Package ‘mlbgameday’

April 2, 2019

Title Tools to Gather Data from Major League Baseball Advanced Media

Version 0.2.0

Description Multi-core processing of data from Major League Baseball Advanced Media <http://gd2.mlb.com/components/game/mlb/>. Additional tools to parallel process 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, markdown, ggplot2, 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 2019-04-02 18:50:07 UTC

R topics documented:

  game_urls ......................................................... 2
  get_payload ...................................................... 2
  mlbgameday ....................................................... 3
  search_gids ...................................................... 4

Index 5
### game_urls

Create urls from game_ids.

**Description**

Create urls from game_ids.

**Usage**

```r
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**

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

Description

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

Author(s)

Kris Eberwein

---

game_ids  A list of user-supplied gameIds.
db_con    A database connection from the DBI package.
overwrite Logical. Should current database be overwritten? Inherited from the dbWriteTable function from the DBI package. The default value is FALSE.
...

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
c1 <- makeCluster(no_cores)
registerDoParallel(c1)

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

stopImplicitCluster()
rm(c1)

## 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)
search_gids | Search the internal game_id data set.

Description
Search the internal game_id data set.

Usage
```r
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), "l" (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
```r
# 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
# 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, 3
mlbgameday-package (mlbgameday), 3

search_gids, 4