Package ‘mnis’

July 3, 2017

Type Package
Title Easy Downloading Capabilities for the Members' Name Information Service
Version 0.2.7
Date 2017-07-03
Author Evan Odell [aut, cre]
Maintainer Evan Odell <evanodell91@gmail.com>
Description An API package for the Members' Name Information Service operated by the UK parliament. Documentation for the API itself can be found here: <http://data.parliament.uk/membersdataplayer/default.aspx>.
URL http://docs.evanodell.com/mnis
BugReports https://github.com/EvanOdell/mnis/issues
License MIT + file LICENSE
LazyData TRUE
LinkingTo Rcpp
Imports httr, jsonlite, utils, dplyr, tibble, stringi, Rcpp (>= 0.12.10)
RoxygenNote 6.0.1
Suggests knitr, rmarkdown, testthat
VignetteBuilder knitr
NeedsCompilation yes
Repository CRAN
Date/Publication 2017-07-03 17:02:18 UTC

R topics documented:

constituency_results_tidy .................................................. 2
date_tidy ........................................................................... 3
mnis ................................................................................. 3
constituency_results_tidy

constituency_results_tidy

Description

constituency_results_tidy

Usage

constituency_results_tidy(results, details)

Arguments

  results The tibble to tidy
  details The list to tidy
**date_tidy**

A function that makes date variables returned from the API datable, ie by converting them to POSIXct. Does the same thing for datetimes.

**Description**

A function that makes date variables returned from the API datable, ie by converting them to POSIXct. Does the same thing for datetimes.

**Usage**

date_tidy(df)

**Arguments**

df The tibble with the undateable dates.

**mnis**

*mnis: Easy Downloading Capabilities for the Member Name Information Service*

**Description**

An API package for the Members’ Name Information Service operated by the UK parliament. The package is intended to simplify pulling data from an API for users unfamiliar with APIs. Documentation for the API itself can be found here: [http://data.parliament.uk/membersdatapackage/default.aspx](http://data.parliament.uk/membersdatapackage/default.aspx).

**Details**

The package includes a built in function to remove a byte-order mark from the API data, and a parameter tidy with each function that converts variable names into an R friendly style, removing non-alphanumeric characters and converting to snake_case when equal to TRUE, its default value.

All functions requests data in JSON format and parse it to a tibble. The exception is mnis_constituency_results which returns a single object containing a list (with constituency details) and a tibble (with election results).

None of the functions included in this package appear to run into the API rate limit, although there may be restrictions to custom requests, which allow a maximum of three parameters.
mnis functions

- `mnis_additional`
- `mnis_all_members`
- `mnis_base`
- `mnis_constituency_results`
- `mnis_department`
- `mnis_eligible`
- `mnis_extra`
- `mnis_full_biog`
- `mnis_general_election_results`
- `mnis_joined_between`
- `mnis_lords_type`
- `mnis_member_date`
- `mnis_party_state`
- `mnis_reference`

---

**mnis_additional**  
A series of basic function for the API lookup. Each function accepts a member’s ID and returns information; if no ID is given basic information on all members of both houses is returned.

---

**Description**

A series of basic function for the API lookup. Each function accepts a member’s ID and returns information; if no ID is given basic information on all members of both houses is returned.

**Usage**

```python
mnis_additional()

mnis_basic_details(ID = NULL, ref_dods = FALSE, tidy = TRUE,  
tidy_style = "snake_case")

mnis.biography_entries(ID = NULL, ref_dods = FALSE, tidy = TRUE,  
tidy_style = "snake_case")

mnis_committees(ID = NULL, ref_dods = FALSE, tidy = TRUE,  
tidy_style = "snake_case")

mnis.addresses(ID = NULL, ref_dods = FALSE, tidy = TRUE,  
tidy_style = "snake_case")
```
mnis_constituencies(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_elections_contested(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_experiences(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_government_posts(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_honours(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_house_memberships(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_statuses(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_staff(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_interests(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_known_as(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_maiden_speeches(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_opposition_posts(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_other_parliaments(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_parliamentary_posts(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_parties(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

mnis_preferred_names(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")
mnis_all_members

Arguments

ID The member ID value. If empty, function calls `mnis_all_members` and returns basic information on all members of both houses.

ref_dods Request based on the DODS membership ID scheme. Defaults to FALSE, where it requests data based on the default membership ID scheme.

tidy If TRUE, fixes the variable names in the tibble to remove non-alphanumeric characters and superfluous text, and convert to a consistent style. Defaults to TRUE.

tidy_style The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".

Value

A tibble with the data corresponding to the particular function called.

See Also

`mnis_full_biog` `mnis_extra`

Examples

```r
## Not run:

x <- mnis_basic_details(house = "all")

## End(Not run)

## Not run:

x <- mnis_additional()

## End(Not run)
```

---

**mnis_all_members**

Returns a tibble with information on all members of both houses or a given house.

Description

Returns a tibble with information on all members of both houses or a given house.

Usage

```r
mnis_all_members(house = "all", party = NULL, tidy = TRUE, tidy_style = "snake_case")
```
Arguments

house  The house to which the member belongs. Accepts one of 'all', 'lords' and 'commons', defaults to 'all'. This parameter is not case sensitive, so using 'commons', 'Commons' or 'cOmMoNS' will result in the same data being returned.

party The party to which a member belongs. Defaults to NULL, in which case all members are returned, subject to other parameters. The party names are not case sensitive, but must be the complete string of the party name, searching and wildcard options are not accepted by the API, e.g. 'green party'.

tidy  Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.

tidy_style  The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".

Value

A tibble with information on all members of the House of Commons and/or the House of Lords that meet the criteria included in the function parameters.

Examples

```r
## Not run:
x <- mnis_all_members(house = 'all', party = NULL, tidy = TRUE, tidy_style="snake_case")
## End(Not run)
```

## mnis_all_reference

Returns a list of 39 tibbles, with the name and data returned from each respective reference function.

Description

Returns a list of 39 tibbles, with the name and data returned from each respective reference function.

Usage

```r
mnis_all_reference(tidy = TRUE, tidy_style = "snake_case")
```

Arguments

- **tidy** Fix the variable names in each tibble to remove special characters and superfluous text, and convert all the variable names to snake_case. Defaults to TRUE.

- **tidy_style** The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".
Value

Returns a list of 39 tibbles, with the name and data returned from each respective reference function.

See Also

`mnis_reference`

Examples

```r
## Not run:

z <- mnis_all_reference()

## End(Not run)
```

---

**mnis_base**

A generic function for the MNIS API lookup. The function requests data in JSON format, but the type of object, and all URLs, paths and parameters are user-defined. `mnis_base` does not include the option to tidy variable names and data types, but the `mnis_tidy` function can be called on the data returned from the API.

---

**Description**

A generic function for the MNIS API lookup. The function requests data in JSON format, but the type of object, and all URLs, paths and parameters are user-defined. `mnis_base` does not include the option to tidy variable names and data types, but the `mnis_tidy` function can be called on the data returned from the API.

**Usage**

`mnis_base(request)`

**Arguments**

- **request** The request query being made to the MNIS URL

**Examples**

```r
## Not run:

x <- mnis_base('House=Commons|IsEligible=true/')

## End(Not run)
```
**mnis_bom**

**Strip out BOM from JSON data**

**Description**

Strip out BOM from JSON data

**Usage**

```r
mnis_bom(x)
```

**Arguments**

- **x**
  
  The GET return to strip BOM out of

---

**mnis_constituency_results**

*Returns a single object containing a list with details of the constituency and a tibble with election results. For constituency IDs, see ref_constituencies.*

**Description**

Returns a single object containing a list with details of the constituency and a tibble with election results. For constituency IDs, see `ref_constituencies`.

**Usage**

```r
mnis_constituency_results(constituency_id = NULL, election_id = 0, tidy = TRUE, tidy_style = "snake_case")
```

**Arguments**

- **constituency_id**
  
  The ID of the constituency to return the data for. This parameter cannot be empty.

- **election_id**
  
  The ID of the election to return the data for. Defaults to 0, which calls the most recent result, either the result of the last general election, or the result of the last byelection held since that election.

- **tidy**
  
  If TRUE, fixes the variable names in the tibble to remove non-alphanumeric characters and superfluous text, and convert to a consistent style. Defaults to TRUE.

- **tidy_style**
  
  The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period_case". Defaults to "snake_case".
Value

A list with details of the constituency, labelled 'details' and a tibble with election results, labelled 'results'. The list and tibble are stored in a single object.

See Also

mnis_reference

Examples

## Not run:

```r
x <- mnis_constituency_results(constituency_id = 3709, election_id = 0)
```

## End(Not run)

---

**mnis_department**

Request data on the holders of cabinet/shadow cabinet positions. Request specific departments by department ID (see **mnis_reference** for the ref_department function to retrieve departmental IDs).

Description

Request data on the holders of cabinet/shadow cabinet positions. Request specific departments by department ID (see **mnis_reference** for the ref_department function to retrieve departmental IDs).

Usage

```r
mnis_department(department_id = 0, bench = "Government", former = TRUE, tidy = TRUE, tidy_style = "snake_case")
```

Arguments

department_id  The department look up. 0 returns the cabinet/shadow cabinet, -1 returns a list of all ministers. Defaults to 0.

bench  Flag to return either Government or Opposition information. Defaults to 'Government'. The API is case sensitive on this parameter, so 'Government' or 'Opposition' will work, but 'government' and 'opposition' will not.

former  Flag to include both current and former ministers/shadow ministers. Defaults to TRUE. If FALSE, only includes current ministers/shadow ministers.

tidy  If TRUE, fixes the variable names in the tibble to remove non-alphanumeric characters and superfluous text, and convert to a consistent style. Defaults to TRUE.

tidy_style  The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period_case". Defaults to "snake_case".
mnis_eligible

Value

A tibble with information on departments and ministers/shadow ministers.

See Also

mnis_reference

Examples

```r
## Not run:
x <- mnis_department(department_id = 0, bench = 'Government', former=TRUE)

## End(Not run)
```

Description

Returns all members who are able to sit in either house, or who are currently ineligible to sit. Members ineligible to sit include but are not necessarily limited to former MPs, members of the judiciary, who are recused from House of Lords duties.

Usage

```r
mnis_eligible(eligible = TRUE, house = "all", party = NULL, tidy = TRUE,
tidy_style = "snake_case")
```

Arguments

- **eligible**: If the member is currently eligible to sit. Accepts TRUE or FALSE. Defaults to TRUE.
- **house**: The house to which the member belongs. Accepts one of `all`, `lords` and `commons`, defaults to `all`. This parameter is not case sensitive, so `commons`, `Commons` and `cOmMONS` will all return the same data.
- **party**: The party to which a member belongs. Defaults to NULL. The party must be fully spelled out (e.g. 'green party'), the API does not accept searches on this parameter. For a tibble of parties, see `ref_parties`. This parameter is not case sensitive.
- **tidy**: Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.
tidy_style  The style to convert variable names to, if tidy = TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case"

Examples

## Not run:
x <- mnis_eligible(eligible=FALSE, house='all', party='labour')
x <- mnis_eligible(eligible=TRUE, house='all', party='green party')
x <- mnis_eligible(house="commons")

## End(Not run)

---

**mnis_extra**  A wrapper for **mnis_additional** functions.

**Description**

It combines the various options of mnis_additional into one dataframe, and the default is similar to **mnis_full_biog**. Variable descriptions are taken from the mnis website: <http://data.parliament.uk/membersdataplatform/memberquery.aspx>.

**Usage**

```r
mnis_extra(ID, ref_dods = FALSE, addresses = TRUE,
           biography_entries = TRUE, committees = TRUE, constituencies = TRUE,
           elections_contested = TRUE, experiences = TRUE, government_posts = TRUE,
           honours = TRUE, house_memberships = TRUE, interests = TRUE,
           known_as = TRUE, maiden_speeches = TRUE, opposition_posts = TRUE,
           other_parliaments = TRUE, parliamentary_posts = TRUE, parties = TRUE,
           preferred_names = TRUE, staff = TRUE, statuses = TRUE, tidy = TRUE,
           tidy_style = "snake_case")
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>The ID number of the member. Defaults to NULL. If NULL, returns</td>
</tr>
<tr>
<td>ref_dods</td>
<td>Request based on the DODS membership ID scheme. Defaults to FALSE. If FALSE, requests data based on the default membership ID scheme.</td>
</tr>
<tr>
<td>addresses</td>
<td>Member address information (e.g. website, twitter, constituency address etc...). Defaults to TRUE. If TRUE, address details are included in the tibble.</td>
</tr>
<tr>
<td>biography_entries</td>
<td>Member biographical information (e.g. countries of interest, policy expertise etc...) Defaults to TRUE. If TRUE, biographical details are included in the tibble.</td>
</tr>
<tr>
<td>committees</td>
<td>Committees a Member sits or has sat on as well details on committee chairing. Defaults to TRUE. If TRUE, committee details are included in the tibble.</td>
</tr>
</tbody>
</table>
constituencies constituencies a Member has represented. Defaults to TRUE. If TRUE, constituency details are included in the tibble.

elections_contested
Elections a Member has contested but not won. Defaults to TRUE. If TRUE, details of unsuccessful election contests are included in the tibble.

experiences Non-parliamentary experience of a Member. Defaults to TRUE. If TRUE, extra-parliamentary experience details are included in the tibble.

government_posts Government posts a Member currently holds. Defaults to TRUE. If TRUE, government posts details are included in the tibble.

honours Honours (e.g. MBE, OBE etc...) held by a Member. Defaults to TRUE. If TRUE, honours details are included in the tibble.

house_memberships House membership list of a Member. Defaults to TRUE. If TRUE, house membership details are included in the tibble.

interests Registered interests (financial) of a Member. Defaults to TRUE. If TRUE, interest details are included in the tibble.

known_as Details of names a Member has chosen to be known as instead of their full title (House of Lords members only). Defaults to TRUE. If TRUE, known as details are included in the tibble.

maiden_speeches Maiden speech dates for a Member. Defaults to TRUE. If TRUE, maiden speech details are included in the tibble.

opposition_posts Opposition posts a Member has held. Defaults to TRUE. If TRUE, opposition post details are included in the tibble.

other_parliaments Other Parliaments that a Member has held a membership of. Defaults to TRUE. If TRUE, details of other parliaments are included in the tibble.

parliamentary_posts Parliamentary posts a Member has held. Defaults to TRUE. If TRUE, parliamentary posts details are included in the tibble.

parties Party affiliations of a Member. Defaults to TRUE. If TRUE, address details are included in the tibble.

preferred_names Full set of data about a Members’ name (e.g. surname, forename, Honorary prefixes, full details of HoL title and rank etc...). Defaults to TRUE. If TRUE, preferred names details are included in the tibble.

staff The staff employed by a Member. Defaults to TRUE. If TRUE, staff details are included in the tibble.

statuses Status history (e.g. suspensions and disqualifications) for a Member. Defaults to TRUE. If TRUE, status details are included in the tibble.

tidy Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.

tidy_style The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period_case". Defaults to "snake_case"
Value

A tibble with the requested data on a given MP.

See Also

mnis_full_biog mnis_basic_details mnis_additional

Examples

## Not run:

x <- mnis_extra(172)

## End(Not run)

mnis_full_biog  

Requests all available biographical information for a given member, and returns it in the form of a tibble.

Description

Requests all available biographical information for a given member, and returns it in the form of a tibble.

Usage

mnis_full_biog(ID = NULL, ref_dods = FALSE, tidy = TRUE, tidy_style = "snake_case")

Arguments

ID

The ID number of the member, using the default MNIS scheme. If ref_dods is TRUE, accepts the Dods monitoring scheme instead. If left empty, returns the same data as mnis_all_members with default parameters.

ref_dods

Request based on the Dods monitoring member ID scheme. Defaults to FALSE. If FALSE, requests using the default MNIS identification scheme.

tidy

Fix the variable names in the tibble to remove non-alphanumeric characters and superfluous text, and convert variable names to a consistent style. Defaults to TRUE.

tidy_style

The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period_case". Defaults to "snake_case".

See Also

mnis_basic_details mnis_additional mnis_extra
mnis_general_election_results

Examples

```r
## Not run:
x <- mnis_full_biog(172)

## End(Not run)
```

**mnis_general_election_results**

*Returns an object containing list with details of the search parameter and a tibble with election results. Accepts queries on location type and name, and the start and end date to return general elections between. The API does not contain data for Norther Ireland.*

**Description**

Returns an object containing list with details of the search parameter and a tibble with election results. Accepts queries on location type and name, and the start and end date to return general elections between. The API does not contain data for Norther Ireland.

**Usage**

```r
mnis_general_election_results(location_type = "Country",
   location_name = "Great Britain", start_date = "1900-01-01",
   end_date = Sys.Date(), tidy = TRUE, tidy_style = "snake_case")
```

**Arguments**

- **location_type** The type of area to return information for. Accepts 'Country', 'Region', 'County', and 'Constituency'. Defaults to 'Country'.
- **location_name** The location to return data for. It can be the name of any Country, Region, County or Constituency. Defaults to 'Great Britain'.
- **start_date** Start date of search. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXx, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. Defaults to '1900-01-01' if no date is selected.
- **end_date** End date of search. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXx, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. Defaults to current date if no date is selected.
- **tidy** Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.
- **tidy_style** The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to 'snake_case'.
mnis_joined_between

Value

Returns a list with details of the search parameter and a tibble with election results.

See Also

mnis_reference

Examples

```r
## Not run:
x <- mnis_general_election_results(location_type = 'Country', location_name = 'England',
  start_date = '2010-01-01', end_date = '2016-01-01')

## End(Not run)
```

---

### mnis.joined_between

Function returns all members who took their seats in the house between two given dates.

Description

Function returns all members who took their seats in the house between two given dates.

Usage

```r
mnis.joined_between(start_date = "1900-01-01", end_date = Sys.Date(),
  house = "all", party = NULL, eligible = "all", tidy = TRUE,
  tidy_style = "snake_case")
```

Arguments

- `start_date`: The start date of the search. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. Defaults to '1900-01-01'.
- `end_date`: The end date of the search. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. Defaults to the current date.
- `house`: The house to which the member belongs. Accepts one of 'all', 'lords' and 'commons', defaults to 'all'.
- `party`: The party to which a member belongs. Defaults to NULL.
- `eligible`: If the member is currently eligible to sit. Accepts one of 'all', 'current', 'former', defaults to 'all'.
- `tidy`: Fix the variable names in the tibble to remove non-alphanumeric characters and superfluous text, and convert variable names to a consistent style. Defaults to TRUE.
- `tidy_style`: The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".
mnis_lords_type

Examples

```r
## Not run:

x <- mnis_joined_between(start_date = '2015-01-01', end_date = '2017-01-01', party = 'labour')
```

## End(Not run)

---

**mnis_lords_type**  
*Calls the API to return a tibble with details on the number of Lords and their affiliations.*

**Description**

Calls the API to return a tibble with details on the number of Lords and their affiliations.

**Usage**

```r
mnis_lords_type(date = Sys.Date(), tidy = TRUE, tidy_style = "snake_case")
```

**Arguments**

- **date**: Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. The API will return data on the state of the House of Lords on that date. Defaults to the current system date.

- **tidy**: If TRUE, fixes the variable names in the tibble to remove non-alphanumeric characters and superfluous text, and convert to a consistent style. Defaults to TRUE.

- **tidy_style**: The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".

**Value**

A tibble with information on the numbers of different types of Lords on a given date.

**See Also**

- [mnis_reference](#)

**Examples**

```r
## Not run:

x <- mnis_lords_type()
```

## End(Not run)
mnis_member_date  

Returns a tibble with a member’s status on a given date.

Description

Returns a tibble with a member’s status on a given date.

Usage

```r
mnis_member_date(id = NULL, date = Sys.Date(), tidy = TRUE, tidy_style = "snake_case")
```

Arguments

- **id**  
The ID of the member. Currently this only accepts IDs from the default membership ID scheme. If empty, the function stops and no data is returned.

- **date**  
Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. Return details on the requested member’s status on that date. Defaults to the current system date.

- **tidy**  
Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.

- **tidy_style**  
The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case"

Value

Returns a tibble with the given member’s status on the given date.

See Also

- `mnis_mps_on_date`

Examples

```r
# Not run:

x <- mnis_member_date(172)

# End(Not run)
mnis_mps_on_date

A tibble with information on all MPs who were members of the House of Commons on the date specified (if only date1 is included as a parameter), or on or between the two dates if both date1 and date2 are specified.

Description

A tibble with information on all MPs who were members of the House of Commons on the date specified (if only date1 is included as a parameter), or on or between the two dates if both date1 and date2 are specified.

Usage

```
mnis_mps_on_date(date1 = Sys.Date(), date2 = NULL, tidy = TRUE, tidy_style = "snake_case")
```

Arguments

date1 The date to return the list of mps from. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with as.Date(). Defaults to current system date.

date2 An optional query parameter. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with as.Date(). If not NULL, the function returns a list of all MPs who were members between date2 and date1. Defaults to NULL.

tidy Fix the variable names in the tibble to remove extra characters, superfluous text and convert variable names to a consistent style. Defaults to TRUE.

tidy_style The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".

Value

A tibble with information on all MPs who were members of the House of Commons on the date specified (if only date1 is included as a parameter), or on or between the two dates if both date1 and date2 are specified.

Examples

```r
## Not run:

x <- mnis_mps_on_date()

## End(Not run)
```
mnis_party_state A tibble with information on the numbers and gender of MPs, by party, for the given date.

Description

A tibble with information on the numbers and gender of MPs, by party, for the given date.

Usage

mnis_party_state(house = "Commons", date = Sys.Date(), tidy = TRUE, tidy_style = "snake_case")

Arguments

- **house**
  - The house of parliament. Accepts either 'Lords' or 'Commons'. Defaults to 'Commons'.

- **date**
  - Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with as.Date(). Defaults to the current system date.

- **tidy**
  - Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.

- **tidy_style**
  - The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".

Value

A tibble with information on the numbers and gender of MPs, by party, by party, for the given date.

See Also

mnis_mps_on_date mnis_peers_on_date

Examples

```r
## Not run:

x <- mnis_party_state('2012-01-12')

## End(Not run)
```
**mnis_peers_on_date**

A tibble with information on all peers who were members of the House of Commons on the date specified (if only date1 is included as a parameter), or on or between the two dates if both date1 and date2 are specified.

**Description**

A tibble with information on all peers who were members of the House of Commons on the date specified (if only date1 is included as a parameter), or on or between the two dates if both date1 and date2 are specified.

**Usage**

```r
mnis_peers_on_date(date1 = Sys.Date(), date2 = NULL, tidy = TRUE, tidy_style = "snake_case")
```

**Arguments**

- **date1**
  
  The date to return the list of peers from. Defaults to current system date. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`.

- **date2**
  
  An optional query parameter. Accepts character values in "YYYY-MM-DD" format, and objects of class Date, POSIXt, POSIXct, POSIXlt or anything else than can be coerced to a date with `as.Date()`. If not NULL, the function returns a list of all peers in the House of Lords between date2 and date1. Defaults to NULL.

- **tidy**
  
  Fix the variable names in the tibble to remove extra characters, superfluous text and convert variable names to a consistent style. Defaults to TRUE.

- **tidy_style**
  
  The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period.case". Defaults to "snake_case".

**Value**

A tibble with information on all Peers who were members of the House of Lords on the date specified (if only date1 is included as a parameter), or on or between the two dates if both date1 and date2 are specified.

**See Also**

`mnis_party_state` `mnis_peers_on_date`
mnis_reference

A series of 39 functions that return tibbles of reference data. This data is useful for providing parameters for other function calls. The functions do not accept any arguments aside from the 'tidy' argument, which defaults to TRUE.

Description
A series of 39 functions that return tibbles of reference data. This data is useful for providing parameters for other function calls. The functions do not accept any arguments aside from the 'tidy' argument, which defaults to TRUE.

Usage

mnis_reference()

ref_address_types(tidy = TRUE, tidy_style = "snake_case")

ref_answering_bodies(tidy = TRUE, tidy_style = "snake_case")

ref_areas(tidy = TRUE, tidy_style = "snake_case")

ref_area_types(tidy = TRUE, tidy_style = "snake_case")

ref_biography_categories(tidy = TRUE, tidy_style = "snake_case")

ref_cabinets(tidy = TRUE, tidy_style = "snake_case")

ref_committees(tidy = TRUE, tidy_style = "snake_case")

ref_committee_types(tidy = TRUE, tidy_style = "snake_case")

ref_constituencies(tidy = TRUE, tidy_style = "snake_case")

ref_constituency_areas(tidy = TRUE, tidy_style = "snake_case")

ref_constituency_types(tidy = TRUE, tidy_style = "snake_case")

ref_countries(tidy = TRUE, tidy_style = "snake_case")
ref_departments(tid = TRUE, tidy_style = "snake_case")
ref_disqualification_types(tid = TRUE, tidy_style = "snake_case")
ref_elections(tid = TRUE, tidy_style = "snake_case")
ref_election_types(tid = TRUE, tidy_style = "snake_case")
ref_end_reasons(tid = TRUE, tidy_style = "snake_case")
ref_experience_types(tid = TRUE, tidy_style = "snake_case")
ref_government_post_departments(tid = TRUE, tidy_style = "snake_case")
ref_government_posts(tid = TRUE, tidy_style = "snake_case")
ref_government_ranks(tid = TRUE, tidy_style = "snake_case")
ref_honourary_prefixes(tid = TRUE, tidy_style = "snake_case")
ref_honour_lists(tid = TRUE, tidy_style = "snake_case")
ref_honours(tid = TRUE, tidy_style = "snake_case")
ref_interest_categories(tid = TRUE, tidy_style = "snake_case")
ref_lords_membership_types(tid = TRUE, tidy_style = "snake_case")
ref_lords_ranks(tid = TRUE, tidy_style = "snake_case")
ref_opposition_post_departments(tid = TRUE, tidy_style = "snake_case")
ref_opposition_posts(tid = TRUE, tidy_style = "snake_case")
ref_opposition_ranks(tid = TRUE, tidy_style = "snake_case")
ref_other_parliaments(tid = TRUE, tidy_style = "snake_case")
ref_parliamentary_posts(tid = TRUE, tidy_style = "snake_case")
ref_parliamentary_ranks(tid = TRUE, tidy_style = "snake_case")
ref_parliament_types(tid = TRUE, tidy_style = "snake_case")
ref_parties(tid = TRUE, tidy_style = "snake_case")
ref_party_sub_types(tid = TRUE, tidy_style = "snake_case")
ref_photo_outputs(tidy = TRUE, tidy_style = "snake_case")

ref_statuses(tidy = TRUE, tidy_style = "snake_case")

ref_titles(tidy = TRUE, tidy_style = "snake_case")

**Arguments**

- **tidy**: Fix the variable names in the tibble to remove special characters and superfluous text, and converts the variable names to a consistent style. Defaults to TRUE.

- **tidy_style**: The style to convert variable names to, if tidy=TRUE. Accepts one of "snake_case", "camelCase" and "period_case". Defaults to "snake_case".

**See Also**

- [mnis_all_reference](#)

**Examples**

```r
## Not run:
ref_address_types()
# The types of addresses available in member's contact details.
# Includes websites and social media, as well as physical addresses.

ref_answering_bodies()
# The bodies that members can address questions to.

ref_areas()
# Geographic areas.

ref_area_types()
# Identifiers for grouping areas (e.g. borough constituencies).

ref_biography_categories()
# Member biography categories.

ref_cabinets()
# Connections that a member has to the cabinet or shadow cabinet.

ref_committees()
# Identifier for parliamentary committees.

ref_committee_types()
# Types of parliamentary committees.

ref_constituencies()
# All constituencies.

ref_constituency_areas()
# The links between constituencies and constituency areas.
```
ref_constituency_types()
# Constituency categories.

ref_countries()
# List of countries that could be listed as members' birthplace.

ref_departments()
# Government and opposition departments.

ref_disqualification_types()
# Types of ways members can be disqualified from sitting in the House.

ref_elections()
# Codes of general and by-elections.

ref_election_types()
# Election categories.

ref_end_reasons()
# Reasons a member may leave the House of Lords or the House of Commons.

ref_experience_types()
# Types of non-parliamentary experience members can list.

ref_government_post_departments()
# All departments that can contain government posts.

ref_government_posts()
# All government posts.

ref_government_ranks()
# All government post ranks.

ref_honourary_prefixes()
# The types of honourary prefixes for members.

ref_honour_lists()
# The types of honour lists that a member may be honoured in.

ref_honours()
# The different honours available to members.

ref_interest_categories()
# The categories available for reporting financial interests.

ref_lords_membership_types()
# Different types of membership of the House of Lords.

ref_lords_ranks()
# Ranks that peers may hold.

ref_opposition_post_departments()
# The link between opposition posts and the government department they shadow.

ref_opposition_posts()
# Opposition posts.

ref_opposition_ranks()
# How opposition posts are ranked.

ref_other_parliaments()
# Other parliaments that a member may have sat in.

ref_parliamentary_posts()
# The different parliamentary posts available.

ref_parliamentary_ranks()
# How those parliamentary posts are ranked.

ref_parliament_types()
# Types of parliaments that parliamentary data may link to.

ref_parties()
# All parties that members can be affiliated with.

ref_party_sub_types()
# Sub-types of parties.

ref_photo_outputs()
# Outputs that a photo of a member may be linked to.

ref_statuses()
# A member's possible current status in the House.

ref_titles()
# Salutary titles.

mnis_reference()
# Returns a list of all possible reference functions.

## End(Not run)

---

**mnis_tidy**

Functions to tidy up the variable names returned from the API, and turn dates and datetimes to POSIXct.

### Description

Functions to tidy up the variable names returned from the API, and turn dates and datetimes to POSIXct.

member_tidy

ref_tidy
tidy_bom

Usage

mnis_tidy(df, tidy_style = "snake_case")

member_tidy(df)

ref_tidy(df, tidy_style)

Arguments

df

The tibble to tidy.

tidy_style

The style to tidy the tibble with.

tidy_bom

A function to strip Byte Order Marks (BOM) out of JSON data returned from the API.

Description

A function to strip Byte Order Marks (BOM) out of JSON data returned from the API.

Usage

tidy_bom(df)

Arguments

df

The GET returned from call to API.
Index

*Topic mnis

- mnis_additional, 4
- mnis_all_members, 6
- mnis_base, 8
- mnis_constituency_results, 9
- mnis_department, 10
- mnis_eligible, 11
- mnis_extra, 12
- mnis_full_biog, 14
- mnis_general_election_results, 15
- mnis_joined_between, 16
- mnis_lords_type, 17
- mnis_member_date, 18
- mnis_party_state, 20
- mnis_reference, 22

- constituency_results_tidy, 2

- date_tidy, 3

- member_tidy (mnis_tidy), 26
- mnis, 3
- mnis-package (mnis), 3
- mnis_additional, 4, 6, 12, 14
- mnis_addresses (mnis_additional), 4
- mnis_all_members, 4, 6, 6, 14
- mnis_all_reference, 7, 24
- mnis_base, 4, 8
- mnis_basic_details, 14
- mnis_basic_details (mnis_additional), 4
- mnis_bio (mnis), 9
- mnis_bio (mnis_additional), 4
- mnis_committees (mnis_additional), 4
- mnis_constituencies (mnis_additional), 4
- mnis_constituency_results, 3, 4, 9
- mnis_department, 4, 10
- mnis_elections_contested (mnis_additional), 4
- mnis_eligible, 4, 11

- mnis_experiences (mnis_additional), 4
- mnis_extra, 4, 6, 12, 14
- mnis_full_biog, 4, 6, 12, 14, 14
- mnis_general_election_results, 4, 15
- mnis_government_posts (mnis_additional), 4
- mnis_honours (mnis_additional), 4
- mnis_house_memberships (mnis_additional), 4
- mnis_interests (mnis_additional), 4
- mnis_joined_between, 4, 16
- mnis_known_as (mnis_additional), 4
- mnis_lords_type, 4, 17
- mnis_maiden_speeches (mnis_additional), 4
- mnis_member_date, 4, 18
- mnis_mps_on_date, 18, 19, 20
- mnis_opposition_posts (mnis_additional), 4
- mnis_other_parliaments (mnis_additional), 4
- mnis_parliamentary_posts (mnis_additional), 4
- mnis_parties (mnis_additional), 4
- mnis_party_state, 4, 20, 21
- mnis_peers_on_date, 20, 21, 21
- mnis_preferred_names (mnis_additional), 4
- mnis_reference, 4, 8, 10, 11, 16, 17, 22
- mnis_staff (mnis_additional), 4
- mnis_statuses (mnis_additional), 4
- mnis_tidy, 8, 26

- ref_address_types (mnis_reference), 22
- ref_answering_bodies (mnis_reference), 22
- ref_area_types (mnis_reference), 22
- ref_areas (mnis_reference), 22
- ref_bio (mnis_reference), 22
- (mnis_reference), 22
INDEX

ref_cabinets (mnis_reference), 22
ref_committee_types (mnis_reference), 22
ref_committees (mnis_reference), 22
ref_constituencies, 9
ref_constituencies (mnis_reference), 22
ref.constituency_areas
    (mnis_reference), 22
ref.constituency_types
    (mnis_reference), 22
ref.departments (mnis_reference), 22
ref.disqualification_types
    (mnis_reference), 22
ref.election_types (mnis_reference), 22
ref.elections (mnis_reference), 22
ref.end_reasons (mnis_reference), 22
ref.experience_types (mnis_reference),
    22
ref.government_post_departments
    (mnis_reference), 22
ref.government_posts (mnis_reference),
    22
ref.government_ranks (mnis_reference),
    22
ref.honour_lists (mnis_reference), 22
ref.honorary_prefixes
    (mnis_reference), 22
ref.honours (mnis_reference), 22
ref.interest_categories
    (mnis_reference), 22
ref.lords_membership_types
    (mnis_reference), 22
ref.lords_ranks (mnis_reference), 22
ref.opposition_post_departments
    (mnis_reference), 22
ref.opposition_posts (mnis_reference),
    22
ref.opposition_ranks (mnis_reference),
    22
ref.other_parliaments (mnis_reference),
    22
ref.parliament_types (mnis_reference),
    22
ref.parliamentary_posts
    (mnis_reference), 22
ref.parliamentary_ranks
    (mnis_reference), 22
ref_parties, 11
ref_parties (mnis_reference), 22
ref.party_sub_types (mnis_reference), 22
ref.photo_outputs (mnis_reference), 22
ref_statuses (mnis_reference), 22
ref.tidy (mnis_tidy), 26
ref.titles (mnis_reference), 22
tidy_bom, 27