

Package ‘gbfs’

September 10, 2019

Type Package

Title Interface with General Bikeshare Feed Specification Files

Version 1.2.0

Description Supplies a set of functions to interface with General Bikeshare Feed Specification .json feeds in R, allowing users to save and accumulate tidy .rds datasets for specified cities/bikeshare programs.

License CC0

Imports dplyr, readr, stringr, jsonlite, lubridate

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests knitr, rmarkdown

URL <https://github.com/ds-civic-data/gbfs>

BugReports <https://github.com/ds-civic-data/gbfs/issues>

NeedsCompilation no

Author Kaelyn Rosenberg [aut],
Simon P. Couch [aut, cre],
Mark Padgham [ctb]

Maintainer Simon P. Couch <simonpatrickcouch@gmail.com>

Repository CRAN

Date/Publication 2019-09-10 08:20:03 UTC

R topics documented:

gbfs	2
get_free_bike_status	2
get_gbfs	3
get_gbfs_cities	4
get_station_information	4
get_station_status	5

get_system_alerts	6
get_system_calendar	6
get_system_hours	7
get_system_information	8
get_system_pricing_plans	9
get_system_regions	10

Index	11
--------------	-----------

gbfs	<i>Package: gbfs</i>
------	----------------------

Description

General Bikeshare Feed Specification

Details

The ‘gbfs’ package supplies a set of functions to interface with General Bikeshare Feed Specification .json feeds in R, allowing users to save and accumulate tidy .rds datasets for specified cities/bikeshare programs.

gbfs Functions

get_gbfs(), get_free_bike_status(), get_station_information(), get_station_status(), get_system_alerts(), get_system_calendar(), get_system_hours(), get_system_information(), get_system_pricing_plans(), get_system_regions()

get_free_bike_status	<i>Grab the free_bike_status feed.</i>
----------------------	--

Description

If the specified file does not exist, get_free_bike_status saves the free_bike_status feed for a given city as a .rds object. If the specified file does exist, get_free_bike_status appends the current free_bike_status feed to the existing file. The resulting dataframe can alternatively be returned (rather than saved) using the ‘output’ argument. Go to ‘<https://github.com/NABSA/gbfs/blob/master/gbfs.md>’ to see metadata for this dataset.

Usage

```
get_free_bike_status(city, directory = NULL,
  file = "free_bike_status.rds", output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See <code>get_gbfs_cities</code> for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of an existing file or new file to be saved. Must end in <code>.rds</code> .
output	The type of output method. If <code>'output = "save"'</code> , the object will be saved as an <code>.rds</code> object at the given path. If <code>'output = "return"'</code> , the output will be returned as a dataframe object. Setting <code>'output = "both"'</code> will do both.

Value

Depends on argument to `'output'`: Either a saved `.rds` object generated from the current `station_information` feed, a dataframe object, or both.

Examples

```
get_free_bike_status(city = "Melbourne", directory = tempdir())
```

<code>get_gbfs</code>	<i>Save gbfs feeds.</i>
-----------------------	-------------------------

Description

`get_gbfs` checks for the existence of gbfs feeds for a given city and saves the feeds as `.rds` objects in a directory that can be specified by the user. Go to `'https://github.com/NABSA/gbfs/blob/master/gbfs.md'` to see metadata for each dataset.

Usage

```
get_gbfs(city, feeds = "all", directory)
```

Arguments

city	A character string or a url to an active gbfs.json feed. See <code>get_gbfs_cities</code> for a current list of available cities.
feeds	A character string specifying which feeds should be saved. Options are <code>"all"</code> , <code>"static"</code> , and <code>"dynamic"</code> .
directory	The name of an existing folder or folder to be created, where the feeds will be saved.

Value

A folder containing the specified feeds saved as `.rds` objects.

Examples

```
get_gbfs(city = "boise", directory = tempdir())
```

get_gbfs_cities	<i>Get table of all cities releasing GBFS feeds</i>
-----------------	---

Description

Get table of all cities releasing GBFS feeds

Usage

```
get_gbfs_cities()
```

Value

A data.frame of all cities issuing GBFS feeds. The ‘Auto-Discovery URL’ column supplies the relevant .json feeds, while the entries in the ‘URL’ column take the user to the public-facing webpage of the programs.

Source

North American Bikeshare Association, General Bikeshare Feed Specification <https://raw.githubusercontent.com/NABSA/gbfs/master/systems.csv>

get_station_information	<i>Grab the station_information feed.</i>
-------------------------	---

Description

get_station_information grabs and tidies the station_information feed for a given city. Go to ‘<https://github.com/NABSA/gbfs/blob/master/gbfs.md>’ to see metadata for this dataset.

Usage

```
get_station_information(city, directory = NULL,
  file = "station_information.rds", output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See get_gbfs_cities for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of the file to be saved (if output is default). Must end in .rds.
output	The type of output method. If ‘output = "save"’, the object will be saved as an .rds object at the given path. If ‘output = "return"’, the output will be returned as a dataframe object. Setting ‘output = "both"’ will do both.

Value

Depends on argument to ‘output’: Either a saved .rds object generated from the current station_information feed, a dataframe object, or both

Examples

```
get_station_information(city =
  "http://biketownpdx.socialbicycles.com/opendata/station_information.json", directory = tempdir())
get_station_information(city = "Battle Creek", directory = tempdir())
```

```
get_station_status      Grab the station_status feed.
```

Description

If the specified file does not exist, get_station_status saves the station_status feed for a given city as a .rds object. If the specified file does exist, get_station_status appends the current station_status feed to the existing file. Go to <https://github.com/NABSA/gbfs/blob/master/gbfs.md> to see metadata for this dataset.

Usage

```
get_station_status(city, directory = NULL, file = "station_status.rds",
  output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See get_gbfs_cities for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of an existing file or new file to be saved. Must end in .rds.
output	The type of output method. If ‘output = "save"’, the object will be saved as an .rds object at the given path. If ‘output = "return"’, the output will be returned as a dataframe object. Setting ‘output = "both"’ will do both.

Value

Depends on argument to ‘output’: Either a saved .rds object generated from the current station_information feed, a dataframe object, or both.

Examples

```
get_station_status(city =
  "http://biketownpdx.socialbicycles.com/opendata/station_status.json", directory = tempdir())
```

get_system_alerts *Grab the system_alerts feed.*

Description

get_system_alerts grabs and tidies the system_alerts feed for a given city. Go to <https://github.com/NABSA/gbfs/blob/master> to see metadata for this dataset.

Usage

```
get_system_alerts(city, directory = NULL, file = "system_alerts.rds",
  output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See get_gbfs_cities for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of the file to be saved. Must end in .rds.
output	The type of output method. If 'output = "save"', the object will be saved as an .rds object at the given path. If 'output = "return"', the output will be returned as a dataframe object. Setting 'output = "both"' will do both.

Value

Depends on argument to 'output': Either a saved .rds object generated from the current system_alerts feed, a dataframe object, or both.

Examples

```
get_system_alerts(city =
  "http://biketownpdx.socialbicycles.com/opendata/system_alerts.json", directory = tempdir())
get_system_alerts(city = "Santa Monica", directory = tempdir())
```

get_system_calendar *Grab the system_calendar feed.*

Description

get_system_calendar grabs and tidies the system_calendar feed for a given city. Go to <https://github.com/NABSA/gbfs/blob/master> to see metadata for this dataset.

Usage

```
get_system_calendar(city, directory = NULL,
  file = "system_calendar.rds", output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See <code>get_gbfs_cities</code> for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of the file to be saved. Must end in <code>.rds</code> .
output	The type of output method. If <code>'output = "save"'</code> , the object will be saved as an <code>.rds</code> object at the given path. If <code>'output = "return"'</code> , the output will be returned as a dataframe object. Setting <code>'output = "both"'</code> will do both.

Value

Depends on argument to `'output'`: Either a saved `.rds` object generated from the current `system_alerts` feed, a dataframe object, or both

Examples

```
get_system_calendar(city =
  "http://biketownpdx.socialbicycles.com/opendata/system_calendar.json", directory = tempdir())
get_system_calendar(city = "Tampa", directory = tempdir())
```

<code>get_system_hours</code>	<i>Grab the system_hours feed.</i>
-------------------------------	------------------------------------

Description

`get_system_hours` grabs and tidies the `system_hours` feed for a given city. Go to `'https://github.com/NABSA/gbfs/blob/master'` to see metadata for this dataset.

Usage

```
get_system_hours(city, directory = NULL, file = "system_hours.rds",
  output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See <code>get_gbfs_cities</code> for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of the file to be saved. Must end in <code>.rds</code> .

output The type of output method. If ‘output = "save"’, the object will be saved as an .rds object at the given path. If ‘output = "return"’, the output will be returned as a dataframe object. Setting ‘output = "both"’ will do both.

Value

Depends on argument to ‘output’: Either a saved .rds object generated from the current system_alerts feed, a dataframe object, or both.

Examples

```
get_system_hours(city =
  "http://biketownpdx.socialbicycles.com/opendata/system_hours.json", directory = tempdir())
get_system_hours(city = "Phoenix", directory = tempdir())
```

```
get_system_information
```

Grab the system_information feed.

Description

get_system_information grabs and tidies the system_information feed for a given city. Go to ‘<https://github.com/NABSA/gbfs/blob/master/gbfs.md>’ to see metadata for this dataset.

Usage

```
get_system_information(city, directory = NULL,
  file = "system_information.rds", output = "save")
```

Arguments

city A character string or a url to an active gbfs.json feed. See get_gbfs_cities for a current list of available cities.

directory The name of an existing folder or folder to be created, where the feed will be saved.

file The name of the file to be saved. Must end in .rds.

output The type of output method. If ‘output = "save"’, the object will be saved as an .rds object at the given path. If ‘output = "return"’, the output will be returned as a dataframe object. Setting ‘output = "both"’ will do both.

Value

Depends on argument to ‘output’: Either a saved .rds object generated from the current system_alerts feed, a dataframe object, or both.

Examples

```
get_system_information(city =  
"http://biketownpdx.socialbicycles.com/opendata/system_information.json", directory = tempdir())  
get_system_information(city = "Omaha", directory = tempdir())
```

get_system_pricing_plans

Grab the system_pricing_plans feed.

Description

get_system_pricing_plans grabs and tidies the system_pricing_plans feed for a given city. Go to <https://github.com/NABSA/gbfs/blob/master/gbfs.md> to see metadata for this dataset.

Usage

```
get_system_pricing_plans(city, directory = NULL,  
file = "system_pricing_plans.rds", output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See get_gbfs_cities for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of the file to be saved. Must end in .rds.
output	The type of output method. If 'output = "save"', the object will be saved as an .rds object at the given path. If 'output = "return"', the output will be returned as a dataframe object. Setting 'output = "both"' will do both.

Value

Depends on argument to 'output': Either a saved .rds object generated from the current system_alerts feed, a dataframe object, or both.

Examples

```
get_system_pricing_plans(city =  
"http://biketownpdx.socialbicycles.com/opendata/system_pricing_plans.json", directory = tempdir())  
get_system_pricing_plans(city = "Houston", directory = tempdir())
```

get_system_regions *Grab the system_regions feed.*

Description

get_system_regions grabs and tidies the system_regions feed for a given city. Go to <https://github.com/NABSA/gbfs/blob/> to see metadata for this dataset.

Usage

```
get_system_regions(city, directory = NULL, file = "system_regions.rds",  
  output = "save")
```

Arguments

city	A character string or a url to an active gbfs.json feed. See get_gbfs_cities for a current list of available cities.
directory	The name of an existing folder or folder to be created, where the feed will be saved.
file	The name of the file to be saved. Must end in .rds.
output	The type of output method. If 'output = "save"', the object will be saved as an .rds object at the given path. If 'output = "return"', the output will be returned as a dataframe object. Setting 'output = "both"' will do both.

Value

Depends on argument to 'output': Either a saved .rds object generated from the current system_alerts feed, a dataframe object, or both.

Examples

```
get_system_regions(city =  
  "http://biketownpdx.socialbicycles.com/opendata/system_regions.json", directory = tempdir())  
get_system_regions(city = "Boston", directory = tempdir())
```

Index

gbfs, [2](#)
gbfs-package (gbfs), [2](#)
get_free_bike_status, [2](#)
get_gbfs, [3](#)
get_gbfs_cities, [4](#)
get_station_information, [4](#)
get_station_status, [5](#)
get_system_alerts, [6](#)
get_system_calendar, [6](#)
get_system_hours, [7](#)
get_system_information, [8](#)
get_system_pricing_plans, [9](#)
get_system_regions, [10](#)