

Package: nsapi (via r-universe)

August 27, 2024

Version 0.1.0.9000

Title Connect to the NS (Dutch Railways) API

Description Access the NS api and download current departure times, disruptions and engineering work, the station list, and travel recommendations from station to station. All results will be returned as a 'data.frame'. NS (Nederlandse Spoorwegen; Dutch Railways) is the largest train travel provider in the Netherlands. for more information about the API itself see <https://www.ns.nl/en/travel-information/ns-api>. To use the API, and this package, you will need to obtain a username and password. More information about authentication and the use of the functions are described in the vignette.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

ByteCompile true

URL <https://github.com/RMHogervorst/nsapi> ,
<https://rmhogervorst.nl/nsapi/>

BugReports <https://github.com/RMHogervorst/nsapi/issues>

Roxygen list(markdown = TRUE)

RoxygenNote 6.1.0

Suggests testthat, knitr, rmarkdown, covr

Imports crul, xml2

VignetteBuilder knitr

Repository <https://rmhogervorst.r-universe.dev>

RemoteUrl <https://github.com/RMHogervorst/nsapi>

RemoteRef HEAD

RemoteSha 9b97dc6c71557892093f616fc51facdf24542646

Contents

check_ns_api_keys	2
datetime	2
get_departures	3
get_disruptions_station	4
get_stationlist	5
get_travel_advise	5
nsapi	6

Index	8
--------------	----------

check_ns_api_keys	<i>Are the username and password set?</i>
-------------------	---

Description

A function to see if your username and password are set for nsapi. Note that I have no clue if the password or username are correct, that is up to you.

Usage

```
check_ns_api_keys()
```

Details

For more details see the help vignette: `vignette("basic_use_nsapi_package", package = "nsapi")` and get your keys here: <https://www.ns.nl/ews-aanvraagformulier/?0>

datetime	<i>Translate date and time into datetime stamp to use in api call</i>
----------	---

Description

The datetime stamp format might be a bit difficult to use. This helper function allows you to supply the date and time and a datetime character string comes out.

Usage

```
datetime(date, time)
```

Arguments

date	date in iso format (year-month-day): 2018-12-29 for example
time	time in standard format (HH:MM): 20:21

Details

For example if it is 21-08-2018, 15.21 hour in the Netherlands than you can add these two items into the function.

Examples

```
## Not run:
get_travel_advise(
  fromStation = "Amsterdam Centraal",
  toStation = "Utrecht Centraal",
  dateTime = datetime("2018-08-21", "15:21"),
  departure = TRUE
)

## End(Not run)
```

get_departures	<i>Get up to date departures from a station</i>
----------------	---

Description

Shows up to date departure times for a station. Displays all departing trains for the next hour. At least 10 departure times will be sent back and at least all the departure times for the next hour. <https://www.ns.nl/en/travel-information/ns-api/documentation-up-to-date-departure-times.html>.

Usage

```
get_departures(station)
```

Arguments

station	station names need to be in Dutch and the NS webservice also accepts short versions:f.i. Groningen or GN
---------	--

Value

a dataframe with date time formatted as time in "Europe/Amsterdam" timezone.

a dataframe with departure times at this moment on the station you chose. Date time columns are formatted as time in "Europe/Amsterdam" timezone.

Examples

```
## Not run:
get_departures("UT")

## End(Not run)
```

`get_disruptions_station`*Get disruptions and engineering work*

Description

These 3 functions call out to find out about disruptions and engineering on the tracks for the current time, for the next 2 weeks or a specific station.

Usage`get_disruptions_station(station)``get_current_disruptions()``get_scheduled_engineering_work()`**Arguments**

`station` optional, station names need to be in Dutch and the NS webservice also accepts short versions:f.i. Groningen or GN

Details

- current disruptions (=unscheduled disruptions + current engineering work)

These are all the disruptions of the railroad at this moment. So both unscheduled work and work that was scheduled and currently underway. Use `get_current_disruptions()`

- scheduled engineering work(=scheduled engineering work)

Get all the scheduled engineering work for the next 2 weeks with `get_scheduled_engineering_work()`. This will exclude work that was unplanned.

- current disruptions for a specific station (=unscheduled disruptions + current engineering work)

Use `get_disruptions_station()` and give a station name as argument.

<https://www.ns.nl/en/travel-information/ns-api/documentation-disruptions-and-maintenance-work.html>

Value

a dataframe with disruptions. Date time columns are formatted as time in "Europe/Amsterdam" timezone.

Functions

- `get_current_disruptions`: current disruptions
- `get_scheduled_engineering_work`: scheduled disruptions

get_stationlist *Get a complete list of all the stations*

Description

This function should not be called too often. The stations will probably not change a lot so it might be better to save it as a dataframe in your local environment for further use.

Usage

```
get_stationlist()
```

Details

The dataframe consists of all the Dutch stations, many German and Belgian stations and bigger stations in other countries in Europe. <https://www.ns.nl/en/travel-information/ns-api/documentation-station-list.html>

get_travel_advise *Get travel advise from one station to another station*

Description

This is equivalent to the [NS reisplanner](#), you give in a from and to station, the timestamp, if you want the time to be your departure or arrival time, and optionally if you have a NS year card (has effect on some travels).

Usage

```
get_travel_advise(fromStation, toStation, dateTime = NULL,
  departure = TRUE, yearCard = FALSE, hslAllowed = FALSE,
  previousAdvices = 4, nextAdvices = 4)
```

Arguments

fromStation	the station to start from, for instance "Rotterdam Centraal"
toStation	the station to end, for instance "Utrecht Centraal"
dateTime	defaults to current time, but you can use a different one: f.i. 2012-02-21T15:50, You can also use the <code>datetime()</code> function.
departure	is the datetime the start or end time? do you want to depart on that date or arrive, defaults to departure
yearCard	if you have a NS year card (jaarabonnement) some trips will be different
hslAllowed	use of the high speed train
previousAdvices	how many advices do you want before the time
nextAdvices	how many advices do you want after

Details

You can also specify how many trips before and after the chosen time you want to collect (defaults to 4, maximum is 5 before, and 5 after).

Some things to consider: station names need to be in Dutch but the NS webservice also accepts shortened versions: "Utrecht Centraal" and "ut" is apparently the same. Station names can be found with the `get_stationlist()` call.

Although the documentation <https://www.ns.nl/en/travel-information/ns-api/documentation-travel-recommen.html> is in English, the returned values are all in Dutch. And I keep the results in Dutch.

Value

A dataframe with travel advises around your chosen date time. Date time columns are formatted as time in "Europe/Amsterdam" timezone.

Examples

```
## Not run:
get_travel_advise("Amsterdam Centraal",
"Utrecht Centraal",dateTime = "2018-08-01T15:21",departure = TRUE)

## End(Not run)
```

nsapi

NSapi package

Description

Access the NS api and download current departure times, disruptions and engineering work, the station list, and travel recommendations from station to station. All results will be returned as a 'data.frame'. NS (Nederlandse Spoorwegen; Dutch Railways) is the largest train travel provider in the Netherlands. for more information about the API itself see <https://www.ns.nl/en/travel-information/ns-api>. To use the API, and this package, you will need to obtain a username and password. More information about authentication and the use of the functions are described in the vignette.

Details

Implemented functions:

- Get travel advise between two stations [get_travel_advise](#).
- Get departures from a particular station [get_departures](#).
- All the stations [get_stationlist](#).
- Get all current disruptions [get_current_disruptions](#).
- Get all scheduled engineering in next two weeks [get_scheduled_engineering_work](#).
- Get all (un-)scheduled disruptions [get_disruptions_station](#).
- Find out if you have configured the API keys correct [check_ns_api_keys](#).

- A small helper function to create datetimestamp [datetime](#).

Image use: I have combined several images from the Noun project:

- hands on circle: Data by Gregor Cresnar from the Noun Project:
- train: railway by BomSymbols from the Noun Project
- laptop Laptop by abdul karim from the Noun Project

Index

check_ns_api_keys, [2](#), [6](#)

datetime, [2](#), [7](#)

get_current_disruptions, [6](#)

get_current_disruptions
(get_disruptions_station), [4](#)

get_departures, [3](#), [6](#)

get_disruptions_station, [4](#), [6](#)

get_scheduled_engineering_work, [6](#)

get_scheduled_engineering_work
(get_disruptions_station), [4](#)

get_stationlist, [5](#), [6](#)

get_travel_advise, [5](#), [6](#)

nsapi, [6](#)

nsapi-package (nsapi), [6](#)