Package 'trainR'

October 14, 2022

Title An Interface to the National Rail Enquiries Systems

Version 0.0.1

Description The goal of 'trainR' is to provide a simple interface to the National Rail Enquiries (NRE) systems. There are few data feeds available, the simplest of them is Darwin, which provides real-time arrival and departure predictions, platform numbers, delay estimates, schedule changes and cancellations. Other data feeds provide historical data, Historic Service Performance (HSP), and much more. 'trainR' simplifies the data retrieval, so that the users can focus on their analyses. For more details visit <https://www.nationalrail.co.uk/46391.aspx>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Depends R (>= 2.10)

URL https://github.com/villegar/trainR/, https://villegar.github.io/trainR/

BugReports https://github.com/villegar/trainR/issues/

Language en-GB

Imports RCurl, dplyr, glue, lubridate, magrittr, purrr, stringr, tibble, tidyr, usethis, xml2

NeedsCompilation no

Author Roberto Villegas-Diaz [aut, cre] (<https://orcid.org/0000-0001-5036-8661>)

Maintainer Roberto Villegas-Diaz <villegas.roberto@hotmail.com>

Repository CRAN

Date/Publication 2021-01-20 11:10:02 UTC

R topics documented:

`GetArrBoardWithDetailsRequest` *Get all public arrivals*

Description

Get all public arrivals for the supplied CRS code within a defined time window, including service details.

Usage

```
GetArrBoardWithDetailsRequest(
  crs,
  filterCrs = NA,
  filterType = "to",
  numRows = 150,
  timeOffset = 0,
  timeWindow = 120,
  token = get_token(),
  url = "https://lite.realtime.nationalrail.co.uk/OpenLDBWS/ldb11.asmx",
  verbose = FALSE
)
```

Arguments

`crs`	(string, 3 characters, alphabetic): The CRS code (see above) of the location for which the request is being made.
`filterCrs`	(string, 3 characters, alphabetic): The CRS code of either an origin or destination location to filter in. Optional.
`filterType`	(string, either "from" or "to"): The type of filter to apply. Filters services to include only those originating or terminating at the `filterCrs` location. Defaults to "to".
`numRows`	(integer, between 0 and 150 exclusive): The number of services to return in the resulting station board.

timeOffset	(integer, between -120 and 120 exclusive): An offset in minutes against the current time to provide the station board for. Defaults to 0.
timeWindow	(integer, between -120 and 120 exclusive): How far into the future in minutes, relative to timeOffset, to return services for. Defaults to 120.
token	Token to access the data feed. The token can be obtained at http://realtime.nationalrail.co.uk/OpenLDBWSRegistration/.
url	Data feed source URL.
verbose	Boolean flag to indicate whether or not to show status messages.

Value

Tibble with arrival records.

Examples

```
## Not run:
rdg_arr <- trainR::GetArrBoardWithDetailsRequest("RDG")
rdg_arr <- trainR::GetArrBoardWithDetailsRequest("RDG", filterCrs = "BRI")
trainR::print(rdg_arr)

## End(Not run)
```

GetArrDepBoardWithDetailsRequest

Get all public arrivals and departures

Description

Get all public arrivals and departures for the supplied CRS code within a defined time window, including service details.

Usage

```
GetArrDepBoardWithDetailsRequest(
  crs,
  filterCrs = NA,
  filterType = "from",
  numRows = 150,
  timeOffset = 0,
  timeWindow = 120,
  token = get_token(),
  url = "https://lite.realtime.nationalrail.co.uk/OpenLDBWS/ldb11.asmx",
  verbose = FALSE
)
```

Arguments

crs	(string, 3 characters, alphabetic): The CRS code (see above) of the location for which the request is being made.
filterCrs	(string, 3 characters, alphabetic): The CRS code of either an origin or destination location to filter in. Optional.
filterType	(string, either "from" or "to"): The type of filter to apply. Filters services to include only those originating or terminating at the filterCrs location. Defaults to "to".
numRows	(integer, between 0 and 150 exclusive): The number of services to return in the resulting station board.
timeOffset	(integer, between -120 and 120 exclusive): An offset in minutes against the current time to provide the station board for. Defaults to 0.
timeWindow	(integer, between -120 and 120 exclusive): How far into the future in minutes, relative to timeOffset, to return services for. Defaults to 120.
token	Token to access the data feed. The token can be obtained at http://realtime.nationalrail.co.uk/OpenLDBWSRegistration/.
url	Data feed source URL.
verbose	Boolean flag to indicate whether or not to show status messages.

Value

Tibble with arrival and departure records.

Examples

```
## Not run:
rdg<- trainR::GetArrDepBoardWithDetailsRequest("RDG")
rdg <- trainR::GetArrDepBoardWithDetailsRequest("RDG", filterCrs = "BRI")
trainR::print(rdg)

## End(Not run)
```

GetDepBoardWithDetailsRequest

Get all public departures

Description

Get all public departures for the supplied CRS code within a defined time window, including service details.

Usage

```
GetDepBoardWithDetailsRequest(
  crs,
  filterCrs = NA,
  filterType = "from",
  numRows = 150,
  timeOffset = 0,
  timeWindow = 120,
  token = get_token(),
  url = "https://lite.realtime.nationalrail.co.uk/OpenLDBWS/ldb11.asmx",
  verbose = FALSE
)
```

Arguments

`crs`	(string, 3 characters, alphabetic): The CRS code (see above) of the location for which the request is being made.
`filterCrs`	(string, 3 characters, alphabetic): The CRS code of either an origin or destination location to filter in. Optional.
`filterType`	(string, either "from" or "to"): The type of filter to apply. Filters services to include only those originating or terminating at the `filterCrs` location. Defaults to "to".
`numRows`	(integer, between 0 and 150 exclusive): The number of services to return in the resulting station board.
`timeOffset`	(integer, between -120 and 120 exclusive): An offset in minutes against the current time to provide the station board for. Defaults to 0.
`timeWindow`	(integer, between -120 and 120 exclusive): How far into the future in minutes, relative to `timeOffset`, to return services for. Defaults to 120.
`token`	Token to access the data feed. The token can be obtained at http://realtime.nationalrail.co.uk/OpenLDBWSRegistration/.
`url`	Data feed source URL.
`verbose`	Boolean flag to indicate whether or not to show status messages.

Value

Tibble with departure records.

Examples

```
## Not run:
rdg_dep <- trainR::GetDepBoardWithDetailsRequest("RDG")
rdg_dep <- trainR::GetDepBoardWithDetailsRequest("RDG", filterCrs = "BRI")
trainR::print(rdg_dep)

## End(Not run)
```

GetServiceDetailsRequest *Get service details*

Description

Get the service details for a specific service identified by a station board. These details are supplied relative to the station board from which the `serviceID` field value was generated. Service details are only available while the service appears on the station board from which it was obtained. This is normally for two minutes after it is expected to have departed, or after a terminal arrival. If a request is made for a service that is no longer available then a `NULL` value is returned.

Usage

```
GetServiceDetailsRequest(
  serviceID,
  token = get_token(),
  url = "https://lite.realtime.nationalrail.co.uk/OpenLDBWS/ldb11.asmx",
  verbose = FALSE
)
```

Arguments

`serviceID`	(string): The LDBWS service ID of the service to request the details of. The service ID is obtained from a service listed in a `StationBoard` object returned from any other request.
`token`	Token to access the data feed. The token can be obtained at http://realtime.nationalrail.co.uk/OpenLDBWSRegistration/.
`url`	Data feed source URL.
`verbose`	Boolean flag to indicate whether or not to show status messages.

Value

Tibble with departure records.

get_token *Get user's token*

Description

Get user's token to access the National Rail Enquiries (NRE) data feeds.

Usage

```
get_token(ENV = "NRE")
```

Arguments

ENV	String with environment variable containing the token to access the NRE data feeds (default = "NRE").

Value

String with token.

`print` *Print Values*

Description

Print Values

Usage

```
print(x, ...)

## S3 method for class 'StationBoard'
print(x, ..., station = NA)

## S3 method for class 'busServices'
print(x, ...)

## S3 method for class 'trainServices'
print(x, ...)

## S3 method for class 'previousCallingPoints'
print(x, ...)

## S3 method for class 'subsequentCallingPoints'
print(x, ...)
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.
station	String to indicate if the destination or origin station should be displayed.

Value

Input data invisibly.

set_token *Configure user's token*

Description

Configure user's token to access the National Rail Enquiries (NRE) data feeds.

Usage

```
set_token()
```

Details

To obtain an access token, you must complete the registration form found at http://realtime.nationalrail.co.uk/OpenLDBWSRegistration/.

Value

Nothing, helper function to set up environment variable.

station_codes *National Rail Enquiries (NRE) Station Codes dataset*

Description

A dataset containing information of 2580 British National Rail Enquiries (NRE) station 3 letter alpha (CRS) codes.

Usage

```
data(station_codes)
```

Format

A data frame with 2580 rows and 2 variables:

name Station name.

crs Station code.

Author(s)

National Rail Enquiries (NRE)

Source

https://www.nationalrail.co.uk/stations_destinations/48541.aspx

Index