NBA信息查询平台

API接口文档

Deworm团队

2015/06/08

更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 王振聪 | 2015/06/08 | 编写API文档 | V1.0测试版 |

目录

[一、引言 3](#_Toc421616693)

[二、API 3](#_Toc421616694)

[（一）Team 3](#_Toc421616695)

[（二）Player 8](#_Toc421616696)

[（三）Match 8](#_Toc421616697)

[（四）通用返回值 8](#_Toc421616698)

[三、API使用示例 9](#_Toc421616699)

[（一）Java 9](#_Toc421616700)

[（二）Python 10](#_Toc421616701)

# 一、引言

NBA信息查询平台API接口文档是NBA信息查询平台对外提供API接口服务的规范化文档，旨在指导API的使用者正确使用API接口服务。

NBA信息查询平台对外提供的API包括球队、球员和比赛的信息和数据查询。客户端可以通过HTTP的Get方法向服务器发送请求，服务器返回的是基于JSON格式的数据（分为JSONObject和JSONArray两种），使用者可自行解析获取到的JSON格式数据。

# 二、API

## （一）Team

1、查询所有球队信息

**API**：http://localhost/team/all/

**功能**：

查询三十支球队的基本信息，包括球队缩写、名称、所在城市、所在联盟、所在分区和建立年份。

**返回值格式**：

返回值是一个JSONArray，代表三十支球队信息的列表；JSONArray里面的每一个元素是一个JSONObject，代表一支球队的基本信息。

**返回值示例**：

[

{

"buildup\_time":"1950",

"division":"Southeast",

"league":"E",

"name":"Atlanta Hawks",

"location":"Atlanta, Georgia",

"abbr":"ATL"

},

...

{

"buildup\_time":"1947",

"division":"Atlantic",

"league":"E",

"name":"Boston Celtics",

"location":"Boston, Massachusetts",

"abbr":"BOS"

}

]

**返回值解释**：

|  |  |  |  |
| --- | --- | --- | --- |
| 键 | 含义 | 数据类型 | 示例 |
| abbr | 球队缩写 | String | ATL |
| name | 球队名称 | String | Atlanta Hawks |
| location | 球队所在城市 | String | Atlanta, Georgia |
| league | 球队所在联盟（东部或西部） | String | E / W |
| division | 球队所在分区（六个分区） | String | Southeast |
| buildup\_time | 球队建立年份 | String | 1947 |

2、查询某球队的信息和数据

**API**：

http://localhost/team?abbr=HOU

[&season=14-15][&total=1][&pergame=1][&advanced=1][&opptotal=1][&opppergame=1]

**功能：**

查询某球队的信息和数据，可根据球队缩写、赛季来查询球队信息、总数据、场均数据、高阶数据、对手总数据、对手场均数据。

**API参数解释**：

|  |  |
| --- | --- |
| 参数 | 解释 |
| abbr | 球队缩写，必选参数，abbr可从上一条API获取，用于指定查询的球队。 |
| season | 赛季，可选参数，参数形式如00-01、14-15，用于指定查询数据的赛季。如无此参数，则返回该球队所有赛季的数据。 |
| total | 球队总数据，可选参数，参数值为0或1，无此参数时默认值为0。参数值为0时，表示不查询总数据，为1则查询。 |
| pergame | 球队场均数据，可选参数，参数值为0或1，无此参数时默认值为0。参数值为0时，表示不查询场均数据，为1则查询。 |
| advanced | 球队高阶数据数据，可选参数，参数值为0或1，无此参数时默认值为0。参数值为0时，表示不查询高阶数据，为1则查询。 |
| opptotal | 对手总数据，可选参数，参数值为0或1，无此参数时默认值为0。参数值为0时，表示不查询对手总数据，为1则查询。 |
| opppergame | 对手场均数据，可选参数，参数值为0或1，无此参数时默认值为0。参数值为0时，表示不查询对手场均数据，为1则查询。 |

备注：若只有abbr参数或只有abbr和season两个参数，则只查询该球队信息。

**返回值格式：**

返回值是一个JSONObject，内含abbr、info、total、pergame、advanced、opptotal、opppergame键，它们的值各自是一个JSONObject。其中total、pergame、advanced、opptotal、opppergame的JSONObject含有一个或多个名为相应赛季的键，它们的值也各自是一个JSONObject。

**返回值示例：**

http://localhost/team?abbr=HOU

&season=14-15&total=1&pergame=0&advanced=0&opptotal=0&opppergame=0

{

"total":

{

"14-15":{

"fg":3032,

"fga":6832,

"ast":1820,

"losses":26,

"fg3\_pct":0.348,

"ft":1525,

"fg2":2099,

"fta":2133,

"fg3":933,

"drb":2624,

"finish":1,

"tov":1366,

"orb":958,

"height":"6-6",

"fg2a":4152,

"fg2\_pct":0.506,

"wins":56,

"fga\_pct":0.444,

"fg3a":2680,

"blk":407,"

trb":3582,

"weight":211,

"stl":777,

"ft\_pct":0.715,

"pts":8522,

"minute":19805,

"num\_of\_game":82,

"pf":1803,

"age":27.6

}

},

"abbr":"HOU",

"info":{

"buildup\_time":"1968",

"division":"Southwest",

"championships":2,

"league":"W",

"record":"2011-1877",

"name":"Houston Rockets",

"location":"Houston, Texas",

"playeroff\_appearance":29

}

}

**返回值解释：**

info：

|  |  |  |  |
| --- | --- | --- | --- |
| 键 | 含义 | 数据类型 | 示例 |
| abbr | 球队缩写 | String | ATL |
| name | 球队名称 | String | Atlanta Hawks |
| location | 球队所在城市 | String | Atlanta, Georgia |
| league | 球队所在联盟（东部或西部） | String | E / W |
| division | 球队所在分区（六个分区） | String | Southeast |
| buildup\_time | 球队建立年份 | String | 1947 |
| playeroff\_appearance | 球队参加季后赛的次数 | Integer | 29 |
| championships | 球队获得季后赛总冠军的次数 | Integer | 2 |

total / pergame：

|  |  |  |
| --- | --- | --- |
| 键 | 含义 | 数据类型 |
| wins | 胜场（仅total） | Integer |
| losses | 输场（仅total） | Integer |
| finish | 常规赛积分榜名次（仅total） | Integer |
| age | 球员平均年龄（仅total） | Double |
| height | 球员平均身高（仅total） | String |
| weight | 球员平均体重（仅total） | Double |
| num\_of\_game | 赛季比赛场数（仅total） | Integer |
| minute | 比赛时间 | Integer/Double |
| fg | 投篮命中数 | Integer/Double |
| fga | 投篮出手数 | Integer/Double |
| fga\_pct | 投篮命中率 | Double |
| fg3 | 三分命中数 | Integer/Double |
| fg3a | 三分出手数 | Integer/Double |
| fg3\_pct | 三分命中率 | Double |
| fg2 | 两分命中数 | Integer/Double |
| fg2a | 两分出手数 | Integer/Double |
| fg2\_pct | 两分命中率 | Double |
| ft | 罚球命中数 | Integer/Double |
| fta | 罚球出手数 | Integer/Double |
| ft\_pct | 罚球命中率 | Double |
| orb | 进攻篮板数 | Integer/Double |
| drb | 防守篮板数 | Integer/Double |
| trb | 总篮板数 | Integer/Double |
| ast | 助攻数 | Integer/Double |
| stl | 抢断数 | Integer/Double |
| blk | 盖帽数 | Integer/Double |
| tov | 失误数 | Integer/Double |
| pf | 犯规数 | Integer/Double |
| pts | 得分 | Integer/Double |

advanced：

|  |  |  |
| --- | --- | --- |
| 键 | 含义 | 数据类型 |
| pw | Pythagorean wins, i.e., expected wins based on points scored and allowed. | Double |
| pl | Pythagorean losses, i.e., expected losses based on points scored and allowed. | Double |
| mov | Margin of Victory. | Double |
| sos | Strength of Schedule. A rating of strength of schedule. | Double |
| srs | Simple Rating System. A team rating that takes into account average point differential and strength of schedule. | Double |
| off\_rtg | Offensive Rating. An estimate of points produced (players) or scored (teams) per 100 possessions. | Double |
| def\_rtg | Defensive Rating. An estimate of points allowed per 100 possessions. | Double |
| pace | Pace Factor. An estimate of possessions per 48 minutes. | Double |
| fta\_per\_fga\_pct | Free Throw Attempt Rate. | Double |
| fg3a\_per\_fga\_pct | 3-Point Attempt Rate. | Double |
| off\_efg\_pct | Offense Effective Field Goal Percentage. | Double |
| off\_tov\_pct | Offense Turnover Percentage. | Double |
| orb\_pct | 进攻篮板率 | Double |
| off\_ft\_rate | Offense Free Throws Per Field Goal Attempt | Double |
| opp\_efg\_pct | Defense Effective Field Goal Percentage. | Double |
| opp\_tov\_pct | Defense Turnover Perce. | Double |
| drb\_pct | 防守篮板率 | Double |
| opp\_ft\_rate | Defense Free Throw per Field Goal Attempt | Double |
| arena | 比赛主场 | String |
| attendance | 观众总人数 | Integer |

opptotal/opppergame：

|  |  |  |
| --- | --- | --- |
| 键 | 含义 | 数据类型 |
| num\_of\_game | 赛季比赛场数（仅total） | Integer |
| minute | 比赛时间 | Integer/Double |
| fg | 对手投篮命中数 | Integer/Double |
| fga | 对手投篮出手数 | Integer/Double |
| fga\_pct | 对手投篮命中率 | Double |
| fg3 | 对手三分命中数 | Integer/Double |
| fg3a | 对手三分出手数 | Integer/Double |
| fg3\_pct | 对手三分命中率 | Double |
| fg2 | 对手两分命中数 | Integer/Double |
| fg2a | 对手两分出手数 | Integer/Double |
| fg2\_pct | 对手两分命中率 | Double |
| ft | 对手罚球命中数 | Integer/Double |
| fta | 对手罚球出手数 | Integer/Double |
| ft\_pct | 对手罚球命中率 | Double |
| orb | 对手进攻篮板数 | Integer/Double |
| drb | 对手防守篮板数 | Integer/Double |
| trb | 对手总篮板数 | Integer/Double |
| ast | 对手助攻数 | Integer/Double |
| stl | 对手抢断数 | Integer/Double |
| blk | 对手盖帽数 | Integer/Double |
| tov | 对手失误数 | Integer/Double |
| pf | 对手犯规数 | Integer/Double |
| pts | 对手得分 | Integer/Double |

## （二）Player

## （三）Match

## （四）通用返回值

|  |  |
| --- | --- |
| 返回的字符串 | 字符串解释 |
| Not support. | 不支持当前的Get请求，请确认是否使用上述正确的API请求 |
| Not found. | 数据库中检索不到当前所查询的数据，请确认API请求 |
| Error. | 服务器发生错误，请联系 |

# 三、API使用示例

为了使API使用者能正确地使用和解析API，下面提供了API的使用示例。

API的使用步骤分为发送HTTP请求、接收JSON格式数据、解析JSON格式数据。

## （一）Java

发送HTTP请求、接收JSON格式数据：可以使用Java自带的HttpURLConnection类或者Apache HTTPClient包来发送HTTP请求并接收数据。

解析JSON格式数据：org.json包。

import org.json.JSONObject;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.net.HttpURLConnection;

import java.net.MalformedURLException;

import java.net.URI;

import java.net.URL;

/\*\*

\* Created by Vboar on 2015/6/8.

\*/

public class Demo {

public static void main(String[] args) {

String content = "";

try {

URL url = new URL("http://localhost/team?abbr=HOU");

HttpURLConnection con = (HttpURLConnection)url.openConnection();

con.connect();

InputStream inputStream = con.getInputStream();

BufferedReader br = new BufferedReader(new InputStreamReader(inputStream));

content = br.readLine();

con.disconnect();

} catch (IOException e) {

e.printStackTrace();

}

JSONObject jo = new JSONObject(content);

String abbr = jo.get("abbr").toString();

JSONObject info = (JSONObject)jo.get("info");

String name = info.get("name").toString();

String location = info.get("location").toString();

String league = info.get("league").toString();

String division = info.get("division").toString();

String buildup\_time = info.get("buildup\_time").toString();

String record = info.get("record").toString();

String playeroff\_appearance = info.get("playeroff\_appearance").toString();

String championships = info.get("championships").toString();

System.out.println("球队：" + abbr);

System.out.println("球队名：" + name);

System.out.println("位置：" + location);

System.out.println("联盟：" + league);

System.out.println("分区：" + division);

System.out.println("成立时间：" + buildup\_time);

System.out.println("历史：" + record);

System.out.println("参加季后赛次数：" + playeroff\_appearance);

System.out.println("获得冠军次数：" + championships);

}

}

## （二）Python

发送HTTP请求、接收JSON格式数据：urllib2模块

解析JSON格式数据：json模块。

# -\*- coding: utf-8 -\*-

\_\_author\_\_ = 'Vboar'

import urllib2

import json

url = 'http://localhost/team?abbr=HOU'

text = urllib2.urlopen(url).read()

json\_object = json.loads(text)

abbr = json\_object['abbr']

info = json\_object['info']

name = info['name']

location = info['location']

league = info['league']

division = info['division']

buildup\_time = info['buildup\_time']

record = info['record']

playeroff\_appearance = info['playeroff\_appearance']

championships = info['championships']

print u'球队：' + abbr

print u'球队名：' + name

print u'位置：' + location

print u'联盟：' + league

print u'分区：' + division

print u'成立时间：' + buildup\_time

print u'历史：' + record

print u'参加季后赛次数：' + playeroff\_appearance.\_\_str\_\_()

print u'获得冠军次数：' + championships.\_\_str\_\_()