Package ‘semnar’

November 3, 2022

Title Constructing and Interacting with Databases of Presentations

Version 0.8.1

Description Provides methods for constructing and maintaining a database of presentations in R. The presentations are either ones that the user gives or gave or presentations at a particular event or event series. The package also provides a plot method for the interactive mapping of the presentations using ‘leaflet’ by grouping them according to country, city, year and other presentation attributes. The markers on the map come with popups providing presentation details (title, institution, event, links to materials and events, and so on).

License GPL-3

Encoding UTF-8

Depends R (>= 4.1.0)

Imports jsonlite, lubridate, parsedate, leaflet, urlshorteneR

RoxygenNote 7.2.1

Suggests covr, tinytest

NeedsCompilation no

Author Ioannis Kosmidis [aut, cre] (<https://orcid.org/0000-0003-1556-0302>)

Maintainer Ioannis Kosmidis <ioannis.kosmidis@warwick.ac.uk>

Repository CRAN

Date/Publication 2022-11-03 11:50:02 UTC

R topics documented:

- seminar-package ............................................ 2
- add_presentation ........................................ 2
- event ................................................... 6
- get_event.semnar ......................................... 7
- get_presenter.semnar ...................................... 9
- guess_address.semnar .................................... 10
- plot.semnar ............................................... 11
- presenter ............................................... 13
- set_event.semnar ......................................... 14
- set_presenter.semnar ..................................... 15
- shorten_url ............................................. 16
Description

Provides methods for constructing and maintaining a database of presentations in R. The presentations are either ones that the user gives or gave or presentations at a particular event or event series. The package also provides a plot method for the interactive mapping of the presentations using `leaflet` by grouping them according to country, city, year and other presentation attributes. The markers on the map come with popups providing presentation details (title, institution, event, links to materials and events, and so on).

See Also

`add_presentation()` `presenter()` `event()` `plot.semnar()`

Usage

```r
add_presentation(
  object,
  presenter = NA,
  presenter_name = NA,
  presenter_midname = NA,
  presenter_surname = NA,
  presenter_affiliation = NA,
  presenter_email = NA,
  presenter_link = NA,
  presenter_address = NA,
  country = NA,
  city = NA,
  state = NA,
  lon = NA,
  lat = NA,
  event = NA,
  title = NA,
  abstract = NA,
```

add_presentation

type = NA,  
link = NA,  
materials = NA,  
institution = NA,  
department = NA,  
school = NA,  
venue = NA,  
address = NA,  
postcode = NA,  
room = NA,  
start = NA,  
end = NA,  
year = NA,  
month = NA,  
day = NA,  
start_hour = NA,  
end_hour = start_hour,  
start_min = 0L,  
end_min = start_min,  
start_sec = 0L,  
end_sec = 0L,  
tag = NA,  
tz = "UTC"
)

Arguments

object either an object an object of class semnar or unspecified (default). See Details.
presenter either NA (default) or an object of class semnar_presenter. In the latter case, all input to the presenter_* arguments below is ignored and populated according to the supplied object.
presenter_name name of the presenter of the presentation; character string or NA (default).
presenter_midname middle name of the presenter of the presentation; character string or NA (default).
presenter_surname surname of the presenter of the presentation; character string or NA (default).
presenter_affiliation affiliation of the presenter of the presentation; character string or NA (default).
presenter_email email of the presenter of the presentation; character string or NA (default).
presenter_link link to the webpage of the presenter of the presentation; character string or NA (default).
presenter_address address of the presenter; character string or NA (default).
country country where the presentation took place; character string or NA (default).
city city where the presentation took place; character string or NA (default).
add_presentation

state
longitude of the venue of the presentation; numeric or NA (default).

lat
latitude of the venue of the presentation; numeric or NA (default).

event
either NA (default) or a character string with the name of the event at which the presentation is/was given or an object of class seminar_event. In the latter case, all input to country, city, state, lon, lat, link, institution, department, school, venue, address, postcode is ignored and populated according to the supplied object.

title
title of the presentation; character string or NA (default).

abstract
abstract of the presentation; character string or NA (default).

type
the type of the talk. Available options are NA (default), "seminar", "webinar", "lecture", "presentation", "talk", "poster".

link
link to the event or seminar/talk page; character string or NA (default).

materials
link to the slides or materials from the seminar/talk; character string or NA (default).

institution
institution at which the event or seminar/talk page took/will take space; character string or NA (default).

department
department at which the event or seminar/talk page took/will take space; character string or NA (default).

school
school at which the event or seminar/talk page took/will take space; character string or NA (default).

venue
venue at which the event or seminar/talk page took/will take space; character string or NA (default).

address
address where the seminar/talk took place; character string or NA (default).

postcode
post code where the seminar/talk took place; character string or NA (default).

room
room at which the event or seminar/talk page took/will take space; character string or NA (default).

start
NA (default) or a character string to be parsed into a calendar date and time using parsedate::parse_date(). If the latter, start overrides any input in year, month, day, start_hour, start_min, start_sec

date
NA (default) or a character string to be parsed into a calendar date and time using parsedate::parse_date(). If the latter, end overrides any input in year, month, day, end_hour, end_min, end_sec.

year
year of the presentation; numeric, e.g. 2019, or NA (default).

month
month of the presentation; numeric (1-12) or NA (default).

day
day of the presentation; numeric (1-31) or NA (default).

start_hour
start hour of the presentation; numeric (1-24) or NA (default).

end_hour
end hour of the presentation; numeric (1-24) or NA (default).

start_min
start minute of the presentation; numeric (0-60) or NA (default).

end_min
end minute of the presentation; numeric (0-60) or NA (default).

start_sec
start second of the presentation; numeric (0-60) or NA (default).

end_sec
end second of the presentation; numeric (0-60) or NA (default).

tag
a tag for the presentation; character string or NA (default).

tz
timezone. Default is "UTC". See DateTimeClasses for details.
**Details**

If object is not specified then `add_presentation()` will create an `semnar` object based on the supplied inputs, otherwise it will add the details of the new presentation on object.

If the start date and end date are specified (either through `start` and `end` or through `year`, `month`, `day`) and no information is provided or can be inferred about start/end times, then a time of 0 hours, 0 minutes and 0 seconds is assumed.

An error is thrown if the start date/time is after the end date/time.

**Value**

A structured `data.frame()` that also inherits from class `semnar`, including the supplied presentation details.

**See Also**

`presenter()`, `plot.semnar()`, `shorten_url()`, `guess_address()`

**Examples**

```r
# Two of my past talks


out
```
Create a `seminar_event` object with event details

Usage

```r
event(
  event = NA,
  country = NA,
  city = NA,
  state = NA,
  lon = NA,
  lat = NA,
  link = NA,
  institution = NA,
  department = NA,
  school = NA,
  venue = NA,
  address = NA,
  postcode = NA
)
```

Arguments

- `event`: either `NA` (default) or a character string with the name of the event at which the presentation is/was given or an object of class `seminar_event`. In the latter case, all input to `country`, `city`, `state`, `lon`, `lat`, `link`, `institution`, `department`, `school`, `venue`, `address`, `postcode` is ignored and populated according to the supplied object.
- `country`: country where the presentation took place; character string or `NA` (default).
- `city`: city where the presentation took place; character string or `NA` (default).
- `state`: state where the presentation took place; character string or `NA` (default).
- `lon`: longitude of the venue of the presentation; numeric or `NA` (default).
- `lat`: latitude of the venue of the presentation; numeric or `NA` (default).
- `link`: link to the event or seminar/talk page; character string or `NA` (default).
- `institution`: institution at which the event or seminar/talk page took/will take place; character string or `NA` (default).
- `department`: department at which the event or seminar/talk page took/will take place; character string or `NA` (default).
- `school`: school at which the event or seminar/talk page took/will take place; character string or `NA` (default).
venue

venue at which the event or seminar/talk page took/will take space; character string or NA (default).

address

address where the seminar/talk took place; character string or NA (default).

postcode

post code where the seminar/talk took place; character string or NA (default).

Value

A structured data.frame() that also inherits from class seminar_event, including the supplied event details.

See Also

get_event() set_event() presenter() get_presenter() set_presenter()

Examples

# A past talk of mine
IK_warwick <- presenter(name = "Ioannis",
surname = "Kosmidis",
affiliation = "University of Warwick",
link = "https://www.ikosmidis.com")

YRM <- event(event = "Young Researchers' Meeting",
country = "England",
city = "Coventry",
state = "NA",
lon = -1.560843, lat = 52.384019,
link = "https://warwick.ac.uk/fac/sci/statistics/news/yrm/",
institution = "University of Warwick",
department = "Department of Statistics",
school = NA,
venue = "Mathemtical Sciences Building",
address = NA,
postcode = NA)

out <- add_presentation(event = YRM,
presenter = IK_warwick,
title = "A workflow that most probably isn't yours",
type = "presentation",
start = "20190528 16:00", end = "20190528 17:00")

get_event.semnar

Get event information from a seminar object

Description

Get event information from a seminar object
get_event.semnar

Usage

```R
## S3 method for class 'semnar'
get_event(object)

get_event(object)
```

Arguments

- `object` either an object an object of class `semnar`.

Value

A list of `semnar_event` objects, with the unique presenters in the object.

See Also

- `presenter()`
- `set_presenter()`

Examples

```R
IK_warwick <- presenter(name = "Ioannis",
surname = "Kosmidis",
affiliation = "University of Warwick",
link = "https://www.ikosmidis.com")

YRM <- event(event = "Young Researchers' Meeting",
country = "England",
city = "Coventry",
state = "NA",
lon = -1.560843, lat = 52.384019,
link = "https://warwick.ac.uk/fac/sci/statistics/news/yrm/",
institution = "University of Warwick",
department = "Department of Statistics",
school = NA,
venue = "Mathematical Sciences Building",
address = NA,
postcode = NA)

out <- add_presentation(presenter = IK_warwick,
event = "A",
country = "Greece",
title = "S") |> 
add_presentation(presenter = IK_warwick,
event = "B",
city = "London",
country = "UK",
title = "T") |> 
add_presentation(presenter = IK_warwick,
event = YRM,
title = "U")

get_event(out)
```
Description

Get presenter information from a `seminar` object

Usage

```r
## S3 method for class 'seminar'
get_presenter(object)

get_presenter(object)
```

Arguments

- `object`: either an object an object of class `seminar`.

Value

A list of `seminar_presenter` objects, with the unique presenters in the object.

See Also

`presenter()`, `set_presenter()`

Examples

```r
out <- add_presentation(presenter_name = "Ioannis",
                        presenter_surname = "Kosmidis",
                        presenter_affiliation = "University of Warwick",
                        presenter_email = "ioannis.kosmidis@warwick.ac.uk",
                        title = "A") |> 
add_presentation(presenter_name = "Ioannis",
                        presenter_surname = "Kosmidis",
                        presenter_affiliation = "University College London",
                        title = "B") |> 
add_presentation(presenter_name = "Ioannis",
                        presenter_surname = "Kosmidis",
                        presenter_affiliation = "University College London",
                        title = "C")

get_presenter(out)
```
guess_address.semnar  
**Guess presentation address in a seminar object using OSM’s API for reverse geocoding**

**Description**

Guess presentation address in a *seminar* object using OSM’s API for reverse geocoding.

**Usage**

```r
## S3 method for class 'seminar'
guess_address(object, all = FALSE)

guess_address(object, all = TRUE)
```

**Arguments**

- **object**
  - an object of class *seminar*. See `add_presentation()`.
- **all**
  - should we be guessing all addresses (TRUE) or only missing ones (FALSE; default)?

**Details**

`guess_address()` is using reverse geocoding through the API at `https://nominatim.openstreetmap.org`. Please check at that link for requests limits.

**Value**

An object of class *seminar* with all (all = TRUE) or the missing presentation addresses (all = FALSE) completed.

**See Also**

- `add_presentation`

**Examples**

```r
out <- add_presentation(country = "England", city = "Coventry",
lon = -1.560843, lat = 52.384019,
event = "Young Researchers Meeting",
title = "A workflow that most probably isn't yours",
link = "https://warwick.ac.uk/fac/sci/statistics/news/yrn/",
type = "presentation", institution = "University of Warwick",
department = "Department of Statistics",
venue = "Mathematical Sciences Building", room = "M1.02",
year = 2019, month = 5, day = 28,
```
```r
start_hour = 16, start_min = 00,
end_hour = 17, end_min = 00) |> 
add_presentation(country = "United States", city = "Stanford",
lon = -122.165330, lat = 37.429464,
event = "useR! 2016",
title = "brglm: Reduced-bias inference in generalized linear models",
materials = "https://bit.ly/2KCBbKg",
type = "presentation", institution = NA, department = NA,
venue = "Stanford Institute for Economic Policy Research",
room = "Siepr 120",
year = 2016, month = 06, day = 29,
start_hour = 14, start_min = 15,
end_hour = 14, end_min = 35)

out$address
# Reverse geocoding
out <- guess_address(out)
out$address
```

---

**Description**

Interactive seminar maps

**Usage**

```r
## S3 method for class 'seminar'
plot(
x,
group = "city",
title = NA,
title_position = "bottomleft",
provider = "OpenStreetMap.Mapnik",
interval = TRUE,
date_format = "dmy",
shorten_url = FALSE,
service = "Is.gd",
width = NULL,
height = NULL,
show_event_url = FALSE,
opacity = 0.2,
past_year_colour = "#737373",
this_year_colour = "#ef3b2c",
...
)
```
Arguments

x an object of class `semnar`. See `add_presentation()`.

group according to what should the seminars be selected on the map? Available options are "none", "year", "month", "presenter", "event", "country", "city" (default), "year+country".

title character string for the title of the map. Default is NA, which produces no title.

title_position the position of the title on the map, if title is not NA. Available options are "bottomleft" (default), "bottomright", "topleft", "topright".

provider the provider of tiles for the base map. See `leaflet::addProviderTiles()`. Default is "OpenStreetMap.Mapnik".

interval Should the start and end times in each popup be displayed as an interval (TRUE; default) or in two separate lines (FALSE)?

date_format In what format should the dates be displayed? Available options are "dmy", "mdy", "ymd", "ymd", where "y" stands for year, "m" stands for month, and "d" stands for day.

shorten_url Should the URL links in "object$link" be shortened? Default is FALSE.

service service to use for shortening URLs. Current options are "Is.gd" (default) and "V.gd". See `shorten_url()`.

width As in `leaflet::leaflet()`. Default is NULL.

height As in `leaflet::leaflet()`. Default is NULL.

show_event_url Should the event or talk/seminar URL ("link" argument in `add_presentation()`) be printed in the popups (TRUE), or the event name ("event" argument in `add_presentation()`) become a hyperlink pointing to the event or talk/seminar URL (FALSE; default).

opacity As in `leaflet::markerOptions()`. Default is 0.2.

past_year_colour Colour to be used for past year's presentations. Default is "#ef3b2c". See Details.

this_year_colour Colour to be used for this year's presentations. Default is "#737373". See Details.

... Arguments to be passed to other methods. Currently unused.

Value

An interactive map (opens in browser) with the locations of the events. When the locations are clicked, pup-ups appear which give access to event details. A legend with the colour-year combination is printed only if `this_year_colour != past_year_colour`. The current year is determined by `lubridate::year(Sys.time())`.

See Also

`add_presentation()`
Examples

```r
```

if (interactive()) {
  plot(out, group = "city", 
       title = "<a href='https://cran.r-project.org/package=semnar'>semnar</a> map")
}
```

presenter

Create a `semnar_presenter()` object with presenter details

Description

Create a `semnar_presenter()` object with presenter details

Usage

```r
presenter(
  name = NA,
  midname = NA,
  surname = NA,
  affiliation = NA,
  link = NA,
  email = NA,
  address = NA
)
```
Arguments

name  name of the presenter; character string or NA (default).
midname  middle name of the presenter; character string or NA (default).
surname  surname of the presenter; character string or NA (default).
affiliation  affiliation of the presenter; character string or NA (default).
link  link to the webpage of the presenter; character string or NA (default).
email  email of the presenter; character string or NA (default).
address  address of the presenter; character string or NA (default).

Value

A structured data.frame() that also inherits from class semnar_presenter, including the supplied presenter details.

See Also

get_presenter set_presenter

Examples

# A past talk of mine
ik_warwick <- presenter(name = "Ioannis", surname = "Kosmidis",
                        affiliation = "University of Warwick", link = "https://www.ikosmidis.com")
out <- add_presentation(country = "England", city = "Coventry",
                         lon = -1.560843, lat = 52.384019,
                         event = "Young Researchers' Meeting",
                         title = "A workflow that most probably isn't yours",
                         presenter = ik_warwick,
                         link = "https://warwick.ac.uk/fac/sci/statistics/news/yrm/",
                         type = "presentation", institution = "University of Warwick",
                         department = "Department of Statistics",
                         venue = "Mathematical Sciences Building", room = "M1.02",
                         year = 2019, month = 5, day = 28,
                         start_hour = 16, start_min = 00,
                         end_hour = 17, end_min = 00)
Usage

```r
## S3 method for class 'semnar'
set_event(object, event)
```

Arguments

- `object`: either an object an object of class `semnar`.
- `event`: an object of class `semnar_event`.

Value

The `semnar` object supplied in `object`, with the event information as in `event`. See the output of `semnar:::get_event_variables()` for what variables are affected.

See Also

`event()`, `get_event()`

Examples

```r
out <- add_presentation(presenter_name = "Ioannis",
  presenter_surname = "Kosmidis",
  presenter_affiliation = "University of Warwick",
  title = "A",
  country = "UK") |> 
add_presentation(presenter_name = "Ioannis",
  presenter_surname = "Kosmidis",
  presenter_affiliation = "University College London",
  title = "B") |> 
add_presentation(presenter_name = "Ioannis",
  presenter_surname = "Kosmidis",
  presenter_affiliation = "University College London",
  title = "C",
  country = "Greece", city = "Athens")
WA <- event("WA", "UK", "Lon", "Lon", NA, NA, NA, "British Library")

out
set_event(out, WA)
```

---

**Description**

Set presenter information from a `semnar` object
Usage

```r
## S3 method for class 'semnar'
set_presenter(object, presenter)

set_presenter(object, presenter)
```

Arguments

- `object`: either an object an object of class `semnar`.
- `presenter`: an object of class `semnar_presenter`.

Value

The `semnar` object supplied in `object`, with the presenter information as in `presenter`. See the output of `semnar:::get_presenter_variables()` for what variables are affected.

See Also

- `presenter` get_presenter

Examples

```r
out <- add_presentation(presenter_name = "Ioannis",
                        presenter_surname = "Kosmidis",
                        presenter_affiliation = "University of Warwick",
                        title = "A") |> 
  add_presentation(presenter_name = "Ioannis",
                   presenter_surname = "Kosmidis",
                   presenter_affiliation = "University College London",
                   title = "B") |> 
  add_presentation(presenter_name = "Ioannis",
                   presenter_surname = "Kosmidis",
                   presenter_affiliation = "University College London",
                   title = "C")

john_doe <- presenter(name = "John",
                      surname = "Doe",
                      affiliation = "Nowhereland",
                      link = "https://johndoe.nowhereland.com",
                      email = "john.doe@nowhereland.com")

out
set_presenter(out, john_doe)
```

shorten_url

Shorten the URLs of any links in a `semnar` object

Description

Shorten the URLs of any links in a `semnar` object
**Usage**

shorten_url(object, service = "V.gd")

```r
## S3 method for class 'seminar'
shorten_url(object, service = "Is.gd")
```

**Arguments**

- **object**: an object of class `seminar`. See `add_presentation()`.
- **service**: service to use for shortening URLs. Current options are "V.gd" (default) and "Is.gd".

**Value**

An object of class `seminar` with any URLs in `object$link` replace with shorter ones according to the value of `service`.

**See Also**

`add_presentation()` `guess_address()`
Index

add_presentation, 2, 10
add_presentation(), 2, 5, 10, 12, 17

data.frame(), 5, 7, 14
DateTimeClasses, 4

event, 6
event(), 2, 15

generate_event(generate_event.semnar), 7
generate_event(), 7, 15
generate_event.semnar, 7
generate_presenter(generate_presenter.semnar), 9
generate_presenter(), 7
generate_presenter.semnar, 9
guess_address(guess_address.semnar), 10
guess_address(), 5, 10, 17
guess_address.semnar, 10

leaflet::addProviderTiles(), 12
leaflet::leaflet(), 12
leaflet::markerOptions(), 12

parsedate::parse_date(), 4
plot.semnar, 11
plot.semnar(), 2, 5
presenter, 13
presenter(), 2, 5, 7–9

seminar, 2, 3, 5, 7–12, 14–17
seminar(add_presentation), 2
seminar-package, 2
seminar_event, 4, 6–8, 15
seminar_event(event), 6
seminar_presenter, 3, 9, 14, 16
seminar_presenter(presenter), 13
set_event(set_event.semnar), 14
set_event(), 7
set_event.semnar, 14
set_presenter(set_presenter.semnar), 15
set_presenter(), 7–9

set_presenter.semnar, 15
shorten_url, 16
shorten_url(), 5, 12