

Package ‘MUCflights’

July 2, 2014

Title Munich Franz-Josef-Strauss Airport Pattern Analysis

Version 0.0-3

Date 2011-02-21

Author The students of the `Advanced R Programming Course' Basil Abou El-Komboz, Andreas Bender, Abdelilah El Hadad, Laura Goeres, Roman Hornung, Max Hughes-Brandl, Christian Lindenlaub, Christina Riedel, Ariane Straub, Florian Wickler under the supervision of Manuel Eugster and Torsten Hothorn

Maintainer Manuel Eugster <Manuel.Eugster@stat.uni-muenchen.de>

Description Functions for downloading flight data from <http://www.munich-airport.de> and for analyzing flight patterns.

License GPL (>= 2)

Depends XML, geosphere (>= 1.2-15), sp, RSQLite, NightDay

URL http://www.youtube.com/watch?v=y3RW_N-TPb4

LazyLoad yes

Repository CRAN

Date/Publication 2011-02-22 08:14:43

NeedsCompilation no

R topics documented:

airports	2
flights	3
getFlights	4
movie	5
routes	5
Index	7

airports

Airports of the World (from <http://www.OpenFlights.org>)

Description

Geographical information on (almost) all commercial airports in the world.

Usage

```
data(airports)
```

Format

A data frame with 6344 observations on the following 11 variables.

AirportID Unique OpenFlights identifier for this airport.

Name Name of airport. May or may not contain the City name.

City Main city served by airport. May be spelled differently from Name.

Country Country or territory where airport is located.

IATA 3-letter FAA code, for airports located in Country "United States of America". 3-letter IATA code, for all other airports.

ICAO 4-letter ICAO code.

Latitude Decimal degrees, usually to six significant digits. Negative is South, positive is North.

Longitude Decimal degrees, usually to six significant digits. Negative is West, positive is East.

Altitude Altitude of the airport in feet.

Timezone Hours offset from UTC. Fractional hours are expressed as decimals, eg. India is 5.5.

DST Daylight savings time. One of E (Europe), A (US/Canada), S (South America), O (Australia), Z (New Zealand), N (None) or U (Unknown).

Details

A data frame containing information on commercial airports as provided by <http://www.OpenFlights.org>.

Source

<http://www.OpenFlights.org/data.html>

Examples

```
data("airports", package = "MUCflights")
head(airports)
```

flights	<i>function to load and format flight data</i>
---------	--

Description

flights is used for loading the format the flight data from selected days.

Usage

```
flights(from = NULL, to = NULL, path = system.file("MUCflights.RData", package = "MUCflights"))
```

Arguments

from	first day which should be loaded
to	last day which should be loaded
path	path the data is located

Details

Information of the flights arrived and departed at Munich Franz-Josef-Strauss airport (from <http://www.munich-airport.de>).

Value

A data frame with the following 18 variables.

lsk	arrival(L) or departure(S)
fnr	number of the flight
lvg	airline
ha1	IATA
ha2	IATA (stopover)
ha3	IATA (stopover)
haf	from/to (German)
hafen	from/to (English)
stt	scheduled time
ett	estimated time
lde	country
len	country
ter	terminal
ber	area
typ	aircraft type
ver	aircraft type-version
saa	
gat	gate

Source

```
http://www.munich-airport.de/de/consumer/fluginfo/abflug/index.jsp http://www.munich-airport.de/de/consumer/fluginfo/ankunft/index.jsp
```

Examples

```
flight.info <- flights(from = "2011-01-07", to = "2011-01-08")
head(flight.info)
```

getFlights	<i>Function for downloading flight data from http://www.munich-airport.de</i>
------------	--

Description

getFlights is used for downloading flight data from <http://www.munich-airport.de>.

Usage

```
getFlights(status = c("ankunft", "abflug"), hour = formatC(c(0, 3:11 * 2),
  flag = "0", width = 2, format = "d"))
```

Arguments

status	specify the downloading data to landing or departure.
hour	lower the downloading period to a certain time.

Value

data.frame

See Also

[htmlTreeParse](#)

Examples

```
## Not run:
flights <- getFlights()
head(flights)

## End(Not run)
```

movie	<i>function to generate a movie of flight data</i>
-------	--

Description

movie.routes is used for generating a movie of flight data

Usage

```
## S3 method for class 'routes'
movie(x, show.progress = TRUE, save = TRUE,
      width = 1024, height = 768, ...)
ffmpeg(dir, ffmpeg = "ffmpeg")
```

Arguments

x	data frame containing routes data
show.progress	if progress bar should be shown
save	if the generated pictures should be saved
width	width of the generated pictures
height	height of the generated pictures
dir	image directory (returned by movie)
ffmpeg	optionally, path to ffmpeg binary
...	NightDay = TRUE if the light-shadow boundary to be drawn in addition. citynames = TRUE if additional city names to be displayed.

Examples

```
myflights <- flights(from = "2011-01-07 11:55:00",
                    to = "2011-01-07 12:05:00")
myroutes <- routes(myflights)
movie(myroutes, save = FALSE, NightDay = TRUE)
```

routes	<i>function to calculate routes for airplanes</i>
--------	---

Description

routes is used for calculation the position matrix and routes for airplanes.

Usage

```
routes(flights.info, start.IATA = "MUC")
```

Arguments

`flights.info` data frame containing flight data
`start.IATA` select the base airport

Value

position matrix

Examples

```
flight.info <- flights(from = "2011-01-07", to = "2011-01-08")  
routes.data <- routes(flight.info)
```

Index

- *Topic **datasets**
 - airports, [2](#)
- *Topic **data**
 - getFlights, [4](#)
- *Topic **downloading**
 - getFlights, [4](#)
- *Topic **format**
 - flights, [3](#)
- *Topic **loading**
 - flights, [3](#)
- *Topic **movie**
 - movie, [5](#)
- *Topic **position matrix**
 - routes, [5](#)

airports, [2](#)

ffmpeg (movie), [5](#)

flights, [3](#)

getFlights, [4](#)

htmlTreeParse, [4](#)

movie, [5](#)

routes, [5](#)