API-NBA

Integration Guide

v. 1.0

Table of Contents

1. Version History

Version	Date	Comment
0.1	May 2022	Initial draft.
1.0	May 2022	Baseline document.

2. Introduction

This document describes the process of integration with API-NBA. This product contains massive data sets with team, player, and game statistics. You can access the information with HTTP GET requests, and use it for sport data analysis, newsfeed implementation, sports app development, and other purposes.

The document describes all required pre-conditions, available methods with request/response samples and parameter descriptions, possible errors, and limitations.

3. Definitions and Abbreviations

While working with API-NBA and reading this document, you may spot the following terms:

Term	Description
NBA	The National Basketball Association. In this document, we refer to the NBA as to the entire organization rather than one league.
Season	API-NBA refers to a season as a year in the *YYYY* format. 2019 season in API-NBA means the 2019/2020 season, 2020 means the 2020/2021, etc.
Leagues	API-NBA refers to leagues as NBA's sub-leagues, such as Orlando summer league, Las Vegas summer league, NBA main league, etc.

4. Pre-Conditions

To start the integration with API-NBA, you should follow the required pre-conditions, including:

- Authorization and Access.
- Requirements.

4.1. Authorization and Access

There are different pricing plans available for users. However, authorization is the same for any plan.

To get access to API-NBA:

1. Register and log in to the RapidAPI hub.
2. Open the API-NBA pricing plan page. Below, you can find actual plans for May 2022.

Objects	Basic $0.00 / mo Subscribe	Pro $15.00 / mo Subscribe	Recommended Ultra $25.00 / mo Subscribe	Mega $35.00 / mo Subscribe
requests	100 / day + $0.0001 each other	5,000 / day Hard Limit	20,000 / day Hard Limit	50,000 / day Hard Limit
Features				
Seasons	✓	✓	✓	✓
Leagues	✓	✓	✓	✓
Teams	✓	✓	✓	✓
Players	✓	✓	✓	✓
Games	✓	✓	✓	✓
GameDetails	✓	✓	✓	✓
Standings	✓	✓	✓	✓
Statistics	✓	✓	✓	✓
Livescore	✓	✓	✓	✓
Fast Support	✓	✓	✓	✓
Rate Limit	10 requests per minute	300 requests per minute	450 requests per minute	900 requests per minute

3. Choose a plan and click *Subscribe*.
4. Provide the necessary information to finish the subscription. These details may vary depending on the pricing plan.
5. After finishing the subscription, you will receive an email with *x-rapidapi-key*. Every request to API-NBA must contain the valid *x-rapidapi-key* in its header.

4.2. Requirements

General requirements for the integration with API-NBA are:

- **REST** is an architectural style.
- Currently, only HTTP **GET** request methods are supported.
- API-NBA v2 is an actual and supported version.
- Every request must contain **two header parameters**:
    - *'x-rapidapi-host': 'api-nba-v1.p.rapidapi.com'* (this value is the same for every request),
    - *'x-rapidapi-key': ' XxXxXxXxXxXxXxXxXxXxXxXx'* (replace the mask with the *x-rapidapi-key* you received).
- If your framework automatically adds extra headers, remove the headers before sending a request to API-NBA.

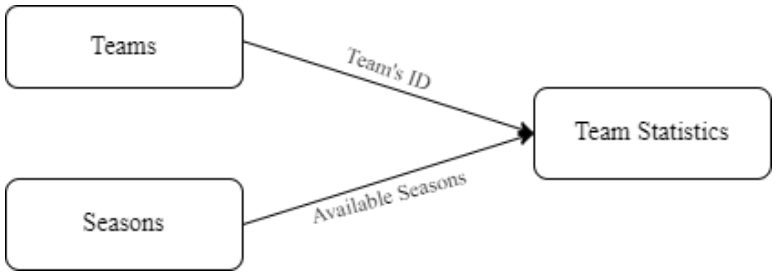
5. Common Flows

There are several flows that you can walk through with API-NBA, including:

5.1. Team's Season Stats

To get the team's whole stats from a particular season:

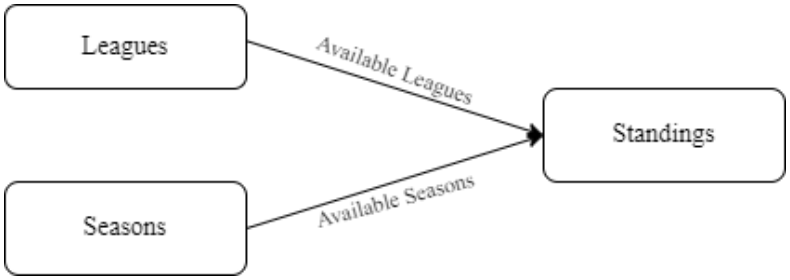
1. Get a *team's ID* with the Teams method.
2. Get a list of available *seasons* with the Seasons method.
3. Using the *team's ID* and the year of the *season* as query parameters, get the team's season stats with the Team Statistics method.



5.2. Season Standings

To get team standings for a particular season:

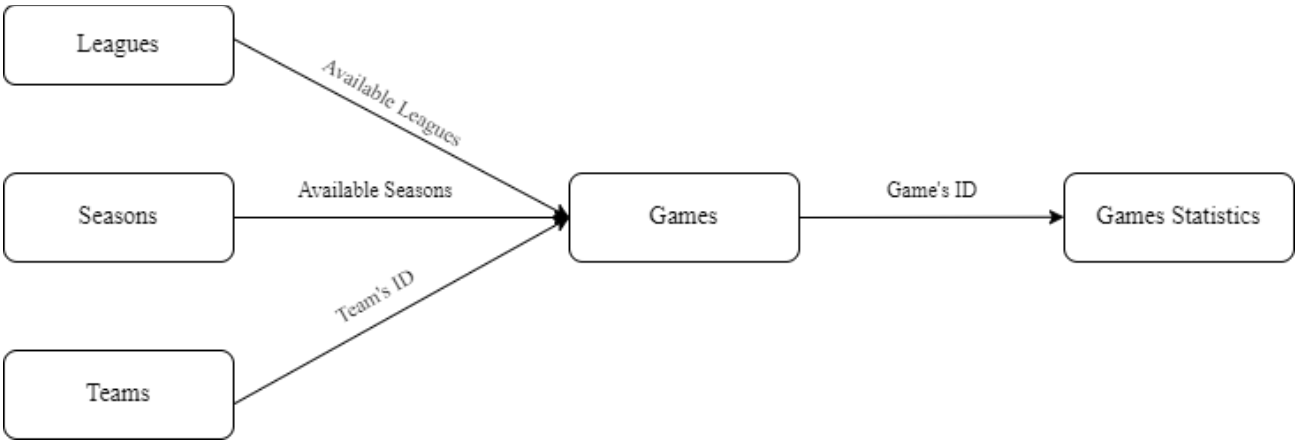
1. Get a list of available *seasons* with the Seasons method.
2. Get a list of available *leagues* with the Leagues method.
3. Using the year of the *season* and the *league's name* as query parameters, get the season standings with the Standings method.



5.3. Game Stats

To get the statistics of a particular game:

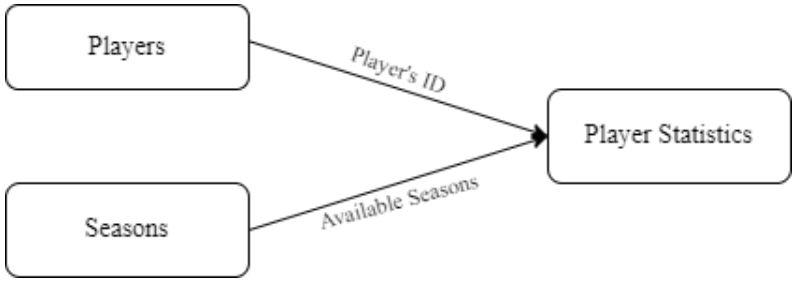
1. Get a list of available *seasons* with the Seasons method.
2. Get a list of available *leagues* with the Leagues method.
3. Get *IDs* of *teams* that played the game with the Teams method.
4. Using the year of the *season*, the *league's name*, and the *team IDs* as query parameters, get the game's basic info with the Games method.
5. Using the *game's ID* the *Games* method response contained, get the game's whole info with the Game Statistics method.



5.4. Player's Season Stats

To get a player's statistics for a particular season:

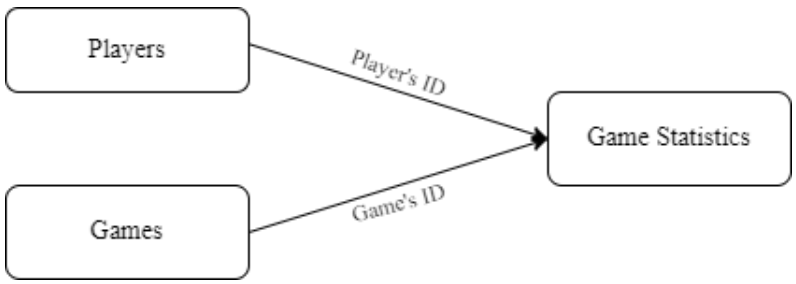
1. Get a list of available *seasons* with the Seasons method.
2. Get a *player's ID* with the Players method.
3. Using the year of the *season* and the *player's ID* as query parameters, get the player's season stats with the Player Statistics method.



5.5. Player's Game Stats

To get a player's statistics for a particular game:

1. Get a *player's ID* with the Players method.
2. Get a *game's ID* with the Games method.
3. Using the *player's* and *game's IDs* as query parameters, get the player's game stats with the Game Statistics method.



6. API Methods

API-NBA supports the following methods:

6.1. Seasons

With this method, you are able to get a list of seasons for which different statistics are available. Later on, you can use the season *YYYY* key as a query parameter.

6.1.1. Seasons Request

Parameter	Value
Request URL	https://api-nba-v1.p.rapidapi.com/seasons
Headers	
x-rapidapi-host	api-nba-v1.p.rapidapi.com
x-rapidapi-key	*x-rapidapi-key* you received

6.1.2. Seasons Response

Parameter	Data Type	Description
get	string	The value is *seasons/* in this case.
parameters	array	Query request parameters. The parameter is empty if there were no query parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	List of available seasons.

6.1.3. Seasons Request Sample

```
curl --request GET \
	--url https://api-nba-v1.p.rapidapi.com/seasons \
	--header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
	--header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.1.4. Seasons Response Sample

```
200 OK
Content Type: application/json
{
 "get": "seasons/",
 "parameters": [],
 "errors": [],
 "results": 4,
 "response": [
```

```
            2018,
            2019,
            2020,
            2021
 ]
}
```


6.2. Leagues

With this method, you are able to get a list of available leagues. For example, *standard* is the main NBA league, *vegas* is the Vegas summer league, etc.

6.2.1. Leagues Request

Parameter	Value
Request URL	https://api-nba-v1.p.rapidapi.com/leagues
Headers	
x-rapidapi-host	api-nba-v1.p.rapidapi.com
x-rapidapi-key	*x-rapidapi-key* you received

6.2.2. Leagues Response

Parameter	Data Type	Description
get	string	The value is *leagues/* in this case.
parameters	array	Query request parameters. The parameter is empty if there were no query parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	List of available leagues.

6.2.3. Leagues Request Sample

```
curl --request GET \
            --url https://api-nba-v1.p.rapidapi.com/leagues \
            --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
            --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.2.4. Leagues Response Sample


```
200 OK
Content Type: application/json
{
 "get": "leagues/",
 "parameters": [],
 "errors": [],
 "results": 6,
 "response": [
            "africa",
            "orlando",
            "sacramento",
            "standard",
            "utah",
            "vegas"
 ]
}
```

6.3. Teams

With this method, you are able to get the information about particular teams.

6.3.1. Teams Request

Parameter	Value	
Request URL	https://api-nba-v1.p.rapidapi.com/teams	
Headers		
x-rapidapi-host	api-nba-v1.p.rapidapi.com	
x-rapidapi-key	*x-rapidapi-key* you received	
Optional Query Parameters		
Parameter	**Data Type**	**Description**
id	integer	ID of a team.
name	string	Full name of a team, for example, *name=Atlanta Hawks*.
code	string (3 characters)	Code of a team, for example, *code=ATL*.
league	string	League where a team plays. You can get a list of leagues with the Leagues method.
conference	string	Conference where a team plays: The possible values are *East* and *West*.
division	string	Division where a team plays. The possible values are: • *Atlantic* • *Central* • *Northwest* • *Pacific* • *Southeast* • *Southwest*
search	string >=3 characters	Partial name of a team, for example, *search=Atlanta*.

6.3.2. Teams Response

Parameter	Data Type	Description
get	string	The value is *teams/* in this case.
parameters	array	Query request parameters. The parameter is empty if there were no query parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The parameter contains the following details for every team: • *id* is a team's unique ID in the system. • *name* is a team's full name, e.g., *San Antonio Spurs*. • *nickname* is a team's short name, e.g., *Spurs*. • *code* is a team's 3-digit code. • *city* is a city where a team is located. • *logo* is a link to a team's logo image. • *allStar* states if it's an all-star team. • *nbaFranchise* states if a team is an NBA franchise. • *leagues* (array) are the leagues where a team plays.

6.3.3. Teams Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/teams?search=Spurs \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.3.4. Teams Response Sample

```
200 OK
Content Type: application/json
{
 "get": "teams/",
 "parameters": {
                    "search":"Spurs"
                  },
 "errors": [],
 "results": 1,
 "response": [
                  0:
                    {
                    "id":31
                    "name":"San Antonio Spurs"
                    "nickname":"Spurs"
                    "code":"SAS"
                    "city":"San Antonio"
                    "logo":"https://upload.wikimedia.org/wikipedia/fr/0/0e/San_Antonio_Spurs_2018.png"
                    "allStar":false
                    "nbaFranchise":true
                    "leagues":{
                                    "standard" :
                                            {
                                            "conference":"West"
                                            "division":"Southwest"
                                             }
                                }
                    }
              ]
}
```

6.4. Team Statistics

With this method, you are able to get team statistics from a particular season.

6.4.1. Team Statistics Request

Parameter	Value	
Request URL	https://api-nba-v1.p.rapidapi.com/teams/statistics	
Headers		
x-rapidapi-host	api-nba-v1.p.rapidapi.com	
x-rapidapi-key	*x-rapidapi-key* you received	
Mandatory Query Parameters		
Parameter	**Data Type**	**Description**

id	integer	ID of a team. You can get a list of teams with the Teams method.
season	integer	Season in the *YYYY* format. You can get a list of seasons with the Seasons method.

6.4.2. Teams Statistics Response

Parameter	Data Type	Description
get	string	The value is *teams/statistics* in this case.
parameters	array	Query request parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The parameter contains the following details for every team: • *games* are total games for a team during a season. • *fastBreakPoint* are total of fast break points. • *pointsInPaint* are total of points in the paint. • *secondChancePoints* are total of second chance points. • *pointsOffTurnovers* are total of points off turnovers. • *points* are total points. • *fgm* are total of field goals made. • *fga* are total of field goal attempts. • *fgp* is a field goal percentage. • *ftm* are total of free throws made. • *fta* are total of free throw attempts. • *ftp* is a free throw percentage. • *tpm* are total of three-points made. • *tpa* are total of three-point attempts. • *tpp* is a three-point percentage. • *offReb* are total of offensive rebounds. • *debReb* are total of defensive rebounds. • *totReb* are total of rebounds. • *assists* are total of assists. • *pFouls* are total of personal fouls. • *steals* are total of steals. • *turnovers* are total of turnovers. • *blocks* are total of blocks. • *plusMinus* is a difference between points and points allowed.

6.4.3. Team Statistics Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/teams/statistics?id=31&season=2020 \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.4.4. Team Statistics Response Sample


```
200 OK
Content Type: application/json
{
 "get": "teams/statistics",
 "parameters": {
                        "id" : "31"
                        "season" : "2020"
                      },
 "errors": [],
 "results": 1,
 "response": [
                        0:
                        {
                        "games" : 76
                        "fastBreakPoints" : 874
                        "pointsInPaint" : 3606
                        "secondChancePoints" : 935
                        "pointsOffTurnovers" : 1197
                        "points" : 8406
                        "fgm" : 3158
                        "fga" : 6888
                        "fgp" : "45.9"
                        "ftm" : 1334
                        "fta" : 1692
                        "ftp" : "79.0"
                        "tpm" : 756
                        "tpa" : 2170
                        "tpp" : "34.6"
                        "offReb" : 715
                        "defReb" : 2627
                        "totReb" : 3342
                        "assists" : 1842
                        "pFouls" : 1376
                        "steals" : 541
                        "turnovers" : 842
                        "blocks" : 386
                        "plusMinus" : -178
                        }
            ]
}
```


6.5. Games

With this method, you can get a list of games according to specific parameters.

6.5.1. Games Request

Parameter	Value
Request URL	https://api-nba-v1.p.rapidapi.com/games
Headers	
x-rapidapi-host	api-nba-v1.p.rapidapi.com

x-rapidapi-key	*x-rapidapi-key* you received	
Optional Query Parameters		
Parameter	**Data Type**	**Description**
id	integer	ID of a game.
date	timestamp	Game's date in the *YYYY-MM-DD* format.
live	string	By setting the parameter to "all" (*live=all*), you get a list of games that are currently live.
league	string	League's name. You can read more about leagues here.
season	integer	Season in the *YYYY* format. You can read more about seasons here.
team	integer	Team's ID. You can read more about teams here.
h2h	string	IDs of teams that played a game. The format is *ID-ID*, for example, *h2h=1-4*.

6.5.2. Games Response

Parameter	Data Type	Description
get	string	The value is *games/* in this case.
parameters	array	Query request parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The parameter contains the following details for every team: • *id* is a game's ID. • *league* is a league's name. • *season* is a season in the *YYYY* format. • *date* is an array of *start*, *end*, and *duration* of a game. • *status* is a game's status. • *periods* show current and total game periods. • *are*na is an arena info, such as its name, city, and country. • *teams* are the details of the teams that played. • *scores* are points and teams' current records. • *officials* are game's officials. • *timeTied* is a number the game was tied. • *leadChanges* is a number of lead changes during the game.

6.5.3. Games Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/games?season=24&date=2021-03-01&team=24 \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.5.4. Games Response Sample

```
200 OK
Content Type: application/json
{
 "get": "games/",
 "parameters": {
                "season":"2020"
```

```
                    "date":"2021-03-01"
                    "team":"24"
                },
"errors": [],
"results": 1,
"response": [
                    0:
                    {
                    "id":8029
                    "league":"standard"
                    "season":2020
                    "date":
                            {
                            "start":"2021-03-01T01:00:00.000Z"
                            "end":"2021-03-01T03:14:00.000Z"
                            "duration":"2:04"
                            }
                    "status":
                            {
                            "clock":NULL
                            "halftime":false
                            "short":3
                            "long":"Finished"
                            }
                    "periods":
                            {
                            "current":4
                            "total":4
                            "endOfPeriod":false
                            }
                    "arena":
                            {
                            "name":"Little Caesars Arena"
                            "city":"Detroit"
                            "state":"MI"
                            "country":"USA"
                            }
                    "teams":
                            {
                                    "visitors":
                                            {
                                            "id":24
                                            "name":"New York Knicks"
                                            "nickname":"Knicks"
                                            "code":"NYK"
                                            "logo":"https://upload.wikimedia.org/
                                                wikipedia/fr/d/dc/NY_Knicks_Logo_2011.png"
                                            }
                                    "home":
                                            {
                                            "id":10
                                            "name":"Detroit Pistons"
                                            "nickname":"Pistons"
                                            "code":"DET"
```

```
                                        "logo":"https://upload.wikimedia.org/wikipedia/commons/thumb/6/6a/
                                            Detroit_Pistons_primary_logo_2017.png/
                                            150px-Detroit_Pistons_primary_logo_2017.png"
                                        }
                        }
                "scores":
                        {
                        "visitors":
                                {
                                "win":18
                                "loss":17
                                "series":
                                        {
                                        "win":1
                                        "loss":0
                                        }
                                "linescore":
                                        [
                                        0:"25"
                                        1:"23"
                                        2:"35"
                                        3:"26"
                                        ]
                                "points":109
                                }
                        "home":
                                {
                                "win":9
                                "loss":25
                                "series":
                                        {
                                        "win":0
                                        "loss":1
                                        }
                                "linescore":
                                        [
                                        0:"20"
                                        1:"17"
                                        2:"27"
                                        3:"26"
                                        ]
                                "points":90
                                }
                        }
                "officials":
                        [
                        0:"Tony Brothers"
                        1:"Marat Kogut"
                        2:"Danielle Scott"
                        ]
                "timesTied":2
                "leadChanges":3
        ]
}
```

6.6. Game Statistics

With this method, you can get the statistics for the teams that played a game.

6.6.1. Game Statistics Request

Parameter	Value	
Request URL	https://api-nba-v1.p.rapidapi.com/games/statistics	
Headers		
x-rapidapi-host	api-nba-v1.p.rapidapi.com	
x-rapidapi-key	*x-rapidapi-key* you received	
Required Query Parameter		
Parameter	**Data Type**	**Description**
id	integer	ID of a game. You can get the ID with the Games method.

6.6.2. Game Statistics Response

Parameter	Data Type	Description
get	string	The value is *games/statistics* in this case.
parameters	array	Query request parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The response contains two arrays: *team* and *statistics*. The *team* array contains: • *id* is a team's ID, • *name* is a team's full name, • *nickname* is a team's nickname, • *code* is a team's 3-symbols code, • *logo* is a link to a team's logo. The *statistics* array contains: • *fastBreakPoint* are total of fast break points. • *pointsInPaint* are total of points in the paint. • *biggestLead* is the team's biggest lead in the game. • *secondChancePoints* are total of second chance points. • *pointsOffTurnovers* are total of points off turnovers. • *longestRun* is the team's biggest run in the game. • *points* are total points. • *fgm* are total of field goals made. • *fga* are total of field goal attempts. • *fgp* is a field goal percentage. • *ftm* are total of free throws made. • *fta* are total of free throw attempts. • *ftp* is a free throw percentage. • *tpm* are total of three-points made. • *tpa* are total of three-point attempts. • *tpp* is a three-point percentage. • *offReb* are total of offensive rebounds. • *debReb* are total of defensive rebounds.

		• *totReb* are total of rebounds. • *assists* are total of assists. • *pFouls* are total of personal fouls. • *steals* are total of steals. • *turnovers* are total of turnovers. • *blocks* are total of blocks. • *plusMinus* is a difference between points and points allowed. • *min* are total minutes played by team's players.

6.6.3. Game Statistics Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/games/statistics?id=10403 \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.6.4. Game Statistics Response Sample

```
200 OK
Content Type: application/json
{
 "get": "games/statistics",
 "parameters": {
                "id": "10403"
              },
 "errors": [],
 "results": 2,
 "response": [
                0:
                {
                "team":
                        {
                        "id":5
                        "name":"Charlotte Hornets"
                        "nickname":"Hornets"
                        "code":"CHA"
                        "logo":"https://upload.wikimedia.org/wikipedia/fr/thumb/f/f3/
                          Hornets_de_Charlotte_logo.svg/1200px-Hornets_de_Charlotte_logo.svg.png"
        }
                "statistics":
                        {
                        "fastBreakPoints":15
                        "pointsInPaint":70
                        "biggestLead":28
                        "secondChancePoints":18
                        "pointsOffTurnovers":24
                        "longestRun":12
                        "points":141
                        "fgm":54
                        "fga":97
                        "fgp":"55.7"
                        "ftm":15
                        "fta":23
                        "ftp":"65.2"
                        "tpm":18
```

```
                    "tpa":42
                    "tpp":"42.9"
                    "offReb":15
                    "defReb":36
                    "totReb":51
                    "assists":36
                    "pFouls":22
                    "steals":13
                    "turnovers":18
                    "blocks":2
                    "plusMinus":"22"
                    "min":"240:00"
                    }
          }
          1:
          {
          "team":
                    {
                    "id":10
                    "name":"Detroit Pistons"
                    "nickname":"Pistons"
                    "code":"DET"
                    "logo":"https://upload.wikimedia.org/wikipedia/
                    commons/thumb/6/6a/Detroit_Pistons_primary_logo_2017.png/
                    150px-Detroit_Pistons_primary_logo_2017.png"
                    }
          "statistics":
                    {
                    "fastBreakPoints":8
                    "pointsInPaint":52
                    "biggestLead":2
                    "secondChancePoints":13
                    "pointsOffTurnovers":24
                    "longestRun":12
                    "points":119
                    "fgm":48
                    "fga":102
                    "fgp":"47.1"
                    "ftm":11
                    "fta":18
                    "ftp":"61.1"
                    "tpm":12
                    "tpa":35
                    "tpp":"34.3"
                    "offReb":16
                    "defReb":29
                    "totReb":45
                    "assists":32
                    "pFouls":20
                    "steals":10
                    "turnovers":18
                    "blocks":6
                    "plusMinus":"-22"
                    "min":"240:00"
                    } } ] }
```

6.7. Players

With this method, you can get data about players.

6.7.1. Players Request

Parameter	Value	
Request URL	https://api-nba-v1.p.rapidapi.com/players	
Headers		
x-rapidapi-host	api-nba-v1.p.rapidapi.com	
x-rapidapi-key	*x-rapidapi-key* you received	
Optional Query Parameter		
Parameter	**Data Type**	**Description**
id	integer	ID of a player.
name	string	Player's name.
team	integer	Team that a player plays for. You can get a list of teams with the Teams method.
season	integer	Season (in the *YYYY* format) when a player performs. You can get a list of seasons with the Seasons method.
country	string	Player's country.
search	string >=3 characters	Partial name of a team, for example, *search=Kobe*.

6.7.2. Players Response

Parameter	Data Type	Description
get	string	The value is *players/* in this case.
parameters	array	Query request parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The parameter contains the following details for every player: • *id* is a player's ID. • *firstname* is a player's first name. • *lastname* is player's last name. • *birth* is a player's birth date and country. • *nba* is a player's first year in the league and total number of seasons. • *height* is player's height, both in feets and meters. • *weight* is a player's weight, both in pounds and kilograms. • *college* is a college/school a player graduated from. • *affiliation* is a player's school. • *leagues* are leagues a player played for.

6.7.3. Players Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/players?name=Rondo \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.7.4. Players Response Sample

```
200 OK
Content Type: application/json
{
 "get": "players",
 "parameters": {
                        "search": "Rondo"
                    },
 "errors": [],
 "results": 1,
 "response": [
                        0:
                        {
                        "id":455
                        "firstname":"Rajon"
                        "lastname":"Rondo"
                        "birth":
                                {
                                "date":"1986-02-22"
                                "country":"USA"
                                }
                        "nba":
                                {
                                "start":2006
                                "pro":15
                                }
                        "height":
                                {
                                "feets":"6"
                                "inches":"1"
                                "meters":"1.85"
                                }
                        "weight":
                                {
                                "pounds":"180"
                                "kilograms":"81.6"
                                }
                        "college":"Kentucky"
                        "affiliation":"Kentucky/USA"
                        "leagues":
                                {
                                        "standard":
                                                {
                                                "jersey":1
                                                "active":true
                                                "pos":"G"
                                                }

                                }
                        }
                ]
}
```

6.8. Player Statistics

With this method, you can get statistics for a player for a particular game.

6.8.1. Player Statistics Request

Parameter	Value	
Request URL	https://api-nba-v1.p.rapidapi.com/players/statistics	
Headers		
x-rapidapi-host	api-nba-v1.p.rapidapi.com	
x-rapidapi-key	*x-rapidapi-key* you received	
Optional Query Parameter		
Parameter	**Data Type**	**Description**
id	integer	ID of a player. You can get the ID with the Players method.
game	integer	ID of a game. You can get the ID with the Games method.
team	integer	Team that a player plays for. You can get a list of teams with the Teams method.
season	integer	Season (in the *YYYY* format) when a player performs. You can get a list of seasons with the Seasons method.

6.8.2. Player Statistics Response

Parameter	Data Type	Description
get	string	The value is *players/statistics* in this case.
parameters	array	Query request parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The parameter contains the following details: • *player* is player's details. • *team* is details for the team that a player plays for. • *game* is a game's ID. • *points* are total points. • *pos* is a player's position. • *min* is total minutes played by a player. • *fgm* are total of field goals made. • *fga* are total of field goal attempts. • *fgp* is a field goal percentage. • *ftm* are total of free throws made. • *fta* are total of free throw attempts. • *ftp* is a free throw percentage. • *tpm* are total of three-points made. • *tpa* are total of three-point attempts. • *tpp* is a three-point percentage. • *offReb* are total of offensive rebounds. • *debReb* are total of defensive rebounds. • *totReb* are total of rebounds. • *assists* are total of assists. • *pFouls* are total of personal fouls. • *steals* are total of steals.

		• *turnovers* are total of turnovers. • *blocks* are total of blocks. • *plusMinus* is a difference between team points and points allowed for the time when a player was on the court.

6.8.3. Player Statistics Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/players/statistics?id=455&game=8125&season=2019 \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.8.4. Player Statistics Response Sample

```
200 OK
Content Type: application/json
{
 "get": "players/statistics ",
 "parameters": {
                "game":"8125"
                "id":"455"
                "season":"2019"
              },
 "errors": [],
 "results": 1,
 "response": [
                0:
                {
                "player":
                        {
                        "id":455
                        "firstname":"Rajon"
                        "lastname":"Rondo"
                        }
                "team":
                        {
                        "id":17
                        "name":"Los Angeles Lakers"
                        "nickname":"Lakers"
                        "code":"LAL"
                        "logo":"https://upload.wikimedia.org/wikipedia/
                        commons/thumb/3/3c/Los_Angeles_Lakers_logo.svg/
                        220px-Los_Angeles_Lakers_logo.svg.png"
                        }
                "game":
                        {
                        "id":8125
                        }
                "points":19
                "pos":NULL
                "min":"30:25"
                "fgm":8
                "fga":11
                "fgp":"72.7"
                "ftm":0
```

```
                    "fta":0
                    "ftp":"0.0"
                    "tpm":3
                    "tpa":4
                    "tpp":"75.0"
                    "offReb":0
                    "defReb":4
                    "totReb":4
                    "assists":4
                    "pFouls":1
                    "steals":1
                    "turnovers":4
                    "blocks":0
                    "plusMinus":"5"
                    "comment":NULL
                    }
                ]
}
```


6.9. Standings

With this method, you can get standings for a particular league, team, and season.

6.9.1. Standings Request

Parameter	Value	
Request URL	https://api-nba-v1.p.rapidapi.com/standings	
Headers		
x-rapidapi-host	api-nba-v1.p.rapidapi.com	
x-rapidapi-key	*x-rapidapi-key* you received	
Mandatory Query Parameter		
Parameter	**Data Type**	**Description**
league	string	League where a team plays.
season	integer	Season (in the *YYYY* format) when a player performs.
Optional Query Parameter		
team	integer	Team's ID. You can find the ID with the Teams method.
conference	string	Conference where a team plays.
division	string	Division where a team plays.

6.9.2. Standings Response

Parameter	Data Type	Description
get	string	The value is *standings/* in this case.
parameters	array	Query request parameters.
errors	array	Error codes. The parameter is empty if there were no errors.
results	integer	Number of results in the *response* array.
response	array	The parameter contains the following details: • *league* is a league where a team plays. • *season* is a season where a team plays. • *team* are team's details. • *conference* is the conference stats for a team.

		• *division* is the division stats for a team. • *win* is the status about team's wins. • *loss* is the status about team's losses. • *gamesBehind* is a number of games a team was behind from the first place at the conference. • *streak* is a current winning or losing streak for a team. • *winStreak* states whether the streak was a winning (*true*) or losing (*false*) one.

6.9.3. Standings Request Sample

```
curl --request GET \
        --url https://api-nba-v1.p.rapidapi.com/standings?league=standard&season=2019&team=19 \
        --header 'x-rapidapi-host: api-nba-v1.p.rapidapi.com' \
        --header 'x-rapidapi-key: XxXxXxXxXxXxXxXxXxXxXxXx'
```

6.9.4. Standings Response Sample

```
200 OK
Content Type: application/json
{
 "get": "players/statistics ",
 "parameters": {
                    "league":"standard"
                    "team":"19"
                    "season":"2019"
                },
 "errors": [],
 "results": 1,
 "response": [
                    0:
                    {
                    "league":"standard"
                    "season":2019
                    "team":{5 items
                    "id":19
                    "name":"Memphis Grizzlies"
                    "nickname":"Grizzlies"
                    "code":"MEM"
                    "logo":"https://upload.wikimedia.org/wikipedia/en/thumb/f/f1/
                            Memphis_Grizzlies.svg/1200px-Memphis_Grizzlies.svg.png"
                    }
                    "conference":
                            {
                            "name":"west"
                            "rank":9
                            "win":20
                            "loss":26
                            }
                    "division":
                            {
                            "name":"southwest"
                            "rank":3
                            "win":4
                            "loss":9
```

```
                        "gamesBehind":"10.5"
                        }
                "win":
                        {
                        "home":20
                        "away":14
                        "total":34
                        "percentage":".466"
                        "lastTen":3
                        }
                "loss":
                        {
                        "home":17
                        "away":22
                        "total":39
                        "percentage":".534"
                        "lastTen":7
                        }
                        "gamesBehind":"19.0"
                        "streak":1
                        "winStreak":true
                }
        ]
}
```

7. Limitations

While working with API-NBA, keep in mind the existing limits:

Limit	Description
Requests per minute	The limit depends on a pricing plan and is equal to: • 10 requests for *basic* users • 300 requests for *pro* users • 450 requests for *ultra* users • 900 requests for *mega* users
Requests per day	The limit depends on a pricing plan and is equal to: • 100 requests for *basic* users (you can exceed the limit for an additional cost) • 5 000 requests for *pro* users • 20 000 requests for *ultra* users • 50 000 requests for *mega* users

8. Possible Errors

While working with API-NBA, you might deal with some errors, including:

Error	Error Description	What Can You Do?
400	Bad Request	Make sure that a request format matches the format described in this document. Make sure that your request contains all the required parameters.
401	Unauthorized	Make sure your request contains an actual and valid *x-rapidapi-key.*
404	Not Found	Re-check a request URL and try again.
405	Method Not Allowed	Make sure you send a result with a GET method since it is the only supported method currently.
429	Too Many Requests	Try again later or change a pricing plan to exceed your limits.
500	Internal Server Error	Try again later.
502	Bad Gateway	Try again later.

In case of any of these errors, a response will contain an error code as an *error* parameter. The parameter matches an HTTP response code.

If you are dealing with an error that is not in the list, please start a new discussion in this tread and we will help you as fast as we can.