_nominal_votes %>%
  dplyr::filter(vote_date >= "2005-03-01", vote_date <= "2007-12-15")
rc <- vote_to_rollcall(votes$senator_vote,
                      legislators = votes$senator_name,
                      bills = votes$bill_id)
```

# Index

## \*Topic **datasets**

- commissions, [9](#)
- sen\_nominal\_votes, [30](#)
- statesBR, [45](#)

cham\_bill\_info, [4](#)  
cham\_bill\_info\_id, [5](#)  
cham\_bills, [3](#)  
cham\_legislator\_list, [6](#)  
cham\_plenary\_bills, [6](#)  
cham\_typeauthors\_bills, [7](#)  
cham\_votes, [8](#)  
cham\_votes\_year, [9](#)  
commissions, [9](#)  
congressbr, [10](#)

sen\_agenda, [10](#)  
sen\_bill\_search, [21](#)  
sen\_bill\_sponsors, [23](#)  
sen\_bills, [11](#)  
sen\_bills\_current, [12](#)  
sen\_bills\_limits, [13](#)  
sen\_bills\_list, [14](#)  
sen\_bills\_locations, [14](#)  
sen\_bills\_passage, [15](#)  
sen\_bills\_passing, [16](#)  
sen\_bills\_situations, [17](#)  
sen\_bills\_status, [17](#)  
sen\_bills\_topics, [18](#)  
sen\_bills\_types, [19](#)  
sen\_bills\_update\_types, [20](#)  
sen\_bills\_updates, [19](#)  
sen\_budget, [24](#)  
sen\_coalition\_info, [25](#)  
sen\_coalitions, [24](#)  
sen\_commission\_positions, [29](#)  
sen\_commissions, [26](#)  
sen\_commissions\_senators, [27](#)  
sen\_commissions\_type, [28](#)  
sen\_nominal\_votes, [30](#)

sen\_parties, [30](#)  
sen\_plenary\_agenda, [31](#)  
sen\_plenary\_leaderships, [32](#)  
sen\_plenary\_result, [32](#)  
sen\_plenary\_sessions, [33](#)  
sen\_senator, [34](#)  
sen\_senator\_bills, [35](#)  
sen\_senator\_commissions, [36](#)  
sen\_senator\_details, [36](#)  
sen\_senator\_legis, [37](#)  
sen\_senator\_list, [38](#)  
sen\_senator\_mandates, [39](#)  
sen\_senator\_suplentes, [40](#)  
sen\_senator\_votes, [41](#)  
sen\_sponsor\_types, [42](#)  
sen\_statement\_list, [42](#)  
sen\_votes, [43](#)  
sen\_votes\_year, [44](#)  
sen\_votes\_year\_file, [45](#)  
statesBR, [45](#)

UF, [46](#)

vote\_to\_rollcall, [46](#)