Package ‘hoopR’

June 18, 2022

Title  Access Men's Basketball Play by Play Data

Version  1.8.0

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://github.com/sportsdataverse/hoopR,
     http://hoopr.sportsdataverse.org/

BugReports  https://github.com/sportsdataverse/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, httr (>= 0.5), janitor, jsonlite, magrittr, progressr (>= 0.6.0), purrr (>= 0.3.0), 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, qs (>= 0.25.1), rmarkdown, RSQLite, stats, stringi, testthat, tibble (>= 3.0), xml2 (>= 1.3), yaml

Encoding  UTF-8

LazyData  true

RoxygenNote  7.2.0

NeedsCompilation  no

Author  Saiem Gilani [aut, cre],
        Jason Lee [ctb],
        Billy Fryer [ctb]
Maintainer  Saiem Gilani <saiem.gilani@gmail.com>
Repository  CRAN
Date/Publication  2022-06-17 23:40:02 UTC

R topics documented:

  allplayers .................................................. 6
  alltime ....................................................... 7
  assists ....................................................... 8
  assist_tracker ................................................ 8
  bs_advv2 ................................................... 9
  bs_defensive ................................................ 10
  bs_ffv2 ..................................................... 11
  bs_match .................................................... 12
  bs_miscv2 .................................................. 12
  bs_pt_v2 ................................................... 13
  bs_scoringv2 ............................................... 14
  bs_similarity ............................................... 15
  bs_summaryv2 ............................................... 16
  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
  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_player_stats ..................................... 35
  espn_mbb_rankings ......................................... 36
  espn_mbb_scoreboard ....................................... 37
  espn_mbb_standings ......................................... 37
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>espn_mbb_teams</td>
<td>38</td>
</tr>
<tr>
<td>espn_mbb_team_box</td>
<td>38</td>
</tr>
<tr>
<td>espn_mbb_team_stats</td>
<td>39</td>
</tr>
<tr>
<td>espn_mbb_wp</td>
<td>40</td>
</tr>
<tr>
<td>espn_nba_betting</td>
<td>41</td>
</tr>
<tr>
<td>espn_nba_game_all</td>
<td>41</td>
</tr>
<tr>
<td>espn_nba_pbp</td>
<td>42</td>
</tr>
<tr>
<td>espn_nba_player_box</td>
<td>43</td>
</tr>
<tr>
<td>espn_nba_player_stats</td>
<td>43</td>
</tr>
<tr>
<td>espn_nba_scoreboard</td>
<td>44</td>
</tr>
<tr>
<td>espn_nba_standings</td>
<td>45</td>
</tr>
<tr>
<td>espn_nba_teams</td>
<td>45</td>
</tr>
<tr>
<td>espn_nba_team_box</td>
<td>46</td>
</tr>
<tr>
<td>espn_nba_team_stats</td>
<td>47</td>
</tr>
<tr>
<td>espn_nba_wp</td>
<td>47</td>
</tr>
<tr>
<td>fantasywidget</td>
<td>48</td>
</tr>
<tr>
<td>franchisehistory</td>
<td>50</td>
</tr>
<tr>
<td>franchiseleaders</td>
<td>50</td>
</tr>
<tr>
<td>franchiseplayers</td>
<td>51</td>
</tr>
<tr>
<td>game_rotation</td>
<td>52</td>
</tr>
<tr>
<td>gl_bs_similarity</td>
<td>52</td>
</tr>
<tr>
<td>homepageleaders</td>
<td>53</td>
</tr>
<tr>
<td>homepagev2</td>
<td>54</td>
</tr>
<tr>
<td>hustle_bs</td>
<td>55</td>
</tr>
<tr>
<td>hustle_p</td>
<td>56</td>
</tr>
<tr>
<td>hustle_pl</td>
<td>58</td>
</tr>
<tr>
<td>hustle_t</td>
<td>60</td>
</tr>
<tr>
<td>hustle_tl</td>
<td>62</td>
</tr>
<tr>
<td>kp_arenas</td>
<td>64</td>
</tr>
<tr>
<td>kp_box</td>
<td>64</td>
</tr>
<tr>
<td>kp_coach_history</td>
<td>65</td>
</tr>
<tr>
<td>kp_conf</td>
<td>66</td>
</tr>
<tr>
<td>kp_confhistory</td>
<td>70</td>
</tr>
<tr>
<td>kp_confstats</td>
<td>71</td>
</tr>
<tr>
<td>kp_efficiency</td>
<td>73</td>
</tr>
<tr>
<td>kp_fanmatch</td>
<td>73</td>
</tr>
<tr>
<td>kp_foul_trouble</td>
<td>74</td>
</tr>
<tr>
<td>kp_fourfactors</td>
<td>75</td>
</tr>
<tr>
<td>kp_gameplan</td>
<td>76</td>
</tr>
<tr>
<td>kp_game_attrs</td>
<td>76</td>
</tr>
<tr>
<td>kp_hca</td>
<td>77</td>
</tr>
<tr>
<td>kp_height</td>
<td>78</td>
</tr>
<tr>
<td>kp_kpoy</td>
<td>78</td>
</tr>
<tr>
<td>kp_minutes_matrix</td>
<td>79</td>
</tr>
<tr>
<td>kp_officials</td>
<td>79</td>
</tr>
<tr>
<td>kp_opptracker</td>
<td>80</td>
</tr>
<tr>
<td>kp_playerstats</td>
<td>81</td>
</tr>
<tr>
<td>kp_player_career</td>
<td>82</td>
</tr>
</tbody>
</table>
R topics documented:

kp_pointdist ................................................. 82
kp_pomeroy_archive_ratings .................................. 83
kp_pomeroy_ratings ............................................ 84
kp_program_ratings ............................................ 85
kp_referee .................................................... 86
kp_teamstats .................................................. 87
kp_team_depth_chart ......................................... 87
kp_team_history ............................................... 88
kp_team_lineups .............................................. 90
kp_team_players .............................................. 91
kp_team_player_stats ......................................... 91
kp_team_schedule ............................................. 92
kp_trends ...................................................... 92
kp_user_pw .................................................... 93
kp_winprob ..................................................... 95
ld_oppptshot ................................................ 95
ld_pbiostats .................................................. 97
ld_pclutch .................................................... 99
ld_pptshot ..................................................... 102
ld_pshotloc ................................................... 104
ld_pstats ...................................................... 107
ld_ptdefend .................................................. 109
ld_ptstats ..................................................... 111
ld_ptteamdefend ............................................. 113
ld_tclutch ................................................... 115
ld_tptshot .................................................... 117
ld_tshotloc ................................................... 119
ld_tstats ....................................................... 121
leaderstiles .................................................. 123
leaguedashlineups .......................................... 124
leagueleaders .................................................. 126
leaguelineupviz ............................................... 127
leagueplayerondetails ........................................ 129
leagueseasonmatchups ....................................... 130
lg_streak ...................................................... 131
load_mbb_pbp ................................................. 136
load_mbb_player_box ....................................... 137
load_mbb_schedule .......................................... 137
load_mbb_team_box ......................................... 138
load_nba_pbp ................................................ 139
load_nba_player_box ....................................... 140
load_nba_schedule .......................................... 141
load_nba_team_box ......................................... 141
l_gamelog ...................................................... 142
l_standings .................................................... 143
l_standingsv3 ................................................ 144
matchupsrollup ............................................... 145
most_recent_mbb_season ..................................... 146
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>most_recent_nba_season</td>
<td>146</td>
</tr>
<tr>
<td>nbagl_pbp</td>
<td>146</td>
</tr>
<tr>
<td>nbagl_players</td>
<td>147</td>
</tr>
<tr>
<td>nbagl_schedule</td>
<td>147</td>
</tr>
<tr>
<td>nbagl_standings</td>
<td>148</td>
</tr>
<tr>
<td>nba_data_pbp</td>
<td>148</td>
</tr>
<tr>
<td>nba_drafthistory</td>
<td>149</td>
</tr>
<tr>
<td>nba_stats_videodetails</td>
<td>150</td>
</tr>
<tr>
<td>nba_stats_videoevents</td>
<td>152</td>
</tr>
<tr>
<td>nba_teams</td>
<td>152</td>
</tr>
<tr>
<td>ncaa_mbb_NET_rankings</td>
<td>153</td>
</tr>
<tr>
<td>pbp</td>
<td>154</td>
</tr>
<tr>
<td>pbps</td>
<td>154</td>
</tr>
<tr>
<td>phyclutch</td>
<td>155</td>
</tr>
<tr>
<td>phygamesplits</td>
<td>157</td>
</tr>
<tr>
<td>phygeneralsplits</td>
<td>158</td>
</tr>
<tr>
<td>phylastngames</td>
<td>160</td>
</tr>
<tr>
<td>phyopponent</td>
<td>162</td>
</tr>
<tr>
<td>physhootingsplits</td>
<td>163</td>
</tr>
<tr>
<td>phyteamerformance</td>
<td>165</td>
</tr>
<tr>
<td>phyyearoveryear</td>
<td>167</td>
</tr>
<tr>
<td>pcareerbycollege</td>
<td>168</td>
</tr>
<tr>
<td>pcareerbycollege_ru</td>
<td>169</td>
</tr>
<tr>
<td>pfantasy</td>
<td>170</td>
</tr>
<tr>
<td>pfantasy_bg</td>
<td>171</td>
</tr>
<tr>
<td>pg_streak</td>
<td>172</td>
</tr>
<tr>
<td>playerawards</td>
<td>177</td>
</tr>
<tr>
<td>playercareerstats</td>
<td>177</td>
</tr>
<tr>
<td>playercompare</td>
<td>178</td>
</tr>
<tr>
<td>playerprofilev2</td>
<td>180</td>
</tr>
<tr>
<td>playervsplayer</td>
<td>180</td>
</tr>
<tr>
<td>po_picture</td>
<td>182</td>
</tr>
<tr>
<td>pptpass</td>
<td>183</td>
</tr>
<tr>
<td>pptreb</td>
<td>184</td>
</tr>
<tr>
<td>pptshotdefend</td>
<td>185</td>
</tr>
<tr>
<td>pptshots</td>
<td>187</td>
</tr>
<tr>
<td>progressively</td>
<td>188</td>
</tr>
<tr>
<td>p_est_metr</td>
<td>189</td>
</tr>
<tr>
<td>p_gamelog</td>
<td>189</td>
</tr>
<tr>
<td>p_gamelogs</td>
<td>190</td>
</tr>
<tr>
<td>p_headshot</td>
<td>192</td>
</tr>
<tr>
<td>p_index</td>
<td>192</td>
</tr>
<tr>
<td>p_n_g</td>
<td>193</td>
</tr>
<tr>
<td>rds_from_url</td>
<td>194</td>
</tr>
<tr>
<td>rejoin_schedules</td>
<td>195</td>
</tr>
<tr>
<td>sc</td>
<td>195</td>
</tr>
<tr>
<td>schedule</td>
<td>197</td>
</tr>
<tr>
<td>scoreboard</td>
<td>197</td>
</tr>
</tbody>
</table>
Index 246

| allplayers | Get NBA Stats API All Players |

Description

Get NBA Stats API All Players

Get NBA Stats API All Players

Usage

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

- `is_only_current_season`
  - Is only current season
- `league_id`
  - League ID
- `season`
  - Season

Value

Return a named list of data frames: CommonAllPlayers

Author(s)

Saiem Gilani

---

**alltime**

Get NBA Stats API All-time Leaders Grid

Description

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, GPILeaders, OREBLeaders, PFLeaders, PTSLeaders, REBLeaders, STLLeaders, TOVLeaders

Author(s)

Saiem Gilani
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

Get NBA Stats API Assist Tracker

**Description**

Get NBA Stats API Assist Tracker

Get NBA Stats API Assist Tracker
Usage

nba_assisttracker(
    league_id = "00",
    per_mode = "PerGame",
    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

Returns a named list of data frames: AssistTracker

Author(s)

Saiem Gilani

bs_advv2

Get NBA Stats API Boxscore Advanced V2

Description

Get NBA Stats API Boxscore Advanced V2
Get NBA Stats API Boxscore Advanced V2

Usage

nba_boxscoreadvancedv2(
    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: PlayerStats, TeamStats

Author(s)

Saiem Gilani

---

**bs_defensive**

Get NBA Stats API Boxscore Defensive

Description

Get NBA Stats API Boxscore Defensive
Get NBA Stats API Boxscore Defensive

Usage

nba_boxscoredefensive(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

Returns a named list of data frames: PlayerDefensiveStats, Table1

Author(s)

Saiem Gilani
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
)

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_match

Get NBA Stats API Boxscore Matchups

Description

Get NBA Stats API Boxscore Matchups
Get NBA Stats API Boxscore Matchups

Usage

nba_boxscorematchups(game_id)

Arguments

game_id  Game ID

Value

Returns a named list of data frames: PlayerMatchupsStats

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

nba_boxscoremiscv2(
    game_id,
    start_period = 0,
    end_period = 14,
    start_range = 0,
    end_range = 0,
    range_type = 0
)
bs_pt_v2

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: 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
Get NBA Stats API Boxscore Player Tracking V2

Usage

nba_boxscoreplayertrackv2(game_id)

Arguments

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

Value

Returns a named list of data frames: PlayerStats, TeamStats

Author(s)

Saiem Gilani
Get NBA Stats API Boxscore Scoring V2

Description

Get NBA Stats API Boxscore Scoring V2

Usage

```r
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>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: sqlPlayersScoring, sqlTeamsScoring

Author(s)

Saiem Gilani
bs_similarity  Get NBA Stats API Boxscore Similarity Score

Description

Get NBA Stats API Boxscore Similarity Score

Usage

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

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

bs_tradv2  Get NBA Stats API Boxscore Traditional V2

Description
Get NBA Stats API Boxscore Traditional V2
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  
)

bs_usagev2

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

bs_usagev2  Get NBA Stats API Boxscore Usage V2

Description

Get NBA Stats API Boxscore Usage V2

Usage

nba_boxscoreusagev2(
    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: sqlPlayersUsage, sqlTeamsUsage

**Author(s)**

Saiem Gilani

---

**check_status**

**Check Status function**

**Description**

Check Status function

**Usage**

check_status(res)

**Arguments**

- res: Response from API

---

**clean_team_names_NCAA_merge**

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

**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

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
csv_from_url

Load .csv / .csv.gz file from a remote connection

Description

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

Usage

csv_from_url(...)

Arguments

...  
Arguments passed on to data.table::fread

input A single character string. The value is inspected and deferred to either
file= (if no \n present), text= (if at least one \n is present) or cmd= (if no
\n is present, at least one space is present, and it isn’t a file name). Exactly
one of input=, file=, text=, or cmd= should be used in the same call.

file  File name in working directory, path to file (passed through
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.

text  The input data itself as a character vector of one or more lines, for example
as returned by readLines().

cmd  A shell command that pre-processes the file; e.g. fread(cmd=paste("grep",word,"filename")).
See Details.

sep  The separator between columns. Defaults to the character in the set [\t
|;] 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 base::readLines does.

sep2  The separator within columns. A list column will be returned where
each cell is a vector of values. This is much faster using less working mem-
ory than strsplit afterwards or similar techniques. For each column sep2
can be different and is the first character in the same set above \[,\t\];,
other than sep, that exists inside each field outside quoted regions in the
sample. NB: sep2 is not yet implemented.

nrows  The maximum number of rows to read. Unlike 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 fread almost
instantly using the large sample of lines. 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.

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 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

Usage

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

Arguments

game_ids    game_ids
league_id   league_id
player_id   player_id
season      season
season_type season_type
team_id     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

Get NBA Stats API Cumulative Player Game Stats

Usage

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

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
</tr>
<tr>
<td>outcome</td>
<td>outcome</td>
</tr>
<tr>
<td>player_id</td>
<td>player_id</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</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>
</tbody>
</table>

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

- game_ids
- league_id
- season
- season_type
- team_id

Value

Returns a named list of data frames: GameByGameStats, TotalTeamStats

Author(s)

Saiem Gilani
cumestatsteamgames

---

**Get NBA Stats API Cumulative Team Game Stats**

**Description**

Get NBA Stats API Cumulative Team Game Stats

**Usage**

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

**Value**

Returns a named list of data frames: CumeStatsTeamGames

**Author(s)**

Saiem Gilani
Get NBA Stats API Draft Board

Description

Get NBA Stats API Draft Board

Usage

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

Arguments

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

Value

Returns a named list of data frames: DraftBoard

Author(s)

Saiem Gilani
Get NBA Stats API Draft Combine Player Anthropological Measurements

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

Arguments

<table>
<thead>
<tr>
<th>league_id</th>
<th>league_id</th>
</tr>
</thead>
<tbody>
<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

Get NBA Stats API Draft Combine Drill Results

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

Arguments

<table>
<thead>
<tr>
<th>league_id</th>
<th>league_id</th>
</tr>
</thead>
<tbody>
<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

---

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

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

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

**Arguments**
- `league_id`  
- `season_year`

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

Author(s)
Saiem Gilani

---

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

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

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

**Arguments**
- `league_id`  
- `season_year`
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

<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_year</td>
<td>season_year</td>
</tr>
</tbody>
</table>

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, Today, 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
espn_mbb_conferences

Examples

try(espn_mbb_betting(game_id = 401256760))

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

game_id Game ID
Value

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

Author(s)

Saiem Gilani

Examples

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

**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

game_id Game ID

Value
A player boxscore data frame

Author(s)
Saiem Gilani

Examples

try(espn_mbb_player_box(game_id = 401256760))

espn_mbb_player_stats
Get ESPN men’s college basketball player stats data

Description
Get ESPN men’s college basketball player stats data

Usage
espn_mbb_player_stats(athlete_id, year, season_type = "regular", total = FALSE)

Arguments

athlete_id Athlete ID
year Year
season_type (character, default: regular): Season type - regular or postseason
total (boolean, default: FALSE): Totals
Value

Returns a tibble with the player stats data

Author(s)

Saiem Gilani

Examples

try(espn_mbb_player_stats(athlete_id = 4433134, year = 2021))

---

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())
espn_mbb_scoreboard  Get ESPN men’s college basketball schedule for a specific year

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 2022-02-15
try(espn_mbb_scoreboard(season = "20220215"))

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
**espn_mbb_teams**

Get ESPN men’s college basketball team names and IDs

**Usage**

```r
espn_mbb_teams()
```

**Value**

A teams data frame

**Author(s)**

Saiem Gilani

**Examples**

```r
try(espn_mbb_teams())
```

---

**espn_mbb_team_box**

Get ESPN men’s college basketball team box scores

**Usage**

```r
espn_mbb_team_box(game_id)
```

**Arguments**

- `game_id` Game ID
Value
A team boxscore data frame

Author(s)
Saiem Gilani

Examples

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

---

**espn_mbb_team_stats** Get ESPN men’s college basketball team stats data

Description
Get ESPN men’s college basketball team stats data

Usage
```
espn_mbb_team_stats(team_id, year, season_type = "regular", total = FALSE)
```

Arguments
- **team_id**: Team ID
- **year**: Year
- **season_type**: (character, default: regular): Season type - regular or postseason
- **total**: (boolean, default: FALSE): Totals

Value
Returns a tibble with the team stats data

Author(s)
Saiem Gilani

Examples

```r
try(espn_mbb_team_stats(team_id = 52, year = 2020))
```
Men’s College Basketball win probability chart data from ESPN

Usage

`espn_mbb_wp(game_id)`

Arguments

- `game_id` *(Integer required): Game ID filter for querying a single game*

Value

`espn_mbb_wp()` - A data frame with 9 variables:

- `game_id`: character. Referencing game ID (should be same as `game_id` from other functions).
- `play_id`: character. Referencing play ID.
- `period`: integer. Clock (time) left in the game.
- `time_left`: character. Clock (time) left in the game.
- `period_seconds_left`: integer. Seconds left in the period.
- `game_seconds_left`: integer. Seconds left in the game.
- `home_win_percentage`: double. The probability of the home team winning the game.
- `away_win_percentage`: double. The probability of the away team winning the game (calculated as 1 - `home_win_percentage` - `tie_percentage`).
- `tie_percentage`: double. The probability of the game ending the final period in a tie.

Examples

`espn_mbb_wp(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

Examples
try(espn_nba_betting(game_id = 401283399))

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

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**

```r
try(espn_nba_player_box(game_id = 401283399))
```

---

**espn_nba_player_stats**  Get ESPN NBA player stats data

**Description**

Get ESPN NBA player stats data

**Usage**

`espn_nba_player_stats(athlete_id, year, season_type = "regular", total = FALSE)`

**Arguments**

- `athlete_id` Athlete ID
- `year` Year
- `season_type` (character, default: regular): Season type - regular or postseason
- `total` (boolean, default: FALSE): Totals
**Value**

Returns a tibble with the player stats data

**Author(s)**

Saiem Gilani

**Examples**

```r
try(espn_nba_player_stats(athlete_id = 4433134, year = 2022))
```

---

**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**

```r
espn_nba_scoreboard(season)
```

**Arguments**

- `season` Either numeric or character (YYYYMMDD)

**Value**

Returns a tibble

**Examples**

```r
# Get schedule from date 2022-02-15 (returns 1000 results, max allowable.)
try(espn_nba_scoreboard (season = "20220215"))
```
**espn_nba_standings**

Get ESPN NBA's Standings

**Description**
Get ESPN NBA's Standings

**Usage**
espn_nba_standings(year)

**Arguments**

<table>
<thead>
<tr>
<th>year</th>
<th>Either numeric or character (YYYY)</th>
</tr>
</thead>
</table>

**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

| game_id       | Game ID |

Value

A team boxscore data frame

Author(s)

Saiem Gilani

Examples

`try(espn_nba_team_box(game_id = 401283399))`
**espn_nba_team_stats**  Get ESPN NBA team stats data

**Description**

Get ESPN NBA team stats data

**Usage**

```
espn_nba_team_stats(team_id, year, season_type = "regular", total = FALSE)
```

**Arguments**

- `team_id`  Team ID
- `year`  Year
- `season_type`  (character, default: regular): Season type - regular or postseason
- `total`  (boolean, default: FALSE): Totals

**Value**

Returns a tibble with the team stats data

**Author(s)**

Saiem Gilani

**Examples**

```
try(espn_nba_team_stats(team_id = 18, year = 2020))
```

---

**espn_nba_wp**  Get NBA win probability chart data from ESPN

**Description**

Get NBA win probability chart data from ESPN

**Usage**

```
espn_nba_wp(game_id)
```

**Arguments**

- `game_id`  (Integer required): Game ID filter for querying a single game
Value

`espn_nba_wp()` - A data frame with 21 variables:

<table>
<thead>
<tr>
<th>col_name</th>
<th>types</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>numeric</td>
</tr>
<tr>
<td>play_id</td>
<td>character</td>
</tr>
<tr>
<td>home_win_percentage</td>
<td>numeric</td>
</tr>
<tr>
<td>away_win_percentage</td>
<td>numeric</td>
</tr>
<tr>
<td>tie_percentage</td>
<td>numeric</td>
</tr>
<tr>
<td>sequence_number</td>
<td>character</td>
</tr>
<tr>
<td>text</td>
<td>character</td>
</tr>
<tr>
<td>away_score</td>
<td>integer</td>
</tr>
<tr>
<td>home_score</td>
<td>integer</td>
</tr>
<tr>
<td>scoring_play</td>
<td>logical</td>
</tr>
<tr>
<td>score_value</td>
<td>integer</td>
</tr>
<tr>
<td>participants</td>
<td>list</td>
</tr>
<tr>
<td>shooting_play</td>
<td>logical</td>
</tr>
<tr>
<td>type_id</td>
<td>character</td>
</tr>
<tr>
<td>type_text</td>
<td>character</td>
</tr>
<tr>
<td>period_number</td>
<td>integer</td>
</tr>
<tr>
<td>period_display_value</td>
<td>character</td>
</tr>
<tr>
<td>clock_display_value</td>
<td>character</td>
</tr>
<tr>
<td>team_id</td>
<td>character</td>
</tr>
<tr>
<td>coordinate_x</td>
<td>integer</td>
</tr>
<tr>
<td>coordinate_y</td>
<td>integer</td>
</tr>
</tbody>
</table>

Examples

`espn_nba_wp(game_id = 401283399)`

---

**fantasywidget**

**Get NBA Stats API Fantasy Widget**

Description

**Get NBA Stats API Fantasy Widget**

Usage

```r
nba_fantasywidget(
    active_players = "N",
```
```python
date_from = "",
date_to = "",
lst_n_games = 0,
league_id = "00",
location = "",
month = "",
opponent_team_id = "",
opponent_team_id = "",
po_round = "",
player_id = "",
position = "",
season = "2019-20",
season_segment = "",
season_type = "Regular Season",
team_id = "",
todays_opponent = 0,
todays_players = "N",
vs_conference = "",
vs_division = ""
)

Arguments

active_players  active_players
date_from       date_from date_from
date_to         date_to date_to
lst_n_games     lst_n_games
league_id       league_id
location        location
month           month
opponent_team_id opponent_team_id
po_round         po_round
player_id        player_id
position         position
season           season
season_segment   season_segment
season_type      season_type
team_id          team_id
todays_opponent  todays_opponent
todays_players   todays_players
vs_conference    vs_conference
vs_division      vs_division
```
franchisehistory

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

Description

Get NBA Stats API Franchise Leaders

Usage

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

Value

Returns a named list of data frames: FantasyWidgetResult

Author(s)

Saiem Gilani
Arguments

<table>
<thead>
<tr>
<th>league_id</th>
<th>league_id</th>
</tr>
</thead>
<tbody>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
</tbody>
</table>

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 Get NBA Stats API Franchise Players

Usage

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

Arguments

<table>
<thead>
<tr>
<th>league_id</th>
<th>league_id</th>
</tr>
</thead>
<tbody>
<tr>
<td>per_mode</td>
<td>per_mode</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: FranchisePlayers

Author(s)

Saiem Gilani
game_rotation  Get NBA Stats API Game Rotation

Description

Get NBA Stats API Game Rotation
Get NBA Stats API Game Rotation

Usage

nba_gamerotation(game_id, league_id = "00", rotation_stat = "PLAYER_PTS")

Arguments

game_id  Game ID
league_id  League ID
rotation_stat  Rotation stat to provide details on: PLAYER_PTS, PT_DIFF, USG_PCT

Value

Returns a named list of data frames: AwayTeam, HomeTeam

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

nba_glalumboxscoresimilarityscore(
  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>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>person_1_id</td>
<td>ID of person 1</td>
</tr>
<tr>
<td>person_1_league_id</td>
<td>League ID of person 1</td>
</tr>
<tr>
<td>person_1_season</td>
<td>Season of person 1</td>
</tr>
<tr>
<td>person_1_season_type</td>
<td>Season type of person 1</td>
</tr>
<tr>
<td>person_2_id</td>
<td>ID of person 2</td>
</tr>
<tr>
<td>person_2_league_id</td>
<td>League ID of person 2</td>
</tr>
<tr>
<td>person_2_season</td>
<td>Season of person 2</td>
</tr>
<tr>
<td>person_2_season_type</td>
<td>Season type of person 2</td>
</tr>
</tbody>
</table>

Value

Returns a named list of data frames: GLeagueAlumBoxScoreSimilarityScores

Author(s)

Saiem Gilani

Description

Get NBA Stats API Homepage Leaders

Usage

```r
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
hustle_bs

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
Get NBA Stats API Hustle Stats Boxscore

Usage

nba_hustlestatsboxscore(game_id)

Arguments

game_id  Game ID

Value

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

Author(s)

Saiem Gilani
Get NBA Stats API League Hustle Stats Player

Usage

```python
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          college
conference       conference
country          country
date_from        date_from
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
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

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

- `college`  college
- `conference`  conference
- `country`  country
- `date_from`  date_from
Value

Returns a named list of data frames: HustleStatsTeam

Author(s)

Saiem Gilani
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

college  college
conference  conference
country  country
date_from  date_from
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**

```r
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**

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

---

**kp_box**  
**Get KenPom Game Box Score**

**Description**

Get KenPom Game Box Score

**Usage**

```r
kp_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))

kp_coach_history  Get KenPom’s coaching resume data

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'))

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.
T0.Pct double.
T0.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]]
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 with 4 columns:

- **Rk**: double.
- **Conference**: character.
- **Rating**: double.
- **Year**: character.

**Examples**

```r
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**

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.

**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:
- Confcharacter.
- Effdouble.
- Eff.Rkdouble.
- Tempodouble.
- Tempo.Rkdouble.
- eFG.Pctdouble.
- eFG.Pct.Rkdouble.
- TO.Pctdouble.
- TO.Pct.Rkdouble.
- OR.Pctdouble.
• 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

kp_efficiency(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 efficiency and tempo ratings

Examples

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

kp_fanmatch

Get FanMatch by date

Description

Get FanMatch by date

Usage

kp_fanmatch(date = "2022-02-22")

Arguments

date Date of games to pull (YYYY-MM-DD)
Value

A data frame 20 columns:

<table>
<thead>
<tr>
<th>col_name</th>
<th>types</th>
</tr>
</thead>
<tbody>
<tr>
<td>prediction</td>
<td>character</td>
</tr>
<tr>
<td>time_et</td>
<td>character</td>
</tr>
<tr>
<td>location</td>
<td>character</td>
</tr>
<tr>
<td>thrill_score</td>
<td>numeric</td>
</tr>
<tr>
<td>comeback</td>
<td>numeric</td>
</tr>
<tr>
<td>excitement</td>
<td>numeric</td>
</tr>
<tr>
<td>road_rk</td>
<td>numeric</td>
</tr>
<tr>
<td>road_team</td>
<td>character</td>
</tr>
<tr>
<td>home_rk</td>
<td>numeric</td>
</tr>
<tr>
<td>home_team</td>
<td>character</td>
</tr>
<tr>
<td>win_rk</td>
<td>numeric</td>
</tr>
<tr>
<td>win_team</td>
<td>character</td>
</tr>
<tr>
<td>win_score</td>
<td>numeric</td>
</tr>
<tr>
<td>loss_rk</td>
<td>numeric</td>
</tr>
<tr>
<td>loss_team</td>
<td>character</td>
</tr>
<tr>
<td>loss_score</td>
<td>numeric</td>
</tr>
<tr>
<td>poss</td>
<td>numeric</td>
</tr>
<tr>
<td>mvp</td>
<td>character</td>
</tr>
<tr>
<td>event</td>
<td>character</td>
</tr>
<tr>
<td>date</td>
<td>character</td>
</tr>
</tbody>
</table>

Examples

```r
try(kp_fanmatch(date="2022-02-22"))
```

kp_foul_trouble

Get 2-Foul Participation Stats

Description

Get 2-Foul Participation Stats

Usage

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

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>min_year</td>
<td>First year of data to pull</td>
</tr>
<tr>
<td>max_year</td>
<td>Last year of data to pull</td>
</tr>
</tbody>
</table>
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

Get KenPom’s team game plan page

**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

Get Game Attributes

**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'
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**

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**

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

kp_kpoy

**Get KPoY Leaders Tables**

**Description**

Get KPoY Leaders Tables

**Usage**

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))
```

---

### kp_minutes_matrix

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

Get officials rankings

**Usage**

```r
kp_officials(year = most_recent_mbb_season())
```
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))
Get Player Stats Leaders by Metric

Usage

```r
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_pointdist

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

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

kp_pointdist

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

Value

Returns a tibble of team points distributions

Examples

```r
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

```r
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.
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

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

<table>
<thead>
<tr>
<th>referee</th>
<th>Referee ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>year</td>
<td>Year of data to pull</td>
</tr>
</tbody>
</table>

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())

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

Value
Returns a tibble of team stats

Examples
try(kp_teamstats(min_year = 2019, max_year =2021))

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

try(kp_team_depth_chart(team = 'Florida St.', year= 2021))
**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.Blk.Pct** double.
- **Def.FG_3A.Pct** double.
- **Def.A.Pct** double.
- **Def.AP** double.
- **Foul2Partic.Pct** double.

**Examples**

```r
try(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_team_schedule  Get team schedule results

Description
Get team schedule results

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

kp_trends  Get Division-I statistical trends

Description
Get Division-I statistical trends

Usage
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

try(kp_trends())

<table>
<thead>
<tr>
<th>kp_user_pw</th>
<th>KenPom Login and Password credentials</th>
</tr>
</thead>
</table>

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

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>user_email</td>
<td>User subscription e-mail</td>
</tr>
<tr>
<td>user_pw</td>
<td>User subscription password</td>
</tr>
</tbody>
</table>

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 KenPom Win Probability

Description
Get KenPom Win Probability

Usage
kp_winprob(game_id, year)

Arguments
- game_id: Game id of game to pull
- year: Year of game to pull

Value
Returns a tibble of game win probabilities

Examples
try(kp_winprob(game_id = 1238, year = 2020))

ld_oppptshot

Get NBA Stats API League Dashboard Player Tracking - Opponent Shots

Description
Get NBA Stats API League Dashboard Player Tracking - Opponent Shots

Usage
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

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>close_def_dist_range</td>
<td>close_def_dist_range</td>
</tr>
<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>division</td>
<td>division</td>
</tr>
<tr>
<td>dribble_range</td>
<td>dribble_range</td>
</tr>
<tr>
<td>game_segment</td>
<td>game_segment</td>
</tr>
<tr>
<td>general_range</td>
<td>general_range</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>
</tbody>
</table>
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
shot_dist_range  shot_dist_range
team_id     team_id
touch_time_range  touch_time_range
vs_conference  vs_conference
vs_division  vs_division

Value

Returns a named list of data frames: LeagueDashPTShots

Author(s)

Saiem Gilani

Description

Get NBA Stats API League Dashboard Player Biographical Stats

Usage

nba_leaguedashplayerbiostats(
  college = "",
  conference = "",
  country = "",
  date_from = "",
  date_to = "",
  division = "",
)

draft_pick = "";
draft_year = "";
game_segment = "";
game_scope = "";
height = "";
lst_n_games = 0;
league_id = "00"
location = "";
month = 0;
opponent_team_id = 0;
outcome = "";
pot_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**

<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>
<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>game_segment</td>
<td>game_segment</td>
</tr>
<tr>
<td>game_scope</td>
<td>game_scope</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>
</tbody>
</table>
opponent_team_id
outcome
po_round
per_mode
period
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
Get NBA Stats API League Dashboard by Player Clutch Splits
Usage

```python
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
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
pace_adjust pace_adjust
plus_minus plus_minus
point_diff point_diff
po_round   po_round
per_mode   per_mode
period     period
player_experience player_experience
player_position player_position
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
touch_time_range touch_time_range
vs_conference vs_conference
vs_division vs_division
weight     weight
Value

Returns a named list of data frames: LeagueDashPlayerClutch

Author(s)

Saiem Gilani

---

**ld_pptshot**

Get NBA Stats API League Dashboard Player Tracking - Player Shots

---

Description

Get NBA Stats API League Dashboard Player Tracking - Player Shots

Get NBA Stats API League Dashboard Player Tracking - Player Shots

Usage

```r
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
college
college
conference
conference
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
general_range
general_range
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
pace_adjust
pace_adjust
po_round
po_round
Value

Returns a named list of data frames: LeagueDashPTShots

Author(s)

Saiem Gilani
Usage

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

Arguments

- **college**: college
- **conference**: conference
- **country**: country
- **date_from**: date_from
**Value**

Returns a named list of data frames: ShotLocations

**Author(s)**

Saiem Gilani

---

**Usage**

```r
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
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
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
pace_adjust pace_adjust
po_round po_round
per_mode per_mode
period period
player_experience
player_experience
player_position
player_position
plus_minus plus_minus
rank rank
ld_ptdefend

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
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**
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 = "", 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
Value

Returns a named list of data frames: LeagueDashPTDefend

Author(s)

Saiem Gilani

Get NBA Stats API League Dashboard Player Tracking - Stats

Usage

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 = "",
op_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
category   category
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
ld_ptteamdefend

period period
player_experience player_experience
player_or_team player_or_team
player_position player_position
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

Get NBA Stats API League Dashboard Player Tracking - Team Defense

Description

Get NBA Stats API League Dashboard Player Tracking - Team Defense

Usage

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

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 = ""
)
```
<table>
<thead>
<tr>
<th>Arguments</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ahead_behind</td>
<td>ahead_behind</td>
</tr>
<tr>
<td>clutch_time</td>
<td>clutch_time</td>
</tr>
<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>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>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>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: LeagueDashTeamClutch

**Author(s)**

Saiem Gilani

---

**1d_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**

```r
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

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
date_from          date_from
date_to            date_to
distance_range     distance_range
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
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

Value

Returns a named list of data frames: ShotLocations

Author(s)

Saiem Gilani
Description

Get NBA Stats API League Dashboard Team Stats

Usage

```r
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`  
- `date_from`  
- `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
leaderstiles          Get NBA Stats API Leaders Tiles

Description

Get NBA Stats API Leaders Tiles

Usage

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

Usage

```python
def 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
leaguedashlineups

- game_segment
- group_quantity
- last_n_games
- league_id
- location
- measure_type
- month
- opponent_team_id
- outcome
- po_round
- pace_adjust
- per_mode
- period
- plus_minus
- rank
- season
- season_segment
- season_type
- shot_clock_range
- team_id
- vs_conference
- vs_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

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

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  conference
date_from date_from
date_to date_to
division division
game_segment  game_segment
group_quantity  group_quantity
last_n_games  last_n_games
league_id  league_id
location  location
measure_type  measure_type
minutes_min  minutes_min
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
team_id  team_id
vs_conference  vs_conference
vs_division  vs_division

Value

Returns a named list of data frames: LeagueLineupViz

Author(s)

Saiem Gilani
leagueplayerondetails  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_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,
    plus_minus = "N",
    rank = "N",
    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
measure_type  measure_type
month  month
leagueseasonmatchups

Value

Returns a named list of data frames: PlayersOnCourtLeaguePlayerDetails

Author(s)

Saiem Gilani

Get NBA Stats API League Season Matchups

Description

Get NBA Stats API League Season Matchups

Usage

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_fga = "", et_fgm = "", et_fg_pct = "", et_minutes = "", et_oreb = "", et_pf = "", et_pts = "", et_reb = "", et_stl = "", et_td = "", et_tov = "",
geame_id = "", gt_ast = "",
gt_blk = "",
gt_dd = "",
gt_dreb = "",
gt_fg3a = "",
gt_fg3m = "",
gt_fg3_pct = "",
gt_fga = "",
gt_fgm = "",
gt_fg_pct = "",
gt_minutes = "",
gt_oreb = "",
gt_pf = "",
gt_pts = "",
gt_reb = "",
gt_stl = "",
gt_td = "",
gt_tov = "",
league_id = "00",
location = "",
lte_g = "", lte_b = "", lte_dd = "", lte_dreb = "", lte_fg3a = "", lte_fg3m = "", lte_fg3_pct = "", lte_fga = "", lte_fgm = "", lte_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

calendar 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
et_fga et_fga
et_fgm et_fgm
et_fg_pct et_fg_pct
et_fta et_fta
et_ftm et_ftm
et_ft_pct et_ft_pct
et_minutes et_minutes
et_oreb 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
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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

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)
...
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**

```r
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**

```r
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
Usage

load_mbb_schedule(
    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_schedule(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 
)
**load_nba_pbp**

**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 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
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

```r
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_leaguegame_log( 
  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

<table>
<thead>
<tr>
<th>counter</th>
<th>counter</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>direction</td>
<td>direction</td>
</tr>
<tr>
<td>league_id</td>
<td>league_id</td>
</tr>
<tr>
<td>player_or_team</td>
<td>player_or_team</td>
</tr>
<tr>
<td>season</td>
<td>season</td>
</tr>
<tr>
<td>season_type</td>
<td>season_type</td>
</tr>
<tr>
<td>sorter</td>
<td>sorter</td>
</tr>
</tbody>
</table>

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

Get NBA Stats API League Standings

Usage

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

Arguments

<table>
<thead>
<tr>
<th>league_id</th>
<th>league_id</th>
</tr>
</thead>
<tbody>
<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

---

l_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

league_id  league_id
season  season
season_type  season_type
season_year  season_year

Value

Return a named list of data frames: Standings

Author(s)

Saiem Gilani
Get NBA Stats API Matchups Rollup

**Description**

Get NBA Stats API Matchups Rollup

Get NBA Stats API Matchups Rollup

**Usage**

```r
def 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_team_id`  
- `league_id`  
- `off_player_id`  
- `off_team_id`  
- `per_mode`  
- `season`  
- `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()
```

**nbagl_pbp**

**Get NBA Data API Play-by-Play for G League Games**

**Description**

Scrapes the NBA Data API for Play By Play for G League games

**Usage**

```r
nbagl_pbp(game_id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>game_id</td>
<td>Game ID - 10 digits, i.e. 0021900001</td>
</tr>
</tbody>
</table>

**Value**

Returns a data frame of play by play

**Author(s)**

Billy Fryer
**nbagl_players**

Get G League Player Data Base from NBA API

**Description**
Scrapes the NBA Data API for All Players in G League History

**Usage**

```r
nbagl_players()
```

**Value**
Returns a data frame of G League Players from NBA Database

**Author(s)**
Billy Fryer

**nbagl_schedule**

Get G League Schedule from NBA API

**Description**
Scrapes the NBA Data API for G League Schedule for a Given Season

**Usage**

```r
nbagl_schedule(season = most_recent_nba_season() - 1)
```

**Arguments**

```r
season    Season - 4 digit, i.e. 2021
```

**Value**
Returns a data frame of the G League Season Schedule

**Author(s)**
Billy Fryer
nbagl_standings  Get Current G League Standings from NBA API

Description
Scrapes the NBA Data API for G League Standings

Usage
nbagl_standings(season = most_recent_nba_season() - 1)

Arguments
season  Season - 4 digit, i.e. 2021

Value
Returns a tibble of the G League Season Standings

Author(s)
Billy Fryer

nba_data_pbp  Get NBA Data API Play-by-Play

Description
Get NBA Data API Play-by-Play

Usage
nba_data_pbp(game_id = "0021900001")

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

Value
Returns a tibble

Author(s)
Saiem Gilani
Get NBA Stats API Draft History

Usage

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

Arguments

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

Value

Returns a named list of data frames: DraftHistory

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**
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

Usage

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

Arguments

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></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

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.

Description

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

**Usage**

```
nba_pbp(game_id, version = "v2", p)
```

**Arguments**

- `game_id` Game ID
- `version` Play-by-play version ("v2" available from 2016-17 onwards)
- `p` Progress bar

**Value**

Returns a data frame: PlayByPlay

**Author(s)**

Jason Lee

---

**pbps**

Get NBA Stats API play-by-play (Multiple Games)

**Description**

Get NBA Stats API play-by-play (Multiple Games)

**Usage**

```
nba_pbps(game_ids = NULL, version = "v2", nest_data = FALSE)
```

**Arguments**

- `game_ids` Game IDs
- `version` Play-by-play version ("v2" available from 2016-17 onwards)
- `nest_data` If TRUE returns nested data by game
pbyclutch

Value

Returns a data frame: PlayByPlay

Author(s)

Jason Lee

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
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

Get NBA Stats API Player Dashboard by General Splits
Usage

```r
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
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: DaysRestPlayerDashboard, LocationPlayerDashboard, MonthPlayerDashboard, OverallPlayerDashboard, PrePostAllStarPlayerDashboard, StartingPosition, WinsLossesPlayerDashboard

Author(s)

Saiem Gilani

Description

Get NBA Stats API Player Dashboard by Last N Games

Usage

```r
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_to
game_segment
game_segment
game_segment
game_segment
game_segment
game_segment
league_id
league_id
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_segment
season_segment
shot_clock_range
shot_clock_range
vs_conference
vs_conference
vs_division
vs_division

Value

Return a named list of data frames: GameNumberPlayerDashboard, Last10PlayerDashboard, Last15PlayerDashboard, Last20PlayerDashboard, Last5PlayerDashboard, OverallPlayerDashboard
Author(s)
Saiem Gilani

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
Get NBA Stats API Player Dashboard by Shooting Splits
Usage

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
Value

Return a named list of data frames: AssistedBy, AssitedShotPlayerDashboard, OverallPlayerDashboard, Shot5FTPPlayerDashboard, Shot8FTPPlayerDashboard, ShotAreaPlayerDashboard, ShotTypePlayerDashboard, ShotTypeSummaryPlayerDashboard

Author(s)

Saiem Gilani

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_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: OverallPlayerDashboard, PointsScoredPlayerDashboard, PointsAgainstPlayerDashboard, ScoreDifferentialPlayerDashboard
Get NBA Stats API Player Dashboard Year over Year

**Usage**

```python
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_to`  
- `game_segment`  
- `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: ByYearPlayerDashboard, OverallPlayerDashboard

**Author(s)**

Saiem Gilani

---

**Description**

Get NBA Stats API Player Career By College

Get NBA Stats API Player Career By College
Usage

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

Arguments

<table>
<thead>
<tr>
<th>college</th>
<th>College Name</th>
</tr>
</thead>
<tbody>
<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

Get NBA Stats API Player Career By College Rollup

Description

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

Description

Get NBA Stats API Player Fantasy Profile

Usage

```r
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
pfantasy_bg

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

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_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_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_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"
stater_bench = "", team_id = "", vs_conference = "", vs_division = "", vs_team_id = "", years_experience = ""
)

Arguments

active_streaks_only  active_streaks_only
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
et_fga  et_fga
et_fgm  et_fgm
et_fg_pct  et_fg_pct
et_fga  et_fga
et_fta  et_fta
et_fpm  et_fpm
et_ft_pct  et_ft_pct
et_minutes  et_minutes
et_oreb  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
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>LtTd</td>
</tr>
<tr>
<td>lt_tov</td>
<td>Lt Tov</td>
</tr>
<tr>
<td>min_games</td>
<td>Min Games</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>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: PlayerGameStreakFinderResults

**Author(s)**

Saiem Gilani
### playerawards

Get NBA Stats API Player Awards

**Description**

Get NBA Stats API Player Awards

**Usage**

```r
defensiveawards(player_id)
```

**Arguments**

- `player_id` Player ID

**Value**

Return a named list of data frames: PlayerAwards

**Author(s)**

Saiem Gilani

---

### playercareerstats

Get NBA Stats API Player Career Stats

**Description**

Get NBA Stats API Player Career Stats

**Usage**

```r
defensiveawards(player_id)
```

**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

---

**Description**

*Get NBA Stats API Player Compare*

*Get NBA Stats API Player Compare*

**Usage**

```r
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

- `conference`
- `date_from`
- `date_to`
- `game_segment`
- `last_n_games`
- `league_id`
- `location`
- `measure_type`
- `month`
- `opponent_team_id`
- `outcome`
- `pace_adjust`
- `per_mode`
- `period`
- `player_id_list`
- `plus_minus`
- `rank`
- `season`
- `season_segment`
- `season_type`
- `shot_clock_range`
- `vs_conference`
- `vs_division`
- `vs_player_id_list`

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**

```r
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_playervsplayer(
    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
Value

Return a named list of data frames: OnOffCourt, Overall, PlayerInfo, ShotAreaOffCourt, ShotAreaOn-Court, ShotAreaOverall, ShotDistanceOffCourt, ShotDistanceOnCourt, ShotDistanceOverall, Vs-PlayerInfo

Author(s)

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

Usage

```python
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 = "0",
    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

player_id: player_id

season: season

season_segment: season_segment
Return a named list of data frames: PassesMade, PassesReceived

Author(s)
Saiem Gilani

```r
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

```python
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

Description

Get NBA Stats API Player Dashboard Player Tracking - Shots

Get NBA Stats API Player Dashboard Player Tracking - Shots

Usage

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_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, Overall, ShotClockShooting, TouchTimeShooting

Author(s)

Saiem Gilani

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.
Value

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

### 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_gameolog

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",
)


```r
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_headshot

**Get NBA Stats API Player Head-shot**

**Description**

Get NBA Stats API Player Head-shot

**Usage**

```python
nba_playerheadshot(player_id = "2544")
```

**Arguments**

- `player_id` - Player ID

**Value**

Returns a url of the png for the player_id selected

**Author(s)**

Saiem Gilani

## p_index

**Get NBA Stats API Player Index**

**Description**

Get NBA Stats API Player Index

**Usage**

```python
nba_playerindex(
    college = "",
    country = "",
    draft_pick = "",
    draft_round = "",
    draft_year = "",
    height = "",
    historical = 1,
    league_id = "00",
    season = "2020-21",
    season_type = "Regular Season",
```
```r
team_id = "0",
weight = ""
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>college</td>
<td>Player College</td>
</tr>
<tr>
<td>country</td>
<td>Player Country</td>
</tr>
<tr>
<td>draft_pick</td>
<td>Draft Pick</td>
</tr>
<tr>
<td>draft_round</td>
<td>Draft Round</td>
</tr>
<tr>
<td>draft_year</td>
<td>Draft Year</td>
</tr>
<tr>
<td>height</td>
<td>Player Height</td>
</tr>
<tr>
<td>historical</td>
<td>Whether to include only current players (0) or all historical (1).</td>
</tr>
<tr>
<td>league_id</td>
<td>League - default: '00'. Other options include '10': WNBA, '20': G-League</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>
<tr>
<td>team_id</td>
<td>Team ID. Default: 0 (all teams).</td>
</tr>
<tr>
<td>weight</td>
<td>Player weight</td>
</tr>
</tbody>
</table>

Value

Return a named list of data frames: PlayerIndex

Author(s)

Saiem Gilani

---

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

```r
nba_playernextngames(
  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

Description

rejoin_schedules (when used from league game finder)

Usage

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

Usage

nba_shotchartdetail(
    context_measure = "FGA",
    date_from = "",
    date_to = "",
    game_id = "",
    game_segment = "",
    last_n_games = 0,
    league_id = "00",
    location = "00",
    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

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

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

Author(s)
Saiem Gilani

tbygamesplits Get NBA Stats API Team Dashboard by Game Splits

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

Usage
```r
tbygamesplits(
  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
**Get NBA Stats API Team Dashboard by General Splits**

**Description**

Get NBA Stats API Team Dashboard by General Splits

**Usage**

```python
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

---

**Get NBA Stats API Team Dashboard by Last N Games**

Description

Get NBA Stats API Team Dashboard by Last N Games

Get NBA Stats API Team Dashboard by Last N Games
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

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</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>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>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>
</tbody>
</table>
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**

```r
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_to  
game_segment  
last_n_games  
league_id  
location  
measure_type  
month  
opponent_team_id  
outcome  
pace_adjust  
plus_minus  
po_round  
per_mode  
period  
rank  
season  
season_segment  
season_type  
shot_clock_range  
team_id  
vs_conference  
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
Get NBA Stats API Team Dashboard by Shooting Splits

Usage
```python
def 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_to`
- `game_segment`
- `last_n_games`
<table>
<thead>
<tr>
<th>column_name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<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>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: AssistedBy, AssitedShotTeamDashboard, OverallTeamDashboard, Shot5FTTeamDashboard, Shot8FTTeamDashboard, ShotAreaTeamDashboard, ShotTypeTeamDashboard

**Author(s)**

Saiem Gilani

---

**tbyteamperformance**  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

Description
Get NBA Stats API Team Dashboard Year over Year

Usage
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
teamdashlineups

Get NBA Stats API Team Dashboard - Lineups

Description

Get NBA Stats API Team Dashboard - Lineups

Get NBA Stats API Team Dashboard - Lineups

Usage

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

Value

Return a named list of data frames: Lineups, Overall

Author(s)

Saiem Gilani
teams_links

Usage

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

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>league_id</td>
<td>League - default: '00'. Other options include '10': WNBA, '20': G-League</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>
<tr>
<td>team_id</td>
<td>Team ID</td>
</tr>
</tbody>
</table>

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_dd = "",
    et_dreb = "",
    et_fg3a = "",
    et_fg3m = "",
    et_fg3_pct = "",
    etc_opp ast = "",
    etc_opp_blk = "",
    etc_opp_dreb = "",
    etc_opp_fg3a = "",
    etc_opp_fg3m = "",
    etc_opp_fg3_pct = "",
    etc_opp_fga = "",
    etc_opp_fgm = "",
    etc_opp_fg_pct = "",
    etc_opp_fta = "",
    etc_opp_ftm = "",
    etc_opp_ft_pct = "",
    etc_opp_oreb = "",
    etc_opp_pf = "",
    etc_opp_pts = "",
    etc_opp_pts2nd_chance = "",
    etc_opp_pts_fb = "",
    etc_opp_pts_off_tov = "",
    etc_opp_pts_paint = "",
    etc_opp_reb = "",
    etc_opp_stl = "",
    etc_opp_tov = ")"
et_fga = "",
et_fgm = "",
et_fg_pct = "",
et_fga = "",
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_pts2nd_chance = "",
eq_pts_fb = "",
eq_pts_off_tov = "",
eq_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_fta = "",
gt_ftm = "",
et_ft_pct = "",
et_minutes = "",
etopp_ast = "",
etopp_blk = "",
etopp_dreb = "",
etopp_fg3a = "",
etopp_fg3m = "",
etopp_fg3_pct = "",
etopp_fga = "",
etopp_fta = "",
etopp_ftm = "",
gt_opp_ft_pct = "",
gt_opp_oreb = "",
gt_opp_pf = "",
gt_opp_pts = "",
gt_opp_pts2nd_chance = "",
gt_opp_pts_fb = "",
gt_opp_pts_off_tov = "",
gt_opp_pts_paint = "",
gt_opp_reb = "",
gt_opp_stl = "",
gt_opp_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_pcnt = "",
lt_fta = "",
lt_ftm = "",
lt_ft_pct = "",
lt_minutes = "",
ltOpp_ast = "",
ltOpp_blk = "",
ltOpp_dreb = "",
ltOpp_fg3a = "",
ltOpp_fg3m = "",
ltOpp_fg3_pct = "",
ltOpp_fga = "",
ltOpp_fgm = "",
ltOpp_fg_pct = ""
lt_opp_fta = "", lt_opp_ftm = "", lt_opp_ft_pct = "", lt_opp oreb = "", lt_opp_pf = "", lt_opp pts = "", lt_opp pts2nd_chance = "", lt_opp pts_fb = "", lt_opp pts_off_tov = "", lt_opp pts_paint = "", lt_opp reb = "", lt_opp stl = "", lt_opp tov = "", lt oreb = "", lt_pf = "", lt pts = "", lt pts2nd_chance = "", lt pts_fb = "", lt pts_off_tov = "", lt pts_paint = "", lt reb = "", lt stl = "", lt td = "", lt 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 ast = "", wrs_opp blk = "", wrs_opp dreb = "", wrs_opp fg3a = "", wrs_opp fg3m = "", wrs_opp fg3_pct = "", wrs_opp fga = "", wrs_opp fgm = "", wrs_opp fg_pct = "", wrs_opp fta = "", wrs_opp ftm = "", wrs_opp ft_pct = "", wrs opp oreb = ""
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_teams_only
btr_oppAst btr_oppAst
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>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>btr-opp-tov</td>
<td>btr_opp_tov</td>
</tr>
<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>division</td>
<td>division</td>
</tr>
<tr>
<td>et_ast</td>
<td>et_ast</td>
</tr>
<tr>
<td>et_BLK</td>
<td>et_BLK</td>
</tr>
<tr>
<td>et_dd</td>
<td>et_dd</td>
</tr>
<tr>
<td>et_dreb</td>
<td>et_dreb</td>
</tr>
<tr>
<td>et_FG3a</td>
<td>et_fG3a</td>
</tr>
<tr>
<td>et_FG3m</td>
<td>et_fG3m</td>
</tr>
<tr>
<td>et_FG3_pct</td>
<td>et_fG3_pct</td>
</tr>
<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>eq_oppp_pts2nd_chance</td>
<td>eq_oppp_pts2nd_chance</td>
</tr>
<tr>
<td>eq_oppp_pts_fb</td>
<td>eq_oppp_pts_fb</td>
</tr>
<tr>
<td>eq_oppp_pts_off_tov</td>
<td>eq_oppp_pts_off_tov</td>
</tr>
<tr>
<td>eq_oppp_pts_paint</td>
<td>eq_oppp_pts_paint</td>
</tr>
<tr>
<td>et_OREB</td>
<td>et_OREB</td>
</tr>
<tr>
<td>et_PF</td>
<td>et_PF</td>
</tr>
<tr>
<td>etPTS</td>
<td>etPTS</td>
</tr>
<tr>
<td>eq_pts2nd_chance</td>
<td>eq_pts2nd_chance</td>
</tr>
<tr>
<td>eq_pts_fb</td>
<td>eq_pts_fb</td>
</tr>
<tr>
<td>eq_pts_off_tov</td>
<td>eq_pts_off_tov</td>
</tr>
<tr>
<td>eq_pts_paint</td>
<td>eq_pts_paint</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>
</tbody>
</table>
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_fg3_pct
gt_opp_fga
gt_opp_fgm
gt_opp_fg_pct
gt_opp_fta
gt_opp_ftm
gt_opp_ft_pct
gt_opp_oreb
gt_opp_pf
gt_opp_pts
gt_opp_pts2nd_chance
gt_opp_pts_fb
gt_opp_pts_off_tov
gt_opp_pts_paint
gt_opp_reb
tg_streak

223

gt_opp_stl
gt_opp_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
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_opp_ast
lt_opp_blk
lt_opp_dreb
lt_opp_fg3a
lt_opp_fg3m
lt_opp_fg3_pct
lt_opp_fga  lt_opp_fga
lt_opp_fgm  lt_opp_fgm
lt_opp_fg_pct  lt_opp_fg_pct
lt_opp_fta  lt_opp_fta
lt_opp_ftm  lt_opp_ftm
lt_opp_ft_pct  lt_opp_ft_pct
lt_opp_oreb  lt_opp_oreb
lt_opp_pf  lt_opp_pf
lt_opp_pts  lt_opp_pts
lt_opp_pts2nd_chance  lt_opp_pts2nd_chance
lt_opp_pts_fb  lt_opp_pts_fb
lt_opp_pts_off_tov  lt_opp_pts_off_tov
lt_opp_pts_paint  lt_opp_pts_paint
lt_opp_reb  lt_opp_reb
lt_opp_stl  lt_opp_stl
lt_opp_tov  lt_opp_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
Value

Return a named list of data frames: TeamGameStreakFinderParametersResults

Author(s)

Saiem Gilani
get

thist_leaders

Get NBA Stats API Team Historical Leaders

Description

Get NBA Stats API Team Historical Leaders

Get NBA Stats API Team Historical Leaders

Usage

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

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_id</td>
<td>season_id</td>
</tr>
<tr>
<td>team_id</td>
<td>team_id</td>
</tr>
</tbody>
</table>

Value

Return a named list of data frames: CareerLeadersByTeam

Author(s)

Saiem Gilani

Get NBA Stats API Team Player Dashboard

Description

Get NBA Stats API Team Player Dashboard

Get NBA Stats API Team Player Dashboard
Usage

```python
tpUsage
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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</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>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>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>
</tbody>
</table>
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,
)```
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_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
Get NBA Stats API Team Player On/Off Details

Usage

```python
def 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_to
game_segment
last_n_games
league_id
location
measure_type
t_details

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: OverallTeamPlayerOnOffDetails, PlayersOffCourtTeamPlayerOnOffDetails, PlayersOnCourtTeamPlayerOnOffDetails

Author(s)

Saiem Gilani

Get NBA Stats API Team Details

Description

Get NBA Stats API Team Details

Usage

nba_teamdetails(team_id = "1610612749")

Arguments

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**

```r
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
Get NBA Stats API Team Game Log

Usage

```r
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

```python
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
t_ptpass

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

---

taget text

Description

Get NBA Stats API Team Dashboard Player Tracking - Passing
Get NBA Stats API Team Dashboard Player Tracking - Passing

Usage

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_to
- last_n_games
- league_id
- location
- month
- opponent_team_id
- outcome
- per_mode
- season
- season_segment
- season_type
- team_id
- vs_conference
- 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
Usage

```python
tptreb

def 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
t_vs_p

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: ClosestDefender10ftPlusShooting, ClosestDefenderShooting, DribbleShooting, GeneralShooting, ShotClockShooting, TouchTimeShooting

Author(s)
Saiem Gilani

Description

Get NBA Stats API Team vs Player

Usage

nba_teamvplayer(
    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_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
team_id
team_id
Value

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

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 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

#### 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)

### Description

Get NBA Stats API Win Probability PBP

Get NBA Stats API Win Probability PBP

### Usage

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

### 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>run_type</td>
<td>Run Type</td>
</tr>
</tbody>
</table>

### Value

Return a named list of data frames:

### Author(s)

Saiem Gilani
year_to_season

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, 83
* Arenas
  kp_arenas, 64
* Betting
  espn_mbb_betting, 32
  espn_nba_betting, 41
* Box
  espn_mbb_player_box, 35
  espn_mbb_team_box, 38
  espn_nba_player_box, 43
  espn_nba_team_box, 46
  kp_box, 64
* 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, 82
* Chart
  espn_mbb_wp, 40
  espn_nba_wp, 47
  kp_team_depth_chart, 87
  kp_team_lineups, 90
* Coach
  kp_coach_history, 65
* Comparison
  kp_confstats, 71
* Conferences
  espn_mbb_conferences, 33
* Conference
  kp_conf, 66
  kp_confhistory, 70
  kp_confstats, 71
* Data
  espn_mbb_wp, 40
  espn_nba_wp, 47
* Depth
  kp_team_depth_chart, 87
  kp_team_lineups, 90
* Efficiency
  kp_efficiency, 73
* Factors
  kp_fourfactors, 75
* FanMatch
  kp_fanmatch, 73
* Foul
  kp_foul_trouble, 74
* Four
  kp_fourfactors, 75
* Game
  espn_mbb_game_all, 33
  espn_nba_game_all, 41
  kp_box, 64
  kp_game_attrs, 76
  kp_gameplan, 76
* HCA
  kp_hca, 77
* History
  kp_coach_history, 65
  kp_confhistory, 70
  kp_team_history, 88
* Internal
  check_status, 18
* Leaders
  kp_kpoy, 78
* Login
  kp_user_pw, 93
* MBB
  espn_mbb_betting, 32
  espn_mbb_player_stats, 35
  espn_mbb_standings, 37
  espn_mbb_team_stats, 39
  espn_mbb_teams, 38
  espn_mbb_wp, 40
* Matrix
base::tempdir, 24
bs_advv2, 9
bs_defensive, 10
bs_ffv2, 11
bs_match, 12
bs_miscv2, 12
bs_pt_v2, 13
bs_scoringv2, 14
bs_similarity, 15
bs_summaryv2, 16
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(), 242, 243
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_player_stats, 35
espn_mbb_rankings, 36
espn_mbb_scoreboard, 37
espn_mbb_standings, 37
espn_mbb_team_box, 38
espn_mbb_team_stats, 39
espn_mbb_teams, 38
espn_mbb_wp, 40
espn_mbb_wp(), 40
espn_nba_betting, 41
espn_nba_game_all, 41
espn_nba_pbp, 42
espn_nba_player_box, 43
espn_nba_player_stats, 43
espn_nba_scoreboard, 44
espn_nba_standings, 45
espn_nba_team_box, 46
espn_nba_team_stats, 47
espn_nba_teams, 45
espn_nba_wp, 47
espn_nba_wp(), 48
fantasywidget, 48
franchisehistory, 50
franchiseleaders, 50
franchiseplayers, 51
game_rotation, 52
gl_bs_similarity, 52
has_kp_user_and_pw (kp_user_pw), 93
homepageleaders, 53
homepagev2, 54
hustle_bs, 55
hustle_p, 56
hustle_pl, 58
hustle_t, 60
hustle_tl, 62
kp_arenas, 64
kp_box, 64
kp_coach_history, 65
kp_conf, 66
kp_confhistory, 70
kp_confstats, 71
kp_efficiency, 73
kp_fanmatch, 73
kp_foul_trouble, 74
kp_fourfactors, 75
kp_gameAttrs, 76
kp_gameplan, 76
kp_hca, 77
kp_height, 78
kp_kpoy, 78
kp_minutes_matrix, 79
kp_officials, 79
kp_opptracker, 80
kp_password (kp_user_pw), 93
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>kp_player_career</td>
<td>82</td>
</tr>
<tr>
<td>kp_playerstats</td>
<td>81</td>
</tr>
<tr>
<td>kp_pointdist</td>
<td>82</td>
</tr>
<tr>
<td>kp_pomeroy_archive_ratings</td>
<td>83</td>
</tr>
<tr>
<td>kp_pomeroy_ratings</td>
<td>84</td>
</tr>
<tr>
<td>kp_program_ratings</td>
<td>85</td>
</tr>
<tr>
<td>kp_referee</td>
<td>86</td>
</tr>
<tr>
<td>kp_team_depth_chart</td>
<td>87</td>
</tr>
<tr>
<td>kp_team_history</td>
<td>88</td>
</tr>
<tr>
<td>kp_team_lineups</td>
<td>90</td>
</tr>
<tr>
<td>kp_team_player_stats</td>
<td>91</td>
</tr>
<tr>
<td>kp_team_players</td>
<td>91</td>
</tr>
<tr>
<td>kp_team_schedule</td>
<td>92</td>
</tr>
<tr>
<td>kp_teamstats</td>
<td>87</td>
</tr>
<tr>
<td>kp_trends</td>
<td>92</td>
</tr>
<tr>
<td>kp_user_email</td>
<td>93</td>
</tr>
<tr>
<td>kp_user_pw</td>
<td>93</td>
</tr>
<tr>
<td>kp_winprob</td>
<td>95</td>
</tr>
<tr>
<td>l_gamelog</td>
<td>142</td>
</tr>
<tr>
<td>l_standings</td>
<td>143</td>
</tr>
<tr>
<td>l_standingsv3</td>
<td>144</td>
</tr>
<tr>
<td>ld_opppshot</td>
<td>95</td>
</tr>
<tr>
<td>ld_pbiostats</td>
<td>97</td>
</tr>
<tr>
<td>ld_pclutch</td>
<td>99</td>
</tr>
<tr>
<td>ld_pptshot</td>
<td>102</td>
</tr>
<tr>
<td>ld_pshotloc</td>
<td>104</td>
</tr>
<tr>
<td>ld_pstats</td>
<td>107</td>
</tr>
<tr>
<td>ld_ptdefend</td>
<td>109</td>
</tr>
<tr>
<td>ld_ptstats</td>
<td>111</td>
</tr>
<tr>
<td>ld_ptteamdefend</td>
<td>113</td>
</tr>
<tr>
<td>ld_tclutch</td>
<td>115</td>
</tr>
<tr>
<td>ld_tptshot</td>
<td>117</td>
</tr>
<tr>
<td>ld_tshotloc</td>
<td>119</td>
</tr>
<tr>
<td>ld_tstats</td>
<td>121</td>
</tr>
<tr>
<td>leaderstiles</td>
<td>123</td>
</tr>
<tr>
<td>leaguedashlineups</td>
<td>124</td>
</tr>
<tr>
<td>leagueleaders</td>
<td>126</td>
</tr>
<tr>
<td>leagueoneupviz</td>
<td>127</td>
</tr>
<tr>
<td>leagueplayerondetails</td>
<td>129</td>
</tr>
<tr>
<td>leagueseasonmatchups</td>
<td>130</td>
</tr>
<tr>
<td>lg_streak</td>
<td>131</td>
</tr>
<tr>
<td>load_mbb_pbp</td>
<td>136</td>
</tr>
<tr>
<td>load_mbb_player_box</td>
<td>137</td>
</tr>
<tr>
<td>load_mbb_schedule</td>
<td>137</td>
</tr>
<tr>
<td>load_mbb_team_box</td>
<td>138</td>
</tr>
<tr>
<td>load_nba_pbp</td>
<td>139</td>
</tr>
<tr>
<td>load_nba_player_box</td>
<td>140</td>
</tr>
<tr>
<td>load_nba_schedule</td>
<td>141</td>
</tr>
<tr>
<td>load_nba_team_box</td>
<td>141</td>
</tr>
<tr>
<td>login(kp_user_pw)</td>
<td>93</td>
</tr>
<tr>
<td>make.names</td>
<td>23</td>
</tr>
<tr>
<td>matchupsrollup</td>
<td>145</td>
</tr>
<tr>
<td>mbb(update_mbb_db)</td>
<td>242</td>
</tr>
<tr>
<td>mbb_db(update_mbb_db)</td>
<td>242</td>
</tr>
<tr>
<td>mbb_pbp_db(update_mbb_db)</td>
<td>242</td>
</tr>
<tr>
<td>most_recent_mbb_season</td>
<td>146</td>
</tr>
<tr>
<td>most_recent_nba_season</td>
<td>146</td>
</tr>
<tr>
<td>nba(update_nba_db)</td>
<td>243</td>
</tr>
<tr>
<td>nba_alltimeleaders (alltime)</td>
<td>7</td>
</tr>
<tr>
<td>nba_assistleaders (assists)</td>
<td>8</td>
</tr>
<tr>
<td>nba_assisttracker (assist_tracker)</td>
<td>8</td>
</tr>
<tr>
<td>nba_boxscoreadvancedv2 (bs_advv2)</td>
<td>9</td>
</tr>
<tr>
<td>nba_boxscoredefensive (bs_defensive)</td>
<td>10</td>
</tr>
<tr>
<td>nba_boxscorefourfactorsv2 (bs_ffv2)</td>
<td>11</td>
</tr>
<tr>
<td>nba_boxscorematchups (bs_match)</td>
<td>12</td>
</tr>
<tr>
<td>nba_boxscoremiscv2 (bs_miscv2)</td>
<td>12</td>
</tr>
<tr>
<td>nba_boxscoreplayertackv2 (bs_pt_v2)</td>
<td>13</td>
</tr>
<tr>
<td>nba_boxscorescoringv2 (bs_scoringv2)</td>
<td>14</td>
</tr>
<tr>
<td>nba_boxscoresimilarityscore (bs_similarity)</td>
<td>15</td>
</tr>
<tr>
<td>nba_boxscoresummaryv2 (bs_summaryv2)</td>
<td>16</td>
</tr>
<tr>
<td>nba_boxscoretraditionalv2 (bs_tradv2)</td>
<td>16</td>
</tr>
<tr>
<td>nba_boxscoreusagev2 (bs_usagev2)</td>
<td>17</td>
</tr>
<tr>
<td>nba_commonallplayers (allplayers)</td>
<td>6</td>
</tr>
<tr>
<td>nba_commonplayerinfo (commonplayerinfo)</td>
<td>19</td>
</tr>
<tr>
<td>nba_commonplayoffseries (commonplayoffseries)</td>
<td>19</td>
</tr>
<tr>
<td>nba_commonroster (commonroster)</td>
<td>20</td>
</tr>
<tr>
<td>nba_cumestatsplayer (cumestatsplayer)</td>
<td>24</td>
</tr>
<tr>
<td>nba_cumestatsplayergames (cumestatsplayergames)</td>
<td>25</td>
</tr>
<tr>
<td>nba_cumestatssteam (cumestatssteam)</td>
<td>26</td>
</tr>
<tr>
<td>nba_cumestatssteamgames (cumestatssteamgames)</td>
<td>27</td>
</tr>
<tr>
<td>nba_data_pbp</td>
<td>148</td>
</tr>
<tr>
<td>nba_db(update_nba_db)</td>
<td>243</td>
</tr>
<tr>
<td>nba_defensehub (defensehub)</td>
<td>31</td>
</tr>
<tr>
<td>nba_draftboard (dboard)</td>
<td>28</td>
</tr>
<tr>
<td>nba_draftcombinedrillresults (dcombine_drill)</td>
<td>29</td>
</tr>
</tbody>
</table>
nba_playerdashptpass (pptpass), 183
nba_playerdashptreb (pttreb), 184
nba_playerdashptshotdefend (ptshotdefend), 185
nba_playerdashptshots (ptshots), 187
nba_playerestimatedmetrics (p_est_metr), 189
nba_playerfantasyprofile (pfantasy), 170
nba_playerfantasyprofilebargraph (pfantasy_bg), 171
nba_playergamelog (p_gamelog), 189
nba_playergamelog (p_gamelogs), 190
nba_playergamestreakfinder (pg_streak), 172
nba_playerheadshot (p_headshot), 192
nba_playerindex (p_index), 192
nba_playernextgames (p_n_g), 193
nba_playerprofilev2 (playerprofilev2), 180
nba_playersvplayer (playersvplayer), 180
nba_playoffpicture (po_picture), 182
nba_schedule (schedule), 197
nba_scoreboard (scoreboard), 197
nba_scoreboardv2 (scoreboardv2), 198
nba_shotchartdetail (sc), 195
nba_shotchartleaguewide (sc_lw), 199
nba_stats_videodetails, 150
nba_stats_videoevents, 152
nba_videodetails
(nba_stats_videodetails), 150
nba_videoevents
(nba_stats_videoevents), 152
nba_videostatus
(nba_stats_videoevents), 152
nba_winprobabilitypbp (winprobabilitypbp), 244
nbagl_pbp, 146
nbagl_players, 147
nbagl_schedule, 147
nbagl_standings, 148
ncaa_mbb_NET_rankings, 153
p_est_metr, 189
p_gamelog, 189
p_gamelogs, 190
p_headshot, 192
p_index, 192
p_n_g, 193
path.expand, 21
pbp, 154
pbps, 154
pbyclutch, 155
pbygamesplits, 157
pbygeneralsplits, 158
pbylastgames, 160
pbyopponent, 162
pbyshootingsplits, 163
teamdashlineups, 213
pbyteamperformance, 165
teaminfo, 214
pbyyearoveryear, 167
teams_links, 215
pcareerbycollege, 168
tg_streak, 216
pcareerbycollege_ru, 169
thist_leaders, 226
pfantasy, 170
tp, 226
pfantasy_bg, 171
tp_onoff_det, 230
g_streak, 172
tp_onoffsummary, 228
pg_streak, 172
readRDS(), 194
playerawards, 177
rejoin_schedules, 195
playercareerstats, 177
sc, 195
playercompare, 178
sc_lw, 199
playerprofilev2, 180
schedule, 197
playersplayer, 180
scoreboard, 197
po_picture, 182
scoreboardv2, 198
pptpass, 183
setindexv, 23
pptreb, 184
setkey, 23
pptshotdefend, 185
t_details, 231
pptshots, 187
t_est_metr, 232
progressively, 188
t_gameolog, 233
t_gameologs, 234
t_ptpass, 235
t_gameonoff, 236
t_ptreb, 236
t_ptshots, 238
t_vs_p, 239
t_yby_stats, 241
tbyclutch, 199
tbygamesplits, 201
tbygeneralsplits, 203
tbylastngames, 204
tbyopponent, 206
tbbygamesplits, 208
tbyteamperformance, 209
tbyyearoveryear, 211
tbyyearoveryear