

Package ‘mlbgameday’

March 15, 2018

Title Tools to Gather Data from Major League Baseball Advanced Media (MLBAM)

Version 0.1.1

Description Multi-core processing of Gameday data from Major League Baseball Advanced Media <<http://gd2.mlb.com/components/game/mlb/>>. Additional tools to parallelize large data sets and write them to a database.

Depends R (>= 3.3.0)

Imports magrittr, xml2, dplyr, stringr, purrr, tidyr, utils, stats, foreach, iterators, parallel, doParallel, DBI

Suggests testthat, knitr, rmarkdown, pitchRx, viridis, ggplot2, plotly, RSQLite, dbplyr

License MIT + file LICENSE

URL <https://github.com/keberwein/mlbgameday>

BugReports <https://github.com/keberwein/mlbgameday/issues>

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

VignetteBuilder knitr

NeedsCompilation no

Author Kris Eberwein [aut, cre]

Maintainer Kris Eberwein <eberwein@knights.ucf.edu>

Repository CRAN

Date/Publication 2018-03-15 22:06:35 UTC

R topics documented:

game_urls	2
get_payload	2
mlbgameday	4
search_gids	4

Index**6**

game_urls	<i>Create urls from game_ids.</i>
-----------	-----------------------------------

Description

Create urls from game_ids.

Usage

```
game_urls(url_gids = NULL, dataset = NULL, ...)
```

Arguments

url_gids	A list of gid urls formatted by the make_gids() function.
dataset	The dataset to be scraped. The default is "inning_all." Other options include, "inning_hit", "players", "game", and "game_events".
...	additional arguments

Examples

```
gids <- make_gids(start = "2016-06-01", end = "2016-06-01", dataset="inning_all")
urls <- game_urls(gids)
```

get_payload	<i>Get Gameday data from MLBAM.</i>
-------------	-------------------------------------

Description

Get Gameday data from MLBAM.

Usage

```
get_payload(start = NULL, end = NULL, league = "mlb", dataset = NULL,
  game_ids = NULL, db_con = NULL, overwrite = FALSE, ...)
```

Arguments

<code>start</code>	A start date passed as a character in ISO 8601 format. "2017-05-01"
<code>end</code>	An end date passed as a character in ISO 8601 format. "2017-09-01"
<code>league</code>	The league to gather gids for. The default is "mlb". Other options include "aaa" and "aa"
<code>dataset</code>	The dataset to be scraped. The default is "inning_all." Other options include, "inning_hit", "linescore".
<code>game_ids</code>	A list of user-supplied gameIds.
<code>db_con</code>	A database connection from the DBI package.
<code>overwrite</code>	Logical. Should current database be overwritten? Inherited from the <code>dbWriteTable</code> function from the DBI package. The default value is FALSE.
<code>...</code>	additional arguments

Examples

```
## Not run:
# Make a request for a single day.
df <- get_payload(start = "2016-06-01", end = "2016-06-01")

# Run larger requests in parallel.
library(doParallel)
library(foreach)

no_cores <- detectCores() - 2
cl <- makeCluster(no_cores)
registerDoParallel(cl)

df <- get_payload(start = "2016-01-01", end = "2017-01-01")

stopImplicitCluster()
rm(cl)

## End(Not run)

# Supply your own custom vector of game ids.

mygids <- search_gids(team = "indians", start = "2016-05-01", end = "2016-05-01")

df <- get_payload(game_ids = mygids)
```

mlbgameday	<i>Tools to Extract Transform and Load Major League Baseball Advanced Media's 'Gameday' Data</i>
------------	--

Description

Tools to Extract Transform and Load Major League Baseball Advanced Media's 'Gameday' Data

Author(s)

Kris Eberwein

search_gids	<i>Search the internal game_id data set.</i>
-------------	--

Description

Search the internal game_id data set.

Usage

```
search_gids(team = NULL, start = NULL, end = NULL, venue = NULL,
            game_type = NULL, home_only = FALSE, away_only = FALSE, ...)
```

Arguments

team	A team name, or character vector of team names, without the city, as found in the team_ids data set.
start	A start date passed as a character in ISO 8601 format. "2017-05-01"
end	An end date passed as a character in ISO 8601 format. "2017-09-01"
venue	The stadium at which the game(s) were played, as found in the venue_ids data set.
game_type	The type of game(s), as found in the game_ids data set. Typical options include "r" (regular), "w" (world series), "1" (league playoffs), "d" (division playoffs), and "s" (spring training).
home_only	Logical. Collect only home games. The default is FALSE.
away_only	Logical. Collect only away games. The default is FALSE.
...	additional arguments

Examples

```
# Collect all of the regular season games for the 2016 Cleveland Indians.
game_ids <- search_gids(team = "indians", start = "2016-01-01", end = "2016-12-31", game_type = "r")
head(game_ids)

## Not run:
# Collect all games from the 2016 World Series
game_ids <- search_gids(start = "2016-10-25", end = "2016-11-02", game_type = "w")

# Collect all regular season games played at Wrigley Field since 2008.
game_ids <- search_gids(venue = "Wrigley Field")

# Fuzzy search results work too.
game_ids <- search_gids(venue = "Wrigley")

## End(Not run)
```

Index

`game_urls`, 2
`get_payload`, 2

`mlbgameday`, 4
`mlbgameday-package (mlbgameday)`, 4

`search_gids`, 4