Package ‘hoopR’

November 10, 2021

Title  Functions to Access Men’s Basketball Play by Play Data
Version  1.5.0
Description  A utility to quickly obtain clean and tidy men’s basketball play by play data. Provides functions to access live play by play and box score data from ESPN<https://www.espn.com> with shot locations when available. It is also a full NBA Stats API<https://www.nba.com/stats/> wrapper. It is also a scraping and aggregating interface for Ken Pomeroy's men's college basketball statistics website<https://kenpom.com>. It provides users with an active subscription the capability to scrape the website tables and analyze the data for themselves.

License  MIT + file LICENSE
URL  https://saiemgilani.github.io/hoopR/,
     https://github.com/saiemgilani/hoopR
BugReports  https://github.com/saiemgilani/hoopR/issues
SystemRequirements  pandoc (>= 1.12.3), pandoc-citeproc
Depends  R (>= 4.0.0)
Imports  cli (>= 1.1.0), data.table (>= 1.14.0), dplyr, furrr, future, glue, janitor, jsonlite, magrittr, progressr (>= 0.6.0), Rcpp (>= 1.0.7), RcppParallel (>= 5.1.4), rlang (>= 0.4.0), rvest (>= 1.0.0), stringr (>= 1.3.0), tidyr (>= 1.0.0), usethis (>= 1.6.0)
Suggests  crayon (>= 1.3.4), curl, DBI, ggplot2, ggrepel, httr (>= 0.5), purrr (>= 0.3.0), qs (>= 0.25.1), rmarkdown, RSQLite, stats, stringi, testthat, tibble (>= 3.0), xml2 (>= 1.3)
Encoding  UTF-8
LazyData  true
RoxygenNote  7.1.2
NeedsCompilation  no
Author  Saiem Gilani [aut, cre]
Maintainer  Saiem Gilani <saiem.gilani@gmail.com>
Repository  CRAN
Date/Publication  2021-11-10 13:30:05 UTC
R topics documented:

allplayers  ................................................................. 6
alltime  ........................................................................... 7
assists ............................................................................ 7
assist_tracker ................................................................. 8
bs_advv2 ......................................................................... 9
bs_defensive ................................................................. 10
bs_ffv2 ........................................................................... 10
bs_match ......................................................................... 10
bs_miscv2 ................................................................. 11
bs_pt_v2 .......................................................................... 11
bs_scoringv2 ............................................................... 13
bs_similarity ................................................................. 14
bs_summaryv2 ............................................................... 15
bs_tradv2 ................................................................. 16
bs_usagerv2 ................................................................. 17
check_status ................................................................. 18
clean_team_names_NCAA_merge ............................................. 18
commonplayerinfo ............................................................ 19
commonplayoffsers ............................................................ 19
commonteamroster ............................................................ 20
csv_from_url ................................................................. 21
cumestatsplayer ............................................................... 24
cumestatsplayergames ....................................................... 25
cumestatsteam ............................................................... 26
cumestatsteamgames .......................................................... 27
dboard ................................................................. 28
dcombine_anthro .............................................................. 29
dcombine_drill ................................................................. 29
dcombine_nsshooting ......................................................... 30
dcombine_ssshooting ......................................................... 30
dcombine_stats ............................................................... 31
defensehub ............................................................... 31
espn_mbb_betting .......................................................... 32
espn_mbb_conferences ..................................................... 33
espn_mbb_game_all ......................................................... 33
espn_mbb_pbp .............................................................. 34
espn_mbb_player_box ....................................................... 35
espn_mbb_rankings .......................................................... 35
espn_mbb_scoreboard ...................................................... 36
espn_mbb_standings ......................................................... 37
espn_mbb_teams ............................................................. 37
espn_mbb_team_box ......................................................... 38
espn_nba_betting ........................................................... 38
espn_nba_game_all ......................................................... 39
espn_nba_pbp .............................................................. 39
espn_nba_player_box ....................................................... 40
topics documented:

- espn_nba_scoreboard ................................................................. 41
- espn_nba_standings ................................................................. 41
- espn_nba_teams ...................................................................... 42
- espn_nba_team_box ................................................................. 42
- fantasywidget ........................................................................... 43
- franchisehistory ...................................................................... 44
- franchiseleaders ..................................................................... 45
- franchiseplayers ...................................................................... 46
- gl_bs_similarity ........................................................................ 46
- homepageleaders .................................................................. 47
- homepagev2 ............................................................................. 48
- hustle_bs ................................................................................. 49
- hustle_p .................................................................................... 50
- hustle_pl ................................................................................... 52
- hustle_t ..................................................................................... 54
- hustle_tl ................................................................................... 56
- kp_arenas ................................................................................. 58
- kp_box ...................................................................................... 58
- kp_coach_history ..................................................................... 59
- kp_conf ..................................................................................... 60
- kp_confhistory .......................................................................... 64
- kp_confstats ............................................................................ 65
- kp_efficiency ............................................................................ 67
- kp_fanmatch ............................................................................. 67
- kp_foul_trouble ......................................................................... 68
- kp_fourfactors ......................................................................... 69
- kp_gameplan ............................................................................. 70
- kp_game_attrs ........................................................................... 70
- kp_hca ...................................................................................... 71
- kp_height ................................................................................. 72
- kp_kpoy ..................................................................................... 72
- kp_minutes_matrix .................................................................... 73
- kp_officials .............................................................................. 73
- kp_opptracker ........................................................................... 74
- kp_playerstats .......................................................................... 75
- kp_player_career ...................................................................... 76
- kp_pointdist ............................................................................ 76
- kp_pomeroy_archive_ratings ....................................................... 77
- kp_pomeroy_ratings .................................................................. 78
- kp_program_ratings .................................................................. 79
- kp_referee ................................................................................ 80
- kp_teamstats ............................................................................. 81
- kp_team_depth_chart ............................................................... 81
- kp_team_history ......................................................................... 82
- kp_team_lineups ........................................................................ 84
- kp_team_players ........................................................................ 85
- kp_team_player_stats ............................................................... 85
- kp_team_schedule ...................................................................... 86
<table>
<thead>
<tr>
<th>R topics documented:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>kp_trends</td>
<td>86</td>
</tr>
<tr>
<td>kp_user_pw</td>
<td>87</td>
</tr>
<tr>
<td>kp_winprob</td>
<td>88</td>
</tr>
<tr>
<td>ld_opppptshot</td>
<td>89</td>
</tr>
<tr>
<td>ld_pbiostats</td>
<td>91</td>
</tr>
<tr>
<td>ld_pclutch</td>
<td>93</td>
</tr>
<tr>
<td>ld_ppptshot</td>
<td>96</td>
</tr>
<tr>
<td>ld_pshotloc</td>
<td>98</td>
</tr>
<tr>
<td>ld_pstats</td>
<td>101</td>
</tr>
<tr>
<td>ld_ptdefend</td>
<td>103</td>
</tr>
<tr>
<td>ld_ptstats</td>
<td>105</td>
</tr>
<tr>
<td>ld_ptteamdefend</td>
<td>107</td>
</tr>
<tr>
<td>ld_tclutch</td>
<td>109</td>
</tr>
<tr>
<td>ld_tptshot</td>
<td>111</td>
</tr>
<tr>
<td>ld_tshotloc</td>
<td>113</td>
</tr>
<tr>
<td>ld_tstats</td>
<td>115</td>
</tr>
<tr>
<td>leaderstiles</td>
<td>117</td>
</tr>
<tr>
<td>leaguedashlineups</td>
<td>118</td>
</tr>
<tr>
<td>leagueleaders</td>
<td>120</td>
</tr>
<tr>
<td>leagueupviz</td>
<td>121</td>
</tr>
<tr>
<td>leagueplayerondetails</td>
<td>123</td>
</tr>
<tr>
<td>leagueseasonmatchups</td>
<td>124</td>
</tr>
<tr>
<td>lg_streak</td>
<td>125</td>
</tr>
<tr>
<td>load_mbb_pbp</td>
<td>130</td>
</tr>
<tr>
<td>load_mbb_player_box</td>
<td>131</td>
</tr>
<tr>
<td>load_mbb_schedule</td>
<td>131</td>
</tr>
<tr>
<td>load_mbb_team_box</td>
<td>132</td>
</tr>
<tr>
<td>load_nba_pbp</td>
<td>133</td>
</tr>
<tr>
<td>load_nba_player_box</td>
<td>134</td>
</tr>
<tr>
<td>load_nba_schedule</td>
<td>135</td>
</tr>
<tr>
<td>load_nba_team_box</td>
<td>135</td>
</tr>
<tr>
<td>l_gamelog</td>
<td>136</td>
</tr>
<tr>
<td>l_standings</td>
<td>137</td>
</tr>
<tr>
<td>l_standingsv3</td>
<td>138</td>
</tr>
<tr>
<td>matchupsrollup</td>
<td>139</td>
</tr>
<tr>
<td>most_recent_mbb_season</td>
<td>140</td>
</tr>
<tr>
<td>most_recent_nba_season</td>
<td>140</td>
</tr>
<tr>
<td>nba_data_pbp</td>
<td>140</td>
</tr>
<tr>
<td>nba_stats_videodetails</td>
<td>141</td>
</tr>
<tr>
<td>nba_stats_videoevents</td>
<td>143</td>
</tr>
<tr>
<td>nba_teams</td>
<td>143</td>
</tr>
<tr>
<td>ncaa_mbb_NET_rankings</td>
<td>144</td>
</tr>
<tr>
<td>pbp</td>
<td>145</td>
</tr>
<tr>
<td>phyclutch</td>
<td>145</td>
</tr>
<tr>
<td>phygamesplits</td>
<td>147</td>
</tr>
<tr>
<td>phygeneralsplits</td>
<td>149</td>
</tr>
<tr>
<td>phylastgames</td>
<td>150</td>
</tr>
<tr>
<td>phyopponent</td>
<td>152</td>
</tr>
</tbody>
</table>
R topics documented:

- phshootingsplits ................................................. 154
- pbyteamperformance ........................................... 155
- pbyyearoveryear ................................................ 157
- pcareerbycollege ............................................. 159
- pcareerbycollege_ru .......................................... 159
- pfantasy ......................................................... 160
- pfantasy_bg ...................................................... 161
- pg_streak ......................................................... 162
- playerawards .................................................... 167
- playercareerstats ............................................. 167
- playercompare .................................................. 168
- playerprofilev2 ............................................... 170
- playersvplayer .................................................. 170
- po_picture ....................................................... 172
- pptpass .......................................................... 173
- pptreb ............................................................ 174
- pptshotdefend .................................................. 175
- pptshots .......................................................... 177
- progressively .................................................... 178
- p_est_metr ....................................................... 179
- p_gamealog ....................................................... 179
- p_gamelogs ....................................................... 180
- p_n_g ............................................................... 182
- rds_from_url .................................................... 182
- rejoin_schedules .............................................. 183
- sc ................................................................. 183
- schedule ........................................................ 185
- scoreboard ....................................................... 185
- scoreboardv2 .................................................... 186
- sc_lw ............................................................... 187
- thyclutch ......................................................... 187
- thygamesplits ................................................... 189
- thygeneralsplits ............................................... 191
- thylastngames .................................................. 192
- thyopponent ..................................................... 194
- thyshootingsplits ............................................. 196
- thteamperformance ............................................ 197
- thyearoveryear .................................................. 199
- teamdashlineups ............................................... 201
- teaminfo ........................................................ 202
- teams_links ...................................................... 203
- tg_streak ......................................................... 204
- thist_leaders .................................................... 214
- tp ................................................................. 214
- tp_onoffsummary ............................................... 216
- tp_onoff_det ..................................................... 218
- t_details .......................................................... 219
- t_est_metr ....................................................... 220
allplayers

Description

Get NBA Stats API All Players
Get NBA Stats API All Players

Usage

```r
nba_commonallplayers(
  is_only_current_season = 0,
  league_id = "00",
  season = "2020-21"
)
```

Arguments

- `is_only_current_season`
- `league_id`
- `season`

Value

Return a named list of data frames: CommonAllPlayers

Author(s)

Saiem Gilani
alltime

---

### alltime

Get NBA Stats API All-time Leaders Grid

---

**Description**

Get NBA Stats API All-time Leaders Grid

Get NBA Stats API All-time Leaders Grid

**Usage**

```r
nba_alltimeleadersgrids(
  league_id = "00",
  per_mode = "PerGame",
  season_type = "Regular Season",
  top_x = 10
)
```

**Arguments**

- `league_id`  
  League - default: '00'. Other options include '10': WNBA, '20': G-League
- `per_mode`  
  Per Mode - PerGame, Totals
- `season_type`  
  Season Type - Regular Season, Playoffs, All-Star
- `top_x`  
  Top X

**Value**

Returns a named list of data frames: ASTLeaders, BLKLeaders, DREBLeaders, FG3ALeaders, FG3MLeaders, FG3_PCTLeaders, FGALeaders, FGMLeaders, FG_PCTLeaders, FTALeaders, FTMLeaders, FT_PCTLeaders, GPLeaders, OREBLeaders, PFLLeaders, PTSLeaders, REBLeaders, STLLeaders, TOVLeaders

**Author(s)**

Saiem Gilani

---

### assists

Get NBA Stats API Assist Leaders

---

**Description**

Get NBA Stats API Assist Leaders

Get NBA Stats API Assist Leaders
Usage

```r
nba_assistleaders(
  league_id = "00",
  per_mode = "PerGame",
  player_or_team = "Team",
  season = "2020-21",
  season_type = "Regular Season"
)
```

Arguments

- **league_id**: League - default: '00'. Other options include '10': WNBA, '20': G-League
- **per_mode**: Per Mode - PerGame, Totals
- **player_or_team**: Player or Team
- **season**: Season - format 2020-21
- **season_type**: Season Type - Regular Season, Playoffs, All-Star

Value

Returns a named list of data frames: AssistLeaders

Author(s)

Saiem Gilani

---

**assist_tracker**

Get NBA Stats API Assist Tracker

Description

Get NBA Stats API Assist Tracker

Usage

```r
nba_assisttracker(
  league_id = "00",
  per_mode = "PerGame",
  season = "2020-21",
  season_type = "Regular Season"
)
```
bs_advv2

Arguments

- `league_id`: League - default: '00'. Other options include '10': WNBA, '20': G-League
- `per_mode`: Per Mode - PerGame, Totals
- `season`: Season - format 2020-21
- `season_type`: Season Type - Regular Season, Playoffs, All-Star

Value

Returns a named list of data frames: AssistTracker

Author(s)

Saeim Gilani

Description

Get NBA Stats API Boxscore Advanced V2

Usage

```r
nba_boxscoreadvancedv2(
  game_id,
  start_period = 0,
  end_period = 14,
  start_range = 0,
  end_range = 0,
  range_type = 0
)
```

Arguments

- `game_id`: Game ID
- `start_period`: start_period
- `end_period`: end_period
- `start_range`: start_range
- `end_range`: end_range
- `range_type`: range_type

Value

Returns a named list of data frames: PlayerStats, TeamStats
Author(s)
Saiem Gilani

**bs_defensive**

Get NBA Stats API Boxscore Defensive

**Description**
Get NBA Stats API Boxscore Defensive

**Usage**
nba_boxscoredefensive(game_id)

**Arguments**
game_id Game ID

**Value**
Returns a named list of data frames: PlayerDefensiveStats, Table1

Author(s)
Saiem Gilani

**bs_ffv2**

Get NBA Stats API Boxscore Four Factors V2

**Description**
Get NBA Stats API Boxscore Four Factors V2

**Usage**
nba_boxscorefourfactorsv2(
  game_id,
  start_period = 0,
  end_period = 14,
  start_range = 0,
  end_range = 0,
  range_type = 0
)

)
bs_match

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game ID</td>
</tr>
<tr>
<td>start_period</td>
<td>start_period</td>
</tr>
<tr>
<td>end_period</td>
<td>end_period</td>
</tr>
<tr>
<td>start_range</td>
<td>start_range</td>
</tr>
<tr>
<td>end_range</td>
<td>end_range</td>
</tr>
<tr>
<td>range_type</td>
<td>range_type</td>
</tr>
</tbody>
</table>

Value

Returns a named list of data frames: sqlPlayersFourFactors, sqlTeamFourFactors

Author(s)

Saiem Gilani
bs_miscv2 Get NBA Stats API Boxscore Misc V2

Description

Get NBA Stats API Boxscore Misc V2
Get NBA Stats API Boxscore Misc V2

Usage

```r
nba_boxscoremiscv2(
  game_id,
  start_period = 0,
  end_period = 14,
  start_range = 0,
  end_range = 0,
  range_type = 0
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game ID</td>
</tr>
<tr>
<td>start_period</td>
<td>start_period</td>
</tr>
<tr>
<td>end_period</td>
<td>end_period</td>
</tr>
<tr>
<td>start_range</td>
<td>start_range</td>
</tr>
<tr>
<td>end_range</td>
<td>end_range</td>
</tr>
<tr>
<td>range_type</td>
<td>range_type</td>
</tr>
</tbody>
</table>

Value

Returns a named list of data frames: sqlPlayersMisc, sqlTeamsMisc

Author(s)

Saiem Gilani
**bs_pt_v2**  
Get NBA Stats API Boxscore Player Tracking V2

**Description**

Get NBA Stats API Boxscore Player Tracking V2

**Usage**

```
nba_boxscoreplayertrackv2(game_id)
```

**Arguments**

- **game_id**  
  Game ID

**Value**

Returns a named list of data frames: PlayerStats, TeamStats

**Author(s)**

Saiem Gilani

---

**bs_scoringv2**  
Get NBA Stats API Boxscore Scoring V2

**Description**

Get NBA Stats API Boxscore Scoring V2

**Usage**

```
nba_boxscorescoringv2(
    game_id,
    start_period = 0,
    end_period = 14,
    start_range = 0,
    end_range = 0,
    range_type = 0
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game ID</td>
</tr>
<tr>
<td>start_period</td>
<td>start_period</td>
</tr>
<tr>
<td>end_period</td>
<td>end_period</td>
</tr>
<tr>
<td>start_range</td>
<td>start_range</td>
</tr>
<tr>
<td>end_range</td>
<td>end_range</td>
</tr>
<tr>
<td>range_type</td>
<td>range_type</td>
</tr>
</tbody>
</table>

Value

Returns a named list of data frames: sqlPlayersScoring, sqlTeamsScoring

Author(s)

Saiem Gilani

bs_similarity  Get NBA Stats API Boxscore Similarity Score

Description

Get NBA Stats API Boxscore Similarity Score

Usage

```r
nba_boxscoresimilarityscore(
  person_1_id,
  person_1_league_id,
  person_1_season,
  person_1_season_type,
  person_2_id,
  person_2_league_id,
  person_2_season,
  person_2_season_type
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>person_1_id</td>
<td>person_1_id</td>
</tr>
<tr>
<td>person_1_league_id</td>
<td>person_1_league_id</td>
</tr>
<tr>
<td>person_1_season</td>
<td>person_1_season</td>
</tr>
</tbody>
</table>
person_1_season_type

person_2_id

person_2_league_id

person_2_season

person_2_season_type

Value

Returns a named list of data frames: BoxScoreSimilarityScores

Author(s)

Saiem Gilani

---

bs_summaryv2  

Get NBA Stats API Boxscore Summary V2

Description

Get NBA Stats API Boxscore Summary V2
Get NBA Stats API Boxscore Summary V2

Usage

nba_boxscoresummaryv2(game_id)

Arguments

game_id  Game ID

Value

Returns a named list of data frames: AvailableVideo, GameInfo, GameSummary, InactivePlayers, LastMeeting, LineScore, Officials, OtherStats, SeasonSeries

Author(s)

Saiem Gilani
Get NBA Stats API Boxscore Traditional V2

Usage

```
nba_boxscoretraditionalv2(
    game_id,
    start_period = 0,
    end_period = 14,
    start_range = 0,
    end_range = 0,
    range_type = 0
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game ID</td>
</tr>
<tr>
<td>start_period</td>
<td>start_period</td>
</tr>
<tr>
<td>end_period</td>
<td>end_period</td>
</tr>
<tr>
<td>start_range</td>
<td>start_range</td>
</tr>
<tr>
<td>end_range</td>
<td>end_range</td>
</tr>
<tr>
<td>range_type</td>
<td>range_type</td>
</tr>
</tbody>
</table>

Value

A list of data frames: PlayerStats, TeamStarterBenchStats, TeamStats

Author(s)

Saiem Gilani
Get NBA Stats API Boxscore Usage V2

**Description**

Get NBA Stats API Boxscore Usage V2

Get NBA Stats API Boxscore Usage V2

**Usage**

```r
nba_boxscoreusagev2(
  game_id,
  start_period = 0,
  end_period = 14,
  start_range = 0,
  end_range = 0,
  range_type = 0
)
```

**Arguments**

- `game_id` Game ID
- `start_period` start_period
- `end_period` end_period
- `start_range` start_range
- `end_range` end_range
- `range_type` range_type

**Value**

Returns a named list of data frames: sqlPlayersUsage, sqlTeamsUsage

**Author(s)**

Saiem Gilani
check_status  

*Description*

Check Status function

*Usage*

```
check_status(res)
```

*Arguments*

- **res**: Response from API

---

clean_team_names_NCAA_merge

*Description*

Clean KenPom Data Frame Team Names to match NCAA Team Names for easier merging

*Usage*

```
clean_team_names_NCAA_merge(df)
```

*Arguments*

- **df**: KenPom dataframe
commonplayerinfo     Get NBA Stats API Player Info

Description
Get NBA Stats API Player Info
Get NBA Stats API Player Info

Usage
nba_commonplayerinfo(league_id = "00", player_id = "2544")

Arguments
league_id     league_id
player_id     player_id

Value
Return a named list of data frames: AvailableSeasons

Author(s)
Saiem Gilani

commonplayoffseries     Get NBA Stats API Playoff Series

Description
Get NBA Stats API Playoff Series
Get NBA Stats API Playoff Series

Usage
nba_commonplayoffseries(league_id = "00", season = "2020-21", series_id = ")

Arguments
league_id     league_id
season        season
series_id     series_id
Value

Return a named list of data frames: PlayoffSeries

Author(s)

Saiem Gilani

------

commonteamroster  Get NBA Stats API Team Roster

Description

Get NBA Stats API Team Roster
Get NBA Stats API Team Roster

Usage

nba_commonteamroster(
  league_id = "00",
  season = "2020-21",
  team_id = "1610612739"
)

Arguments

league_id  league_id
season     season
team_id    team_id

Value

Return a named list of data frames: Coaches, CommonTeamRoster

Author(s)

Saiem Gilani
Description

This is a thin wrapper on data.table::fread

Usage

\texttt{csv\_from\_url(...)}

Arguments

- \texttt{...} Arguments passed on to \texttt{data.table::fread}
  - \texttt{input} A single character string. The value is inspected and deferred to either
    - \texttt{file=} (if no `\n` present), \texttt{text=} (if at least one `\n` is present) or \texttt{cmd=} (if no `\n` is present, at least one space is present, and it isn’t a file name). Exactly one of \texttt{input=}, \texttt{file=}, \texttt{text=} or \texttt{cmd=} should be used in the same call.
  - \texttt{file} File name in working directory, path to file (passed through \texttt{path.expand} for convenience), or a URL starting http://, file://, etc. Compressed files with extension `.gz` and `.bz2` are supported if the R.utils package is installed.
  - \texttt{text} The input data itself as a character vector of one or more lines, for example as returned by \texttt{readLines()}
  - \texttt{cmd} A shell command that pre-processes the file; e.g. \texttt{fread(cmd=paste("grep",word,"filename"))}. See Details.
  - \texttt{sep} The separator between columns. Defaults to the character in the set \[,,\t,\mid,;:], that separates the sample of rows into the most number of lines with the same number of fields. Use `NULL` or `""` to specify no separator; i.e. each line a single character column like \texttt{base::readLines} does.
  - \texttt{sep2} The separator \texttt{within} columns. A \texttt{list} column will be returned where each cell is a vector of values. This is much faster using less working memory than \texttt{strsplit} afterwards or similar techniques. For each column \texttt{sep2} can be different and is the first character in the same set above \[,,\t,\mid,\]:, other than \texttt{sep}, that exists inside each field outside quoted regions in the sample. \texttt{NB: sep2 is not yet implemented.}
  - \texttt{nrows} The maximum number of rows to read. Unlike \texttt{read.table}, you do not need to set this to an estimate of the number of rows in the file for better speed because that is already automatically determined by \texttt{fread} almost instantly using the large sample of lines. \texttt{nrows=0} returns the column names and typed empty columns determined by the large sample; useful for a dry run of a large file or to quickly check format consistency of a set of files before starting to read any of them.
  - \texttt{header} Does the first data line contain column names? Defaults according to whether every non-empty field on the first data line is type character. If so, or \texttt{TRUE} is supplied, any empty column names are given a default name.
na.strings A character vector of strings which are to be interpreted as NA values. By default, "," for columns of all types, including type character is read as NA for consistency. "", is unambiguous and read as an empty string. To read ,NA, as NA, set na.strings="NA". To read ,, as blank string ",", set na.strings=NULL. When they occur in the file, the strings in na.strings should not appear quoted since that is how the string literal "," , "NA", is distinguished from ,NA,, for example, when na.strings="NA".

stringsAsFactors Convert all character columns to factors?

verbose Be chatty and report timings?

skip If 0 (default) start on the first line and from there finds the first row with a consistent number of columns. This automatically avoids irregular header information before the column names row. skip>0 means ignore the first skip rows manually. skip="string" searches for "string" in the file (e.g. a substring of the column names row) and starts on that line (inspired by read.xls in package gdata).

select A vector of column names or numbers to keep, drop the rest. select may specify types too in the same way as colClasses; i.e., a vector of colname=type pairs, or a list of type=col(s) pairs. In all forms of select, the order that the columns are specified determines the order of the columns in the result.

drop Vector of column names or numbers to drop, keep the rest.

colClasses As in utils::read.csv; i.e., an unnamed vector of types corresponding to the columns in the file, or a named vector specifying types for a subset of the columns by name. The default, NULL means types are inferred from the data in the file. Further, data.table supports a named list of vectors of column names or numbers where the list names are the class names; see examples. The list form makes it easier to set a batch of columns to be a particular class. When column numbers are used in the list form, they refer to the column number in the file not the column number after select or drop has been applied. If type coercion results in an error, introduces NAs, or would result in loss of accuracy, the coercion attempt is aborted for that column with warning and the column's type is left unchanged. If you really desire data loss (e.g. reading 3.14 as integer) you have to truncate such columns afterwards yourself explicitly so that this is clear to future readers of your code.

integer64 "integer64" (default) reads columns detected as containing integers larger than 2^31 as type bit64:::integer64. Alternatively, "double" | "numeric" reads as utils::read.csv does; i.e., possibly with loss of precision and if so silently. Or, "character".

dec The decimal separator as in utils::read.csv. If not "." (default) then usually ",". See details.

col.names A vector of optional names for the variables (columns). The default is to use the header column if present or detected, or if not "V" followed by the column number. This is applied after check.names and before key and index.

check.names default is FALSE. If TRUE then the names of the variables in the data.table are checked to ensure that they are syntactically valid variable
names. If necessary they are adjusted (by `make.names`) so that they are, and also to ensure that there are no duplicates.

*encoding* default is "unknown". Other possible options are "UTF-8" and "Latin-1". Note: it is not used to re-encode the input, rather enables handling of encoded strings in their native encoding.

*quote* By default ("\""), if a field starts with a double quote, fread handles embedded quotes robustly as explained under Details. If it fails, then another attempt is made to read the field *as is*, i.e., as if quotes are disabled. By setting `quote="", the field is always read as if quotes are disabled. It is not expected to ever need to pass anything other than \" to quote; i.e., to turn it off.

*strip.white* default is TRUE. Strips leading and trailing whitespaces of unquoted fields. If FALSE, only header trailing spaces are removed.

*fill* logical (default is FALSE). If TRUE then in case the rows have unequal length, blank fields are implicitly filled.

*blank.lines.skip* logical, default is FALSE. If TRUE blank lines in the input are ignored.

*key* Character vector of one or more column names which is passed to `setkey`. It may be a single comma separated string such as key="x,y,z", or a vector of names such as key=c("x","y","z"). Only valid when argument `data.table=TRUE`. Where applicable, this should refer to column names given in `col.names`.

*index* Character vector or list of character vectors of one or more column names which is passed to `setindexv`. As with key, comma-separated notation like index="x,y,z" is accepted for convenience. Only valid when argument `data.table=TRUE`. Where applicable, this should refer to column names given in `col.names`.

*showProgress* TRUE displays progress on the console if the ETA is greater than 3 seconds. It is produced in fread’s C code where the very nice (but R level) `txtProgressBar` and `tkProgressBar` are not easily available.

*data.table* TRUE returns a `data.table`. FALSE returns a `data.frame`. The default for this argument can be changed with `options(datatable.fread.datatable=FALSE)`.

*nThread* The number of threads to use. Experiment to see what works best for your data on your hardware.

*logical01* If TRUE a column containing only 0s and 1s will be read as logical, otherwise as integer.

*keepLeadingZeros* If TRUE a column containing numeric data with leading zeros will be read as character, otherwise leading zeros will be removed and converted to numeric.

*yaml* If TRUE, fread will attempt to parse (using `yaml.load`) the top of the input as YAML, and further to glean parameters relevant to improving the performance of fread on the data itself. The entire YAML section is returned as parsed into a list in the `yaml_metadata` attribute. See Details.

*autostart* Deprecated and ignored with warning. Please use `skip` instead.

*tmpdir* Directory to use as the `tmpdir` argument for any `tempfile` calls, e.g. when the input is a URL or a shell command. The default is `tempdir()`
which can be controlled by setting TMPDIR before starting the R session; see base::tempdir.

tz Relevant to datetime values which have no Z or UTC-offset at the end, i.e. unmarked datetime, as written by utils::write.csv. The default tz="UTC" reads unmarked datetime as UTC POSIXct efficiently. tz="" reads unmarked datetime as type character (slowly) so that as.POSIXct can interpret (slowly) the character datetimes in local timezone; e.g. by using "POSIXct" in colClasses=. Note that fwrite() by default writes datetime in UTC including the final Z and therefore fwrite's output will be read by fread consistently and quickly without needing to use tz= or colClasses=. If the TZ environment variable is set to "UTC" (or "" on non-Windows where unset vs "" is significant) then the R session’s timezone is already UTC and tz="" will result in unmarked datetimes being read as UTC POSIXct. For more information, please see the news items from v1.13.0 and v1.14.0.

Value

a dataframe as created by data.table::fread()

cumestatsplayer Get NBA Stats API Cumulative Player Stats

Description
Get NBA Stats API Cumulative Player Stats
Get NBA Stats API Cumulative Player Stats

Usage

nba_cumestatsplayer(
  game_ids = "0022000756",
  league_id = "00",
  player_id = "1629611",
  season = "2020-21",
  season_type = "Regular Season",
  team_id = ""
)

Arguments

  game_ids  league_id
  player_id season
  season_type team_id
Value

Return a named list of data frames: GameByGameStats, TotalPlayerStats

Author(s)

Saiem Gilani

---

**cumestatsplayergames**  Get NBA Stats API Cumulative Player Game Stats

---

Description

Get NBA Stats API Cumulative Player Game Stats

Usage

```r
nba_cumestatsplayergames(
  league_id = "00",
  location = "",
  outcome = "",
  player_id = "2544",
  season = "2020-21",
  season_type = "Regular Season",
  vs_conference = "",
  vs_division = "",
  vs_team_id = ""
)
```

Arguments

- `league_id`  league_id
- `location`  location
- `outcome`  outcome
- `player_id`  player_id
- `season`  season
- `season_type`  season_type
- `vs_conference`  vs_conference
- `vs_division`  vs_division
- `vs_team_id`  vs_team_id

Value

Returns a named list of data frames: CumeStatsPlayerGames
Author(s)
Saiem Gilani

Get NBA Stats API Cumulative Team Stats

Description
Get NBA Stats API Cumulative Team Stats

Usage
nba_cumestatsteam(
  game_ids = "0022000756",
  league_id = "00",
  season = "2020-21",
  season_type = "Regular Season",
  team_id = ""
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_ids</td>
<td>game_ids</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
</tbody>
</table>

Value
Returns a named list of data frames: GameByGameStats, TotalTeamStats

Author(s)
Saiem Gilani
Description

Get NBA Stats API Cumulative Team Game Stats

Usage

```r
nba_cumestatsteamgames(
  league_id = "00",
  location = "",
  outcome = "",
  season = "2020-21",
  season_id = "",
  season_type = "Regular Season",
  team_id = 1610612739,
  vs_conference = "",
  vs_division = "",
  vs_team_id = ""
)
```

Arguments

- `league_id`  
- `location`  
- `outcome`  
- `season`  
- `season_id`  
- `season_type`  
- `team_id`  
- `vs_conference`  
- `vs_division`  
- `vs_team_id`  

Value

Returns a named list of data frames: CumeStatsTeamGames

Author(s)

Saiem Gilani
Get NBA Stats API Draft Board

Usage

```r
nba_draftboard(
  league_id = "00",
  college = "",
  overall_pick = "",
  round_pick = "",
  round_num = "",
  season = "2019",
  team_id = "",
  top_x = ""
)
```

Arguments

- `league_id`  
- `college`  
- `overall_pick`  
- `round_pick`  
- `round_num`  
- `season`  
- `team_id`  
- `top_x`

Value

Returns a named list of data frames: DraftBoard

Author(s)

Saiem Gilani
**dcombine_anthro**

Get NBA Stats API Draft Combine Player Anthropological Measurements

**Description**

Get NBA Stats API Draft Combine Player Anthropological Measurements

**Usage**

```r
nba_draftcombineplayeranthro(league_id = "00", season_year = "2020")
```

**Arguments**

- `league_id`  
  - league_id
- `season_year`  
  - season_year

**Value**

Returns a named list of data frames: Results

**Author(s)**

Saiem Gilani

---

**dcombine_drill**

Get NBA Stats API Draft Combine Drill Results

**Description**

Get NBA Stats API Draft Combine Drill Results

**Usage**

```r
nba_draftcombinedrillresults(league_id = "00", season_year = "2020")
```

**Arguments**

- `league_id`  
  - league_id
- `season_year`  
  - season_year

**Value**

Returns a named list of data frames: Results
**Description**

Get NBA Stats API Draft Combine Non-Stationary Shooting
Get NBA Stats API Draft Combine Non-Stationary Shooting

**Usage**

```
nba_draftcombinenonstationaryshooting(league_id = "00", season_year = "2020")
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>season_year</td>
<td>season_year</td>
</tr>
</tbody>
</table>

**Value**

Returns a named list of data frames: Results

**Author(s)**

Saiem Gilani

---

**Description**

Get NBA Stats API Draft Combine - Spot Shooting
Get NBA Stats API Draft Combine - Spot Shooting

**Usage**

```
nba_draftcombinespotshooting(league_id = "00", season_year = "2020")
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>season_year</td>
<td>season_year</td>
</tr>
</tbody>
</table>

**Author(s)**

Saiem Gilani
dcombine_stats

Value
Returns a named list of data frames: Results

Author(s)
Saiem Gilani

dcombine_stats  Get NBA Stats API Draft Combine Stats

Description
Get NBA Stats API Draft Combine Stats
Get NBA Stats API Draft Combine Stats

Usage
nba_draftcombinestats(league_id = "00", season_year = "2020")

Arguments
league_id  league_id
season_year  season_year

Value
Returns a named list of data frames: DraftCombineStats

Author(s)
Saiem Gilani

defensehub  Get NBA Stats API Defense Hub

Description
Get NBA Stats API Defense Hub
Get NBA Stats API Defense Hub
Usage

nba_defensehub(
    league_id = "00",
    game_scope = "Season",
    player_or_team = "Team",
    player_scope = "All Players",
    season = "2020-21",
    season_type = "Regular Season"
)

Arguments

league_id League - default: '00'. Other options include '10': WNBA, '20': G-League

game_scope Game Scope - Season, Last 10, ,Yesterday, Finals

player_or_team Player or Team

player_scope Player Scope - All Players, Rookies

season Season - format 2020-21

season_type Season Type - Regular Season, Playoffs

Value


Author(s)

Saiem Gilani

---

espn_mbb_betting

Get ESPN MBB’s Betting information

Description

Get ESPN MBB’s Betting information

Usage

espn_mbb_betting(game_id)

Arguments

game_id Game ID

Value

Returns a named list of data frames: pickcenter, againstTheSpread, predictor_df
**espn_mbb_conferences**

*Get ESPN conference names and ids*

**Description**
Get ESPN conference names and ids

**Usage**
espn_mbb_conferences()

**Value**
A conferences data frame

**Author(s)**
Saiem Gilani

**Examples**

try(espn_mbb_conferences())

---

**espn_mbb_game_all**

*Get ESPN men’s college basketball data (Pbp, Team and Player Box)*

**Description**
Get ESPN men’s college basketball data (Pbp, Team and Player Box)

**Usage**
espn_mbb_game_all(game_id)

**Arguments**

<table>
<thead>
<tr>
<th>game_id</th>
<th>Game ID</th>
</tr>
</thead>
</table>

---

try(espn_mbb_betting(game_id = 401256760))
Value
A named list of data frames: Plays, Team, Player

Author(s)
Saiem Gilani

Examples

```r
try(espn_mbb_game_all(game_id = 401256760))
```

---

espn_mbb_pbp | Get ESPN men's college basketball PBP data

Description
Get ESPN men's college basketball PBP data

Usage
```r
espn_mbb_pbp(game_id)
```

Arguments

game_id | Game ID

Value
A play-by-play data frame.

Author(s)
Saiem Gilani

Examples

```r
try(espn_mbb_pbp(game_id = 401256760))
```
espn_mbb_player_box

Get ESPN men's college basketball player box scores

Description
Get ESPN men’s college basketball player box scores

Usage
espn_mbb_player_box(game_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game ID</td>
</tr>
</tbody>
</table>

Value
A player boxscore data frame

Author(s)
Saiem Gilani

Examples

try(espn_mbb_player_box(game_id = 401256760))

espn_mbb_rankings
Get men's college basketball AP and Coaches Poll rankings from ESPN

Description
Get men’s college basketball AP and Coaches Poll rankings from ESPN

Usage
espn_mbb_rankings()

Value
Returns a tibble
Author(s)
Saiem Gilani

Examples

# Get current AP and Coaches Poll rankings
try(espn_mbb_rankings())

Description
Get ESPN men’s college basketball schedule for a specific year

Usage
espn_mbb_scoreboard(season)

Arguments
season Either numeric or character

Value
Returns a tibble

Examples

# Get schedule from date 2021-02-15
try(espn_mbb_scoreboard (season = "20210215"))
espn_mbb_standings

Get ESPN men’s college basketball standings

Description
Get ESPN men’s college basketball standings

Usage
espn_mbb_standings(year)

Arguments
year Either numeric or character (YYYY)

Value
A standings data frame

Examples
try(espn_mbb_standings(2021))

espn_mbb_teams
Get ESPN men’s college basketball team names and ids

Description
Get ESPN men’s college basketball team names and ids

Usage
espn_mbb_teams()

Value
A teams data frame

Author(s)
Saiem Gilani

Examples
try(espn_mbb_teams())
espn_mbb_team_box  Get ESPN men’s college basketball team box scores

Description
Get ESPN men’s college basketball team box scores

Usage
espn_mbb_team_box(game_id)

Arguments

game_id  Game ID

Value
A team boxscore data frame

Author(s)
Saiem Gilani

Examples

try(espn_mbb_team_box(game_id = 401256760))

espn_nba_betting  Get ESPN NBA’s Betting information

Description
Get ESPN NBA’s Betting information

Usage
espn_nba_betting(game_id)

Arguments

game_id  Game ID

Value
Returns a named list of data frames: pickcenter, againstTheSpread, predictor_df
espn_nba_game_all

Examples

try(espn_nba_betting(game_id = 401283399))

Description

Get ESPN NBA game data (Pbp, Team and Player Box)

Usage

espn_nba_game_all(game_id)

Arguments

game_id Game ID

Value

A named list of data frames: Plays, Team, Player

Author(s)

Saiem Gilani

Examples

try(espn_nba_game_all(game_id = 401283399))

espn_nba_pbp

Get ESPN NBA PBP data

Description

Get ESPN NBA PBP data

Usage

espn_nba_pbp(game_id)
Arguments

game_id       Game ID

Value

A play-by-play data frame.

Author(s)

Saiem Gilani

Examples

try(espn_nba_pbp(game_id = 401283399))

---

espn_nba_player_box

Get ESPN NBA player box scores

Description

Get ESPN NBA player box scores

Usage

espn_nba_player_box(game_id)

Arguments

game_id       Game ID

Value

A player boxscore data frame

Author(s)

Saiem Gilani

Examples

try(espn_nba_player_box(game_id = 401283399))
espn_nba_scoreboard  

Get ESPN men’s NBA schedule for a specific year

Description

Get ESPN men’s NBA schedule for a specific year

Usage

espn_nba_scoreboard(season)

Arguments

season  Either numeric or character (YYYYMMDD)

Value

Returns a tibble

Examples

# Get schedule from date 2021-02-15 (returns 1000 results, max allowable.)
try(espn_nba_scoreboard (season = "20210215"))

espn_nba_standings  

Get ESPN NBA’s Standings

Description

Get ESPN NBA’s Standings

Usage

espn_nba_standings(year)

Arguments

year  Either numeric or character (YYYY)

Value

A standings data frame
Examples

try(espn_nba_standings(year = 2021))

---

espn_nba_teams

*Get ESPN NBA team names and ids*

**Description**

Get ESPN NBA team names and ids

**Usage**

espn_nba_teams()

**Value**

A teams data frame

**Author(s)**

Saiem Gilani

**Examples**

try(espn_nba_teams())

---

espn_nba_team_box

*Get ESPN NBA team box scores*

**Description**

Get ESPN NBA team box scores

**Usage**

espn_nba_team_box(game_id)

**Arguments**

<table>
<thead>
<tr>
<th>game_id</th>
<th>Game ID</th>
</tr>
</thead>
</table>


Value
A team boxscore data frame

Author(s)
Saiem Gilani

Examples

try(espn_nba_team_box(game_id = 401283399))
Arguments

- active_players
- date_from
- date_to
- last_n_games
- league_id
- location
- month
- opponent_team_id
- po_round
- player_id
- position
- season
- season_segment
- season_type
- team_id
- todays_opponent
- todays_players
- vs_conference
- vs_division

Value

Returns a named list of data frames: FantasyWidgetResult

Author(s)

Saiem Gilani

---

franchisehistory Get NBA Stats API Franchise History

Description

Get NBA Stats API Franchise History

Usage

nba_franchisehistory(league_id = "00")
Arguments

league_id  league_id

Value

Returns a named list of data frames: DefunctTeams, FranchiseHistory

Author(s)

Saiem Gilani

---

franchiseleaders  Get NBA Stats API Franchise Leaders

Description

Get NBA Stats API Franchise Leaders

Get NBA Stats API Franchise Leaders

Usage

nba_franchiseleaders(league_id = "00", team_id = "1610612739")

Arguments

league_id  league_id

team_id  team_id

Value

Returns a named list of data frames: FranchiseLeaders

Author(s)

Saiem Gilani
franchiseplayers Get NBA Stats API Franchise Players

Description
Get NBA Stats API Franchise Players

Usage
nba_franchiseplayers(
  league_id = "00",
  per_mode = "Totals",
  season_type = "Regular Season",
  team_id = "1610612739"
)

Arguments
  league_id  league_id
  per_mode   per_mode
  season_type season_type
  team_id    team_id

Value
Returns a named list of data frames: FranchisePlayers

Author(s)
Saiem Gilani

gl_bs_similarity Get NBA Stats API G-League Alum Boxscore Similarity Score

Description
Get NBA Stats API G-League Alum Boxscore Similarity Score

Get NBA Stats API G-League Alum Boxscore Similarity Score
Usage

```r
nba_glaumboxscoresimilarityscore(
  person_1_id,
  person_1_league_id,
  person_1_season,
  person_1_season_type,
  person_2_id,
  person_2_league_id,
  person_2_season,
  person_2_season_type
)
```

Arguments

- `person_1_id`
- `person_1_league_id`
- `person_1_season`
- `person_1_season_type`
- `person_2_id`
- `person_2_league_id`
- `person_2_season`
- `person_2_season_type`

Value

Returns a named list of data frames: `GLeagueAlumBoxScoreSimilarityScores`

Author(s)

Saiem Gilani

Description

Get NBA Stats API Homepage Leaders
Get NBA Stats API Homepage Leaders
Usage

nba_homepageleaders(
    league_id = "00",
    game_scope = "Season",
    player_or_team = "Team",
    player_scope = "All Players",
    season = "2020-21",
    season_type = "Regular Season",
    stat_category = "Points"
)

Arguments

league_id League - default: '00'. Other options include '10': WNBA, '20': G-League

game_scope Game Scope - Season, Last 10, Yesterday, Finals

player_or_team Player or Team

player_scope Player Scope - All Players, Rookies

season Season - format 2020-21

season_type Season Type - Regular Season, Playoffs

stat_category Stat Category: Points, Rebounds, Assists, Defense, Clutch, Playmaking, Efficiency, Fast Break, Scoring Breakdown

Value

Returns a named list of data frames: HomePageLeaders, LeagueAverage, LeagueMax

Author(s)

Saiem Gilani

Get NBA Stats API HomepageV2 Leaders

Description

Get NBA Stats API HomepageV2 Leaders
Get NBA Stats API HomepageV2 Leaders
Usage

```r
nba_homepagev2(
  league_id = "00",
  game_scope = "Season",
  player_or_team = "Team",
  player_scope = "All Players",
  season = "2020-21",
  season_type = "Regular Season",
  stat_type = "Traditional"
)
```

Arguments

- **league_id**: League - default: '00'. Other options include '10': WNBA, '20': G-League
- **game_scope**: Game Scope - Season, Last 10, Yesterday, Finals
- **player_or_team**: Player or Team
- **player_scope**: Player Scope - All Players, Rookies
- **season**: Season - format 2020-21
- **season_type**: Season Type - Regular Season, Playoffs
- **stat_type**: Stat Type - Traditional, Advanced, Tracking

Value


Author(s)

Saiem Gilani

---

**hustle_bs**  Get NBA Stats API Hustle Stats Boxscore

Description

Get NBA Stats API Hustle Stats Boxscore

**Usage**

```r
nba_hustlestatsboxscore(game_id)
```

**Arguments**

- **game_id**: Game ID
Value

Returns a named list of data frames: HustleStatsAvailable, PlayerStats, TeamStats

Author(s)

Saiem Gilani

---

**hustle_p**  
Get NBA Stats API League Hustle Stats Player

Description

Get NBA Stats API League Hustle Stats Player

Usage

```r
nba_leaguehustlestatsplayer(
  college = "",
  conference = "",
  country = "",
  date_from = "",
  date_to = "",
  division = "",
  draft_pick = "",
  draft_year = "",
  height = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  per_mode = "Totals",
  player_experience = "",
  player_position = "",
  season = "2020-21",
  season_segment = "",
  season_type = "Regular Season",
  team_id = "",
  vs_conference = "",
  vs_division = "",
  weight = ""
)
```
Arguments

- college
- conference
- country
- date_from
- date_to
- division
- draft_pick
- draft_year
- height
- last_n_games
- league_id
- location
- month
- opponent_team_id
- outcome
- po_round
- per_mode
- player_experience
- player_position
- season
- season_segment
- season_type
- team_id
- vs_conference
- vs_division
- weight

Value

Returns a named list of data frames: HustleStatsPlayer

Author(s)

Saiem Gilani
Get NBA Stats API League Hustle Stats Player Leaders

Usage

nba_leaguehustlestatsplayerleaders(
    college = "",
    conference = "",
    country = "",
    date_from = "",
    date_to = "",
    division = "",
    draft_pick = "",
    draft_year = "",
    height = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    per_mode = "Totals",
    player_experience = "",
    player_position = "",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    team_id = "",
    vs_conference = "",
    vs_division = "",
    weight = ""
)

Arguments

college college
conference conference
country country
date_from date_from
date_to date_to
division division
draft_pick draft_pick
draft_year draft_year
height height
last_n_games last_n_games
league_id league_id
location location
month month
opponent_team_id opponent_team_id
outcome outcome
po_round po_round
per_mode per_mode
player_experience player_experience
player_position player_position
season season
season_segment season_segment
season_type season_type
team_id team_id
vs_conference vs_conference
vs_division vs_division
weight weight

Value

Returns a named list of data frames: PlayerChargesDrawnLeaders, PlayerContestedShotsLeaders, PlayerDeflectionsLeaders, PlayerLooseBallLeaders, PlayerScreenAssistLeaders, Table5

Author(s)

Saiem Gilani
hustle_t

Get NBA Stats API League Hustle Stats Team

Description

Get NBA Stats API League Hustle Stats Team

Get NBA Stats API League Hustle Stats Team

Usage

```python
nba_leaguehustlestatsteam(
    college = "",
    conference = "",
    country = "",
    date_from = "",
    date_to = "",
    division = "",
    draft_pick = "",
    draft_year = "",
    height = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    per_mode = "Totals",
    player_experience = "",
    player_position = "",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    team_id = "",
    vs_conference = "",
    vs_division = "",
    weight = ""
)
```

Arguments

<table>
<thead>
<tr>
<th>college</th>
<th>college</th>
</tr>
</thead>
<tbody>
<tr>
<td>conference</td>
<td>conference</td>
</tr>
<tr>
<td>country</td>
<td>country</td>
</tr>
<tr>
<td>date_from</td>
<td>date_from</td>
</tr>
</tbody>
</table>
### hustle_t

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>date_to</td>
<td>date_to</td>
</tr>
<tr>
<td>division</td>
<td>division</td>
</tr>
<tr>
<td>draft_pick</td>
<td>draft_pick</td>
</tr>
<tr>
<td>draft_year</td>
<td>draft_year</td>
</tr>
<tr>
<td>height</td>
<td>height</td>
</tr>
<tr>
<td>last_n_games</td>
<td>last_n_games</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>month</td>
<td>month</td>
</tr>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>player_experience</td>
<td>player_experience</td>
</tr>
<tr>
<td>player_position</td>
<td>player_position</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
<tr>
<td>weight</td>
<td>weight</td>
</tr>
</tbody>
</table>

### Value

Returns a named list of data frames: HustleStatsTeam

### Author(s)

Saiem Gilani
hustle_tl

Get NBA Stats API League Hustle Stats Team Leaders

Description

Get NBA Stats API League Hustle Stats Team Leaders

Usage

nba_leaguehustlestatsteamleaders(
    college = "",
    conference = "",
    country = "",
    date_from = "",
    date_to = "",
    division = "",
    draft_pick = "",
    draft_year = "",
    height = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    per_mode = "Totals",
    player_experience = "",
    player_position = "",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    team_id = "",
    vs_conference = "",
    vs_division = "",
    weight = ""
)

Arguments

<table>
<thead>
<tr>
<th>college</th>
<th>college</th>
</tr>
</thead>
<tbody>
<tr>
<td>conference</td>
<td>conference</td>
</tr>
<tr>
<td>country</td>
<td>country</td>
</tr>
<tr>
<td>date_from</td>
<td>date_from</td>
</tr>
</tbody>
</table>
Value

Returns a named list of data frames: Table5, Table6, TeamChargesDrawnLeaders, TeamContested-ShotsLeaders, TeamDeflectionsLeaders, TeamLooseBallLeaders, TeamScreenAssistLeaders

Author(s)

Saiem Gilani
kp_arenas

Get Home Court Arenas

Description
Get Home Court Arenas

Usage
kp_arenas(year = most_recent_mbb_season())

Arguments
year Year of data to pull

Value
A data frame with 6 columns:
Rk double.
Team character.
Conf character.
Arena character.
Alternate character.
Year double.

Examples
try(kp_arenas(year=2021))

kp_box

Get Game Box Score

Description
Get Game Box Score

Usage
cpy_box(game_id, year)
**kp_coach_history**

**Arguments**
- `game_id` Game id of game to pull
- `year` Year of game to pull

**Value**
Returns a tibble of game box scores

**Examples**
```
try(kp_box(game_id = 6, year = 2021))
```

**Description**
Get KenPom’s coaching resume data

**Usage**
```
kp_coach_history(coach)
```

**Arguments**
- `coach` Coach filter to select.

**Value**
A data frame with 30 columns:
- `Year` double.
- `Team.Rk` double.
- `Team` character.
- `Coach` character.
- `Conf` character.
- `W-L` character.
- `AdjT` double.
- `AdjO` double.
- `AdjD` double.
- `Off.eFG.Pct` double.
- `Off.TO.Pct` double.
Off.OR.Pct double.
Off.FTRate double.
Off.FG_2.Pct double.
Off.FT.Pct double.
Off.FG_3A.Pct double.
Off.A.Pct double.
Off.APL double.
Def.eFG.Pct double.
Def.TO.Pct double.
Def.OR.Pct double.
Def.FTRate double.
Def.FG_2.Pct double.
Def.FG_3.Pct double.
Def.FG_3A.Pct double.
Def.A.Pct double.
Def.AP double.
Foul2Partic.Pct double.

Examples

try(kp_coach_history(coach = 'Leonard Hamilton'))

---

date  

kp_conf  

Get KenPom's conference-wide stats

Description

Get KenPom's conference-wide stats

Usage

kp_conf(year, conf)
Arguments

**year**
Year (YYYY)

**conf**
If you try to use a conference that doesn’t exist for a given season, like 'IND' and '2018', you’ll get an empty table, as kenpom.com doesn’t serve 404 pages for invalid table queries like that.
No filter applied by default.

Value

A list of 7 dataframes accessible via `[[1]], [[2]], ..., [[7]]`
- **First data frame, accessible via `[[1]]`**
  A data frame with 15 columns:
  - Team character.
  - Overall character.
  - Conf character.
  - AdjEM double.
  - AdjEM.Rk double.
  - AdjO double.
  - AdjO.Rk double.
  - AdjD double.
  - AdjD.Rk double.
  - AdjT double.
  - AdjT.Rk double.
  - ConfSOS double.
  - ConfSOS.Rk double.
  - NextGame character.
  - Year character.

- **Second data frame, accessible via `[[2]]`**
  A data frame with 20 columns:
  - Team character.
  - OE double.
  - OE.Rk double.
  - eFG.Pct double.
A data frame with 20 columns:

Team character.
DE double.
DE.Rk double.
eFG.Pct double.
eFG.Pct.Rk double.
TO.Pct double.
TO.Pct.Rk double.
OR.Pct double.
OR.Pct.Rk double.
FTR double.
FTR.Rk double.
FG2.Pct double.
FG3.Pct double.
FT.Pct double.
FT.Pct.Rk double.
Tempo double.
Tempo.Rk double.
Year character.

Third data frame, accessible via \[[3]\]

Third data frame, accessible via [[3]]
kp_conf

Year character.

Fourth data frame, accessible via [[4]]
A data frame with 3 columns:

Rk integer.
Player character.
Year character.

Fifth data frame, accessible via [[5]]
A data frame with 4 columns:

Stat character.
Value double.
Rk double.
Year character.

Sixth data frame, accessible via [[6]]
A data frame with 5 columns:

Stat character.
Count character.
Value double.
Rk double.
Year character.

Seventh data frame, accessible via [[7]]
A data frame 4 columns:

Rk double.
Conference character.
Rating double.
Year character.

Examples

try(kp_conf(year = 2020, conf = 'ACC'))
kp_confhistory

Get KenPom’s historical conference ratings

Description

Get KenPom’s historical conference ratings

Usage

kp_confhistory(conf)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>conf</td>
<td>Used to limit to players in a specific conference. Allowed values are: 'A10', 'ACC', 'AE', 'AMER', 'ASUN', 'B10', 'B12', 'BE', 'BSKY', 'BSTH', 'BW', 'CAA', 'CUSA', 'HORIZ', 'IND', 'IVY', 'MAAC', 'MAC', 'MEAC', 'MVC', 'MWC', 'NEC', 'OVC', 'P12', 'PAT', 'SB', 'SC', 'SEC', 'SLND', 'SUM', 'SWAC', 'WAC', 'WCC'. If you try to use a conference that doesn’t exist for a given season, like 'IND' and '2018', you’ll get an empty table, as kenpom.com doesn’t serve 404 pages for invalid table queries like that. No filter applied by default.</td>
</tr>
</tbody>
</table>

Value

A data frame with 23 columns:

- Year - integer
- Rank - character
- Tempo - double
- Efficiency - double
- eFG.Pct - double
- TO.Pct - double
- OR.Pct - double
- FTR - double
- FG2.Pct - double
- FG3.Pct - double
- FT.Pct - double
- FG3A.Pct - double
- A.Pct - double
- Blk.Pct - double
- Stl.Pct - double
- HomeRecord - character
kp_confstats

- Bids- character
- S16- character
- F4- character
- CH- character
- RegSeasonChamp- character
- TourneyChamp- character
- BestTeam- character

Examples

```
try(kp_confhistory(conf = 'ACC'))
```

---

kp_confstats | Get KenPom’s conference comparison stats

### Description

Get KenPom’s conference comparison stats

### Usage

```
kp_confstats(year = most_recent_mbb_season())
```

### Arguments

- `year` Year (YYYY)

### Value

A data frame with 35 columns:

- Conf character.
- Eff double.
- Eff.Rk double.
- Tempo double.
- Tempo.Rk double.
- eFG.Pct double.
- eFG.Pct.Rk double.
- TO.Pct double.
- TO.Pct.Rk double.
- OR.Pct double.
• OR.Pct.Rkdouble.
• FTRatedouble.
• FTRate.Rkdouble.
• Blk.Pctdouble.
• Stl.Pctdouble.
• FG_2.Pctdouble.
• FG_3.Pctdouble.
• FT.Pctdouble.
• FT.Pct.Rkdouble.
• FG_3A.Pctdouble.
• FG_3A.Pct.Rkdouble.
• A.Pctdouble.
• A.Pct.Rkdouble.
• HomeWLcharacter.
• HomeWL.Pctdouble.
• HomeWL.Rkdouble.
• Closedouble.
• Close.Rkdouble.
• Blowoutsdouble.
• Blowouts.Rkdouble.
• Yearcharacter.

Examples

try(kp_confstats(year=most_recent_mbb_season()))
### kp_efficiency

**Get KenPom Efficiency and Tempo Summary**

**Description**

Get KenPom Efficiency and Tempo Summary

**Usage**

\[
\text{kp\_efficiency}(\text{min\_year}, \text{max\_year} = \text{most\_recent\_mbb\_season()})
\]

**Arguments**

- \text{min\_year}  
  First year of data to pull
- \text{max\_year}  
  Last year of data to pull

**Value**

Returns a tibble of efficiency and tempo ratings

**Examples**

\[
\text{try}(\text{kp\_efficiency}(\text{min\_year} = 2020, \text{max\_year} = 2021))
\]

---

### kp_fanmatch

**Get FanMatch by date**

**Description**

Get FanMatch by date

**Usage**

\[
\text{kp\_fanmatch}(\text{date} = \text{"2020-02-12"})
\]

**Arguments**

- \text{date}  
  Date of games to pull (YYYY-MM-DD)
kp_foul_trouble

Value

A data frame 16 columns:

- Prediction character.
- Time(ET) character.
- Location character.
- ThrillScore double.
- Comeback double.
- Excitement double.
- WinRk character.
- WinTeam character.
- WinScore double.
- LossRk character.
- LossTeam character.
- LossScore double.
- Poss double.
- MVP character.
- Event character.
- Date character.

Examples

```
try(kp_fanmatch(date="2020-03-10"))
```

kp_foul_trouble Get 2-Foul Participation Stats

Description

Get 2-Foul Participation Stats

Usage

```
kp_foul_trouble(min_year, max_year = most_recent_mbb_season())
```

Arguments

- `min_year`: First year of data to pull
- `max_year`: Last year of data to pull
kp_fourfactors

Value

Returns a tibble of foul participation stats

Examples

try(kp_foul_trouble(min_year = 2020, max_year = most_recent_mbb_season()))

kp_fourfactors  Get Four Factors Data

Description

Get Four Factors Data

Usage

kp_fourfactors(min_year, max_year = most_recent_mbb_season())

Arguments

| min_year | First year of data to pull |
| max_year | Last year of data to pull |

Value

Returns a tibble of four factors ratings

Examples

try(kp_fourfactors(min_year = 2020, max_year = 2021))
### kp_gameplan

#### Description
Get KenPom’s team game plan page

#### Usage
 kp_gameplan(team, year = 2021)

#### Arguments
- **team**: Team filter to select.
- **year**: Year of data to pull

#### Value
Returns a tibble of team game plans

#### Examples
```r
try(kp_gameplan(team="Florida St.", year=2021))
```

### kp_game_attrs

#### Description
Get Game Attributes

#### Usage
 kp_game_attrs(year = most_recent_mbb_season(), attr = "Excitement")

#### Arguments
- **year**: Year of data to pull
- **attr**: Game Attribute, valid values include:
  - 'Excitement'
  - 'Tension'
  - 'Dominance'
  - 'MinWp'
  - 'FanMatch'
  - 'Upsets'
  - 'Busts'
  - 'Comeback'
  - 'Window'

#### Examples
```r
```
kp_hca

Value

Returns a tibble with game attributes

Examples

```
try(kp_game_attrs(year=2021, attr = "Excitement"))
```

kp_hca

Get Home Court Advantage Estimates

Description

Get Home Court Advantage Estimates

Usage

```
kp_hca()
```

Value

A data frame with 14 columns:

- Team character.
- Conf character.
- HCA double.
- HCA.Rk double.
- PF double.
- PF.Rk double.
- Pts double.
- Pts.Rk double.
- NST double.
- NST.Rk double.
- Blk double.
- Blk.Rk double.
- Elev double.
- Elev.Rk double.

Examples

```
try(kp_hca())
```
kp_height  
*Get Heights, Experience, Bench and Continuity Data*

**Description**
Get Heights, Experience, Bench and Continuity Data

**Usage**

```r
kp_height(min_year, max_year = most_recent_mbb_season())
```

**Arguments**

- `min_year`: First year of data to pull
- `max_year`: Last year of data to pull

**Value**
Returns a tibble of heights

**Examples**

```r
try(kp_height(min_year = 2020, max_year = 2021))
```

kp_kpoy  
*Get KPoY Leaders Tables*

**Description**
Get KPoY Leaders Tables

**Usage**

```r
kp_kpoy(year = most_recent_mbb_season())
```

**Arguments**

- `year`: Year of data to pull (earliest year of data available: 2011)

**Value**
Returns a list of tibbles: "kPoY Rating", "Game MVP Leaders"
kp_minutes_matrix

**Examples**

```r
try(kp_kpoy(year=2021))
```

---

**Description**
Get Minutes Matrix from Expanded Player Page

**Usage**

```r
kp_minutes_matrix(team, year = 2021)
```

**Arguments**

- `team` Team filter to select.
- `year` Year of data to pull

**Value**
Returns a tibble of minutes matrix data

**Examples**

```r
try(kp_minutes_matrix(team = 'Florida St.', year = 2021))
```

---

kp_officials

**Description**
Get officials rankings

**Usage**

```r
kp_officials(year = most_recent_mbb_season())
```
kp_opptracker

Arguments

year Year of data to pull

Value

A data frame with 7 columns:

Rk integer.
OfficialName character.
RefRating double.
Gms double.
Last.Game character.
Last.Game.1 character.
Year double.

Examples

try(kp_officials(year = 2021))

kp_opptracker Get KenPom’s team opponent tracker page

Description

Get KenPom’s team opponent tracker page

Usage

kp_opptracker(team, year = 2021)

Arguments

team Team filter to select.
year Year of data to pull

Value

Returns a tibble of team opponent tracker data

Examples

try(kp_opptracker(team = 'Florida St.', year = 2021))
kp_playerstats

Get Player Stats Leaders by Metric

Description

Get Player Stats Leaders by Metric

Usage

kp_playerstats(
  metric = "eFG",
  conf = NULL,
  conf_only = FALSE,
  year = most_recent_mbb_season()
)

Arguments

metric
  Used to get leaders for different metrics. Available values are:
  'ORtg', 'Min', 'eFG', 'Poss', 'Shots', 'OR', 'DR', 'TO', 'ARate', 'Blk',
  'FTRate', 'Stl', 'TS', 'FC40', 'FD40', '2P', '3P', 'FT'.
  Default is 'eFG'. 'ORtg' returns a list of four dataframes, as there are four tables:
  players that used more than 28 percent of possessions, more than 24 percent
  of possessions, more than 20 percent of possessions, and with no possession
  restriction.

conf
  Used to limit to players in a specific conference. Allowed values are: 'A10',
  'ACC', 'AE', 'AMER',
  'ASUN', 'B10', 'B12', 'BE', 'BSKY', 'BSTH', 'BW', 'CAA', 'CUSA', 'HORZ',
  'IND', 'IVY',
  'MAAC', 'MAC', 'MEAC', 'MVC', 'MWC', 'NEC', 'OVC', 'P12', 'PAT',
  'SB', 'SC', 'SEC', 'SLND',
  'SUM', 'SWAC', 'WAC', 'WCC'.
  If you try to use a conference that doesn't exist for a given season, like 'IND'
  and '2018',
  you'll get an empty table, as kenpom.com doesn't serve 404 pages for invalid
  table queries like that.
  No filter applied by default.

conf_only
  Used to define whether stats should reflect conference games only.
  Only available if specific conference is defined. Only available for season after
  2013, FALSE by default.

year
  Year of data to pull (earliest year of data available: 2004)

Value

Returns a tibble of player stats
kp_player_career

Get KenPom's player career stats from the player page

Description

Get KenPom's player career stats from the player page

Usage

kp_player_career(player_id)

Arguments

player_id Player Id filter to select.

Value

Returns a tibble of team player career stats

Examples

try(kp_player_career(player_id = '41180'))

kp_pointdist

Get Team Points Distribution

Description

Get Team Points Distribution

Usage

kp_pointdist(min_year, max_year = most_recent_mbb_season())

Arguments

min_year First year of data to pull
max_year Last year of data to pull

Examples

try(kp_pointdist(metric = 'eFG', conf_only = FALSE, year=2021))
kp_pomeroy_archive_ratings

Value
Returns a tibble of team points distributions

Examples

try(kp_pointdist(min_year = 2020, max_year = 2021))

kp_pomeroy_archive_ratings

Get KenPom’s ratings archive pages

Description
Get KenPom’s ratings archive pages

Usage
kp_pomeroy_archive_ratings(date)

Arguments
date Date (YYYY-MM-DD)

Value
A data frame with 22 columns:
AdjEM.Rk double.
Team character.
Conf character.
AdjEM double.
AdjO double.
AdjO.Rk double.
AdjD double.
AdjD.Rk double.
AdjT double.
AdjT.Rk double.
Final.Rk double.
Final.AdjEM double.
Final.AdjO double.
Final.AdjO.Rk double.
kp_pomeroy_ratings

Final.AdjD double.
Final.AdjD.Rk double.
Final.AdjT double.
Final.AdjT.Rk double.
Rk.Chg double.
EM.Chg double.
AdjT.Chg double.
NCAA_Seed double.

Examples

```r
try(kp_pomeroy_archive_ratings(date='2018-11-22'))
```

kp_pomeroy_ratings  Get KenPom Ratings

Description

Get KenPom Ratings

Usage

```r
kp_pomeroy_ratings(min_year, max_year = most_recent_mbb_season())
```

Arguments

- `min_year`: First year of data to pull
- `max_year`: Last year of data to pull

Value

Returns a tibble of ratings

Examples

```r
try(kp_pomeroy_ratings(min_year = 2020, max_year = 2021))
```
kp_program_ratings  Get KenPom’s program ratings

Description
Get KenPom’s program ratings

Usage
kp_program_ratings()

Value
A data frame with 17 columns:
Rk  double.
Team  character.
Conf  character.
Rtg  double.
Best.Rk  double.
Best.Yr  double.
Worst.Rk  double.
Worst.Yr  double.
KP.Median  double.
Top10  double.
Top25  double.
Top50  double.
CH  double.
F4  double.
S16  double.
R1  double.
Chg  double.

Examples
try(kp_program_ratings())
kp_referee  Get referee game log

Description
Get referee game log

Usage
kp_referee(referee, year)

Arguments
referee  Referee ID
year  Year of data to pull

Value
A data frame with 11 columns:
game_number  integer.
date  character.
time_et)  character.
game  character.
location  character.
venue  character.
conference  character.
thrill_score  double.
referee_name  character.
ref_rank  integer.
year  integer.

Examples
try(kp_referee(referee = 714363, year = 2021))
**kp_teamstats**  
*Get Team Stats*

**Description**  
Get Team Stats

**Usage**  
kp_teamstats(min_year, max_year = most_recent_mbb_season(), defense = FALSE)

**Arguments**
- **min_year**  
  First year of data to pull
- **max_year**  
  Last year of data to pull
- **defense**  
  Choose whether to pull offense (default) with FALSE or defense with TRUE

**Value**  
Returns a tibble of team stats

**Examples**
try(kp_teamstats(min_year = 2019, max_year = 2021, defense = FALSE))

---

**kp_team_depth_chart**  
*Get Depth Chart Last 5 Games from Team Page*

**Description**  
Get Depth Chart Last 5 Games from Team Page

**Usage**  
kp_team_depth_chart(team, year = 2021)

**Arguments**
- **team**  
  Team filter to select.
- **year**  
  Year of data to pull
Value

A data frame with 12 columns:

- **PG** character. DESCRIPTION.
- **PG.Minpct** character. DESCRIPTION.
- **SG** character. DESCRIPTION.
- **SG.Minpct** character. DESCRIPTION.
- **SF** character. DESCRIPTION.
- **SF.Minpct** character. DESCRIPTION.
- **PF** character. DESCRIPTION.
- **PF.Minpct** character. DESCRIPTION.
- **C** character. DESCRIPTION.
- **C.Minpct** character. DESCRIPTION.
- **Team** character. DESCRIPTION.
- **Year** double. DESCRIPTION.

Examples

```r
try(kp_team_depth_chart(team = 'Florida St.', year = 2021))
```

---

### kp_team_history

*Get KenPom's team efficiency history*

**Description**

Get KenPom's team efficiency history

**Usage**

```r
kp_team_history(team)
```

**Arguments**

- **team** Team filter to select.
Value

A data frame with 30 columns:

- Year double.
- Team.Rk double.
- Team character.
- Coach character.
- Conf character.
- W-L character.
- AdjT double.
- AdjO double.
- AdjD double.
- Off.eFG.Pct double.
- Off.TO.Pct double.
- Off.OR.Pct double.
- Off.FTRate double.
- Off.FG_2.Pct double.
- Off.FT.Pct double.
- Off.FG_3A.Pct double.
- Off.A.Pct double.
- Off.APL double.
- Def.eFG.Pct double.
- Def.TO.Pct double.
- Def.OR.Pct double.
- Def.FTRate double.
- Def.FG_2.Pct double.
- Def.FG_3.Pct double.
- Def.FG_3A.Pct double.
- Def.A.Pct double.
- Def.AP double.
- Foul2Partic.Pct double.

Examples

```r
kp_team_history(team = 'Florida St.')
```
kp_team_lineups

Get Lineups Last 5 Games from Team Page

Description
Get Lineups Last 5 Games from Team Page

Usage
kp_team_lineups(team, year = 2021)

Arguments
team Team filter to select.
year Year of data to pull

Value
A data frame with 9 columns:
Rk character. DESCRIPTION.
PG character. DESCRIPTION.
SG character. DESCRIPTION.
SF character. DESCRIPTION.
PF character. DESCRIPTION.
C character. DESCRIPTION.
Minpct character. DESCRIPTION.
Team character. DESCRIPTION.
Year double. DESCRIPTION.

Examples
try(kp_team_lineups(team = 'Florida St.', year = 2021))
kp_team_players

Get KenPom’s player stats from the team page

Description
Get KenPom’s player stats from the team page

Usage
kp_team_players(team, year = 2021)

Arguments
- team: Team filter to select.
- year: Year of data to pull

Value
Returns a tibble of team player data

Examples

try(kp_team_players(team = 'Florida St.', year = 2021))

kp_team_player_stats

Get Team Player Stats

Description
Get Team Player Stats

Usage
kp_team_player_stats(team, year = 2021)

Arguments
- team: Team filter to select.
- year: Year of data to pull

Value
Returns a tibble of team player stats data
**kp_trends**

**Examples**

```r
try(kp_team_player_stats(team = 'Florida St.', year = 2021))
```

---

**kp_team_schedule**

*Get team schedule results*

**Description**

Get team schedule results

**Usage**

```r
kp_team_schedule(team, year = 2022)
```

**Arguments**

- **team**: Team filter to select.
- **year**: Year of data to pull

**Value**

Returns a tibble of team schedules

**Examples**

```r
try(kp_team_schedule(team = 'Florida St.', year = 2022))
```

---

**kp_trends**

*Get Division-I statistical trends*

**Description**

Get Division-I statistical trends

**Usage**

```r
kp_trends()
```
**Value**

A data frame with 19 columns:

- Season double.
- Efficiency double.
- Tempo double.
- eFG.Pct double.
- TO.Pct double.
- OR.Pct double.
- FTRate double.
- FG_2.Pct double.
- FG_3A.Pct double.
- FT.Pct double.
- A.Pct double.
- Blk.Pct double.
- Stl.Pct double.
- NonStl.Pct double.
- AvgHgt double.
- Continuity double.
- HomeWin.Pct double.
- PPG double.

**Examples**

```r
try(kp_trends())
```

---

**kp_user_pw** | KenPom Login and Password credentials

**Description**

Save your KenPom login e-mail and password as the system environment variables KP_USER and KP_PW.

Requires a subscription to KenPom.com
Usage

```r
login(user_email = Sys.getenv("KP_USER"), user_pw = Sys.getenv("KP_PW"))

kp_user_email()

kp_password()

has_kp_user_and_pw()
```

Arguments

- **user_email**  User subscription e-mail
- **user_pw** User subscription password

Details

**Using your KenPom subscription with the package:**

Run `usethis::edit_r_environ()` and THEN paste the following in the new script that pops up (without quotations)

```r
KP_USER = YOUR-EMAIL@DOMAIN.COM
KP_PW = XXX-YOUR-PASSWORD-XXX
```

You can save the login information for consistent usage by adding

```r
KP_USER = YOUR-EMAIL@DOMAIN.COM
KP_PW = XXX-YOUR-PASSWORD-XXX
```

to your `.Renviron` file (easily accessed via `usethis::edit_r_environ()`)

For less consistent usage:

At the beginning of every session or within an R environment, save your login e-mail and password as the environment variables

```r
Sys.setenv(KP_USER = "YOUR-EMAIL@DOMAIN.COM")
Sys.setenv(KP_PW = "XXX-YOUR-PASSWORD-XXX")
```

---

**kp_winprob**

*Get Win Probability*

---

**Description**

Get Win Probability
**Usage**

```r
cpy_winprob(game_id, year)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game id of game to pull</td>
</tr>
<tr>
<td>year</td>
<td>Year of game to pull</td>
</tr>
</tbody>
</table>

**Value**

Returns a tibble of game win probabilities

**Examples**

```r
try(kp_winprob(game_id = 1238, year = 2020))
```

---

**Description**

Get NBA Stats API League Dashboard Player Tracking - Opponent Shots

**Usage**

```r
nba_leaguedashoppptshot(
  close_def_dist_range = "",
  conference = "",
  date_from = "",
  date_to = "",
  division = "",
  dribble_range = "",
  game_segment = "",
  general_range = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  pace_adjust = "N",
)```
per_mode = "Totals",
period = 0,
plus_minus = "N",
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
shot_dist_range = "",
team_id = "",
touch_time_range = "",
vs_conference = "",
vs_division = ""
)

Arguments

close_def_dist_range
close_def_dist_range
conference conference
date_from date_from
date_to date_to
division division
dribble_range dribble_range
game_segment game_segment
geneneral_range general_range
last_n_games last_n_games
league_id league_id
location location
measure_type measure_type
month month
opponent_team_id opponent_team_id
outcome outcome
po_round po_round
pace_adjust pace_adjust
per_mode per_mode
period period
plus_minus plus_minus
rank rank
season season
season_segment season_segment
Get NBA Stats API League Dashboard Player Biographical Stats

Usage

```r
nba_leaguedashplayerbiostats(
  college = "",
  conference = "",
  country = "",
  date_from = "",
  date_to = "",
  division = "",
  draft_pick = "",
  draft_year = "",
  game_segment = "",
  game_scope = "",
  height = "",
  last_n_games = 0,
  league_id = "00",
  location = ",",
  month = 0,
  opponent_team_id = 0,
)```
outcome = "", 
po_round = "", 
per_mode = "Totals", 
period = "", 
player_experience = "", 
player_position = "", 
season = "2020-21", 
season_segment = "", 
season_type = "Regular Season", 
shot_clock_range = "", 
starter_bench = "", 
team_id = "", 
touch_time_range = "", 
vs_conference = "", 
vs_division = "", 
weight = ""
)

Arguments

college          college
conference       conference
country          country
date_from        date_from
date_to          date_to
division         division
draft_pick       draft_pick
draft_year       draft_year
game_segment     game_segment
game_scope       game_scope
height           height
last_n_games     last_n_games
league_id        league_id
location         location
month            month
opponent_team_id  opponent_team_id
outcome          outcome
po_round         po_round
per_mode         per_mode
period           period
player_experience player_experience
player_position
season
season_segment
season_type
shot_clock_range
starter_bench
team_id
touch_time_range
vs_conference
vs_division
weight

Value
Returns a named list of data frames: LeagueDashPlayerBioStats

Author(s)
Saiem Gilani

---

ld_pclutch  Get NBA Stats API League Dashboard by Player Clutch Splits

Description
Get NBA Stats API League Dashboard by Player Clutch Splits

Usage
```r
nba_leaguedashplayerclutch(
    ahead_behind = "Ahead or Behind",
    clutch_time = "Last 5 Minutes",
    college = "",
    conference = "",
    country = "",
    date_from = "",
    date_to = "",
    division = "",
    draft_pick = "",
    draft_year = "",
)```

game_scope = "";
game_segment = "";
height = "";
last_n_games = 0;
league_id = "00";
location = "";
measure_type = "Base";
month = 0;
opponent_team_id = 0;
outcome = "";
pace_adjust = "N";
plus_minus = "N";
point_diff = 5;
po_round = "";
per_mode = "Totals";
period = 0;
player_experience = "";
player_position = "";
rank = "N";
season = "2020-21";
season_segment = "";
season_type = "Regular Season";
shot_clock_range = "";
starter_bench = "";
team_id = "";
touch_time_range = "";
vs_conference = "";
vs_division = "";
weight = ""

Arguments

ahead_behind ahead_behind
clutch_time clutch_time
college college
conference conference
country country
date_from date_from
date_to date_to
division division
draft_pick draft_pick
draft_year draft_year
game_scope game_scope
game_segment game_segment
ld_pclutch

<table>
<thead>
<tr>
<th>height</th>
<th>height</th>
</tr>
</thead>
<tbody>
<tr>
<td>last_n_games</td>
<td>last_n_games</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>measure_type</td>
<td>measure_type</td>
</tr>
<tr>
<td>month</td>
<td>month</td>
</tr>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>pace_adjust</td>
<td>pace_adjust</td>
</tr>
<tr>
<td>plus_minus</td>
<td>plus_minus</td>
</tr>
<tr>
<td>point_diff</td>
<td>point_diff</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>period</td>
<td>period</td>
</tr>
<tr>
<td>player_experience</td>
<td>player_experience</td>
</tr>
<tr>
<td>player_position</td>
<td>player_position</td>
</tr>
<tr>
<td>rank</td>
<td>rank</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>shot_clock_range</td>
<td>shot_clock_range</td>
</tr>
<tr>
<td>starter_bench</td>
<td>starter_bench</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
<tr>
<td>touch_time_range</td>
<td>touch_time_range</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
<tr>
<td>weight</td>
<td>weight</td>
</tr>
</tbody>
</table>

**Value**

Returns a named list of data frames: LeagueDashPlayerClutch

**Author(s)**

Saiem Gilani
Get NBA Stats API League Dashboard Player Tracking - Player Shots

Usage

```python
def nba_leaguedashplayerptshot(
    close_def_dist_range = '',
    college = '',
    conference = '',
    country = '',
    date_from = '',
    date_to = '',
    distance_range = '',
    division = '',
    draft_pick = '',
    draft_year = '',
    dribble_range = '',
    game_scope = '',
    game_segment = '',
    general_range = '',
    height = '',
    last_n_games = 0,
    league_id = '00',
    location = '',
    measure_type = 'Base',
    month = 0,
    opponent_team_id = 0,
    outcome = '',
    pace_adjust = 'N',
    po_round = '',
    per_mode = 'Totals',
    period = 0,
    player_experience = '',
    player_position = '',
    season = '2020-21',
    season_segment = '',
    season_type = 'Regular Season',
    shot_clock_range = '',
    shot_dist_range = '',
    starter_bench = '',
    team_id = '',
)
```
touch_time_range = "",
vs_conference = "",
vs_division = "",
weight = ""
)

Arguments

close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range
close_def_dist_range

college
college
college
college
college
college
college
college
college
college
college
college
college
college
college
college

date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from
date_from

date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to
date_to

distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range
distance_range

division
division
division
division
division
division
division
division
division
division
division
division
division
division
division
division

draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick
draft_pick

draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year
draft_year

dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range
dribble_range

game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope
game_scope

game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment

general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range
general_range

height
height
height
height
height
height
height
height
height
height
height
height
height
height
height
height

last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games
last_n_games

league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id
league_id

location
location
location
location
location
location
location
location
location
location
location
location
location
location
location
location

measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type
measure_type

month
month
month
month
month
month
month
month
month
month
month
month
month
month
month
month

opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id
opponent_team_id

outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome
outcome

pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust
pace_adjust

po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round
po_round

per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode
per_mode

period
period
period
period
period
period
period
period
period
period
period
period
period
period
period
period

player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience
player_experience

player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position
player_position

season
season
season
season
season
season
season
season
season
season
season
season
season
season
season
season
season_segment  season_segment
season_type      season_type
shot_clock_range shot_clock_range
shot_dist_range  shot_dist_range
starter_bench   starter_bench
team_id          team_id
touch_time_range touch_time_range
vs_conference   vs_conference
vs_division     vs_division
weight          weight

Value

Returns a named list of data frames: LeagueDashPTShots

Author(s)

Saiem Gilani

Get NBA Stats API League Dashboard Player Shot Locations

Description

Get NBA Stats API League Dashboard Player Shot Locations
Get NBA Stats API League Dashboard Player Shot Locations

Usage

nba_leaguedashplayershotlocations(
    college = "",
    conference = "",
    country = "",
    date_from = "",
    date_to = "",
    distance_range = "By Zone",
    division = "",
    draft_pick = "",
    draft_year = "",
    dribble_range = "",
    game_scope = "",
    game_segment = "",
)
ld_pshotloc

height = "",
last_n_games = 0,
league_id = "00",
location = "",
measure_type = "Base",
month = 0,
opponent_team_id = 0,
outcome = "",
op_round = "",
pace_adjust = "N",
per_mode = "Totals",
period = 0,
player_experience = "",
player_position = "",
plus_minus = "N",
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
starter_bench = "",
team_id = "",
vs_conference = "",
vs_division = "",
weight = ""
)

Arguments

college
college
country
country
date_from
date_from
date_to
date_to
distance_range
distance_range
division
division
draft_pick
draft_pick
draft_year
draft_year
dribble_range
dribble_range
game_scope
game_scope
game_segment
game_segment
height
height
last_n_games
last_n_games
league_id
league_id
location | location
measure_type | measure_type
month | month
opponent_team_id | opponent_team_id
outcome | outcome
po_round | po_round
pace_adjust | pace_adjust
per_mode | per_mode
period | period
player_experience | player_experience
player_position | player_position
plus_minus | plus_minus
rank | rank
season | season
season_segment | season_segment
season_type | season_type
shot_clock_range | shot_clock_range
starter_bench | starter_bench
team_id | team_id
vs_conference | vs_conference
vs_division | vs_division
weight | weight

**Value**

Returns a named list of data frames: ShotLocations

**Author(s)**

Saiem Gilani
Get NBA Stats API League Dashboard Player Stats

Usage

nba_leaguedashplayerstats(
    college = "",  
    conference = "",  
    country = "",  
    date_from = "",  
    date_to = "",  
    division = "",  
    draft_pick = "",  
    draft_year = "",  
    game_scope = "",  
    game_segment = "",  
    height = "",  
    last_n_games = 0,  
    league_id = "00",  
    location = "",  
    measure_type = "Base",  
    month = 0,  
    opponent_team_id = 0,  
    outcome = "",  
    pace_adjust = "N",  
    po_round = "",  
    per_mode = "Totals",  
    period = 0,  
    player_experience = "",  
    player_position = "",  
    plus_minus = "N",  
    rank = "N",  
    season = "2020-21",  
    season_segment = "",  
    season_type = "Regular Season",  
    shot_clock_range = "",  
    starter_bench = "",  
    team_id = "",  
    two_way = "",  
    vs_conference = "",  
    vs_division = "",  
)
weight = ""
}

Arguments

college    college
college    college
conference  conference
conference  conference
country    country
country    country
date_from  date_from
date_from  date_from
date_to    date_to
date_to    date_to
division    division
division    division
draft_pick  draft_pick
draft_pick  draft_pick
draft_year  draft_year
draft_year  draft_year
game_scope  game_scope
game_scope  game_scope
game_segment  game_segment
game_segment  game_segment
height    height
height    height
last_n_games  last_n_games
last_n_games  last_n_games
league_id    league_id
league_id    league_id
location    location
location    location
measure_type  measure_type
measure_type  measure_type
month    month
month    month
opponent_team_id  opponent_team_id
opponent_team_id  opponent_team_id
outcome    outcome
outcome    outcome
pace_adjust  pace_adjust
pace_adjust  pace_adjust
po_round  po_round
po_round  po_round
per_mode  per_mode
per_mode  per_mode
period    period
period    period
player_experience  player_experience
player_experience  player_experience
player_position  player_position
player_position  player_position
plus_minus  plus_minus
plus_minus  plus_minus
rank    rank
rank    rank
season    season
season    season
season_segment  season_segment
season_segment  season_segment
season_type  season_type
season_type  season_type
shot_clock_range  shot_clock_range
shot_clock_range  shot_clock_range
starter_bench  starter_bench
starter_bench  starter_bench
team_id  team_id
two_way  two_way
vs_conference  vs_conference
vs_division  vs_division
weight  weight

Value
Returns a named list of data frames: LeagueDashPlayerStats

Author(s)
Saiem Gilani

---

**ld_ptdefend**

**Get NBA Stats API League Dashboard Player Tracking - Defense**

**Description**

Get NBA Stats API League Dashboard Player Tracking - Defense
Get NBA Stats API League Dashboard Player Tracking - Defense

**Usage**

```r
nba_leaguedashptdefend(
  college = "",
  conference = "",
  country = "",
  date_from = "",
  date_to = "",
  defense_category = "Overall",
  division = "",
  draft_pick = "",
  draft_year = "",
  game_segment = "",
  height = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  per_mode = "Totals",
  period = "",
  player_experience = "",
)```
```python
player_id = "",
player_position = "",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
starter_bench = "",
team_id = "",
vs_conference = "",
vs_division = "",
weight = ""
)

Arguments

college
college
country
country
date_from
date_from
date_to
date_to
defense_category
defense_category
division
division
draft_pick
draft_pick
draft_year
draft_year
game_segment
game_segment
height
height
last_n_games
last_n_games
league_id
league_id
location
location
month
month
opponent_team_id
opponent_team_id
outcome
outcome
po_round
po_round
per_mode
per_mode
period
period
player_experience
player_experience
player_id
player_id
player_position
player_position
season
season
```
Value

Returns a named list of data frames: LeagueDashPTDefend

Author(s)

Saiem Gilani

---

**ld_ptstats**  
Get NBA Stats API League Dashboard Player Tracking - Stats

**Description**

Get NBA Stats API League Dashboard Player Tracking - Stats

Get NBA Stats API League Dashboard Player Tracking - Stats

**Usage**

```r
nba_leaguedashptstats(
  college = "",
  conference = "",
  country = "",
  date_from = "",
  date_to = "",
  division = "",
  draft_pick = "",
  draft_year = "",
  game_scope = "",
  height = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  per_mode = "Totals",
)```
period = "", 
player_experience = "", 
player_or_team = "Team", 
player_position = "", 
pt_measure_type = "SpeedDistance", 
season = "2020-21", 
season_segment = "", 
season_type = "Regular Season", 
starter_bench = "", 
team_id = "", 
vs_conference = "", 
vs_division = "", 
weight = ""
)

Arguments

college
college
country
country
date_from
date_from
date_to
date_to
division
division
draft_pick
draft_pick
draft_year
draft_year
game_scope
game_scope
height
height
last_n_games
last_n_games
league_id
league_id
location
location
month
month
opponent_team_id
opponent_team_id
outcome
outcome
po_round
po_round
per_mode
per_mode
period
period
player_experience
player_experience
player_or_team
player_or_team
player_position
player_position
ld_ptteamdefend

pt_measure_type  pt_measure_type
season          season
season_segment  season_segment
season_type     season_type
starter_bench   starter_bench
team_id         team_id
vs_conference   vs_conference
vs_division     vs_division
weight          weight

Author(s)

Saiem Gilani

| ld_ptteamdefend | Get NBA Stats API League Dashboard Player Tracking - Team Defense |

Description

Get NBA Stats API League Dashboard Player Tracking - Team Defense
Get NBA Stats API League Dashboard Player Tracking - Team Defense

Usage

```python
def nba_leaguedashptteamdefend(
    conference = "",
    date_from = "",
    date_to = "",
    defense_category = "Overall",
    division = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    per_mode = "Totals",
    period = "",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
)```
team_id = "",
vs_conference = "",
vs_division = ""
)

Arguments
conference conference
date_from date_from
date_to date_to
defense_category defense_category
division division
game_segment game_segment
last_n_games last_n_games
league_id league_id
location location
month month
opponent_team_id opponent_team_id
outcome outcome
po_round po_round
per_mode per_mode
period period
season season
season_segment season_segment
season_type season_type
team_id team_id
vs_conference vs_conference
vs_division vs_division

Author(s)
Saiem Gilani
Get NBA Stats API League Dashboard by Team Clutch Splits

Description

Get NBA Stats API League Dashboard by Team Clutch Splits

Usage

```python
nba_leaguedashteamclutch(
    ahead_behind = "Ahead or Behind",
    clutch_time = "Last 5 Minutes",
    conference = "",
    date_from = "",
    date_to = "",
    division = "",
    game_scope = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    point_diff = 5,
    po_round = "",
    per_mode = "Totals",
    period = 0,
    player_experience = "",
    player_position = "",
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    starter_bench = "",
    team_id = "",
    vs_conference = "",
    vs_division = ""
)
```
Arguments

- ahead_behind
- clutch_time
- conference
- date_from
- date_to
- division
- game_scope
- game_segment
- last_n_games
- league_id
- location
- measure_type
- month
- opponent_team_id
- outcome
- pace_adjust
- plus_minus
- point_diff
- po_round
- per_mode
- period
- player_experience
- player_position
- rank
- season
- season_segment
- season_type
- shot_clock_range
- starter_bench
- team_id
- vs_conference
- vs_division
Value

Returns a named list of data frames: LeagueDashTeamClutch

Author(s)

Saiem Gilani

---

ld_tptshot  Get NBA Stats API League Dashboard Player Tracking - Team Shots

Description

Get NBA Stats API League Dashboard Player Tracking - Team Shots
Get NBA Stats API League Dashboard Player Tracking - Team Shots

Usage

nba_leaguedashteamptshot(
  close_def_dist_range = "",
  conference = "",
  date_from = "",
  date_to = "",
  division = "",
  dribble_range = "",
  game_segment = "",
  general_range = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  per_mode = "Totals",
  period = 0,
  season = "2020-21",
  season_segment = "",
  season_type = "Regular Season",
  shot_clock_range = "",
  shot_dist_range = "",
  team_id = "",
  touch_time_range = "",
  vs_conference = "",
  vs_division = ""
)
Arguments

- close_def_dist_range
- conference
- date_from
- date_to
- division
- dribble_range
- game_segment
- general_range
- last_n_games
- league_id
- location
- month
- opponent_team_id
- outcome
- po_round
- per_mode
- period
- season
- season_segment
- season_type
- shot_clock_range
- shot_dist_range
- team_id
- touch_time_range
- vs_conference
- vs_division

Value

Returns a named list of data frames: LeagueDashPTShots

Author(s)

Saiem Gilani
ld_tshotloc  Get NBA Stats API League Dashboard Team Shot Locations

Description

Get NBA Stats API League Dashboard Team Shot Locations

Usage

```python
nba_leaguedashteamshotlocations(
    conference = "",
    date_from = "",
    date_to = "",
    distance_range = "By Zone",
    division = "",
    game_scope = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    pace_adjust = "N",
    per_mode = "Totals",
    period = 0,
    player_experience = "",
    player_position = "",
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    starter_bench = "",
    team_id = "",
    vs_conference = "",
    vs_division = ""
)
```

Arguments

conference  conference
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>date_from</td>
<td>date_from</td>
</tr>
<tr>
<td>date_to</td>
<td>date_to</td>
</tr>
<tr>
<td>distance_range</td>
<td>distance_range</td>
</tr>
<tr>
<td>division</td>
<td>division</td>
</tr>
<tr>
<td>game_scope</td>
<td>game_scope</td>
</tr>
<tr>
<td>game_segment</td>
<td>game_segment</td>
</tr>
<tr>
<td>last_n_games</td>
<td>last_n_games</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>measure_type</td>
<td>measure_type</td>
</tr>
<tr>
<td>month</td>
<td>month</td>
</tr>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>pace_adjust</td>
<td>pace_adjust</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>period</td>
<td>period</td>
</tr>
<tr>
<td>player_experience</td>
<td>player_experience</td>
</tr>
<tr>
<td>player_position</td>
<td>player_position</td>
</tr>
<tr>
<td>plus_minus</td>
<td>plus_minus</td>
</tr>
<tr>
<td>rank</td>
<td>rank</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>shot_clock_range</td>
<td>shot_clock_range</td>
</tr>
<tr>
<td>starter_bench</td>
<td>starter_bench</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
</tbody>
</table>

**Value**

Returns a named list of data frames: ShotLocations

**Author(s)**

Saiem Gilani
ld_tstats

Get NBA Stats API League Dashboard Team Stats

Description

Get NBA Stats API League Dashboard Team Stats

Get NBA Stats API League Dashboard Team Stats

Usage

nba_leaguedashteamstats(
    conference = "",
    date_from = "",
    date_to = "",
    division = "",
    game_scope = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    pace_adjust = "N",
    per_mode = "Totals",
    period = 0,
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    starter_bench = "",
    team_id = "",
    two_way = "",
    vs_conference = "",
    vs_division = ""
)

Arguments

conference  conference
date_from  date_from
date_to  date_to
division  division
game_scope  game_scope
game_segment  game_segment
last_n_games  last_n_games
league_id  league_id
location  location
measure_type  measure_type
month  month
opponent_team_id  opponent_team_id
outcome  outcome
po_round  po_round
pace_adjust  pace_adjust
per_mode  per_mode
period  period
plus_minus  plus_minus
rank  rank
season  season
season_segment  season_segment
season_type  season_type
shot_clock_range  shot_clock_range
starter_bench  starter_bench
team_id  team_id
two_way  two_way
vs_conference  vs_conference
vs_division  vs_division

Value

Returns a named list of data frames: LeagueDashTeamStats

Author(s)

Saiem Gilani
**description**

**get NBA stats API Leaders Tiles**

**Usage**

```r
nba_leaderstiles(
  league_id = "00",
  game_scope = "Season",
  player_or_team = "Team",
  player_scope = "All Players",
  season = "2020-21",
  season_type = "Regular Season",
  stat = "PTS"
)
```

**Arguments**

- **league_id**
  - League - default: '00'. Other options include '10': WNBA, '20': G-League
- **game_scope**
  - Game Scope - Season, Last 10, Yesterday, Finals
- **player_or_team**
  - Player or Team
- **player_scope**
  - Player Scope - All Players, Rookies
- **season**
  - Season - format 2020-21
- **season_type**
  - Season Type - Regular Season, Playoffs
- **stat**
  - Stat - PTS, REB, AST, FG_PCT, FT_PCT, FG3_PCT, STL, BLK

**Value**

Returns a named list of data frames: AllTimeSeasonHigh, LastSeasonHigh, LeadersTiles, LowSeasonHigh,

**Author(s)**

Saiem Gilani
leaguedashlineups  Get NBA Stats API League Dashboard Lineups

Description

Get NBA Stats API League Dashboard Lineups

Get NBA Stats API League Dashboard Lineups

Usage

nba_leaguedashlineups(
    conference = "",
    date_from = "",
    date_to = "",
    division = "",
    game_segment = "",
    group_quantity = 5,
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    pace_adjust = "N",
    per_mode = "Totals",
    period = 0,
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = "",
    vs_conference = "",
    vs_division = ""
)

Arguments

conference  conference
date_from  date_from
date_to  date_to
division  division
Value

Returns a named list of data frames: Lineups

Author(s)

Saiem Gilani
**leagueleaders**

Get NBA Stats API League Leaders

**Description**

Get NBA Stats API League Leaders

Get NBA Stats API League Leaders

**Usage**

```r
nba_leagueleaders(
active_flag = "",
league_id = "00",
per_mode = "Totals",
scope = "S",
season = "2020-21",
season_type = "Regular Season",
stat_category = "PTS"
)
```

**Arguments**

- **active_flag**: Active Flag
- **league_id**: League - default: '00'. Other options include '10': WNBA, '20': G-League
- **per_mode**: Per Mode - Totals, PerGame, Per48
- **scope**: Scope - RS, S, Rookies
- **season**: Season - format 2020-21
- **season_type**: Season Type - Regular Season, Playoffs
- **stat_category**: Stat Category: PTS, REB, AST, FG_PCT, FT_PCT, FG3_PCT, STL, BLK

**Value**

Returns a named list of data frames: LeagueLeaders

**Author(s)**

Saiem Gilani
Get NBA Stats API League Lineup Visual Data

Usage

```python
def nba_leaguelineupviz(
    conference = '',
    date_from = '',
    date_to = '',
    division = '',
    game_segment = '',
    group_quantity = 5,
    last_n_games = 0,
    league_id = "00",
    location = '',
    measure_type = "Base",
    minutes_min = 10,
    month = 0,
    opponent_team_id = 0,
    outcome = '',
    po_round = '',
    pace_adjust = "N",
    per_mode = "Totals",
    period = 0,
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_segment = '',
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = '',
    vs_conference = '',
    vs_division = ''
)
```

Arguments

- **conference**
- **date_from**
- **date_to**
- **division**
<table>
<thead>
<tr>
<th>game_segment</th>
<th>game_segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>group_quantity</td>
<td>group_quantity</td>
</tr>
<tr>
<td>last_n_games</td>
<td>last_n_games</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>measure_type</td>
<td>measure_type</td>
</tr>
<tr>
<td>minutes_min</td>
<td>minutes_min</td>
</tr>
<tr>
<td>month</td>
<td>month</td>
</tr>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>pace_adjust</td>
<td>pace_adjust</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>period</td>
<td>period</td>
</tr>
<tr>
<td>plus_minus</td>
<td>plus_minus</td>
</tr>
<tr>
<td>rank</td>
<td>rank</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>shot_clock_range</td>
<td>shot_clock_range</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
</tbody>
</table>

**Value**

Returns a named list of data frames: LeagueLineupViz

**Author(s)**

Saiem Gilani
Get NBA Stats API League Player On/Off Details

Description

Get NBA Stats API League Player On/Off Details
Get NBA Stats API League Player On/Off Details

Usage

```python
nba_leagueplayerondetails(
    date_from = "",    # date_from
    date_to = "",     # date_to
    game_segment = "", # game_segment
    last_n_games = 0, # last_n_games
    league_id = "00", # league_id
    location = "",    # location
    measure_type = "Base", # measure_type
    month = 0,        # month
    opponent_team_id = 0, # opponent_team_id
    outcome = "",     # outcome
    pace_adjust = "N", # pace_adjust
    per_mode = "Totals", # per_mode
    period = 0,       # period
    plus_minus = "N",  # plus_minus
    rank = "N",       # rank
    season = "2020-21", # season
    season_segment = "", # season_segment
    season_type = "Regular Season", # season_type
    team_id = "1610612749", # team_id
    vs_conference = "", # vs_conference
    vs_division = ""   # vs_division
)
```

Arguments

date_from    date_from
date_to      date_to
game_segment game_segment
last_n_games last_n_games
league_id    league_id
location     location
measure_type measure_type
month        month
leagueseasonmatchups

**opponent_team_id, outcome, pace_adjust, per_mode, period, plus_minus, rank, season, season_segment, season_type, team_id, vs_conference, vs_division**

**Value**

Returns a named list of data frames: PlayersOnCourtLeaguePlayerDetails

**Author(s)**

Saiem Gilani

---

**leagueseasonmatchups**  Get NBA Stats API League Season Matchups

**Description**

Get NBA Stats API League Season Matchups

Get NBA Stats API League Season Matchups

**Usage**

```r
nba_leagueseasonmatchups(
  def_player_id = "",  
  def_team_id = "",  
  league_id = "00",  
  off_player_id = "",  
  off_team_id = "",  
  per_mode = "Totals",  
  season = "2020-21",  
  season_type = "Regular Season"
)
```
Arguments

- def_player_id
- def_team_id
- league_id
- off_player_id
- off_team_id
- per_mode
- season
- season_type

Value

Returns a named list of data frames: SeasonMatchups

Author(s)

Saiem Gilani

---

**lg_streak**

Get NBA Stats API League Game Streak Finder

Description

Get NBA Stats API League Game Streak Finder

Usage

```r
nba_leaguegamefinder(
  conference = "",
  date_from = "",
  date_to = "",
  division = "",
  draft_year = "",
  draft_team_id = "",
  draft_round = "",
  draft_number = "",
  et_ast = "",
  et_blk = "",
  et_dd = "",
  et_dreb = "",
  et_fg3a = "",
  et_fg3m = "",
  et_fg3_pct = "",
)```
et_fga = "", et_fgm = "", et_fg_pct = "", et_fta = "", et_ftm = "", et_ft_pct = "", et_minutes = "", et_oreb = "", et_pf = "", et_pts = "", et_reb = "", et_stl = "", et_td = "", et_tov = "", game_id = "", gt_ast = "", gt_blk = "", gt_dd = "", gt_dreb = "", gt_fg3a = "", gt_fg3m = "", gt_fg3_pct = "", gt_fga = "", gt fgm = "", gt_fg_pct = "", gt_fta = "", gt_ftm = "", gt_ft_pct = "", gt_minutes = "", gt_oreb = "", gt_pf = "", gt_pts = "", gt_reb = "", gt_stl = "", gt_td = "", gt_tov = "", league_id = "00", location = "", lt_ast = "", lt_blk = "", lt_dd = "", lt_dreb = "", lt_fg3a = "", lt_fg3m = "", lt_fg3_pct = "", lt_fga = "", lt_fgm = "", lt_fg_pct = "",
lt_fta = "",
lt_ftm = "",
lt_ft_pct = "",
lt_minutes = "",
lt_oreb = "",
lt_pf = "",
lt_pts = "",
lt_reb = "",
lt_stl = "",
lt_td = "",
lt_tov = "",
outcome = "",
po_round = "",
player_id = "",
player_or_team = "T",
rookie_year = "",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
starter_bench = "",
team_id = "",
vs_conference = "",
vs_division = "",
vs_team_id = "",
years_experience = ""
)

Arguments

conference conference
date_from date_from
date_to date_to
division division
draft_year draft_year
draft_team_id draft_team_id
draft_round draft_round
draft_number draft_number
et_ast et_ast
et_blk et_blk
et_dd et_dd
et_dreb et_dreb
et_fg3a et_fg3a
et_fg3m et_fg3m
et_fg3_pct et_fg3_pct
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>et_fga</td>
<td>et_fga</td>
</tr>
<tr>
<td>et_fgm</td>
<td>et_fgm</td>
</tr>
<tr>
<td>et_fg_pct</td>
<td>et_fg_pct</td>
</tr>
<tr>
<td>et_fta</td>
<td>et_fta</td>
</tr>
<tr>
<td>et_ftm</td>
<td>et_ftm</td>
</tr>
<tr>
<td>et_ft_pct</td>
<td>et_ft_pct</td>
</tr>
<tr>
<td>et_minutes</td>
<td>et_minutes</td>
</tr>
<tr>
<td>et_oreb</td>
<td>et_oreb</td>
</tr>
<tr>
<td>et_pf</td>
<td>et_pf</td>
</tr>
<tr>
<td>et_pts</td>
<td>et_pts</td>
</tr>
<tr>
<td>et_reb</td>
<td>et_reb</td>
</tr>
<tr>
<td>et_stl</td>
<td>et_stl</td>
</tr>
<tr>
<td>et_td</td>
<td>et_td</td>
</tr>
<tr>
<td>et_tov</td>
<td>et_tov</td>
</tr>
<tr>
<td>game_id</td>
<td>game_id</td>
</tr>
<tr>
<td>gt_ast</td>
<td>gt_ast</td>
</tr>
<tr>
<td>gt_blk</td>
<td>gt_blk</td>
</tr>
<tr>
<td>gt_dd</td>
<td>gt_dd</td>
</tr>
<tr>
<td>gt_dreb</td>
<td>gt_dreb</td>
</tr>
<tr>
<td>gt_fg3a</td>
<td>gt_fg3a</td>
</tr>
<tr>
<td>gt_fg3m</td>
<td>gt_fg3m</td>
</tr>
<tr>
<td>gt_fg3_pct</td>
<td>gt_fg3_pct</td>
</tr>
<tr>
<td>gt_fga</td>
<td>gt_fga</td>
</tr>
<tr>
<td>gt_fgm</td>
<td>gt_fgm</td>
</tr>
<tr>
<td>gt_fg_pct</td>
<td>gt_fg_pct</td>
</tr>
<tr>
<td>gt_fta</td>
<td>gt_fta</td>
</tr>
<tr>
<td>gt_ftm</td>
<td>gt_ftm</td>
</tr>
<tr>
<td>gt_ft_pct</td>
<td>gt_ft_pct</td>
</tr>
<tr>
<td>gt_minutes</td>
<td>gt_minutes</td>
</tr>
<tr>
<td>gt_oreb</td>
<td>gt_oreb</td>
</tr>
<tr>
<td>gt_pf</td>
<td>gt_pf</td>
</tr>
<tr>
<td>gt_pts</td>
<td>gt_pts</td>
</tr>
<tr>
<td>gt_reb</td>
<td>gt_reb</td>
</tr>
<tr>
<td>gt_stl</td>
<td>gt_stl</td>
</tr>
<tr>
<td>gt_td</td>
<td>gt_td</td>
</tr>
<tr>
<td>gt_tov</td>
<td>gt_tov</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>lt_ast</td>
<td>lt_ast</td>
</tr>
<tr>
<td>lt_blk</td>
<td>lt_blk</td>
</tr>
<tr>
<td>lt_dd</td>
<td>lt_dd</td>
</tr>
<tr>
<td>lt_dreb</td>
<td>lt_dreb</td>
</tr>
<tr>
<td>lt_fg3a</td>
<td>lt_fg3a</td>
</tr>
<tr>
<td>lt_fg3m</td>
<td>lt_fg3m</td>
</tr>
<tr>
<td>lt_fg3_pct</td>
<td>lt_fg3_pct</td>
</tr>
<tr>
<td>lt_fga</td>
<td>lt_fga</td>
</tr>
<tr>
<td>lt_fgm</td>
<td>lt_fgm</td>
</tr>
<tr>
<td>lt_fg_pct</td>
<td>lt_fg_pct</td>
</tr>
<tr>
<td>lt_fta</td>
<td>lt_fta</td>
</tr>
<tr>
<td>lt_ftm</td>
<td>lt_ftm</td>
</tr>
<tr>
<td>lt_ft_pct</td>
<td>lt_ft_pct</td>
</tr>
<tr>
<td>lt_minutes</td>
<td>lt_minutes</td>
</tr>
<tr>
<td>lt_oreb</td>
<td>lt_oreb</td>
</tr>
<tr>
<td>lt_pf</td>
<td>lt_pf</td>
</tr>
<tr>
<td>lt_pts</td>
<td>lt_pts</td>
</tr>
<tr>
<td>lt_reb</td>
<td>lt_reb</td>
</tr>
<tr>
<td>lt_stl</td>
<td>lt_stl</td>
</tr>
<tr>
<td>lt_td</td>
<td>lt_td</td>
</tr>
<tr>
<td>lt_tov</td>
<td>lt_tov</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>player_id</td>
<td>player_id</td>
</tr>
<tr>
<td>player_or_team</td>
<td>player_or_team</td>
</tr>
<tr>
<td>rookie_year</td>
<td>rookie_year</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>starter_bench</td>
<td>starter_bench</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
<tr>
<td>vs_team_id</td>
<td>vs_team_id</td>
</tr>
<tr>
<td>years_experience</td>
<td>years_experience</td>
</tr>
</tbody>
</table>
Value

Return a named list of data frames: LeagueGameFinderResults

Author(s)

Saiem Gilani

load_mbb_pbp  Load hoopR men’s college basketball play-by-play

Description

helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots

Usage

load_mbb_pbp(
  seasons = most_recent_mbb_season(),
  ..., 
  dbConnection = NULL,
  tablename = NULL
)

Arguments

seasons  A vector of 4-digit years associated with given men’s college basketball seasons. (Min: 2006)
...
  Additional arguments passed to an underlying function that writes the season data into a database (used by update_mbb_db()).

dbConnection  A DBIConnection object, as returned by

tablename  The name of the play by play data table within the database

Value

Returns a tibble

Examples

load_mbb_pbp(2021)
load_mbb_player_box  

Load hoopR men’s college basketball player box scores

Description

helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots

Usage

load_mbb_player_box(
  seasons = most_recent_mbb_season(),
  ..., 
  dbConnection = NULL,
  tablename = NULL 
)

Arguments

seasons A vector of 4-digit years associated with given men’s college basketball seasons. (Min: 2003)
... Additional arguments passed to an underlying function that writes the season data into a database (used by update_mbb_db()).
dbConnection A DBIConnection object, as returned by
tablename The name of the play by play data table within the database

Value

Returns a tibble

Examples

load_mbb_player_box(2021)

load_mbb_schedule  

Load hoopR men’s college basketball schedule

Description

helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots
load_mbb_team_box

Usage

load_mbb_team_box(
    seasons = most_recent_mbb_season(),
    ..., 
    dbConnection = NULL, 
    tablename = NULL 
)

Arguments

seasons  A vector of 4-digit years associated with given men's college basketball seasons.  
(Min: 2002)
...
Additional arguments passed to an underlying function that writes the season 
  data into a database (used by update_mbb_db()).
dbConnection  A DBIConnection object, as returned by 
tablename  The name of the play by play data table within the database

Value

Returns a tibble

Examples

load_mbb_team_box(2021)

load_mbb_team_box  Load hoopR men’s college basketball team box scores

Description

helper that loads multiple seasons from the data repo either into memory or writes it into a db using 
some forwarded arguments in the dots

Usage

load_mbb_team_box(
    seasons = most_recent_mbb_season(),
    ..., 
    dbConnection = NULL, 
    tablename = NULL 
)
Arguments

- **seasons**: A vector of 4-digit years associated with given men's college basketball seasons. (Min: 2003)
- **...** Additional arguments passed to an underlying function that writes the season data into a database (used by `update_mbb_db()`).
- **dbConnection**: A DBIConnection object, as returned by
- **tablename**: The name of the play by play data table within the database

Value

Returns a tibble

Examples

```r
load_mbb_team_box(2021)
```

---

**load_nba_pbp**

*Load hoopR NBA play-by-play*

Description

helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots

Usage

```r
load_nba_pbp(
  seasons = most_recent_nba_season(),
  ...,
  dbConnection = NULL,
  tablename = NULL
)
```

Arguments

- **seasons**: A vector of 4-digit years associated with given NBA seasons. (Min: 2002)
- **...** Additional arguments passed to an underlying function that writes the season data into a database (used by `update_nba_db()`).
- **dbConnection**: A DBIConnection object, as returned by
- **tablename**: The name of the play by play data table within the database

Value

Returns a tibble
load_nba_player_box

Examples

load_nba_pbp(2021)

load_nba_player_box

Load hoopR NBA player box scores

Description

helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots

Usage

load_nba_player_box(
  seasons = most_recent_nba_season(),
  ..., 
  dbConnection = NULL,
  tablename = NULL
)

Arguments

seasons A vector of 4-digit years associated with given NBA seasons. (Min: 2002)
... Additional arguments passed to an underlying function that writes the season data into a database (used by update_nba_db()).
dbConnection A DBIConnection object, as returned by
tablename The name of the play by play data table within the database

Value

Returns a tibble

Examples

load_nba_player_box(2021)
load_nba_schedule Load hoopR NBA schedules

Description
helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots

Usage
load_nba_schedule(
  seasons = most_recent_nba_season(),
  ..., 
  dbConnection = NULL,
  tablename = NULL
)

Arguments
seasons A vector of 4-digit years associated with given NBA seasons. (Min: 2002)
... Additional arguments passed to an underlying function that writes the season data into a database (used by update_nba_db()).
dbConnection A DBIConnection object, as returned by
tablename The name of the play by play data table within the database

Value
Returns a tibble

Examples
load_nba_schedule(2021)

load_nba_team_box Load hoopR NBA team box scores

Description
helper that loads multiple seasons from the data repo either into memory or writes it into a db using some forwarded arguments in the dots
Usage

```r
load_nba_team_box(
  seasons = most_recent_nba_season(),
  ...,
  dbConnection = NULL,
  tablename = NULL
)
```

Arguments

- **seasons**: A vector of 4-digit years associated with given NBA seasons. (Min: 2002)
- **...**: Additional arguments passed to an underlying function that writes the season data into a database (used by `update_nba_db()`).
- **dbConnection**: A `DBIConnection` object, as returned by
- **tablename**: The name of the play by play data table within the database

Value

Returns a tibble

Examples

```r
load_nba_team_box(2021)
```

---

**l_gamelog**

Get NBA Stats API League Game Log

Description

Get NBA Stats API League Game Log

Usage

```r
nba_leaguegamelog(
  counter = 0,
  date_from = "",
  date_to = "",
  direction = "ASC",
  league_id = "00",
  player_or_team = "T",
  season = "2020-21",
  season_type = "Regular Season",
  sorter = "DATE"
)
```
Arguments

- counter
- date_from
- date_to
- direction
- league_id
- player_or_team
- season
- season_type
- sorter

Value

Return a named list of data frames: LeagueGameLog

Author(s)

Saiem Gilani

---

1_standings

Get NBA Stats API League Standings

Description

Get NBA Stats API League Standings

Usage

```r
nba_leaguestandings(
  league_id = "00",
  season = "2020-21",
  season_type = "Regular Season",
  season_year = ""
)
```

Arguments

- league_id
- season
- season_type
- season_year
l_standingsv3

Value

Return a named list of data frames: Standings

Author(s)

Saiem Gilani

1_standingsv3  Get NBA Stats API League Standings V3

Description

Get NBA Stats API League Standings V3

Usage

nba_leaguestandingsv3(
  league_id = "00",
  season = "2020-21",
  season_type = "Regular Season",
  season_year = ""
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>season_year</td>
<td>season_year</td>
</tr>
</tbody>
</table>

Value

Return a named list of data frames: Standings

Author(s)

Saiem Gilani
matchupsrollup

Get NBA Stats API Matchups Rollup

Description

Get NBA Stats API Matchups Rollup

Get NBA Stats API Matchups Rollup

Usage

```r
nba_matchupsrollup(
  def_player_id = "",
  def_team_id = "",
  league_id = "00",
  off_player_id = "",
  off_team_id = "",
  per_mode = "Totals",
  season = "2020-21",
  season_type = "Regular Season"
)
```

Arguments

- `def_player_id`: def_player_id
- `def_team_id`: def_team_id
- `league_id`: league_id
- `off_player_id`: off_player_id
- `off_team_id`: off_team_id
- `per_mode`: per_mode
- `season`: season
- `season_type`: season_type

Value

Returns a named list of data frames: MatchupsRollup

Author(s)

Saiem Gilani
### most_recent_mbb_season

**Most Recent Men’s College Basketball Season**

**Description**

Most Recent Men’s College Basketball Season

**Usage**

```r
most_recent_mbb_season()
```

---

### most_recent_nba_season

**Most Recent NBA Season**

**Description**

Most Recent NBA Season

**Usage**

```r
most_recent_nba_season()
```

---

### nba_data_pbp

**Get NBA Data API Play-by-Play**

**Description**

Get NBA Data API Play-by-Play

**Usage**

```r
nba_data_pbp(game_id = "0021900001")
```

**Arguments**

- `game_id` Game ID - 10 digits, i.e. "0021900001"

**Value**

Returns a tibble

**Author(s)**

Saiem Gilani
nba_stats_videodetails

Get NBA Stats API Video Details

Description

Get NBA Stats API Video Details

Usage

```python
nba_videodetails(
    ahead_behind = "",
    clutch_time = "",
    context_filter = "",
    context_measure = "FGA",
    date_from = "",
    date_to = "",
    end_period = "",
    end_range = "",
    game_id = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    period = 0,
    player_id = "2544",
    point_diff = "",
    position = "",
    range_type = "",
    rookie_year = "",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    start_period = "",
    start_range = "",
    team_id = "1610612739",
    vs_conference = "",
    vs_division = ""
)
```

Arguments

ahead_behind    ahead_behind
clutch_time  clutch_time
context_filter context_filter
context_measure context_measure
date_from date_from
date_to date_to
end_period end_period
end_range end_range
game_id game_id
game_segment game_segment
last_n_games last_n_games
league_id league_id
location location
month month
opponent_team_id opponent_team_id
outcome outcome
period period
player_id player_id
point_diff point_diff
position position
range_type range_type
rookie_year rookie_year
season season
season_segment season_segment
season_type season_type
start_period start_period
start_range start_range
team_id team_id
vs_conference vs_conference
vs_division vs_division

Value
Return a list of tibbles

Author(s)
Saiem Gilani
nba_stats_videoevents  Get NBA Stats API Video Events

Description
Get NBA Stats API Video Events
Get NBA Stats API Video Events
Get NBA Stats API Video Events
Get NBA Stats API Video Events
Get NBA Stats API Video Events

Usage
nba_videoevents(game_id = "0021700807", game_event_id = "0")
nba_videostatus(game_date = "2020-08-16", league_id = "00")

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>game_id</td>
</tr>
<tr>
<td>game_event_id</td>
<td>game_event_id</td>
</tr>
<tr>
<td>game_date</td>
<td>game_date</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
</tbody>
</table>

Value
Return a list of tibbles
Return a list of tibbles: VideoStatus

Author(s)
Saiem Gilani
Saiem Gilani

nba_teams  NBA Stats API’s Teams Dictionary

Description
NBA Stats API’s Teams Dictionary

Usage
nba_teams
Format

A data frame with 30 rows and 10 variables:

- TeamID character.
- TeamCity character.
- TeamName character.
- TeamSlug double.
- Conference character.
- Division character.
- LeagueID character.
- SeasonID character.
- SEASON character.
- TEAMNAMEFULL character.

Get men’s college basketball NET rankings for the current date from the NCAA website

Usage

ncaa_mbb_NET_rankings()

Value

Returns a tibble

Author(s)

Saiem Gilani

Examples

# Get current NCAA NET rankings
try(ncaa_mbb_NET_rankings())
pbp  Get NBA Stats API play-by-play

Description
Get NBA Stats API play-by-play
Get NBA Stats API play-by-play

Usage
nba_pbp(game_id, version = "v2")

Arguments
  game_id    Game ID
  version    Play-by-play version ("v2" available from 2016-17 onwards)

Value
Returns a named list of data frames: PlayByPlay

Author(s)
Saiem Gilani

pbyclutch  Get NBA Stats API Player Dashboard by Clutch Splits

Description
Get NBA Stats API Player Dashboard by Clutch Splits
Get NBA Stats API Player Dashboard by Clutch Splits

Usage
nba_playerdashboardbyclutch(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
)
outcome = "", 
po_round = "", 
pace_adjust = "N", 
per_mode = "Totals", 
period = 0, 
player_id = "2544", 
plus_minus = "N", 
rank = "N", 
season = "2020-21", 
season_segment = "", 
season_type = "Regular Season", 
shot_clock_range = "", 
vs_conference = "", 
vs_division = ""
)

Arguments

date_from, date_from

date_to, date_to

game_segment, game_segment

last_n_games, last_n_games

league_id, league_id

location, location

measure_type, measure_type

month, month

opponent_team_id, opponent_team_id

outcome, outcome

po_round, po_round

pace_adjust, pace_adjust

per_mode, per_mode

period, period

player_id, player_id

plus_minus, plus_minus

rank, rank

season, season

season_segment, season_segment

season_type, season_type

shot_clock_range, shot_clock_range

vs_conference, vs_conference

vs_division, vs_division
Value

Return a named list of data frames: Last10Sec3Point2PlayerDashboard, Last10Sec3PointPlayerDashboard, Last1Min5PointPlayerDashboard, Last1MinPlusMinus5PointPlayerDashboard, Last30Sec3Point2PlayerDashboard, Last30Sec3PointPlayerDashboard, Last3Min5PointPlayerDashboard, Last3MinPlusMinus5PointPlayerDashboard, Last5Min5PointPlayerDashboard, Last5MinPlusMinus5PointPlayerDashboard, OverallPlayerDashboard

Author(s)

Saiem Gilani

pbygamesplits Get NBA Stats API Player Dashboard by Game Splits

Description

Get NBA Stats API Player Dashboard by Game Splits

Usage

nba_playerdashboardbygamesplits(
  date_from = "", 
  date_to = "", 
  game_segment = "", 
  last_n_games = 0, 
  league_id = "00", 
  location = "", 
  measure_type = "Base", 
  month = 0, 
  opponent_team_id = 0, 
  outcome = "", 
  po_round = "", 
  pace_adjust = "N", 
  per_mode = "Totals", 
  period = 0, 
  player_id = "2544", 
  plus_minus = "N", 
  rank = "N", 
  season = "2020-21", 
  season_segment = "", 
  season_type = "Regular Season", 
  shot_clock_range = "", 
  vs_conference = "", 
  vs_division = "" 
)
Arguments

date_from  date_from
date_to    date_to
game_segment  game_segment
last_n_games  last_n_games
league_id    league_id
location    location
measure_type  measure_type
month      month
opponent_team_id  opponent_team_id
outcome     outcome
po_round    po_round
pace_adjust  pace_adjust
per_mode    per_mode
period      period
player_id    player_id
plus_minus  plus_minus
rank        rank
season      season
season_segment  season_segment
season_type  season_type
shot_clock_range  shot_clock_range
vs_conference  vs_conference
vs_division  vs_division

Value

Return a named list of data frames: ByActualMarginPlayerDashboard, ByHalfPlayerDashboard, ByPeriodPlayerDashboard, ByScoreMarginPlayerDashboard, OverallPlayerDashboard

Author(s)

Saiem Gilani
pbygeneralsplits  Get NBA Stats API Player Dashboard by General Splits

Description

Get NBA Stats API Player Dashboard by General Splits

Usage

```r
def nba_playerdashboardbygeneralsplits(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    pace_adjust = "N",
    per_mode = "Totals",
    period = 0,
    player_id = "2544",
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    vs_conference = "",
    vs_division = ""
)```

Arguments

date_from  date_from
date_to  date_to
game_segment  game_segment
last_n_games  last_n_games
league_id  league_id
location  location
measure_type  measure_type
<table>
<thead>
<tr>
<th>month</th>
<th>month</th>
</tr>
</thead>
<tbody>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>pace_adjust</td>
<td>pace_adjust</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>period</td>
<td>period</td>
</tr>
<tr>
<td>player_id</td>
<td>player_id</td>
</tr>
<tr>
<td>plus_minus</td>
<td>plus_minus</td>
</tr>
<tr>
<td>rank</td>
<td>rank</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>shot_clock_range</td>
<td>shot_clock_range</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
</tbody>
</table>

**Value**

Return a named list of data frames: DaysRestPlayerDashboard, LocationPlayerDashboard, MonthPlayerDashboard, OverallPlayerDashboard, PrePostAllStarPlayerDashboard, StartingPosition, WinsLossesPlayerDashboard

**Author(s)**

Saiem Gilani

---

**pbylastngames** | Get NBA Stats API Player Dashboard by Last N Games

**Description**

Get NBA Stats API Player Dashboard by Last N Games

Get NBA Stats API Player Dashboard by Last N Games
Usage

nba_playerdashboardbylastngames(
    date_from = "",  
    date_to = "",  
    game_segment = "",  
    last_n_games = 0,  
    league_id = "00",  
    location = "",  
    measure_type = "Base",  
    month = 0,  
    opponent_team_id = 0,  
    outcome = "",  
    po_round = "",  
    pace_adjust = "N",  
    per_mode = "Totals",  
    period = 0,  
    player_id = "2544",  
    plus_minus = "N",  
    rank = "N",  
    season = "2020-21",  
    season_segment = "",  
    season_type = "Regular Season",  
    shot_clock_range = "",  
    vs_conference = "",  
    vs_division = ""
)

Arguments

date_from  date_from
date_to  date_to
game_segment  game_segment
last_n_games  last_n_games
league_id  league_id
location  location
measure_type  measure_type
month  month
opponent_team_id  opponent_team_id
outcome  outcome
po_round  po_round
pace_adjust  pace_adjust
per_mode  per_mode
period  period
Value

Return a named list of data frames: GameNumberPlayerDashboard, Last10PlayerDashboard, Last15PlayerDashboard, Last20PlayerDashboard, Last5PlayerDashboard, OverallPlayerDashboard

Author(s)

Saiem Gilani

Description

Get NBA Stats API Player Dashboard by Opponent

Get NBA Stats API Player Dashboard by Opponent

Usage

nba_playerdashboardbyopponent(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  pace_adjust = "N",
  per_mode = "Totals",
  period = 0,
player_id = "2544",
plus_minus = "N",
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
vs_conference = "",
vs_division = ""
)

Arguments

date_from          date_from
date_to            date_to

game_segment       game_segment
last_n_games       last_n_games
league_id          league_id
location           location
measure_type       measure_type
month              month
opponent_team_id   opponent_team_id
outcome            outcome
po_round           po_round
pace_adjust        pace_adjust
per_mode           per_mode
period             period
player_id          player_id
plus_minus         plus_minus
rank               rank
season             season
season_segment     season_segment
season_type        season_type
shot_clock_range   shot_clock_range
vs_conference      vs_conference
vs_division        vs_division

Value

Return a named list of data frames: ConferencePlayerDashboard, DivisionPlayerDashboard, OpponentPlayerDashboard, OverallPlayerDashboard
Author(s)

Saiem Gilani

---

**pbyshootingsplits**

Get NBA Stats API Player Dashboard by Shooting Splits

---

Description

Get NBA Stats API Player Dashboard by Shooting Splits

Usage

```python
nba_playerdashboardbyshootingsplits(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    pace_adjust = "N",
    per_mode = "Totals",
    period = 0,
    player_id = "2544",
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    vs_conference = "",
    vs_division = ""
)
```

Arguments

date_from   date_from
date_to     date_to
game_segment  game_segment
last_n_games  last_n_games
league_id | league_id
---|---
location | location
measure_type | measure_type
month | month
opponent_team_id | opponent_team_id
outcome | outcome
po_round | po_round
pace_adjust | pace_adjust
per_mode | per_mode
period | period
player_id | player_id
plus_minus | plus_minus
rank | rank
season | season
season_segment | season_segment
season_type | season_type
shot_clock_range | shot_clock_range
vs_conference | vs_conference
vs_division | vs_division

**Value**

Return a named list of data frames: AssistedBy, AssistedShotPlayerDashboard, OverallPlayerDashboard, Shot5FTPlayerDashboard, Shot8FTPlayerDashboard, ShotAreaPlayerDashboard, ShotTypePlayerDashboard, ShotTypeSummaryPlayerDashboard

**Author(s)**

Saiem Gilani

---

**pbyteamperformance**  
Get NBA Stats API Player Dashboard by Team Performance

---

**Description**

Get NBA Stats API Player Dashboard by Team Performance
Get NBA Stats API Player Dashboard by Team Performance
Usage

nba_playerdashboardbyteamperformance(
    date_from = "",  
    date_to = "",  
    game_segment = "", 
    last_n_games = 0,  
    league_id = "00",  
    location = "",  
    measure_type = "Base", 
    month = 0,  
    opponent_team_id = 0,  
    outcome = "", 
    po_round = "", 
    pace_adjust = "N", 
    per_mode = "Totals", 
    period = 0,  
    player_id = "2544",  
    plus_minus = "N", 
    rank = "N", 
    season = "2020-21",  
    season_segment = "",  
    season_type = "Regular Season",  
    shot_clock_range = "",  
    vs_conference = "", 
    vs_division = "" 
)

Arguments

date_from    date_from
date_to      date_to
game_segment game_segment
last_n_games last_n_games
league_id    league_id
location     location
measure_type measure_type
month        month
opponent_team_id opponent_team_id
outcome      outcome
po_round     po_round
pace_adjust  pace_adjust
per_mode     per_mode
period       period
Value

Return a named list of data frames: OverallPlayerDashboard, PointsScoredPlayerDashboard, PointsAgainstPlayerDashboard, ScoreDifferentialPlayerDashboard

Author(s)

Saiem Gilani

Description

Get NBA Stats API Player Dashboard Year over Year

Usage

```r
nba_playerdashboardbyyearoveryear(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  po_round = "",
  pace_adjust = "N",
  per_mode = "Totals",
  period = 0,
)```

player_id = "2544",
plus_minus = "N",
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
vs_conference = "",
vs_division = ""
}

Arguments

date_from   date_from
date_to     date_to
game_segment game_segment
last_n_games last_n_games
league_id   league_id
location    location
measure_type measure_type
month       month
opponent_team_id

outcome   outcome
po_round  po_round
pace_adjust   pace_adjust
per_mode  per_mode
period     period
player_id  player_id
plus_minus  plus_minus
rank       rank
season     season
season_segment season_segment
season_type season_type
shot_clock_range

Value

Return a named list of data frames: ByYearPlayerDashboard, OverallPlayerDashboard

Author(s)

Saiem Gilani
**Description**

Get NBA Stats API Player Career By College

**Usage**

```r
nba_playercareerbycollege(
  college = "Florida State",
  league_id = "00",
  per_mode = "Totals",
  season = "2020-21",
  season_type = "Regular Season"
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>college</td>
<td>College Name</td>
</tr>
<tr>
<td>league_id</td>
<td>League - default: '00'. Other options include '10': WNBA, '20': G-League</td>
</tr>
<tr>
<td>per_mode</td>
<td>Per Mode - PerGame, Totals</td>
</tr>
<tr>
<td>season</td>
<td>Season - format 2020-21</td>
</tr>
<tr>
<td>season_type</td>
<td>Season Type - Regular Season, Playoffs, All-Star</td>
</tr>
</tbody>
</table>

**Value**

Return a named list of data frames: PlayerCareerByCollege

**Author(s)**

Saiem Gilani

---

**pcareerbycollege_ru**  
Get NBA Stats API Player Career By College Rollup

**Description**

Get NBA Stats API Player Career By College Rollup

Get NBA Stats API Player Career By College Rollup
Usage

nba_playercareerbycollegerollup(
    league_id = "00",
    per_mode = "Totals",
    season = "2020-21",
    season_type = "Regular Season"
)

Arguments

- **league_id**: League - default: '00'. Other options include '10': WNBA, '20': G-League
- **per_mode**: Per Mode - PerGame, Totals
- **season**: Season - format 2020-21
- **season_type**: Season Type - Regular Season, Playoffs, All-Star

Value

Return a named list of data frames: East, Midwest, South, West

Author(s)

Saiem Gilani

---

pfantasy | Get NBA Stats API Player Fantasy Profile

Description

Get NBA Stats API Player Fantasy Profile

Usage

nba_playerfantasyprofile(
    league_id = "00",
    measure_type = "Base",
    pace_adjust = "N",
    per_mode = "Totals",
    player_id = "2544",
    plus_minus = "N",
    rank = "N",
    season = "2020-21",
    season_type = "Regular Season"
)
Arguments

league_id  League - default: '00'. Other options include '10': WNBA, '20': G-League
measure_type  measure_type
pace_adjust  Pace Adjustment - Y/N
per_mode  Per Mode - PerGame, Totals
player_id  Player ID
plus_minus  Plus Minus - Y/N
rank  Rank - Y/N
season  Season - format 2020-21
season_type  Season Type - Regular Season, Playoffs, All-Star

Value

Return a named list of data frames: DaysRestModified, LastNGames, Location, Opponent, Overall

Author(s)

Saiem Gilani

---

pfantasy_bg  Get NBA Stats API Player Fantasy Profile Bar Graph

Description

Get NBA Stats API Player Fantasy Profile Bar Graph
Get NBA Stats API Player Fantasy Profile Bar Graph

Usage

nba_playerfantasyprofilebargraph(
  league_id = "00",
  player_id = "2544",
  season = "2020-21",
  season_type = "Regular Season"
)

Arguments

league_id  League - default: '00'. Other options include '10': WNBA, '20': G-League
player_id  Player ID
season  Season - format 2020-21
season_type  Season Type - Regular Season, Playoffs, All-Star
**Value**

Return a named list of data frames: `LastFiveGamesAvg`, `SeasonAvg`

**Author(s)**

Saiem Gilani

---

**Description**

Get NBA Stats API Player Game Streak Finder

**Usage**

```r
nba_playergamestreakfinder(
  active_streaks_only = "",
  conference = "",
  date_from = "",
  date_to = "",
  division = "",
  draft_year = "",
  draft_team_id = "",
  draft_round = "",
  draft_number = "",
  et_ast = "",
  et_blk = "",
  et_dd = "",
  et_dreb = "",
  et_fg3a = "",
  et_fg3m = "",
  et_fg3_pct = "",
  et_fga = "",
  et_fgm = "",
  et_fg_pct = "",
  et_fta = "",
  et_ftm = "",
  et_ft_pct = "",
  et_minutes = "",
  et_oreb = "",
  et_pf = "",
  et_pts = "",
  et_reb = "",
  et_stl = "",
)```
et_td = "",  
et_tov = "",  
game_id = "",  
gt_ast = "",  
gt_blk = "",  
gt_dd = "",  
gt_dreb = "",  
gt_fg3a = "",  
gt_fg3m = "",  
gt_fg3_pct = "",  
gt_fga = "",  
gt_fgm = "",  
gt_fg_pct = "",  
gt_fta = "",  
gt_ftm = "",  
gt_ft_pct = "",  
gt_minutes = "",  
gt_oreb = "",  
gt_pf = "",  
gt_pts = "",  
gt_reb = "",  
gt_stl = "",  
gt_td = "",  
gt_tov = "",  
league_id = "00",  
location = "",  
l_t_ast = "",  
l_t_blk = "",  
l_t_dd = "",  
l_t_dreb = "",  
l_t_fg3a = "",  
l_t_fg3m = "",  
l_t_fg3_pct = "",  
l_t_fga = "",  
l_t_fgm = "",  
l_t_fg_pct = "",  
l_t_fta = "",  
l_t_ftm = "",  
l_t_ft_pct = "",  
l_t_minutes = "",  
l_t_oreb = "",  
l_t_pf = "",  
l_t_pts = "",  
l_t_reb = "",  
l_t_stl = "",  
l_t_td = "",  
l_t_tov = "",  
min_games = "",  

outcome = "", 
po_round = "", 
player_id = "", 
rookie_year = "", 
season = "2020-21", 
season_segment = "", 
season_type = "Regular Season", 
starter_bench = "", 
team_id = "", 
vs_conference = "", 
vs_division = "", 
vs_team_id = "", 
years_experience = ""
)

Arguments

active_streaks_only
active_streaks_only

calendar
conference
date_from
date_to
division
draft_year
draft_team_id
draft_round
draft_number
et_ast
et_blk
et_dd
et_dreb
et_fg3a
et_fg3m
et_fg3_pct
et_fga
et_fgm
et_fg_pct
et_fta
et_ftm
et_ft_pct
et_minutes
et_oreb
et_pf  et_pf
et_pts et_pts
et_reb et_reb
et_stl et_stl
et_td  et_td
et_tov et_tov
game_id game_id
gt_ast  gt_ast
gt_blk  gt_blk
gt_dd  gt_dd
gt_dreb gt_dreb
gt_fg3a gt_fg3a
gt_fg3m gt_fg3m
gt_fg3_pct gt_fg3_pct
gt_fga  gt_fga
gt_fgm  gt_fgm
gt_fg_pct gt_fg_pct
gt_fta  gt_fta
gt_ftm  gt_ftm
gt_ft_pct gt_ft_pct
gt_minutes gt_minutes
gt_oreb gt_oreb
gt_pf   gt_pf
gt_pts  gt_pts
gt_reb  gt_reb
gt_stl  gt_stl
gt_td   gt_td
gt_tov  gt_tov
league_id league_id
location location
lt_ast  lt_ast
lt_blk  lt_blk
lt_dd   lt_dd
lt_dreb lt_dreb
lt_fg3a lt_fg3a
lt_fg3m lt_fg3m
lt_fg3_pct lt_fg3_pct
lt_fga   lt_fga
lt_fgm   lt_fgm
lt_fg_pct lt_fg_pct
lt_fta   lt_fta
lt_ftm   lt_ftm
lt_ft_pct lt_ft_pct
lt_minutes lt_minutes
lt_oreb lt_oreb
lt_pf   lt_pf
lt_pts   lt_pts
lt_reb   lt_reb
lt_stl   lt_stl
lt_td   lt_td
lt_tov   lt_tov
min_games min_games
outcome  outcome
po_round po_round
player_id player_id
rookie_year rookie_year
season  season
season_segment season_segment
season_type season_type
starter_bench starter_bench
team_id team_id
vs_conference vs_conference
vs_division vs_division
vs_team_id vs_team_id
years_experience

Value

Return a named list of data frames: PlayerGameStreakFinderResults

Author(s)

Saiem Gilani
playerawards

---

**playerawards**

Get NBA Stats API Player Awards

**Description**

Get NBA Stats API Player Awards
Get NBA Stats API Player Awards

**Usage**

```r
nba_playerawards(player_id)
```

**Arguments**

- `player_id`
  - Player ID

**Value**

Return a named list of data frames: PlayerAwards

**Author(s)**

Saiem Gilani

---

playercareerstats

---

**playercareerstats**

Get NBA Stats API Player Career Stats

**Description**

Get NBA Stats API Player Career Stats
Get NBA Stats API Player Career Stats

**Usage**

```r
nba_playercareerstats(
  league_id = "00",
  per_mode = "Totals",
  player_id = "2544"
)
```

**Arguments**

- `league_id`
  - League - default: '00’. Other options include '10': WNBA, '20': G-League
- `per_mode`
  - Per Mode - PerGame, Totals
- `player_id`
  - Player ID
Value

Return a named list of data frames: CareerTotalsAllStarSeason, CareerTotalsCollegeSeason, CareerTotalsPostSeason, CareerTotalsRegularSeason, SeasonRankingsPostSeason, SeasonRankingsRegularSeason, SeasonTotalsAllStarSeason, SeasonTotalsCollegeSeason, SeasonTotalsPostSeason, SeasonTotalsRegularSeason

Author(s)

Saiem Gilani

---

playercompare  Get NBA Stats API Player Compare

Description

Get NBA Stats API Player Compare

Get NBA Stats API Player Compare

Usage

```r
def nba_playercompare(
  conference = "",
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  pace_adjust = "N",
  per_mode = "Totals",
  period = 0,
  player_id_list = "202681,203078,2544,201567,203954",
  plus_minus = "N",
  rank = "N",
  season = "2020-21",
  season_segment = "",
  season_type = "Regular Season",
  shot_clock_range = "",
  vs_conference = "",
  vs_division = "",
  vs_player_id_list = "201566,201939,201935,201142,203076"
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>conference</td>
<td>conference</td>
</tr>
<tr>
<td>date_from</td>
<td>date_from</td>
</tr>
<tr>
<td>date_to</td>
<td>date_to</td>
</tr>
<tr>
<td>game_segment</td>
<td>game_segment</td>
</tr>
<tr>
<td>last_n_games</td>
<td>last_n_games</td>
</tr>
<tr>
<td>league_id</td>
<td>League - default: '00'. Other options include '10': WNBA, '20': G-League</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>measure_type</td>
<td>measure_type</td>
</tr>
<tr>
<td>month</td>
<td>month</td>
</tr>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>pace_adjust</td>
<td>pace_adjust</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>period</td>
<td>period</td>
</tr>
<tr>
<td>player_id_list</td>
<td>Player ID</td>
</tr>
<tr>
<td>plus_minus</td>
<td>plus_minus</td>
</tr>
<tr>
<td>rank</td>
<td>rank</td>
</tr>
<tr>
<td>season</td>
<td>Season - format 2020-21</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>Season Type - Regular Season, Playoffs, All-Star</td>
</tr>
<tr>
<td>shot_clock_range</td>
<td>shot_clock_range</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
<tr>
<td>vs_player_id_list</td>
<td>vs_player_id_list</td>
</tr>
</tbody>
</table>

Value

Return a named list of data frames: Individual, OverallCompare

Author(s)

Saiem Gilani
playerprofilev2  

Get NBA Stats API Player Profile V2

Description

Get NBA Stats API Player Profile V2

Usage

nba_playerprofilev2(league_id = "", per_mode = "Totals", player_id = "2544")

Arguments

- league_id: League - default: '00'. Other options include '10': WNBA, '20': G-League
- per_mode: Season - format 2020-21
- player_id: Player ID

Value

Return a named list of data frames: CareerHighs, CareerTotalsAllStarSeason, CareerTotalsCollegeSeason, CareerTotalsPostSeason, CareerTotalsPreseason, CareerTotalsRegularSeason, NextGame, SeasonHighs, SeasonRankingsPostSeason, SeasonRankingsRegularSeason, SeasonTotalsAllStarSeason, SeasonTotalsCollegeSeason, SeasonTotalsPostSeason, SeasonTotalsPreseason, SeasonTotalsRegularSeason

Author(s)

Saiem Gilani

playervsplayer  

Get NBA Stats API Player vs Player

Description

Get NBA Stats API Player vs Player

Get NBA Stats API Player vs Player
Usage

```python
nba_playervplayer(
    date_from = "", 
    date_to = "", 
    game_segment = "", 
    last_n_games = 0, 
    league_id = "00", 
    location = "", 
    measure_type = "Base", 
    month = 0, 
    opponent_team_id = 0, 
    outcome = "", 
    pace_adjust = "N", 
    per_mode = "Totals", 
    period = 0, 
    player_id = "2544", 
    plus_minus = "N", 
    rank = "N", 
    season = "2020-21", 
    season_segment = "", 
    season_type = "Regular Season", 
    vs_conference = "", 
    vs_division = "", 
    vs_player_id = "203076"
)
```

Arguments

date_from  date_from

date_to  date_to

game_segment  game_segment

last_n_games  last_n_games

league_id  League - default: '00'. Other options include '10': WNBA, '20': G-League

location  location

measure_type  measure_type

month  month

opponent_team_id  opponent_team_id

outcome  outcome

pace_adjust  pace_adjust

per_mode  per_mode

period  period

player_id  Player ID

plus_minus  plus_minus
Get NBA Stats API Playoff Picture

Description

Get NBA Stats API Playoff Picture
Get NBA Stats API Playoff Picture

Usage

nba_playoffpicture(league_id = "00", season_id = "22020")

Arguments

league_id league_id
season_id season_id

Value

Return a named list of data frames: EastConfPlayoffPicture, EastConfRemainingGames, EastConfStandings, WestConfPlayoffPicture, WestConfRemainingGames, WestConfStandings

Author(s)

Saiem Gilani
Get NBA Stats API Player Dashboard Player Tracking - Passing

Usage

```python
def nba_playerdashptpass(
    date_from = "",  
    date_to = "",  
    last_n_games = 0,  
    league_id = "00",  
    location = "",  
    month = 0,  
    opponent_team_id = 0,  
    outcome = "",  
    per_mode = "Totals",  
    player_id = "2544",  
    season = "2020-21",  
    season_segment = "",  
    season_type = "Regular Season",  
    team_id = "",  
    vs_conference = "",  
    vs_division = ""
):
    pass
```

Arguments

- `date_from`  
- `date_to`  
- `last_n_games`  
- `league_id`  
- `location`  
- `month`  
- `opponent_team_id`  
- `outcome`  
- `per_mode`  
- `player_id`  
- `season`  
- `season_segment`
season_type  season_type  
team_id  team_id  
vs_conference  vs_conference  
vs_division  vs_division  

Value

Return a named list of data frames: PassesMade, PassesReceived

Author(s)

Saiem Gilani

Get NBA Stats API Player Dashboard Player Tracking - Rebounding

Usage

nba_playerdashptreb(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  per_mode = "Totals",
  period = 0,
  player_id = "2544",
  season = "2020-21",
  season_segment = "",
  season_type = "Regular Season",
  team_id = "0",
  vs_conference = "",
  vs_division = ""
)
Arguments

date_from  date_from
date_to  date_to
game_segment  game_segment
last_n_games  last_n_games
league_id  league_id
location  location
month  month
opponent_team_id  opponent_team_id
outcome  outcome
per_mode  per_mode
period  period
player_id  player_id
season  season
season_segment  season_segment
season_type  season_type
team_id  team_id
vs_conference  vs_conference
vs_division  vs_division

Value

Return a named list of data frames: NumContestedRebounding, OverallRebounding, RebDistanceRebounding, ShotDistanceRebounding, ShotTypeRebounding

Author(s)

Saiem Gilani

Get NBA Stats API Player Dashboard Player Tracking - Defense

Description

Get NBA Stats API Player Dashboard Player Tracking - Defense
Get NBA Stats API Player Dashboard Player Tracking - Defense
Usage

nba_playerdashptshotdefend(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    per_mode = "Totals",
    period = 0,
    player_id = "2544",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    team_id = "0",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from       date_from
date_to         date_to
game_segment    game_segment
last_n_games    last_n_games
league_id       league_id
location        location
month           month
opponent_team_id opponent_team_id
outcome         outcome
per_mode        per_mode
period          period
player_id       player_id
season          season
season_segment  season_segment
season_type     season_type
team_id         team_id
vs_conference   vs_conference
vs_division     vs_division
Value

Return a named list of data frames: DefendingShots

Author(s)

Saiem Gilani

---

**Get NBA Stats API Player Dashboard Player Tracking - Shots**

**Usage**

```r
nba_playerdashptshots(
  date_from = ",",
  date_to = ",",
  game_segment = ",",
  last_n_games = 0,
  league_id = "00",
  location = ",",
  month = 0,
  opponent_team_id = 0,
  outcome = ",",
  per_mode = "Totals",
  period = 0,
  player_id = "2544",
  season = "2020-21",
  season_segment = ",",
  season_type = "Regular Season",
  team_id = "0",
  vs_conference = ",",
  vs_division = ","
)
```

**Arguments**

- `date_from`  
- `date_to`  
- `game_segment`  
- `last_n_games`  
- `league_id`
progressively

location location
month month
opponent_team_id opponent_team_id
outcome outcome
per_mode per_mode
period period
player_id player_id
season season
season_segment season_segment
season_type season_type
team_id team_id
vs_conference vs_conference
vs_division vs_division

Value

Return a named list of data frames: ClosestDefender10ftPlusShooting, ClosestDefenderShooting, DribbleShooting, GeneralShooting, Overall, ShotClockShooting, TouchTimeShooting

Author(s)

Saiem Gilani

progressively Progressively

Description

This function helps add progress-reporting to any function - given function f() and progressor p(), it will return a new function that calls f() and then (on-exiting) will call p() after every iteration.

Usage

progressively(f, p = NULL)

Arguments

f a function to add progressr functionality to.
p a progressor function as created by progressr::progressor()

Details

This is inspired by purrr’s safely, quietly, and possibly function decorators.
**p_est_metr**

**Value**

a function that does the same as `f` but it calls `p()` after iteration.

---

**p_est_metr**

Get NBA Stats API Player Estimated Metrics

**Description**

Get NBA Stats API Player Estimated Metrics

**Usage**

```r
nba_playerestimatedmetrics(
  league_id = "00",
  season = "2020-21",
  season_type = "Regular Season"
)
```

**Arguments**

- `league_id` League - default: '00'. Other options include '10': WNBA, '20': G-League
- `season` Season - format 2020-21
- `season_type` Season Type - Regular Season, Playoffs, All-Star

**Value**

Return a named list of data frames: PlayerEstimatedMetrics

**Author(s)**

Saiem Gilani

---

**p_game_log**

Get NBA Stats API Player Game Log

**Description**

Get NBA Stats API Player Game Log

Get NBA Stats API Player Game Log
Usage

nba_playergamelog(
    date_from = "",  
    date_to = "",  
    league_id = "00",  
    player_id = "2544",  
    season = "2020-21",  
    season_type = "Regular Season"  
)

Arguments

date_from    date_from
date_to      date_to
league_id    League - default: '00'. Other options include '10': WNBA, '20': G-League
player_id    Player ID
season       Season - format 2020-21
season_type  Season Type - Regular Season, Playoffs, All-Star

Value

Return a named list of data frames: PlayerGameLog

Author(s)

Saiem Gilani

Description

Get NBA Stats API Player Game Logs

Usage

nba_playergamelogs(
    date_from = "",  
    date_to = "",  
    game_segment = "",  
    last_n_games = 0,  
    league_id = "00",  
    location = "",  
    measure_type = "Base",  
)
month = 0,
opponent_team_id = 0,
outcome = "",
op_round = "",
per_mode = "Totals",
period = 0,
player_id = "2544",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
team_id = "",
vs_conference = "",
vs_division = ""
)

Arguments

date_from        date_from
date_to          date_to

game_segment     game_segment
last_n_games     last_n_games
league_id        League - default: '00'. Other options include '10': WNBA, '20': G-League
location        location
measure_type     measure_type
month            month
opponent_team_id opponent_team_id
outcome          outcome
po_round         po_round
per_mode         per_mode
period           period
player_id        Player ID
season           Season - format 2020-21
season_segment   season_segment
season_type      Season Type - Regular Season, Playoffs, All-Star
team_id          team_id
vs_conference    vs_conference
vs_division      vs_division

Value

Return a named list of data frames: PlayerGameLogs

Author(s)

Saiem Gilani
p_n_g  

Get NBA Stats API Player Next N Games

Description

Get NBA Stats API Player Next N Games
Get NBA Stats API Player Next N Games

Usage

nba_player_nextngames(
  league_id = "", 
  number_of_games = 2147483647,  
  player_id = "2544",  
  season = "2020-21",  
  season_type = "Regular Season"
)

Arguments

league_id      League - default: '00'. Other options include '10': WNBA, '20': G-League
number_of_games N in number of games
player_id      Player ID
season         Season - format 2020-21
season_type    Season Type - Regular Season, Playoffs, All-Star

Value

Return a named list of data frames: NextNGames

Author(s)

Saiem Gilani

rds_from_url  

Load .rds file from a remote connection

Description

Load .rds file from a remote connection

Usage

rds_from_url(url)
Arguments
url a character url

Value
a dataframe as created by `readRDS()`

---

**rejoin_schedules**

rejoin schedules (when used from league game finder)

**Description**
rejoin schedules (when used from league game finder)

**Usage**

```r
rejoin_schedules(df)
```

**Arguments**

- `df` data frame pulled from `nba_leaguegamefinder()`

---

**sc**

Get NBA Stats API Shot Chart Detail

**Description**
Get NBA Stats API Shot Chart Detail
Get NBA Stats API Shot Chart Detail

**Usage**

```r
nba_shotchartdetail(
  context_measure = "FGA",
  date_from = "",
  date_to = "",
  game_id = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  period = 0,
```
player_id = "202696",
player_position = "",
rookie_year = "",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
team_id = 0,
vs_conference = "",
vs_division = ""
)

**Arguments**

- context_measure
- date_from
- date_to
- game_id
- game_segment
- last_n_games
- league_id
- location
- month
- opponent_team_id
- outcome
- period
- player_id
- player_position
- rookie_year
- season
- season_segment
- season_type
- team_id
- vs_conference
- vs_division

**Value**

Return a named list of data frames: LeagueAverages, Shot_Chart_Detail

**Author(s)**

Saiem Gilani
**schedule**

---

**Get NBA Stats API Schedule**

**Description**

Get NBA Stats API Schedule

**Usage**

```r
nba_schedule(season = 2021, league = "NBA")
```

**Arguments**

- `season`: Season - 4 digit integer corresponding to the first year in the season format 2020-21
- `league`: League - default: 'NBA'. Other options include 'all'

**Value**

Returns a tibble

**Author(s)**

Saiem Gilani

---

**scoreboard**

---

**Get NBA Stats API Scoreboard**

**Description**

Get NBA Stats API Scoreboard

**Usage**

```r
nba_scoreboard(league_id = "00", game_date = "2021-07-20", day_offset = 0)
```

**Arguments**

- `league_id`: League - default: '00'. Other options include '10': WNBA, '20': G-League
- `game_date`: Game Date
- `day_offset`: Day Offset (integer 0,-1)
Value

Return a named list of data frames: Available, EastConfStandingsByDay, GameHeader, LastMeeting, LineScore, SeriesStandings, WestConfStandingsByDay

Author(s)

Saiem Gilani

---

**scoreboardv2**

**Get NBA Stats API Scoreboard V2**

Description

Get NBA Stats API Scoreboard V2

Get NBA Stats API Scoreboard V2

Usage

```r
nba_scoreboardv2(league_id = "00", game_date = "2021-07-20", day_offset = 0)
```

Arguments

- **league_id**: League - default: '00'. Other options include '10': WNBA, '20': G-League
- **game_date**: Game Date
- **day_offset**: Day Offset (integer 0,-1)

Value

Return a named list of data frames: Available, EastConfStandingsByDay, GameHeader, LastMeeting, LineScore, SeriesStandings, TeamLeaders, TicketLinks, WestConfStandingsByDay, WinProbability

Author(s)

Saiem Gilani
sc_lw

Get NBA Stats API Shot Chart League-Wide

Description

Get NBA Stats API Shot Chart League-Wide
Get NBA Stats API Shot Chart League-Wide

Usage

nba_shotchartleaguewide(league_id = "00", season = "2020-21")

Arguments

league_id League - default: '00'. Other options include '10': WNBA, '20': G-League
season

Value

Return a named list of data frames: League_Wide

Author(s)

Saiem Gilani

tbyclutch

Get NBA Stats API Team Dashboard by Clutch Splits

Description

Get NBA Stats API Team Dashboard by Clutch Splits
Get NBA Stats API Team Dashboard by Clutch Splits

Usage

nba_teamdashboardbyclutch(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
outcome = "",
pace_adjust = "N",
plus_minus = "N",
po_round = "",
per_mode = "Totals",
period = 0,
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
team_id = "1610612749",
vs_conference = "",
vs_division = ""
)

Arguments

date_from
date_from
date_to
date_to
game_segment
game_segment
last_n_games
last_n_games
league_id
league_id
location
location
measure_type
measure_type
month
month
opponent_team_id
opponent_team_id
outcome
outcome
pace_adjust
pace_adjust
plus_minus
plus_minus
po_round
po_round
per_mode
per_mode
period
period
rank
rank
season
season
season_segment
season_segment
season_type
season_type
shot_clock_range
shot_clock_range
team_id
team_id
vs_conference
vs_conference
vs_division
vs_division
**Value**

Return a named list of data frames: Last10Sec3Point2TeamDashboard, Last10Sec3PointTeamDashboard, Last1Min5PointTeamDashboard, Last1MinPlusMinus5PointTeamDashboard, Last30Sec3Point2TeamDashboard, Last30Sec3PointTeamDashboard, Last3Min5PointTeamDashboard, Last3MinPlusMinus5PointTeamDashboard, Last5Min5PointTeamDashboard, Last5MinPlusMinus5PointTeamDashboard, OverallTeamDashboard

**Author(s)**

Saiem Gilani

---

### nba_teamdashboardbygamesplits

**Get NBA Stats API Team Dashboard by Game Splits**

**Description**

Get NBA Stats API Team Dashboard by Game Splits

**Usage**

```r
nba_teamdashboardbygamesplits(
  date_from = "", 
  date_to = "", 
  game_segment = "", 
  last_n_games = 0, 
  league_id = "00", 
  location = "", 
  measure_type = "Base", 
  month = 0, 
  opponent_team_id = 0, 
  outcome = "", 
  pace_adjust = "N", 
  plus_minus = "N", 
  po_round = "", 
  per_mode = "Totals", 
  period = 0, 
  rank = "N", 
  season = "2020-21", 
  season_segment = "", 
  season_type = "Regular Season", 
  shot_clock_range = "", 
  team_id = "1610612749", 
  vs_conference = "", 
  vs_division = ""
)
```
Arguments

date_from
date_from
date_to
date_to
game_segment
game_segment
last_n_games
last_n_games
league_id
league_id
location
location
measure_type
measure_type
month
month
opponent_team_id
opponent_team_id
outcome
outcome
pace_adjust
pace_adjust
plus_minus
plus_minus
po_round
po_round
per_mode
per_mode
period
period
rank
rank
season
season
season_segment
season_segment
season_type
season_type
shot_clock_range
shot_clock_range
team_id
team_id
vs_conference
vs_conference
vs_division
vs_division

Value

Return a named list of data frames: ByActualMarginTeamDashboard, ByHalfTeamDashboard,
ByPeriodTeamDashboard, ByScoreMarginTeamDashboard, OverallTeamDashboard

Author(s)

Saiem Gilani
tbygeneralsplits Get NBA Stats API Team Dashboard by General Splits

Description

Get NBA Stats API Team Dashboard by General Splits
Get NBA Stats API Team Dashboard by General Splits

Usage

nba_teamdashboardbygeneralsplits(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    po_round = "",
    per_mode = "Totals",
    period = 0,
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from    date_from
date_to      date_to
game_segment game_segment
last_n_games last_n_games
league_id    league_id
location     location
measure_type measure_type
Value

Return a named list of data frames: DaysRestTeamDashboard, LocationTeamDashboard, MonthTeamDashboard, OverallTeamDashboard, PrePostAllStarTeamDashboard, WinsLossesTeamDashboard

Author(s)

Saiem Gilani
Usage

nba_teamdashboardbylastngames(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    po_round = "",
    per_mode = "Totals",
    period = 0,
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from       date_from
date_to         date_to
game_segment    game_segment
last_n_games    last_n_games
league_id       league_id
location        location
measure_type    measure_type
month           month
opponent_team_id opponent_team_id
outcome         outcome
pace_adjust     pace_adjust
plus_minus      plus_minus
po_round        po_round
per_mode        per_mode
period  period
rank    rank
season  season
season_segment  season_segment
season_type  season_type
shot_clock_range  shot_clock_range
team_id  team_id
vs_conference  vs_conference
vs_division  vs_division

Value
Return a named list of data frames: GameNumberTeamDashboard, Last10TeamDashboard, Last15TeamDashboard,
Last20TeamDashboard, Last5TeamDashboard, OverallTeamDashboard

Author(s)
Saiem Gilani

---

tbyopponent  Get NBA Stats API Team Dashboard by Opponent

Description
Get NBA Stats API Team Dashboard by Opponent
Get NBA Stats API Team Dashboard by Opponent

Usage
nba_teamdashboardbyopponent(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    po_round = "",
    per_mode = "Totals",
)
period = 0,  
rank = "N",  
season = "2020-21",  
season_segment = ","  
season_type = "Regular Season",  
shot_clock_range = ","  
team_id = "1610612749",  
vs_conference = ","  
vs_division = ","
)

Arguments

date_from          date_from
date_to            date_to
game_segment       game_segment
last_n_games       last_n_games
league_id          league_id
location           location
measure_type       measure_type
month              month
opponent_team_id   opponent_team_id
outcome            outcome
pace_adjust        pace_adjust
plus_minus         plus_minus
po_round           po_round
per_mode           per_mode
period             period
rank               rank
season             season
season_segment     season_segment
season_type        season_type
shot_clock_range   shot_clock_range
team_id            team_id
vs_conference      vs_conference
vs_division        vs_division

Value

Return a named list of data frames: ConferenceTeamDashboard, DivisionTeamDashboard, OpponentTeamDashboard, OverallTeamDashboard
Author(s)

Saiem Gilani

tbyshootingsplits  Get NBA Stats API Team Dashboard by Shooting Splits

Description

Get NBA Stats API Team Dashboard by Shooting Splits

Usage

nba_teamdashboardbyshootingsplits(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    po_round = "",
    per_mode = "Totals",
    period = 0,
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from  date_from
date_to   date_to
game_segment  game_segment
last_n_games  last_n_games
league_id  league_id
location    location
measure_type measure_type
month       month
opponent_team_id opponent_team_id
outcome     outcome
pace_adjust pace_adjust
plus_minus  plus_minus
po_round    po_round
per_mode    per_mode
period      period
rank        rank
season      season
season_segment season_segment
season_type season_type
shot_clock_range shot_clock_range
team_id     team_id
vs_conference vs_conference
vs_division vs_division

**Value**

Return a named list of data frames: AssistedBy, AssitedShotTeamDashboard, OverallTeamDashboard, Shot5FTTeamDashboard, Shot8FTTeamDashboard, ShotAreaTeamDashboard, ShotTypeTeamDashboard

**Author(s)**

Saiem Gilani

---

**Get NBA Stats API Team Dashboard by Team Performance**

**Description**

Get NBA Stats API Team Dashboard by Team Performance
Get NBA Stats API Team Dashboard by Team Performance
Usage

nba_teamdashboardbyteamperformance(
    date_from = "", 
    date_to = "", 
    game_segment = "", 
    last_n_games = 0, 
    league_id = "00", 
    location = "", 
    measure_type = "Base", 
    month = 0, 
    opponent_team_id = 0, 
    outcome = "", 
    pace_adjust = "N", 
    plus_minus = "N", 
    po_round = "", 
    per_mode = "Totals", 
    period = 0, 
    rank = "N", 
    season = "2020-21", 
    season_segment = "", 
    season_type = "Regular Season", 
    shot_clock_range = "", 
    team_id = "1610612749", 
    vs_conference = "", 
    vs_division = ""
)

Arguments

date_from  date_from
date_to    date_to
game_segment  game_segment
last_n_games  last_n_games
league_id    league_id
location    location
measure_type  measure_type
month    month
opponent_team_id  opponent_team_id
outcome    outcome
pace_adjust  pace_adjust
plus_minus  plus_minus
po_round  po_round
per_mode  per_mode
Return a named list of data frames: OverallTeamDashboard, PointsScoredTeamDashboard, PointsAgainstTeamDashboard, ScoreDifferentialTeamDashboard

Author(s)
Saiem Gilani

---

<table>
<thead>
<tr>
<th>tbyyearoveryear</th>
<th>Get NBA Stats API Team Dashboard Year over Year</th>
</tr>
</thead>
</table>

Description
Get NBA Stats API Team Dashboard Year over Year

Usage
```r
nba_teamdashboardbyyearoveryear(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  pace_adjust = "N",
  plus_minus = "N",
  po_round = "",
  per_mode = "Totals",
)```
period = 0,
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
team_id = "1610612749",
vs_conference = "",
vs_division = ""
)

Arguments

date_from                  date_from
date_to                    date_to
game_segment               game_segment
last_n_games               last_n_games
league_id                  league_id
location                  location
measure_type               measure_type
month                      month
opponent_team_id           opponent_team_id
outcome                    outcome
pace_adjust                pace_adjust
plus_minus                 plus_minus
po_round                   po_round
per_mode                   per_mode
period                     period
rank                       rank
season                     season
season_segment             season_segment
season_type                season_type
shot_clock_range           shot_clock_range
team_id                    team_id
vs_conference              vs_conference
vs_division                vs_division

Value

Return a named list of data frames: ByYearTeamDashboard, OverallTeamDashboard

Author(s)

Saiem Gilani
Description

Get NBA Stats API Team Dashboard - Lineups

Usage

```python
nba_teamdashlineups(
    date_from = "",
    date_to = "",
    game_id = "",
    game_segment = "",
    group_quantity = 5,
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    po_round = "",
    per_mode = "Totals",
    period = 0,
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)
```

Arguments

date_from    date_from
date_to      date_to
game_id      game_id
game_segment game_segment
group_quantity group_quantity
last_n_games  last_n_games
league_id     league_id
location     location
measure_type measure_type
month        month
opponent_team_id opponent_team_id
outcome      outcome
pace_adjust pace_adjust
plus_minus   plus_minus
po_round     po_round
per_mode     per_mode
period       period
rank         rank
season       season
season_segment season_segment
season_type  season_type
shot_clock_range shot_clock_range
team_id      team_id
vs_conference vs_conference
vs_division  vs_division

Value
Return a named list of data frames: Lineups, Overall

Author(s)
Saiem Gilani

Description
Get NBA Stats API Team Common Info
Get NBA Stats API Team Common Info
Usage

```r
nba_teaminfocommon(
    league_id = "00",
    season = "2020-21",
    season_type = "Regular Season",
    team_id = "1610612749"
)
```

Arguments

- **league_id**
  League - default: '00'. Other options include '10': WNBA, '20': G-League
- **season**
  Season - format 2020-21
- **season_type**
  Season Type - Regular Season, Playoffs, All-Star
- **team_id**
  Team ID

Value

Return a named list of data frames: AvailableSeasons, TeamInfoCommon, TeamSeasonRanks

Author(s)

Saiem Gilani

---

**teams_links**

*Men’s College Basketball KenPom Teams Dictionary Team link KenPom reference lookup for the package*

Description

Men’s College Basketball KenPom Teams Dictionary Team link KenPom reference lookup for the package

Usage

```r
teams_links
```

Format

A data frame with 357 rows and 6 variables:

- Team character.
- Team.link character.
- team.link.ref character.
- Year double.
- Conf character.
- Conf.link character.
- conf.link.ref character.
Get NBA Stats API Team Game Streak Finder

Usage

nba_teamgamestreakfinder(
    active_streaks_only = "",
    active_teams_only = "",
    btr_opp_ast = "",
    btr_opp_blk = "",
    btr_opp_dreb = "",
    btr_opp_fg3a = "",
    btr_opp_fg3m = "",
    btr_opp_fg3_pct = "",
    btr_opp_fga = "",
    btr_opp_fgm = "",
    btr_opp_fg_pct = "",
    btr_opp_fta = "",
    btr_opp_ftm = "",
    btr_opp_ft_pct = "",
    btr_opp_oreb = "",
    btr_opp_pf = "",
    btr_opp_pts = "",
    btr_opp_pts2nd_chance = "",
    btr_opp_pts_fb = "",
    btr_opp_pts_off_tov = "",
    btr_opp_pts_paint = "",
    btr_opp_reb = "",
    btr_opp_stl = "",
    btr_opp_tov = "",
    conference = "",
    date_from = "",
    date_to = "",
    division = "",
    et_ast = "",
    et_blk = "",
    et_dre = "",
    et_dreb = "",
    et_fg3a = "",
    et_fg3m = "",
    et_fg3_pct = "",
)
et_fga = "", et_fgm = "", et_fg_pct = "", et_fta = "", et_ftm = "", et_ft_pct = "", et_minutes = "", eq OPP pts2nd_chance = "", eq OPP pts_fb = "", eq OPP pts_off_tov = "", eq OPP pts_paint = "", et_oreb = "", et_pf = "", et_pts = "", eq OPP pts2nd_chance = "", eq OPP pts_fb = "", eq OPP pts_off_tov = "", eq OPP pts_paint = "", et_reb = "", et_stl = "", et_td = "", et_tov = "", game_id = "", gt_ast = "", gt_blk = "", gt_dd = "", gt_dreb = "", gt_fg3a = "", gt_fg3m = "", gt_fg3_pct = "", gt_fga = "", gt_fgm = "", gt_fg_pct = "", gt_fta = "", gt_ftm = "", gt_ft_pct = "", gt_minutes = "", gt_opp_ast = "", gt_opp_blk = "", gt_opp_dreb = "", gt_opp_fg3a = "", gt_opp_fg3m = "", gt_opp_fg3pct = "", gt_opp_fga = "", gt_opp_fgm = "", gt_opp_fg_pct = "", gt_opp_fta = "", gt_opp_ftm = "", }
gt_oppp_ft_pct = "",
gt_oppp_oreb = "",
gt_oppp_pf = "",
gt_oppp_pts = "",
gt_oppp_pts2nd_chance = "",
gt_oppp_pts_fb = "",
gt_oppp_pts_off_tov = "",
gt_oppp_pts_paint = "",
gt_oppp_reb = "",
gt_oppp_stl = "",
gt_oppp_tov = "",
gt_oreb = "",
gt_pf = "",
gt_pts = "",
gt_pts2nd_chance = "",
gt_pts_fb = "",
gt_pts_off_tov = "",
gt_pts_paint = "",
gt_reb = "",
gt_stl = "",
gt_td = "",
gt_tov = "",
lstreak = "",
league_id = "00",
location = "",
lt_ast = "",
lt_blk = "",
lt_dd = "",
lt_dreb = "",
lt_fg3a = "",
lt_fg3m = "",
lt_fg3_pct = "",
lt_fga = "",
lt_fgm = "",
lt_fg_pct = "",
lt_ft_pct = "",
lt_fgm = "",
lt_fga = "",
lt_fg3a = "",
lt_fg3m = "",
lt_fg3_pct = "",
lt_oppp_fga = "",
lt_oppp_fgm = "",
lt_oppp_fg3a = "",
lt_oppp_fg3m = "",
lt_oppp_fg3_pct = "",
lt_oppp_fga = "",
lt_oppp_fgm = "",
lt_oppp_fg3a = "",
lt_oppp_fg3m = "",
lt_oppp_fg3_pct = "",
lt_oppp_fga = "",
lt_oppp_fgm = "",
lt_oppp_fg3m = "",
lt_oppp_fg3a = "",
lt_oppp_fg3_pct = "",
lt_oppp_dreb = "",
lt_oppp_blk = "",
wrs_oppp_dreb = "",
wrs_oppp_blk = "",
wrs_oppp_fga = "",
wrs_oppp_fgm = "",
wrs_oppp_fg3m = "",
wrs_oppp_fg3a = "",
wrs_oppp_fg3_pct = "",
wrs_oppp_fta = "",
wrs_oppp_ftm = "",
wrs_oppp_ft_pct = "",
wstreak = "",
wrs_opp_oreb = "",
wrs_opp_pf = "",
lt_opp_oreb = "",
lt_opp_pf = "",
lw_opp_pts2nd_chance = "",
lw_opp_pts_fb = "",
lw_opp_pts_off_tov = "",
lw_opp_pts_paint = "",
lw_reb = "",
lw_tov = "",
lw_pts = "",
lt_oreb = "",
lt_pf = "",
lt_pts = "",
lt_opp_oreb = "",
lt_opp_pf = "",
lt_opp_pts2nd_chance = "",
lt_opp_pts_fb = "",
lt_opp_pts_off_tov = "",
lt_opp_pts_paint = "",
lt_reb = "",
lw_opp_stl = "",
lw_tov = "",
min_games = "",
outcome = "",
po_round = "",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
team_id = "",
vs_conference = "",
vs_division = "",
vs_team_id = "",
wstreak = "",
wrs_opp_oreb = "",
wrs_opp_pf = "",
wrs_opp_pts2nd_chance = "",
wrs_opp_pts_fb = "",
wrs_opp_pts_off_tov = "",
wrs_opp_pts_paint = "",
wrs_reb = "",
season = "2020-21",
wrs_opp_pf = "",
wrs_opp_pts = "",
wrs_opp_pts2nd_chance = "",
wrs_opp_pts_fb = "",
wrs_opp_pts_off_tov = "",
wrs_opp_pts_paint = "",
wrs_opp_reb = "",
wrs_opp_stl = "",
wrs_opp_tov = ""
)

Arguments

active_streaks_only
    active_streaks_only
active_teams_only
    active_teams_only
btr_opp_ast  btr_opp_ast
btr_opp_blk  btr_opp_blk
btr_opp_dreb btr_opp_dreb
btr_opp_fg3a btr_opp_fg3a
btr_opp_fg3m btr_opp_fg3m
btr_opp_fg3_pct btr_opp_fg3_pct
btr_opp_fga  btr_opp_fga
btr_opp_fgm  btr_opp_fgm
btr_opp_fg_pct btr_opp_fg_pct
btr_opp_fta  btr_opp_fta
btr_opp_ftm  btr_opp_ftm
btr_opp_ft_pct btr_opp_ft_pct
btr_opp_oreb btr_opp_oreb
btr_opp_pf  btr_opp_pf
btr_opp_pts  btr_opp_pts
btr_opp_pts2nd_chance btr_opp_pts2nd_chance
btr_opp_pts_fb btr_opp_pts_fb
btr_opp_pts_off_tov btr_opp_pts_off_tov
btr_opp_pts_paint btr_opp_pts_paint
btr_opp_reb  btr_opp_reb
btr_opp_stl  btr_opp_stl
<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>btr_opponent_tov</td>
<td>conference</td>
</tr>
<tr>
<td>date_from</td>
<td>date_from</td>
</tr>
<tr>
<td>date_to</td>
<td>date_to</td>
</tr>
<tr>
<td>division</td>
<td>division</td>
</tr>
<tr>
<td>et_assists</td>
<td>et_assists</td>
</tr>
<tr>
<td>et_blocks</td>
<td>et_blocks</td>
</tr>
<tr>
<td>et_defensive_rebounds</td>
<td>et_defensive_rebounds</td>
</tr>
<tr>
<td>et_dribbles</td>
<td>et_dribbles</td>
</tr>
<tr>
<td>et_field_goals_3_pointers</td>
<td>et_field_goals_3_pointers</td>
</tr>
<tr>
<td>et_field_goals_3_make</td>
<td>et_field_goals_3_make</td>
</tr>
<tr>
<td>et_field_goals_3_percent</td>
<td>et_field_goals_3_percent</td>
</tr>
<tr>
<td>et_field_goals Attempts</td>
<td>et_field_goals Attempts</td>
</tr>
<tr>
<td>et_field_goals made</td>
<td>et_field_goals made</td>
</tr>
<tr>
<td>et_field_goals_percent</td>
<td>et_field_goals_percent</td>
</tr>
<tr>
<td>et_minutes</td>
<td>et_minutes</td>
</tr>
<tr>
<td>eq_opponent_points_2nd_chance</td>
<td>eq_opponent_points_2nd_chance</td>
</tr>
<tr>
<td>eq_opponent_points_free</td>
<td>eq_opponent_points_free</td>
</tr>
<tr>
<td>eq_opponent_points_off_tov</td>
<td>eq_opponent_points_off_tov</td>
</tr>
<tr>
<td>eq_opponent_points_paint</td>
<td>eq_opponent_points_paint</td>
</tr>
<tr>
<td>et_offensive_rebounds</td>
<td>et_offensive_rebounds</td>
</tr>
<tr>
<td>et_points</td>
<td>et_points</td>
</tr>
<tr>
<td>eq_points_2nd_chance</td>
<td>eq_points_2nd_chance</td>
</tr>
<tr>
<td>eq_points_free</td>
<td>eq_points_free</td>
</tr>
<tr>
<td>eq_points_off_tov</td>
<td>eq_points_off_tov</td>
</tr>
<tr>
<td>eq_points_paint</td>
<td>eq_points_paint</td>
</tr>
<tr>
<td>et_rebounds</td>
<td>et_rebounds</td>
</tr>
<tr>
<td>et_stars</td>
<td>et_stars</td>
</tr>
<tr>
<td>et_turnovers</td>
<td>et_turnovers</td>
</tr>
<tr>
<td>et_treys</td>
<td>et_treys</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>game_id</td>
<td>game_id</td>
</tr>
<tr>
<td>gt_ast</td>
<td>gt_ast</td>
</tr>
<tr>
<td>gt_blk</td>
<td>gt_blk</td>
</tr>
<tr>
<td>gt_dd</td>
<td>gt_dd</td>
</tr>
<tr>
<td>gt_dreb</td>
<td>gt_dreb</td>
</tr>
<tr>
<td>gt_fg3a</td>
<td>gt_fg3a</td>
</tr>
<tr>
<td>gt_fg3m</td>
<td>gt_fg3m</td>
</tr>
<tr>
<td>gt_fg3_pct</td>
<td>gt_fg3_pct</td>
</tr>
<tr>
<td>gt_fga</td>
<td>gt_fga</td>
</tr>
<tr>
<td>gt_fgm</td>
<td>gt_fgm</td>
</tr>
<tr>
<td>gt_fg_pct</td>
<td>gt_fg_pct</td>
</tr>
<tr>
<td>gt_fta</td>
<td>gt_fta</td>
</tr>
<tr>
<td>gt_ftm</td>
<td>gt_ftm</td>
</tr>
<tr>
<td>gt_ft_pct</td>
<td>gt_ft_pct</td>
</tr>
<tr>
<td>gt_minutes</td>
<td>gt_minutes</td>
</tr>
<tr>
<td>gt_oppp førstast</td>
<td>gt_oppp førstast</td>
</tr>
<tr>
<td>gt_oppp_blk</td>
<td>gt_oppp_blk</td>
</tr>
<tr>
<td>gt_oppp_dreb</td>
<td>gt_oppp_dreb</td>
</tr>
<tr>
<td>gt_oppp_fg3a</td>
<td>gt_oppp_fg3a</td>
</tr>
<tr>
<td>gt_oppp_fg3m</td>
<td>gt_oppp_fg3m</td>
</tr>
<tr>
<td>gt_oppp_fg3_pct</td>
<td>gt_oppp_fg3_pct</td>
</tr>
<tr>
<td>gt_oppp_fga</td>
<td>gt_oppp_fga</td>
</tr>
<tr>
<td>gt_oppp_fgm</td>
<td>gt_oppp_fgm</td>
</tr>
<tr>
<td>gt_oppp_fg_pct</td>
<td>gt_oppp_fg_pct</td>
</tr>
<tr>
<td>gt_oppp_fta</td>
<td>gt_oppp_fta</td>
</tr>
<tr>
<td>gt_oppp_ftm</td>
<td>gt_oppp_ftm</td>
</tr>
<tr>
<td>gt_oppp_ft_pct</td>
<td>gt_oppp_ft_pct</td>
</tr>
<tr>
<td>gt_oppp_oreb</td>
<td>gt_oppp_oreb</td>
</tr>
<tr>
<td>gt_oppp_pf</td>
<td>gt_oppp_pf</td>
</tr>
<tr>
<td>gt_oppp_3pts</td>
<td>gt_oppp_3pts</td>
</tr>
<tr>
<td>gt_oppp_3pts2nd_chance</td>
<td>gt_oppp_3pts2nd_chance</td>
</tr>
<tr>
<td>gt_oppp_3pts_fba</td>
<td>gt_oppp_3pts_fba</td>
</tr>
<tr>
<td>gt_oppp_3pts_off_tov</td>
<td>gt_oppp_3pts_off_tov</td>
</tr>
<tr>
<td>gt_oppp_3pts_paint</td>
<td>gt_oppp_3pts_paint</td>
</tr>
<tr>
<td>gt_oppp_3pts_reb</td>
<td>gt_oppp_3pts_reb</td>
</tr>
</tbody>
</table>
gt_opp_stl  gt_opp_stl
gt_opp_tov  gt_opp_tov
gt_oreb   gt_oreb
gt_pf     gt_pf
gt_pts    gt_pts
gt_pts2nd_chance     gt_pts2nd_chance
gt_pts_fb  gt_pts_fb
gt_pts_off_tov gt_pts_off_tov
gt_pts_paint gt_pts_paint
gt_reb   gt_reb
gt_stl    gt_stl
gt_td     gt_td
gt_tov    gt_tov
lstreak lstreak
league_id league_id
location location
lt_ast    lt_ast
lt_blk    lt_blk
lt_dd     lt_dd
lt_dreb   lt_dreb
lt_fg3a   lt_fg3a
lt_fg3m   lt_fg3m
lt_fg3_pct lt_fg3_pct
lt_fga    lt_fga
lt_fgm    lt_fgm
lt_fg_pct lt_fg_pct
lt_fta    lt_fta
lt_ftm    lt_ftm
lt_ft_pct lt_ft_pct
lt_minutes lt_minutes
lt_opp_ast lt_opp_ast
lt_opp_blk lt_opp_blk
lt_opp_dreb lt_opp_dreb
lt_opp_fg3a lt_opp_fg3a
lt_opp_fg3m lt_opp_fg3m
lt_opp_fg3_pct lt_opp_fg3_pct
lt_oppp_fga  lt_oppp_fga
lt_oppp_fgm  lt_oppp_fgm
lt_oppp_fg_pct lt_oppp_fg_pct
lt_oppp_fta  lt_oppp_fta
lt_oppp_ftm  lt_oppp_ftm
lt_oppp_ft_pct lt_oppp_ft_pct
lt_oppp_oreb lt_oppp_oreb
lt_oppp_pf  lt_oppp_pf
lt_oppp_pts  lt_oppp_pts
lt_oppp_pts2nd_chance lt_oppp_pts2nd_chance
lt_oppp_pts_fb lt_oppp_pts_fb
lt_oppp_pts_off_tov lt_oppp_pts_off_tov
lt_oppp_pts_paint lt_oppp_pts_paint
lt_oppp_reb lt_oppp_reb
lt_oppp_stl lt_oppp_stl
lt_oppp_tov lt_oppp_tov
lt_oreb lt_oreb
lt_pf lt_pf
lt_pts lt_pts
lt_pts2nd_chance lt_pts2nd_chance
lt_pts_fb lt_pts_fb
lt_pts_off_tov lt_pts_off_tov
lt_pts_paint lt_pts_paint
lt_reb lt_reb
lt_stl lt_stl
lt_td lt_td
lt_tov lt_tov
min_games min_games
outcome outcome
po_round po_round
season season
season_segment season_segment
season_type season_type
team_id team_id
```
vs_conference  vs_conference  
vs_division    vs_division    
vs_team_id     vs_team_id     
wstreak        wstreak        
wrs_opponent_asts wrs_opponent_asts 
wrs_opponent_blocks wrs_opponent_blocks 
wrs_opponent_dreb   wrs_opponent_dreb   
wrs_opponent_fg3as  wrs_opponent_fg3as  
wrs_opponent_fg3ms  wrs_opponent_fg3ms  
wrs_opponent_fg3_pcts wrs_opponent_fg3_pcts  
wrs_opponent_fgas  wrs_opponent_fgas  
wrs_opponent_fgam wrs_opponent_fgam  
wrs_opponent_fg_pcts wrs_opponent_fg_pcts  
wrs_opponent_ftas wrs_opponent_ftas  
wrs_opponent_ftms wrs_opponent_ftms  
wrs_opponent_ft_pcts wrs_opponent_ft_pcts  
wrs_opponent_orebs wrs_opponent_orebs  
wrs_opponent_pf wrs_opponent_pf  
wrs_opponent_pts wrs_opponent_pts  
wrs_opponent_pts2nd_chance wrs_opponent_pts2nd_chance  
wrs_opponent_pts_fb wrs_opponent_pts_fb  
wrs_opponent_pts_off_tov wrs_opponent_pts_off_tov  
wrs_opponent_pts_paint wrs_opponent_pts_paint  
wrs_opponent_reb wrs_opponent_reb  
wrs_opponent_stls wrs_opponent_stls  
wrs_opponent_tovs wrs_opponent_tovs  
```

**Value**

Return a named list of data frames: TeamGameStreakFinderParametersResults

**Author(s)**

Saiem Gilani
thist_leaders  

Get NBA Stats API Team Historical Leaders

Description

Get NBA Stats API Team Historical Leaders

Usage

nba_teamhistoricalleaders(
    league_id = "00",
    season_id = "2020",
    team_id = "1610612749"
)

Arguments

league_id  league_id
season_id  season_id
team_id  team_id

Value

Return a named list of data frames: CareerLeadersByTeam

Author(s)

Saiem Gilani

________
tp  

Get NBA Stats API Team Player Dashboard

Description

Get NBA Stats API Team Player Dashboard

Get NBA Stats API Team Player Dashboard
Usage

nba_teamplayerdashboard(
    date_from = "", 
    date_to = "", 
    game_segment = "", 
    last_n_games = 0, 
    league_id = "00", 
    location = "", 
    measure_type = "Base", 
    month = 0, 
    opponent_team_id = 0, 
    outcome = "", 
    pace_adjust = "N", 
    plus_minus = "N", 
    po_round = "", 
    per_mode = "Totals", 
    period = 0, 
    rank = "N", 
    season = "2020-21", 
    season_segment = "", 
    season_type = "Regular Season", 
    shot_clock_range = "", 
    team_id = "1610612749", 
    vs_conference = "", 
    vs_division = ""
)

Arguments

date_from       date_from
date_to         date_to
game_segment    game_segment
last_n_games    last_n_games
league_id       league_id
location        location
measure_type    measure_type
month           month
opponent_team_id opponent_team_id
outcome         outcome
pace_adjust     pace_adjust
plus_minus      plus_minus
po_round        po_round
per_mode        per_mode
period  period
rank  rank
season  season
season_segment  season_segment
season_type  season_type
shot_clock_range  shot_clock_range
team_id  team_id
vs_conference  vs_conference
vs_division  vs_division

Value
Return a named list of data frames: PlayersSeasonTotals, TeamOverall

Author(s)
Saiem Gilani

---

**tp_onoffsummary**
Get NBA Stats API Team Player On/Off Summary

**Description**
Get NBA Stats API Team Player On/Off Summary

**Usage**
```r
nba_teamplayeronoffsummary(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  pace_adjust = "N",
  plus_minus = "N",
  po_round = "",
  per_mode = "Totals",
  period = 0,
)```
tp_onoffsummary

```r
rank = "N",
season = "2020-21",
season_segment = "",
season_type = "Regular Season",
shot_clock_range = "",
team_id = "1610612749",
vs_conference = "",
vs_division = ""
```

Arguments

date_from date_from
date_to date_to
game_segment game_segment
last_n_games last_n_games
league_id league_id
location location
measure_type measure_type
month month
opponent_team_id opponent_team_id
outcome outcome
pace_adjust pace_adjust
plus_minus plus_minus
po_round po_round
per_mode per_mode
period period
rank rank
season season
season_segment season_segment
season_type season_type
shot_clock_range shot_clock_range
team_id team_id
vs_conference vs_conference
vs_division vs_division

Value

Return a named list of data frames: OverallTeamPlayerOnOffSummary, PlayersOffCourtTeamPlayerOnOffSummary, PlayersOnCourtTeamPlayerOnOffSummary

Author(s)

Saiem Gilani
tp_onoff_det  Get NBA Stats API Team Player On/Off Details

Description

Get NBA Stats API Team Player On/Off Details
Get NBA Stats API Team Player On/Off Details

Usage

nba_teamplayeronoffdetails(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    pace_adjust = "N",
    plus_minus = "N",
    po_round = "",
    per_mode = "Totals",
    period = 0,
    rank = "N",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    shot_clock_range = "",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from  date_from
date_to    date_to
game_segment  game_segment
last_n_games  last_n_games
league_id    league_id
location    location
measure_type  measure_type
```r
t_details

<table>
<thead>
<tr>
<th>month</th>
<th>month</th>
</tr>
</thead>
<tbody>
<tr>
<td>opponent_team_id</td>
<td>opponent_team_id</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>pace_adjust</td>
<td>pace_adjust</td>
</tr>
<tr>
<td>plus_minus</td>
<td>plus_minus</td>
</tr>
<tr>
<td>po_round</td>
<td>po_round</td>
</tr>
<tr>
<td>per_mode</td>
<td>per_mode</td>
</tr>
<tr>
<td>period</td>
<td>period</td>
</tr>
<tr>
<td>rank</td>
<td>rank</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_segment</td>
<td>season_segment</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>shot_clock_range</td>
<td>shot_clock_range</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
<tr>
<td>vs_conference</td>
<td>vs_conference</td>
</tr>
<tr>
<td>vs_division</td>
<td>vs_division</td>
</tr>
</tbody>
</table>

Value

Return a named list of data frames: OverallTeamPlayerOnOffDetails, PlayersOffCourtTeamPlayerOnOffDetails, PlayersOnCourtTeamPlayerOnOffDetails

Author(s)

Saiem Gilani

---

t_details

Get NBA Stats API Team Details

Description

Get NBA Stats API Team Details

Usage

```r
nba_teamdetails(team_id = "1610612749")
```

Arguments

```r
team_id Team ID
```
Value

Return a named list of data frames: TeamAwardsChampionships, TeamAwardsConf, TeamAwardsDiv, TeamBackground, TeamHistory, TeamHof, TeamRetired, TeamSocialSites

Author(s)

Saiem Gilani

t_est_metr Get NBA Stats API Team Estimated Metrics

Description

Get NBA Stats API Team Estimated Metrics

Get NBA Stats API Team Estimated Metrics

Usage

nba_teamestimatedmetrics(
  league_id = "00",
  season = "2020-21",
  season_type = "Regular Season"
)

Arguments

league_id League - default: '00'. Other options include '10': WNBA, '20': G-League
season Season - format 2020-21
season_type Season Type - Regular Season, Playoffs, All-Star

Value

Return a named list of data frames: TeamEstimatedMetrics

Author(s)

Saiem Gilani
Description

Get NBA Stats API Team Game Log

Usage

nba_teamgamelog(
    date_from = "", 
    date_to = "", 
    league_id = "00", 
    season = "2020-21", 
    season_type = "Regular Season", 
    team_id = "1610612749"
)

Arguments

date_from    date_from
date_to      date_to
league_id    League - default: '00'. Other options include '10': WNBA, '20': G-League
season       Season - format 2020-21
season_type  Season Type - Regular Season, Playoffs, All-Star
team_id      Team ID

Value

Return a named list of data frames: TeamGameLog

Author(s)

Saiem Gilani
t_gamelogs

Get NBA Stats API Team Game Logs

Description

Get NBA Stats API Team Game Logs
Get NBA Stats API Team Game Logs

Usage

nba_teamgamelogs(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    measure_type = "Base",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    po_round = "",
    per_mode = "Totals",
    period = 0,
    player_id = "",
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from    date_from
date_to      date_to
game_segment game_segment
last_n_games last_n_games
league_id    League - default: '00'. Other options include '10': WNBA, '20': G-League
location    location
measure_type measure_type
month        month
opponent_team_id opponent_team_id
outcome  outcome
po_round  po_round
per_mode  per_mode
period    period
player_id Player ID
season    Season - format 2020-21
season_segment season_segment
season_type  Season Type - Regular Season, Playoffs, All-Star
team_id    team_id
vs_conference vs_conference
vs_division vs_division

Value
Return a named list of data frames: TeamGameLogs

Author(s)
Saiem Gilani

---

**t_ptpass**

Get NBA Stats API Team Dashboard Player Tracking - Passing

---

**Description**

Get NBA Stats API Team Dashboard Player Tracking - Passing

Get NBA Stats API Team Dashboard Player Tracking - Passing

**Usage**

```r
nba_teamdashptpass(
  date_from = "",
  date_to = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  per_mode = "Totals",
  season = "2020-21",
  season_segment = "",
  season_type = "Regular Season",
  team_id = "1610612749",
  vs_conference = "",
  vs_division = ""
)
```
Arguments

date_from    date_from
date_to    date_to
last_n_games    last_n_games
league_id    league_id
location    location
month    month
opponent_team_id    opponent_team_id
outcome    outcome
per_mode    per_mode
season    season
season_segment    season_segment
season_type    season_type
team_id    team_id
vs_conference    vs_conference
vs_division    vs_division

Value

Return a named list of data frames: PassesMade, PassesReceived

Author(s)

Saiem Gilani

---

t_ptreb    Get NBA Stats API Team Dashboard Player Tracking - Rebounding

Description

Get NBA Stats API Team Dashboard Player Tracking - Rebounding
Get NBA Stats API Team Dashboard Player Tracking - Rebounding
t_ptreb

Usage

nba_teamdashptreb(
    date_from = "",
    date_to = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "",
    month = 0,
    opponent_team_id = 0,
    outcome = "",
    per_mode = "Totals",
    period = 0,
    season = "2020-21",
    season_segment = "",
    season_type = "Regular Season",
    team_id = "1610612749",
    vs_conference = "",
    vs_division = ""
)

Arguments

date_from date_from
date_to date_to
game_segment game_segment
last_n_games last_n_games
league_id league_id
location location
month month
opponent_team_id opponent_team_id
outcome outcome
per_mode per_mode
period period
season season
season_segment season_segment
season_type season_type
team_id team_id
vs_conference vs_conference
vs_division vs_division
Value
Return a named list of data frames: NumContestedRebounding, OverallRebounding, RebDistanceRebounding, ShotDistanceRebounding, ShotTypeRebounding

Author(s)
Saiem Gilani

t_ptshots Get NBA Stats API Team Dashboard Player Tracking - Shots

Description
Get NBA Stats API Team Dashboard Player Tracking - Shots
Get NBA Stats API Team Dashboard Player Tracking - Shots

Usage
nba_teamdashptshots(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "00",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
  per_mode = "Totals",
  period = 0,
  season = "2020-21",
  season_segment = "",
  season_type = "Regular Season",
  team_id = "1610612749",
  vs_conference = "",
  vs_division = ""
)

Arguments
date_from       date_from
date_to         date_to
game_segment    game_segment
last_n_games    last_n_games
league_id       league_id
Value

Return a named list of data frames: ClosestDefender10ftPlusShooting, ClosestDefenderShooting, DribbleShooting, GeneralShooting, ShotClockShooting, TouchTimeShooting

Author(s)

Saiem Gilani

---

Get NBA Stats API Team vs Player

Description

Get NBA Stats API Team vs Player
Get NBA Stats API Team vs Player

Usage

nba_teamvsplayer(
  date_from = "",
  date_to = "",
  game_segment = "",
  last_n_games = 0,
  league_id = "00",
  location = "",
  measure_type = "Base",
  month = 0,
  opponent_team_id = 0,
  outcome = "",
)
po_round = "",  
pace_adjust = "N",  
per_mode = "Totals",  
period = 0,  
player_id = "",  
plus_minus = "N",  
rank = "N",  
season = "2020-21",  
season_segment = "",  
season_type = "Regular Season",  
shot_clock_range = "",  
team_id = "1610612749",  
vs_conference = "",  
vs_division = "",  
vs_player_id = "2544"  
)

Arguments

date_from
date_to
game_segment
game_segment
last_n_games
league_id
location
measure_type
month
month
opponent_team_id
opponent_team_id
outcome
outcome
po_round
po_round
pace_adjust
pace_adjust
per_mode
per_mode
period
period
player_id
Player ID
plus_minus
plus_minus
rank
rank
season
season
season_segment
season_segment
season_type
season_type
shot_clock_range
shot_clock_range
team_id
team_id
Return a named list of data frames: OnOffCourt, Overall, ShotAreaOffCourt, ShotAreaOnCourt, ShotAreaOverall, ShotDistanceOffCourt, ShotDistanceOnCourt, ShotDistanceOverall, vsPlayerOverall

Author(s)
Saiem Gilani
update_mbb_db  Update or create a hoopR play-by-play database

Description

update_mbb_db() updates or creates a database with hoopR play by play data of all completed and available games since 2006.

Usage

update_mbb_db(
  dbdir = ".",
  dbname = "hoopR_db",
  tblname = "hoopR_mbb_pbp",
  force_rebuild = FALSE,
  db_connection = NULL
)

Arguments

dbdire Directory in which the database is or shall be located

dbname File name of an existing or desired SQLite database within dbdir

tblname The name of the play by play data table within the database

force_rebuild Hybrid parameter (logical or numeric) to rebuild parts of or the complete play by play data table within the database (please see details for further information)

db_connection A DBIConnection object, as returned by DBI::dbConnect() (please see details for further information)

Details

This function creates and updates a data table with the name tblname within a SQLite database (other drivers via db_connection) located in dbdir and named dbname. The data table combines all play by play data for every available game back to the 2006 season and adds the most recent completed games as soon as they are available for hoopR.

The argument force_rebuild is of hybrid type. It can rebuild the play by play data table either for the whole hoopR era (with force_rebuild = TRUE) or just for specified seasons (e.g. force_rebuild = c(2019, 2020)). Please note the following behavior:

- force_rebuild = TRUE: The data table with the name tblname will be removed completely and rebuilt from scratch. This is helpful when new columns are added during the Off-Season.
- force_rebuild = c(2019, 2020): The data table with the name tblname will be preserved and only rows from the 2019 and 2020 seasons will be deleted and re-added. This is intended to be used for ongoing seasons because ESPN’s data provider can make changes to the underlying data during the week.
The parameter `db_connection` is intended for advanced users who want to use other DBI drivers, such as MariaDB, Postgres or odbc. Please note that the arguments `dbdir` and `dbname` are dropped in case a `db_connection` is provided but the argument `tblname` will still be used to write the data table into the database.

**Value**

Returns a logical value (TRUE/FALSE)

---

**update_nba_db**

**Update or create a hoopR play-by-play database**

**Description**

`update_nba_db()` updates or creates a database with hoopR play by play data of all completed and available games since 2002.

**Usage**

```r
update_nba_db(
  dbdir = ".", 
  dbname = "hoopR_db", 
  tblname = "hoopR_nba_pbp", 
  force_rebuild = FALSE, 
  db_connection = NULL
)
```

**Arguments**

- `dbdir` Directory in which the database is or shall be located
- `dbname` File name of an existing or desired SQLite database within `dbdir`
- `tblname` The name of the play by play data table within the database
- `force_rebuild` Hybrid parameter (logical or numeric) to rebuild parts of or the complete play by play data table within the database (please see details for further information)
- `db_connection` A DBIConnection object, as returned by `DBI::dbConnect()` (please see details for further information)

**Details**

This function creates and updates a data table with the name `tblname` within a SQLite database (other drivers via `db_connection`) located in `dbdir` and named `dbname`. The data table combines all play by play data for every available game back to the 2002 season and adds the most recent completed games as soon as they are available for hoopR.

The argument `force_rebuild` is of hybrid type. It can rebuild the play by play data table either for the whole hoopR era (with `force_rebuild = TRUE`) or just for specified seasons (e.g. `force_rebuild = c(2019,2020)`). Please note the following behavior:
- `force_rebuild = TRUE`: The data table with the name `tblname` will be removed completely and rebuilt from scratch. This is helpful when new columns are added during the Off-Season.
- `force_rebuild = c(2019, 2020)`: The data table with the name `tblname` will be preserved and only rows from the 2019 and 2020 seasons will be deleted and re-added. This is intended to be used for ongoing seasons because ESPN's data provider can make changes to the underlying data during the week.

The parameter `db_connection` is intended for advanced users who want to use other DBI drivers, such as MariaDB, Postgres or odbc. Please note that the arguments `dbdir` and `dbname` are dropped in case a `db_connection` is provided but the argument `tblname` will still be used to write the data table into the database.

**Value**

Returns a logical value (TRUE/FALSE)

---

<table>
<thead>
<tr>
<th>winprobabilitypbp</th>
<th>Get NBA Stats API Win Probability PBP</th>
</tr>
</thead>
</table>

**Description**

Get NBA Stats API Win Probability PBP

**Usage**

```
nba_winprobabilitypbp(game_id = "0021700807", run_type = "each second")
```

**Arguments**

- `game_id` Game ID
- `run_type` Run Type

**Value**

Return a named list of data frames:

**Author(s)**

Saiem Gilani
| year_to_season | year to season (XXXX -> XXXX-YY) |

**Description**

year to season (XXXX -> XXXX-YY)

**Usage**

year_to_season(year)

**Arguments**

| year | Four digit year (XXXX) |
Index

* Archive
  kp_pomeroy_archive_ratings, 77
* Arenas
  kp_arenas, 58
* Betting
  espn_mbb_betting, 32
  espn_nba_betting, 38
* Box
  espn_mbb_player_box, 35
  espn_mbb_team_box, 38
  espn_nba_player_box, 40
  espn_nba_team_box, 42
  kp_box, 58
* CBB
  espn_mbb_conferences, 33
  espn_mbb_game_all, 33
  espn_mbb_pbp, 34
  espn_mbb_player_box, 35
  espn_mbb_team_box, 38
* Career
  kp_player_career, 76
* Chart
  kp_team_depth_chart, 81
  kp_team_lineups, 84
* Coach
  kp_coach_history, 59
* Comparison
  kp_confstats, 65
* Conferences
  espn_mbb_conferences, 33
* Conference
  kp_conf, 60
  kp_confhistory, 64
  kp_confstats, 65
* Depth
  kp_team_depth_chart, 81
  kp_team_lineups, 84
* Efficiency
  kp_efficiency, 67
* Factors
  kp_fourfactors, 69
* FanMatch
  kp_fanmatch, 67
* Foul
  kp_foul_trouble, 68
* Four
  kp_fourfactors, 69
* Game
  espn_mbb_game_all, 33
  espn_nba_game_all, 39
  kp_box, 58
  kp_game_attrs, 70
  kp_gameplan, 70
* HCA
  kp_hca, 71
* History
  kp_coach_history, 59
  kp_confhistory, 64
  kp_team_history, 82
* Internal
  check_status, 18
* Leaders
  kp_kpoy, 72
* Login
  kp_user_pw, 87
* MBB
  espn_mbb_betting, 32
  espn_mbb_standings, 37
  espn_mbb_teams, 37
* Matrix
  kp_minutes_matrix, 73
* Minutes
  kp_minutes_matrix, 73
* NBA
  espn_nba_betting, 38
  espn_nba_game_all, 39
  espn_nba_pbp, 39
  espn_nba_player_box, 40
espn_nba_standings, 41
espn_nba_team_box, 42
espn_nba_teams, 42
* Opponent
  kp_opptracker, 74
* PBP
  espn_mbb_pbp, 34
espn_nba_pbp, 39
* Plan
  kp_gameplan, 70
* Player
  espn_mbb_player_box, 35
espn_nba_player_box, 40
kp_player_career, 76
kp_playerstats, 75
kp_team_player_stats, 85
kp_team_players, 85
* Points
  kp_pointdist, 76
* Probability
  kp_winprob, 88
* Program
  kp_program_ratings, 79
* Ratings
  kp_pomeroy_archive_ratings, 77
kp_pomeroy_ratings, 78
kp_program_ratings, 79
* Refs
  kp_officials, 73
kp_referee, 80
* Roster
  kp_height, 72
* Schedule
  kp_team_schedule, 86
* Score
  kp_box, 58
* Standings
  espn_mbb_standings, 37
espn_nba_standings, 41
* Stats
  kp_conf, 60
kp_player_career, 76
kp_team_player_stats, 85
kp_team_players, 85
* Teams
  espn_mbb_teams, 37
espn_nba_teams, 42
* Team
  espn_mbb_team_box, 38
espn_nba_team_box, 42
kp_team_history, 82
kp_team_player_stats, 85
kp_team_players, 85
kp_team_schedule, 86
kp_teamstats, 81
* Tracker
  kp_opptracker, 74
* Trends
  kp_trends, 86
* Trouble
  kp_foul_trouble, 68
* Util
  clean_team_names_NCAA_merge, 18
* Win
  kp_winprob, 88
* datasets
  nba_teams, 143
teams_links, 203
allplayers, 6
alltime, 7
assist_tracker, 8
assists, 7
base::tempdir, 24
bs_advv2, 9
bs_defensive, 10
bs_ffv2, 10
bs_match, 11
bs_miscv2, 12
bs_pt_v2, 13
bs_scoringv2, 13
bs_similarity, 14
bs_summaryv2, 15
bs_tradv2, 16
bs_usagev2, 17
check_status, 18
clean_team_names_NCAA_merge, 18
commonplayerinfo, 19
commonplayoffseries, 19
commonteamroster, 20
csv_from_url, 21
cumestatsplayer, 24
cumestatsplayergames, 25
cumestatsteam, 26
cumestatsteamgames, 27
data.table::fread, 21
data.table::fread(), 24
DBI::dbConnect(), 230, 231
dboard, 28
dcombine_anthro, 29
dcombine_drill, 29
dcombine_nsshooting, 30
dcombine_sshooting, 30
dcombine_stats, 31
defensehub, 31
espn_mbb_betting, 32
espn_mbb_conferences, 33
espn_mbb_game_all, 33
espn_mbb_pbp, 34
espn_mbb_player_box, 35
espn_mbb_rankings, 35
espn_mbb_scoreboard, 36
espn_mbb_standings, 37
espn_mbb_teams, 37
espn_nba_betting, 38
espn_nba_game_all, 39
espn_nba_pbp, 39
espn_nba_player_box, 40
espn_nba_scoreboard, 41
espn_nba_standings, 41
espn_nba_team_box, 42
espn_nba_teams, 42
fantasywidget, 43
franchisehistory, 44
franchiseleaders, 45
franchiseplayers, 46
gl_bs_similarity, 46
has_kp_user_and_pw(kp_user_pw), 87
homepageleaders, 47
homepagev2, 48
hustle_bs, 49
hustle_p, 50
hustle_pl, 52
hustle_t, 54
hustle_tl, 56
kp_arenas, 58
kp_box, 58
kp_coach_history, 59
kp_conf, 60
kp_confhistory, 64
kp_confstats, 65
kp_efficiency, 67
kp_fanmatch, 67
kp_foul_trouble, 68
kp_fourfactors, 69
kp_game_attrs, 70
kp_gameplan, 70
kp_hca, 71
kp_height, 72
kp_kpooy, 72
kp_minutes_matrix, 73
kp_officials, 73
kp_opptracker, 74
kp_password(kp_user_pw), 87
kp_player_career, 76
kp_playerstats, 75
kp_pointdist, 76
kp_pomeroy_archive_ratings, 77
kp_pomeroy_ratings, 78
kp_program_ratings, 79
kp_referee, 80
kp_team_depth_chart, 81
kp_team_history, 82
kp_team_lineups, 84
kp_team_player_stats, 85
kp_team_players, 85
kp_team_schedule, 86
kp_teamstats, 81
kp_trends, 86
kp_user_email(kp_user_pw), 87
kp_user_pw, 87
kp_winprob, 88
l_gamelog, 136
l_standings, 137
l_standingsv3, 138
1d_oppptshot, 89
1d_pbiostats, 91
1d_pclutch, 93
1d_ppptshot, 96
1d_pshotloc, 98
1d_pstats, 101
1d_ptdefend, 103
1d_ptstats, 105
1d_ptteamdefend, 107
1d_tclutch, 109
1d_tptshot, 111
INDEX

ld_tshotloc, 113
ld_tstats, 115
leaderstiles, 117
leaguedashlineups, 118
leagueleaders, 120
leagueineuupviz, 121
leagueplayerondetails, 123
leagueseasonmatchups, 124
lg_streak, 125
load_mbb_pbp, 130
load_mbb_player_box, 131
load_mbb_schedule, 131
load_mbb_team_box, 132
load_nba_pbp, 133
load_nba_player_box, 134
load_nba_schedule, 135
load_nba_team_box, 135
login (kp_user_pw), 87
make.names, 23
matchupsrollup, 139
mbb (update_mbb_db), 230
mbb_db (update_mbb_db), 230
mbb_pbp_db (update_mbb_db), 230
most_recent_mbb_season, 140
most_recent_nba_season, 140

nba (update_nba_db), 231
nba_alltimeleadersgrids (alltime), 7
nba_assistleaders (assists), 7
nba_assisttracker (assist_tracker), 8
nba_boxscoreadvancedv2 (bs_advv2), 9
nba_boxscoredefensive (bs_defensive), 10
nba_boxscorefourfactorsv2 (bs_ffv2), 10
nba_boxscorematchups (bs_match), 11
nba_boxscoremiscv2 (bs_miscv2), 12
nba_boxscoreplayertrackv2 (bs_pt_v2), 13
nba_boxscorescoringv2 (bs_scoringv2), 13
nba_boxscoresimilarityscore (bs_similarity), 14
nba_boxscoresummaryv2 (bs_summaryv2), 15
nba_boxscoretraditionalv2 (bs_tradv2), 16
nba_boxscoreusagev2 (bs_usagev2), 17
nba_commonallplayers (allplayers), 6
nba_commonplayerinfo (commonplayerinfo), 19
nba_commonplayoffseries (commonplayoffseries), 19
nba_commonteamroster (commonteamroster), 20
nba_cumestatsplayer (cumestatsplayer), 24
nba_cumestatsplayergames (cumestatsplayergames), 25
nba_cumestatsteam (cumestatsteam), 26
nba_cumestatsteamgames (cumestatsteamgames), 27
nba_data_pbp, 140
nba_db (update_nba_db), 231
nba_defensehub (defenseHub), 31
nba_draftboard (dboard), 28
nba_draftcombinedrillresults (dcombine_drill), 29
nba_draftcombinenonstationaryshooting (dcombine_nsshooting), 30
nba_draftcombineplayeranthro (dcombine_anthro), 29
nba_draftcombinespotsighting (dcombine_sshooting), 30
nba_draftcombinestats (dcombine_stats), 31
nba_fantasywidget (fantasywidget), 43
nba_franchisehistory (franchisehistory), 44
nba_franchiseleaders (franchiseleaders), 45
nba_franchiseplayers (franchiseplayers), 46
nba_glalumboxscoresimilarityscore (gl_bs_similarity), 46
nba_homepageleaders (homepageleaders), 47
nba_homepagev2 (homepagev2), 48
nba_hustlestatsboxscore (hustle_bs), 49
nba_leaderstiles (leaderstiles), 117
nba_leaguedashlineups (leaguedashlineups), 118
nba_leaguedashoppptshot (ld_oppptshot), 89
nba_leaguedashplayerbiostats (ld_pbiostats), 91
nba_leaguedashplayerclutch (ld_pclutch), 93
nba_leaguedashplayerptshot (ld_pptshot), 96
nba_leaguedashplayershotlocations
(ld_pshotloc), 98
nba_leaguedashplayerstats (ld_pstats),
  101
nba_leaguedashptdefend (ld_ptdefend),
  103
nba_leaguedashptstats (ld_pstats), 105
nba_leaguedashptteamdefend
  (ld_ptteamdefend), 107
nba_leaguedashteamclutch (ld_tclutch),
  109
nba_leaguedashteamptshot (ld_tptshot),
  111
nba_leaguedashteamshotlocations
  (ld_tshotloc), 113
nba_leaguedashteamstats (ld_tstats), 115
nba_leaguegamefinder (lg_streak), 125
nba_leaguegamelog (l_gamelog), 136
nba_leaguehustlestatsplayer (hustle_p), 50
nba_leaguehustlestatsplayerleaders
  (hustle_pl), 52
nba_leaguehustlestatssteam (hustle_t), 54
nba_leaguehustlestatsteamleaders
  (hustle_tl), 56
nba_leagueleaders (leagueleaders), 120
nba_leaguelineupviz (leaguelineupviz), 121
nba_leagueplayerondetails
  (leagueplayerondetails), 123
nba_leagueseasonmatchups
  (leagueseasonmatchups), 124
nba_leaguestandings (l_standings), 137
nba_leaguestandingsv3 (l_standingsv3), 138
nba_matchupsrollup (matchupsrollup), 139
nba_pbp (pbp), 145
nba_pbp_db (update_nba_db), 231
nba_playerawards (playerawards), 167
nba_playercareerbycollege
  (pcareerbycollege), 159
nba_playercareerbycollege_rollup
  (pcareerbycollege_ru), 159
nba_playercareerstats
  (playercareerstats), 167
nba_playercompare (playercompare), 168
nba_playerdashboardbyclutch
  (pbyclutch), 145
nba_playerdashboardbygamesplits
  (pbygamesplits), 147
nba_playerdashboardbygeneralsplits
  (pbygeneralsplits), 149
nba_playerdashboardbylastngames
  (pbylastngames), 150
nba_playerdashboardbyopponent
  (pbyopponent), 152
nba_playerdashboardbyshottingsplits
  (pbyshottingsplits), 154
nba_playerdashboardbyteamperformance
  (pbyteamperformance), 155
nba_playerdashboardbyyearoveryear
  (pbyyearoveryear), 157
nba_playerdashptpass (pptpass), 173
nba_playerdashptreb (pptreb), 174
nba_playerdashptshotdefend
  (pptshotdefend), 175
nba_playerdashptshots (pptshots), 177
nba_playerestimatedmetrics
  (p_est_metr), 179
nba_playerfantasyprofile (pfantasy), 160
nba_playerfantasyprofilebargraph
  (pfantasy_bg), 161
nba_playergamelog (p_gamelog), 179
nba_playergamelogs (p_gamelogs), 180
nba_playergamelog (l_gamelog), 179
nba_playergamestreakfinder (pg_streak), 162
nba_playergamedetails (p_n_g), 182
nba_playerprofilev2 (playerprofilev2), 170
nba_playervsplayer (playervsplayer), 170
nba_playoffpicture (po_picture), 172
nba_scheduledisplay (schedule), 185
nba_schedule (schedule), 185
nba_scoreboard (scoreboard), 185
nba_scoreboardv2 (scoreboardv2), 186
nba_shotchartdetail (sc), 183
nba_shotchartleaguewide (sc_lw), 187
nba_stats_videodetails, 141
nba_stats_videoevents, 143
nba_teamdashboardbyclutch (tbyclutch),
  187
nba_teamdashboardbygamesplits
  (tbygamesplits), 189
nba_teamdashboardbygeneralsplits
  (tbygeneralsplits), 191
nba_teamdashboardbylastngames
  (tbylastngames), 192
nba_teamdashboardbyopponent
tbyshootingsplits, 196
byteamperformance, 197
tbyyearoveryear, 199
teamdashlineups, 201
teaminfo, 202
teams_links, 203
tg_streak, 204
thist_leaders, 214
tp, 214
tp_onoff_det, 218
tp_onoffsummary, 216
update_mbb_db, 230
update_nba_db, 231
utils::read.csv, 22
utils::write.csv, 24
winprobabilitypbp, 232
yaml.load, 23
year_to_season, 233