apisports Documentation

Release 0.3.1

MikeSmithEU

Jun 14, 2021

CONTENTS

A simple python library for easy querying of APISports data.

CHAPTER ONE

EXAMPLE USAGE

```
from apisports import Football

api = Football(api_key='XXXXXXXX')

# get all players for Chelsea FC (id=49) for the 2020 season
# and sort by age

players = api.players(season=2020, team=49)

if not players.ok:
    print("something went wrong: " + players.error_description)
else:
    sorted_by_age = sorted(players, key=lambda player : player['player']['age'])
    def format_player(player):
        return "{firstname} {lastname}: {age}".format(
            firstname = player['player']['firstname'],
            lastname = player['player']['lastname'],
            age = player['player']['age']
        )

    for player in sorted_by_age:
        print(format_player(player))
```

outputs:

```
Harvey Vale: 18
Dynel Simeu: 19
Myles Peart-Harris: 19
Ian Maatsen: 19
Valentino Livramento: 19
Lewis Bate: 19
Karlo Žiger: 20
Henry Lawrence: 20
Billy Gilmour: 20
Faustino Anjorin: 20
Armando Broja: 20
Marc Guehi: 21
Conor Gallagher: 21
Tariq Lamptey: 21
Juan Familia-Castillo: 21
Ethan Ampadu: 21
Callum Hudson-Odoi: 21
Jamie Cumming: 22
Luke McCormick: 22
```

(continues on next page)

(continued from previous page)

```
Kai Havertz: 22
Mason Mount: 22
Reece James: 22
Christian Pulisic: 23
Nathan Baxter: 23
Isaiah Brown: 24
Fikayo Tomori: 24
Jake Clarke-Salter: 24
Tammy Bakumo-Abraham: 24
Ruben Loftus-Cheek: 25
Robert Kenedy Nunes do Nascimento: 25
Ben Chilwell: 25
Timo Werner: 25
Andreas Bødtker Christensen: 25
Lewis Baker: 26
Kepa Arrizabalaga Revuelta: 27
Emerson Palmieri dos Santos: 27
Mateo Kovačić: 27
Kurt Happy Zouma: 27
Ross Barkley: 28
Hakim Ziyech: 28
Antonio Rüdiger: 28
Michy Batshuayi Tunga: 28
Davide Zappacosta: 29
Edouard Mendy: 29
Jorge Luiz Frello Filho: 30
N'Golo Kanté: 30
Victor Moses: 31
Marcos Alonso Mendoza: 31
César Azpilicueta Tanco: 32
Willian Borges da Silva: 33
Pedro Eliezer Rodríguez Ledesma: 34
Olivier Giroud: 35
Thiago Emiliano da Silva: 37
Wilfredo Daniel Caballero: 40
```

CHAPTER

TWO

INSTALLATION

Note: Assumes Python 3.6 or greater is already installed on the system.

Versions are published on PyPi and can be easily installed using `pip`, this will also install all requirements:

```
python3 -m pip install apisports
```

2.1 Sports Classes

2.1.1 Baseball

class apisports.**Baseball**(*host=None*, *api_key=None*, *session=None*)

countries(***kwargs*)

Get the list of available countries.

The `name` and `code` fields can be used in other endpoints as filters.

All the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the country
- **name** (*string*) – The name of the country
- **code** (*string*) – The code of the country
- **search** (*string*) –

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

games(***kwargs*)

For all requests to games you can add the query parameter `timezone` to your request in order to retrieve the list of games in the time zone of your choice like *"Europe/London"*

To know the list of available time zones you have to use the endpoint timezone

Available status

- NS : Not Started

- IN1 : Inning 1
- IN2 : Inning 2
- IN3 : Inning 3
- IN4 : Inning 4
- IN5 : Inning 5
- IN6 : Inning 6
- IN7 : Inning 7
- IN8 : Inning 8
- IN9 : Inning 9
- POST : Postponed
- CANC : Cancelled
- INTR : Interrupted
- ABD : Abandoned
- FT : Finished

Games are updated every 15 seconds

This endpoint requires at least one parameter.

Parameters

- **`id`** (*`integer`*) – The id of the game
- **`date`** (*`string`*) – A valid date
- **`league`** (*`integer`*) – The id of the league
- **`season`** (*`integer`*) – The season of the league
- **`team`** (*`integer`*) – The id of the team
- **`timezone`** (*`string`*) – A valid timezone

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

`games_h2h`(***kwargs*)

Get heads to heads between two teams.

Parameters

- **`h2h`** (*`string`*) – The ids of the teams (id-id)
- **`date`** (*`string`*) – A valid date (YYYY-MM-DD)
- **`league`** (*`integer`*) – The id of the league
- **`season`** (*`integer`*) – The season of the league
- **`timezone`** (*`string`*) – A valid timezone

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

leagues(***kwargs*)

Get the list of available leagues and cups.

The league `id` are **unique** in the API and leagues keep it across all `seasons`

Most of the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the league
- **name** (*string*) – The name of the league
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country
- **type** (*string*) – The type of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds(***kwargs*)

Get odds from games or leagues.

We provide pre-match odds between 1 and 7 days before the game.

We keep a 7-day history *(The availability of odds may vary according to the leagues, seasons, games and bookmakers)*

Odds are updated once a day

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **game** (*integer*) – The id of the game
- **bookmaker** (*integer*) – The id of the bookmaker
- **bet** (*integer*) – The id of the bet

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds_bets(***kwargs*)

Get all available bets.

All bets `id` can be used in endpoint odds as filters

Parameters

- **id** (*integer*) – The id of the bet
- **search** (*string*) – The name of the bet

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds_bookmakers(***kwargs*)

Get all available bookmakers.

All bookmakers `id` can be used in endpoint odds as filters.

Parameters

- **id** (*integer*) – The id of the bookmaker
- **search** (*string*) – The name of the bookmaker

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

seasons(***kwargs*)

All seasons are only **4-digit keys**, so for a league whose season is `2018-2019` the season in the API will be `2018`.

All `seasons` can be used in other endpoints as filters.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings(***kwargs*)

Get the standings for a league.

Return a table of one or more rankings according to the league / cup. Some competitions have several rankings in a year, regular season, pre season etc…

To know the list of available stages or groups you have to use the endpoint standings/stages or standings/groups

Standings are updated every hours

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **stage** (*string*) – A valid stage
- **group** (*string*) – A valid group

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_groups(***kwargs*)

Get the list of available groups for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_stages (***kwargs*)

Get the list of available stages for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams (***kwargs*)

Get data about teams.

The team `id` are **unique** in the API and teams keep it among all the leagues/cups in which they participate.

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the team
- **name** (*string*) – The name of the team
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams_statistics (***kwargs*)

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league
- **team** (*integer*) – The id of the team
- **date** (*string*) – A date limit (YYYY-MM-DD)

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

timezone (***kwargs*)

Get the list of available timezone to be used in the fixtures endpoint.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

2.1.2 Basketball

class apisports.**Basketball**(*host=None*, *api_key=None*, *session=None*)

countries(***kwargs*)

Get the list of available countries.

The `id` `name` and `code` fields can be used in other endpoints as filters.

All the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the country
- **name** (*string*) – The name of the country
- **code** (*string*) – The code of the country
- **search** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

games(***kwargs*)

For all requests to games you can add the query parameter `timezone` to your request in order to retrieve the list of games in the time zone of your choice like *"Europe/London"*

To know the list of available time zones you have to use the endpoint timezone

Available status

- NS : Not Started
- Q1 : Quarter 1 (In Play)
- Q2 : Quarter 2 (In Play)
- Q3 : Quarter 3 (In Play)
- Q4 : Quarter 4 (In Play)
- OT : Over Time (In Play)
- BT : Break Time (In Play)
- HT : Halftime (In Play)
- FT : Game Finished (Game Finished)
- AOT : After Over Time (Game Finished)
- POST : Game Postponed
- CANC : Game Cancelled

Games are updated every 15 seconds

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the game

- **date** (*string*) – A valid date
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **timezone** (*string*) – A valid timezone

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

games_h2h (***kwargs*)

Get heads to heads between two teams.

Parameters

- **h2h** (*string*) – The ids of the teams (id-id)
- **date** (*string*) – A valid date (YYYY-MM-DD)
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **timezone** (*string*) – A valid timezone

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

leagues (***kwargs*)

Get the list of available leagues and cups.

The league `id` are **unique** in the API and leagues keep it across all `seasons`

Most of the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the league
- **name** (*string*) – The name of the league
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country
- **type** (*string*) – The type of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) – The code of the country
- **code** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds (***kwargs*)

Get odds from games or leagues.

We provide pre-match odds between 1 and 7 days before the game.

We keep a 1-Month history *(The availability of odds may vary according to the leagues, seasons, games and bookmakers)*

Odds are updated once a day

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **game** (*integer*) – The id of the game
- **bookmaker** (*integer*) – The id of the bookmaker
- **bet** (*integer*) – The id of the bet

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds_bets (***kwargs*)

Get all available bets.

All bets id can be used in endpoint odds as filters

Parameters

- **id** (*integer*) – The id of the bet
- **search** (*string*) – The name of the bet

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds_bookmakers (***kwargs*)

Get all available bookmakers.

All bookmakers id can be used in endpoint odds as filters.

Parameters

- **id** (*integer*) – The id of the bookmaker
- **search** (*string*) – The name of the bookmaker

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

seasons (***kwargs*)

All seasons can be used in other endpoints as filters.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings (***kwargs*)

Get the standings for a league.

Return a table of one or more rankings according to the league / cup. Some competitions have several rankings in a year, regular season, pre season etc. . .

To know the list of available stages or groups you have to use the endpoint standings/stages or standings/groups

Standings are updated every hours

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **stage** (*string*) – A valid stage
- **group** (*string*) – A valid group

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_groups(***kwargs*)

Get the list of available groups for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_stages(***kwargs*)

Get the list of available stages for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams(***kwargs*)

Get data about teams.

The team `id` are **unique** in the API and teams keep it among all the leagues/cups in which they participate.

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the team
- **name** (*string*) – The name of the team
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams_statistics(***kwargs*)

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league
- **team** (*integer*) – The id of the team
- **date** (*string*) – A Limit Date (YYYY-MM-DD)

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

timezone(***kwargs*)

Get the list of available timezone to be used in the games endpoint.

This endpoint does not require any parameters.

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

2.1.3 Football

class apisports.**Football**(*host=None*, *api_key=None*, *session=None*)

bets(***kwargs*)

Get all available bets.

All bets `id` can be used in endpoint odds as filters.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **id** (*string*) – The id of the bet name
- **search** (*string*) – The name of the bet

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

bookmakers(***kwargs*)

Get all available bookmakers.

All bookmakers `id` can be used in endpoint odds as filters.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **id** (*integer*) – The id of the bookmaker
- **search** (*string*) – The name of the bookmaker

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

coachs (***kwargs*)

Get all the information about the coachs and their careers.

To get the photo of a coach you have to call the following url: `https://media.api-sports.io/football/coachs/{coach_id}.png`

Update Frequency : This endpoint is updated every day.

Recommended Calls : 1 call per day.

Parameters

- **id** (*integer*) – The id of the coach
- **team** (*integer*) – The id of the team
- **search** (*string*) – The name of the coach

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

countries (***kwargs*)

Get the list of available countries.

The `name` and `code` fields can be used in other endpoints as filters.

To get the flag of a country you have to call the following url: `https://media.api-sports.io/flags/{country_code}.svg`

Examples available in Request samples "Use Cases".

All the parameters of this endpoint can be used together.

Update Frequency : This endpoint is updated each time a new league from a country not covered by the API is added.

Recommended Calls : 1 call per day.

Parameters

- **name** (*string*) – The name of the country
- **code** (*string*) – The Alpha2 code of the country (FR, GB, IT . . .)
- **search** (*string*) – The name of the country

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

fixtures (***kwargs*)

For all requests to fixtures you can add the query parameter `timezone` to your request in order to retrieve the list of matches in the time zone of your choice like *"Europe/London"*

To know the list of available time zones you have to use the endpoint timezone

Some leagues have only final result check our coverage page to know which ones

Available fixtures status

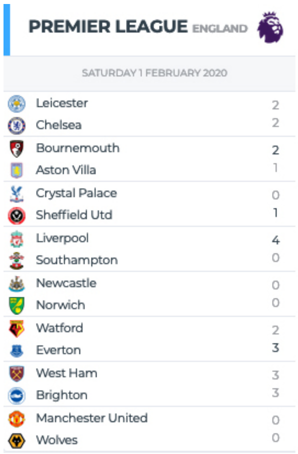
- TBD : Time To Be Defined
- NS : Not Started
- 1H : First Half, Kick Off

- HT : Halftime
- 2H : Second Half, 2nd Half Started
- ET : Extra Time
- P : Penalty In Progress
- FT : Match Finished
- AET : Match Finished After Extra Time
- PEN : Match Finished After Penalty
- BT : Break Time (in Extra Time)
- SUSP : Match Suspended
- INT : Match Interrupted
- PST : Match Postponed
- CANC : Match Cancelled
- ABD : Match Abandoned
- AWD : Technical Loss
- WO : WalkOver

Update Frequency : This endpoint is updated every 15 seconds.

Recommended Calls : 1 call per minute for the leagues, teams, fixtures who have at least one fixture in progress otherwise 1 call per day.

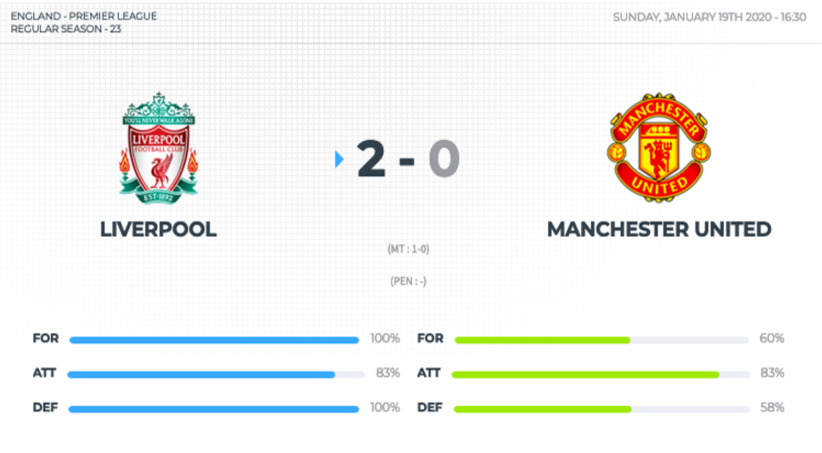
Here are several examples of what can be achieved



PREMIER LEAGUE ENGLAND

SATURDAY 1 FEBRUARY 2020

Leicester	2
Chelsea	2
Bournemouth	2
Aston Villa	1
Crystal Palace	0
Sheffield Utd	1
Liverpool	4
Southampton	0
Newcastle	0
Norwich	0
Watford	2
Everton	3
West Ham	3
Brighton	3
Manchester United	0
Wolves	0



ENGLAND - PREMIER LEAGUE
REGULAR SEASON - 23

SUNDAY, JANUARY 19TH 2020 - 16:30

LIVERPOOL 2 - 0 MANCHESTER UNITED

(MT : 1-0)

(PEN : -)

FOR	100%	FOR	60%
ATT	83%	ATT	83%
DEF	100%	DEF	58%

Parameters

- **id** (*integer*) – The id of the fixture
- **live** (*string*) –
- **date** (*string*) – A valid date (YYYY-MM-DD)
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **team** (*integer*) – The id of the team
- **last** (*integer*) – For the X last fixtures

- **next** (*integer*) – For the X next fixtures
- **from** (*string*) – A valid date (YYYY-MM-DD)
- **to** (*string*) – A valid date (YYYY-MM-DD)
- **round** (*string*) – The round of the fixture
- **status** (*string*) – The status short of the fixture
- **timezone** (*string*) – A valid timezone from the endpoint *Timezone*

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

fixtures_events (***kwargs*)

Get the events from a fixture.

Available events

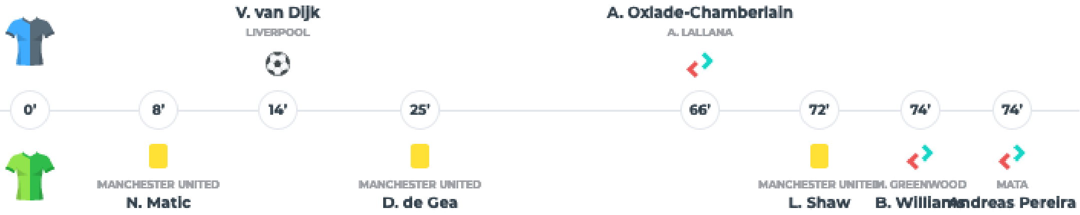
- *VAR events are available from the 2020-2021 season.*

Update Frequency : This endpoint is updated every 15 seconds.

Recommended Calls : 1 call per minute for the fixtures in progress otherwise 1 call per day.

You can also retrieve all the events of the fixtures in progress thanks to the endpoint `fixtures?live=all`

Here is an example of what can be achieved



Parameters

- **fixture** (*integer*) – The id of the fixture
- **team** (*integer*) – The id of the team
- **player** (*integer*) – The id of the player
- **type** (*string*) – The type

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

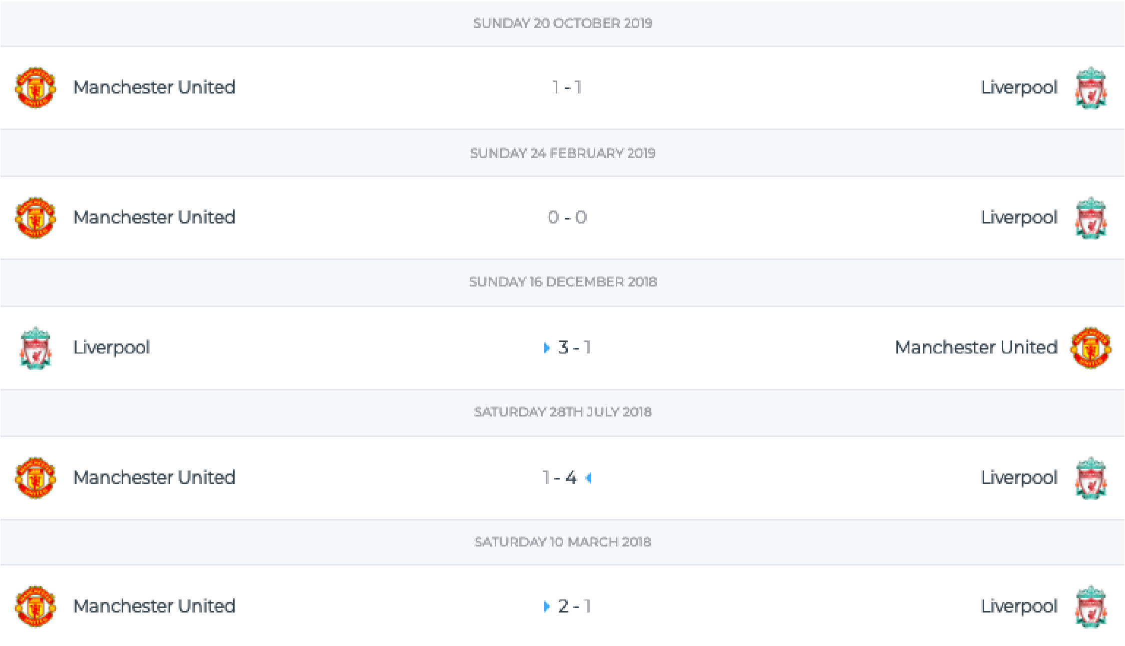
fixtures_headtohead (***kwargs*)

Get heads to heads between two teams.

Update Frequency : This endpoint is updated every 15 seconds.

Recommended Calls : 1 call per minute for the leagues, teams, fixtures who have at least one fixture in progress otherwise 1 call per day.

Here is an example of what can be achieved



	SUNDAY 20 OCTOBER 2019	
Manchester United	1 - 1	Liverpool
	SUNDAY 24 FEBRUARY 2019	
Manchester United	0 - 0	Liverpool
	SUNDAY 16 DECEMBER 2018	
Liverpool	▸ 3 - 1	Manchester United
	SATURDAY 28TH JULY 2018	
Manchester United	1 - 4 ◂	Liverpool
	SATURDAY 10 MARCH 2018	
Manchester United	▸ 2 - 1	Liverpool

Parameters

- **h2h** (*string*) – The ids of the teams (ID-ID)
- **date** (*string*) – (YYYY-MM-DD)
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **last** (*integer*) – For the X last fixtures
- **next** (*integer*) – For the X next fixtures
- **from** (*string*) – (YYYY-MM-DD)
- **to** (*string*) – (YYYY-MM-DD)
- **status** (*string*) – The status short of the fixture
- **timezone** (*string*) – A valid timezone from the endpoint *Timezone*

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

fixtures_lineups (***kwargs*)

Get the lineups for a fixture.

Available datas

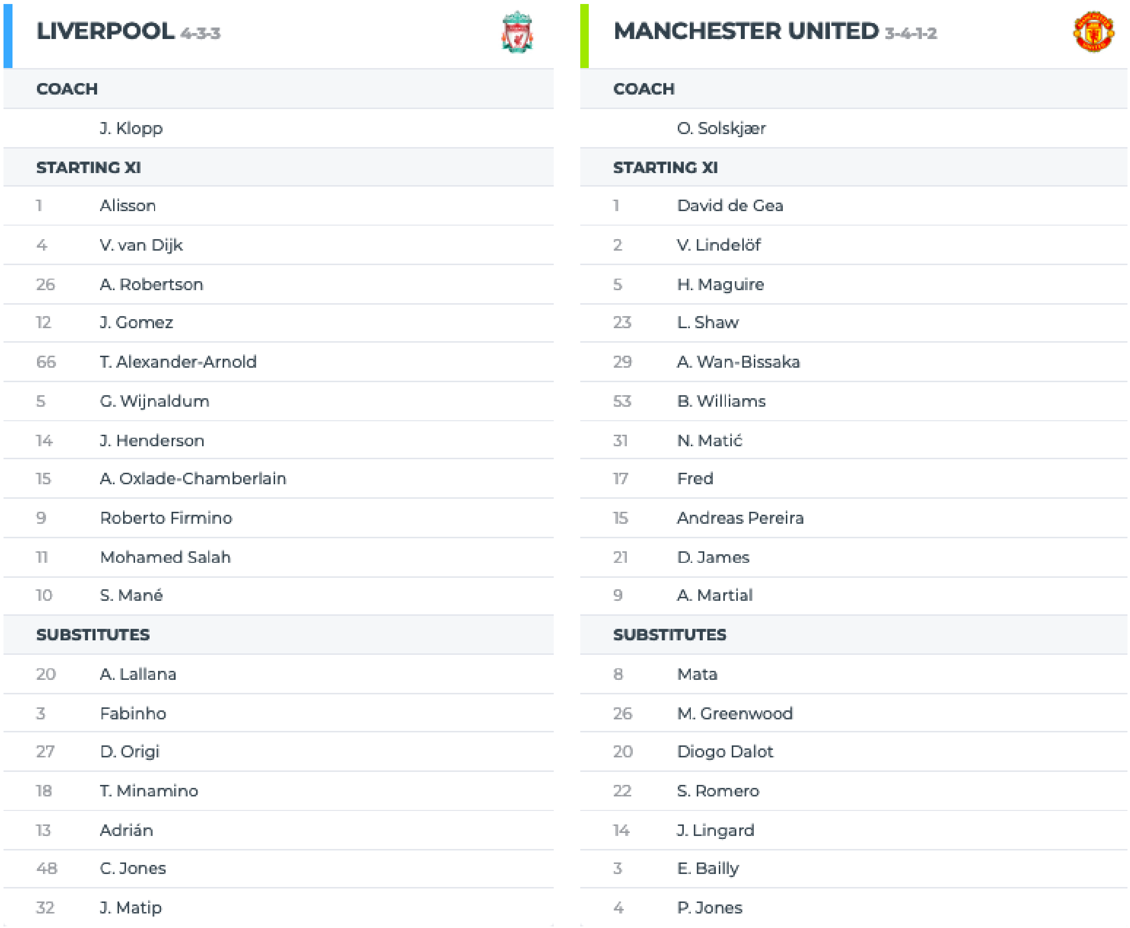
- Formation
- Coach
- Start XI
- Substitutes

Lineups are available between 20 and 40 minutes before the fixture.

Update Frequency : This endpoint is updated every 15 minutes.

Recommended Calls : 1 call every 15 minutes for the fixtures in progress otherwise 1 call per day.

Here is an example of what can be achieved



LIVERPOOL 4-3-3	
COACH	
	J. Klopp
STARTING XI	
1	Alisson
4	V. van Dijk
26	A. Robertson
12	J. Gomez
66	T. Alexander-Arnold
5	G. Wijnaldum
14	J. Henderson
15	A. Oxlade-Chamberlain
9	Roberto Firmino
11	Mohamed Salah
10	S. Mané
SUBSTITUTES	
20	A. Lallana
3	Fabinho
27	D. Origi
18	T. Minamino
13	Adrián
48	C. Jones
32	J. Matip

MANCHESTER UNITED 3-4-1-2	
COACH	
	O. Solskjær
STARTING XI	
1	David de Gea
2	V. Lindelöf
5	H. Maguire
23	L. Shaw
29	A. Wan-Bissaka
53	B. Williams
31	N. Matić
17	Fred
15	Andreas Pereira
21	D. James
9	A. Martial
SUBSTITUTES	
8	Mata
26	M. Greenwood
20	Diogo Dalot
22	S. Romero
14	J. Lingard
3	E. Bailly
4	P. Jones

Parameters

- **`fixture`** (*integer*) – The id of the fixture
- **`team`** (*integer*) – The id of the team
- **`player`** (*integer*) – The id of the player
- **`type`** (*string*) – The type

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

`fixtures_players`(***kwargs*)

Get the players statistics from one fixture.

Update Frequency : This endpoint is updated every minute.

Recommended Calls : 1 call every minute for the fixtures in progress otherwise 1 call per day.

Parameters

- **`fixture`** (*integer*) – The id of the fixture
- **`team`** (*integer*) – The id of the team

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

fixtures_rounds(***kwargs*)

Get the rounds for a league or a cup.

The `round` can be used in endpoint fixtures as filters

Examples available in Request samples "Use Cases".

Update Frequency : This endpoint is updated every day.

Recommended Calls : 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **current** (*boolean*) – The current round only

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

fixtures_statistics(***kwargs*)

Get the statistics for one fixture.

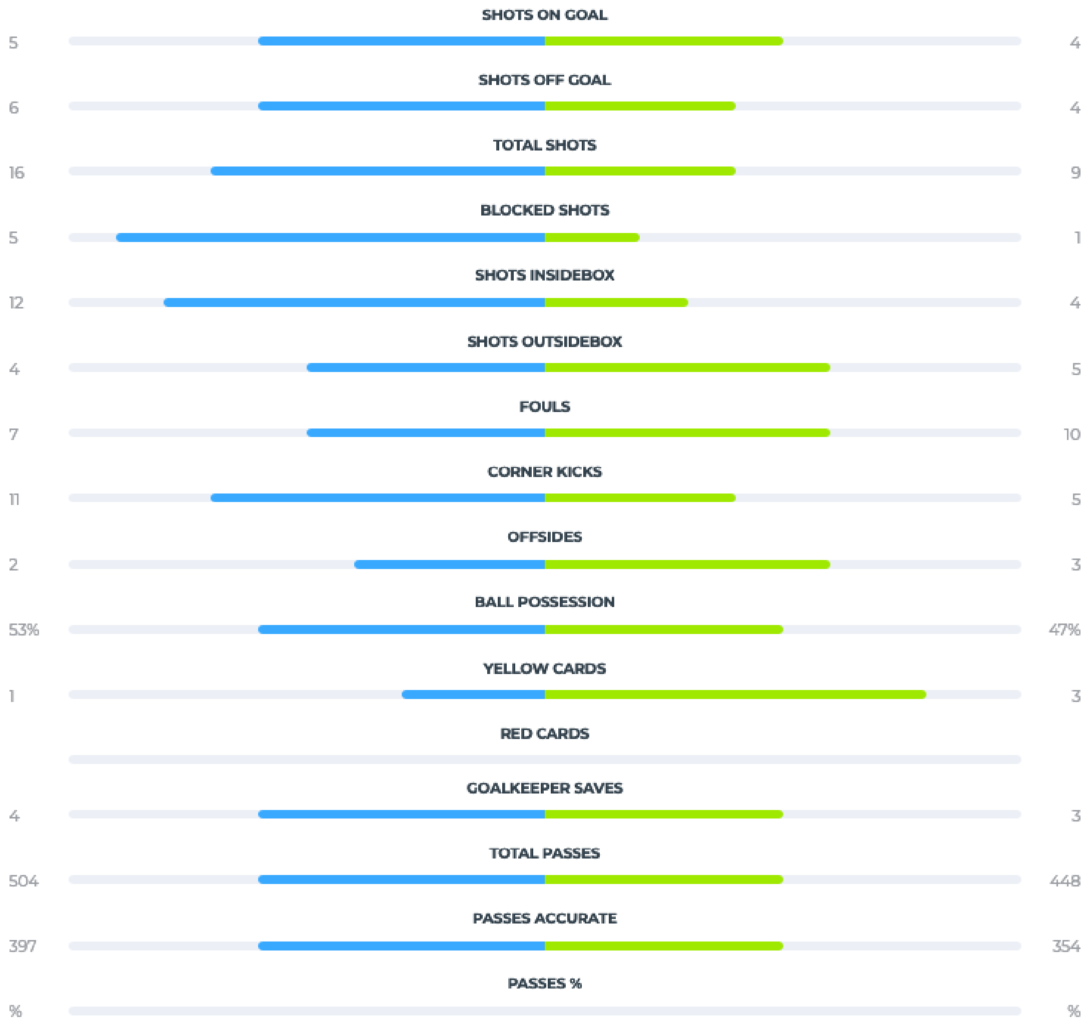
Available statistics

- Shots on Goal
- Shots off Goal
- Shots insidebox
- Shots outsidebox
- Total Shots
- Blocked Shots
- Fouls
- Corner Kicks
- Offsides
- Ball Possession
- Yellow Cards
- Red Cards
- Goalkeeper Saves
- Total passes
- Passes accurate
- Passes %

Update Frequency : This endpoint is updated every minute.

Recommended Calls : 1 call every minute for the teams or fixtures who have at least one fixture in progress otherwise 1 call per day.

Here is an example of what can be achieved



Parameters

- **fixture** (*integer*) – The id of the fixture
- **team** (*integer*) – The id of the team
- **type** (*string*) – The type of statistics

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

injuries(***kwargs*)

Get the list of players not participating in the fixtures for various reasons such as `suspended`, `injured` for example.

Being a new endpoint, the data is only available from April 2021.

There are two types:

- `Missing Fixture` : The player will not play the fixture.
- `Questionable` : The information is not yet 100% sure, the player may eventually play the fixture.

Examples available in Request samples “Use Cases”.

All the parameters of this endpoint can be used together.

This endpoint requires at least one parameter.

Update Frequency : This endpoint is updated every 4 hours.

Recommended Calls : 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league, **required** with *league*, *team* and *player* parameters (YYYY)
- **fixture** (*integer*) – The id of the fixture
- **team** (*integer*) – The id of the team
- **player** (*integer*) – The id of the player
- **date** (*string*) – A valid date (YYYY-MM-DD)
- **timezone** (*string*) – A valid timezone from the endpoint *Timezone*

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

leagues(***kwargs*)

Get the list of available leagues and cups.

The league `id` are **unique** in the API and leagues keep it across all `seasons`

To get the logo of a competition you have to call the following url: `https://media.api-sports.io/football/leagues/{league_id}.png`

This endpoint also returns the `coverage` of each competition, which makes it possible to know what is available for that league or cup.

Example :

```
"coverage": {
  "fixtures": {
      "events": true,
      "lineups": true,
      "statistics_fixtures": false,
      "statistics_players": false
  },
  "standings": true,
  "players": true,
  "top_scorers": true,
  "top_assists": true,
  "top_cards": true,
  "injuries": true,
  "predictions": true,
  "odds": false
}
```

In this example we can deduce that the competition does not have the following features: `statistics_fixtures`, `statistics_players`, `odds` because it is set to `False`.

The coverage of a competition can vary from season to season.

Examples available in Request samples "Use Cases".

Most of the parameters of this endpoint can be used together.

Update Frequency : This endpoint is updated several times a day.

Recommended Calls : 1 call per hour.

Parameters

- **id** (*integer*) – The id of the league
- **name** (*string*) – The name of the league
- **country** (*string*) – The country name of the league
- **code** (*string*) – The Alpha2 code of the country (FR, GB, IT…)
- **season** (*integer*) – The season of the league (YYYY)
- **team** (*integer*) – The id of the team
- **type** (*string*) – The type of the league
- **current** (*string*) – The state of the league
- **search** (*string*) – The name or the country of the league
- **last** (*integer*) – The X last leagues/cups added in the API

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds(***kwargs*)

Get odds from fixtures, leagues or date.

This endpoint uses a **pagination system**, you can navigate between the different pages thanks to the `page` parameter.

Pagination : 10 results per page.

We provide pre-match odds between 1 and 14 days before the fixture.

We keep a 7-days history *(The availability of odds may vary according to the leagues, seasons, fixtures and bookmakers)*

Update Frequency : This endpoint is updated every day.

Recommended Calls : 1 call per day.

Parameters

- **fixture** (*integer*) – The id of the fixture
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **date** (*string*) – A valid date (YYYY-MM-DD)
- **timezone** (*string*) – A valid timezone from the endpoint *Timezone*
- **page** (*integer*) – Use for the pagination
- **bookmaker** (*integer*) – The id of the bookmaker
- **bet** (*integer*) – The id of the bet

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds_mapping(***kwargs*)

Get the list of available fixtures `id` for the endpoint odds.

All fixtures, leagues `id` and `date` can be used in endpoint odds as filters.

This endpoint uses a **pagination system**, you can navigate between the different pages thanks to the `page` parameter.

Pagination : 100 results per page.

Update Frequency : This endpoint is updated every day.

Recommended Calls : 1 call per day.

Parameters **page** (*integer*) – Use for the pagination

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

players(***kwargs*)

Get players statistics.

The players `id` are unique in the API and players keep it among all the teams they have been in

The statistics are calculated according to the team `id`, league `id` and `season`.

You can find the available `seasons` by using the endpoint players seasons.

To get the photo of a player you have to call the following url: `https://media.api-sports.io/football/players/{player_id}.png`

This endpoint uses a **pagination system**, you can navigate between the different pages thanks to the `page` parameter.

Pagination : 20 results per page.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **id** (*integer*) – The id of the player
- **team** (*integer*) – The id of the team
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY | Requires the fields Id, League or Team)
- **search** (*string*) – The name of the player (Requires the fields League or Team)
- **page** (*integer*) – Use for the pagination

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

players_seasons(***kwargs*)

Get all available seasons for players statistics.

Update Frequency : This endpoint is updated every day.

Recommended Calls : 1 call per day.

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

players_topassists(***kwargs*)

Get the 20 best players assists for a league or cup.

How it is calculated:

- 1 : The player that has delivered the higher number of goal assists
- 2 : The player that has scored the higher number of goals
- 3 : The player that has scored the fewer number of penalties
- 4 : The player that assists in the higher number of matches
- 5 : The player that played the fewer minutes
- 6 : The player that received the fewer number of red cards
- 7 : The player that received the fewer number of yellow cards

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

players_topredcards(***kwargs*)

Get the 20 players with the most red cards for a league or cup.

How it is calculated:

- 1 : The player that received the higher number of red cards
- 2 : The player that received the higher number of yellow cards
- 3 : The player that assists in the higher number of matches
- 4 : The player that played the fewer minutes

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

players_topscorers(***kwargs*)

Get the 20 best players for a league or cup.

How it is calculated:

- 1 : The player that has scored the higher number of goals

- 2 : The player that has scored the fewer number of penalties
- 3 : The player that has delivered the higher number of goal assists
- 4 : The player that scored their goals in the higher number of matches
- 5 : The player that played the fewer minutes
- 6 : The player that plays for the team placed higher on the table
- 7 : The player that received the fewer number of red cards
- 8 : The player that received the fewer number of yellow cards

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

players_topyellowcards(***kwargs*)

Get the 20 players with the most yellow cards for a league or cup.

How it is calculated:

- 1 : The player that received the higher number of yellow cards
- 2 : The player that received the higher number of red cards
- 3 : The player that assists in the higher number of matches
- 4 : The player that played the fewer minutes

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

predictions(***kwargs*)

Get predictions about a fixture.

The predictions are made using several algorithms including the poisson distribution, comparison of team statistics, last matches, players etc…

Bookmakers odds are not used to make these predictions

Also provides some comparative statistics between teams

Available Predictions

- Match winner : Id of the team that can potentially win the fixture
- Win or Draw : If `True` indicates that the designated team can win or draw

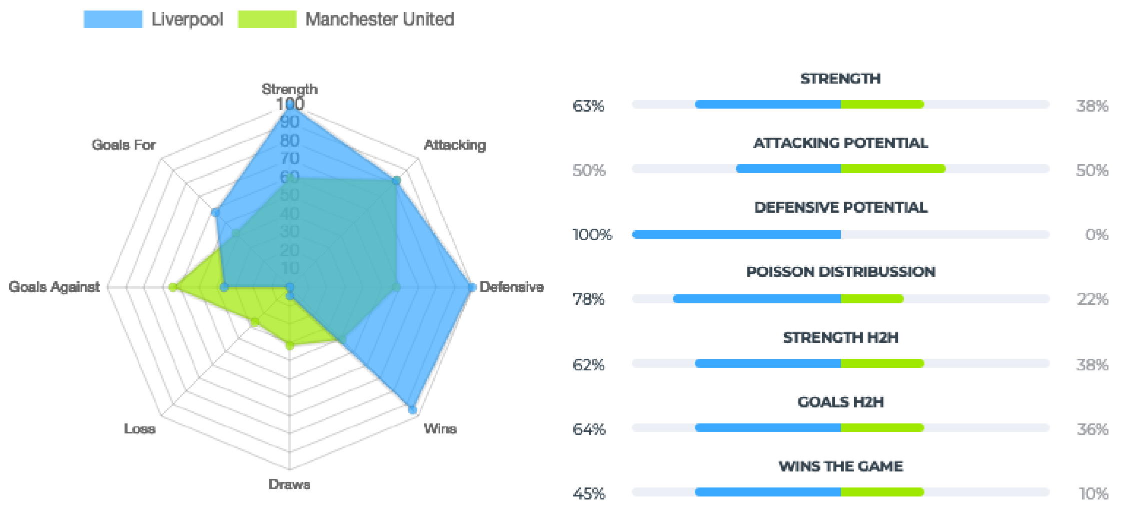
- Under / Over : -1.5 / -2.5 / -3.5 / -4.5 / +1.5 / +2.5 / +3.5 / +4.5 ⋆
- Goals Home : -1.5 / -2.5 / -3.5 / -4.5 ⋆
- Goals Away -1.5 / -2.5 / -3.5 / -4.5 ⋆
- Advice *(Ex : Deportivo Santani or draws and -3.5 goals)*

⋆ **-1.5** means that there will be a maximum of **1.5** goals in the fixture, i.e : **1** goal

Update Frequency : This endpoint is updated every hour.

Recommended Calls : 1 call per hour for the fixtures in progress otherwise 1 call per day.

Here is an example of what can be achieved



Parameters **`fixture`** (*`integer`*) – The id of the fixture

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

`seasons`(***kwargs*)

Get the list of available seasons.

All seasons are only **4-digit keys**, so for a league whose season is `2018-2019` like the English Premier League (EPL), the `2018-2019` season in the API will be `2018`.

All `seasons` can be used in other endpoints as filters.

This endpoint does not require any parameters.

Update Frequency : This endpoint is updated each time a new league is added.

Recommended Calls : 1 call per day.

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

`sidelined`(***kwargs*)

Get all available sidelined for a player or a coach.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **player** (*integer*) – The id of the player
- **coach** (*integer*) – The id of the coach

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings(***kwargs*)

Get the standings for a league or a team.

Return a table of one or more rankings according to the league / cup.

Some competitions have several rankings in a year, group phase, opening ranking, closing ranking etc…

Examples available in Request samples “Use Cases”.

Most of the parameters of this endpoint can be used together.

Update Frequency : This endpoint is updated every hour.

Recommended Calls : 1 call per hour for the leagues or teams who have at least one fixture in progress otherwise 1 call per day.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **team** (*integer*) – The id of the team

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams(***kwargs*)

Get the list of available teams.

The team `id` are **unique** in the API and teams keep it among all the leagues/cups in which they participate.

To get the logo of a team you have to call the following url: `https://media.api-sports.io/football/teams/{team_id}.png`

Examples available in Request samples “Use Cases”.

All the parameters of this endpoint can be used together.

This endpoint requires at least one parameter.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **id** (*integer*) – The id of the team
- **name** (*string*) – The name of the team
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **country** (*string*) – The country name of the team
- **search** (*string*) – The name or the country name of the team

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

teams_statistics(***kwargs*)

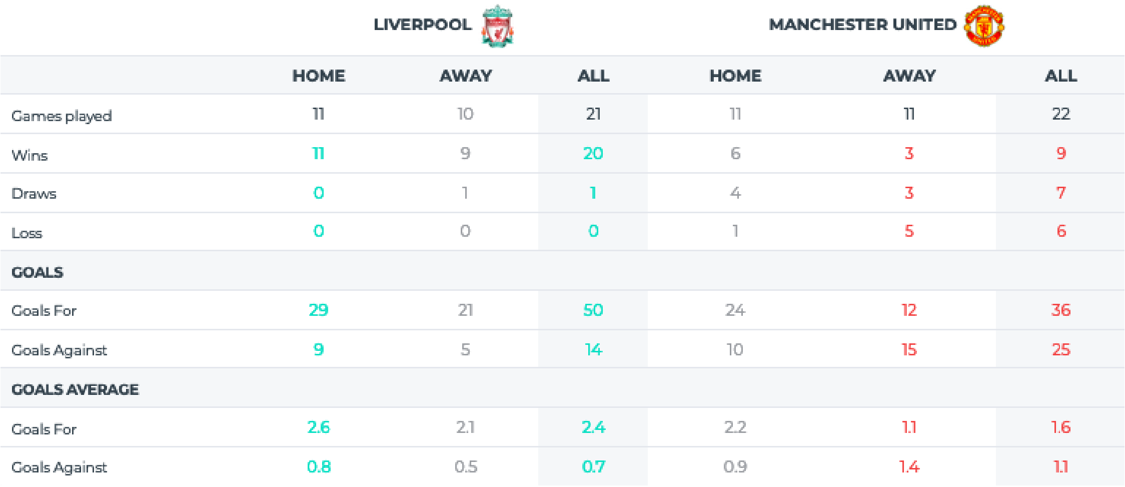
Returns the statistics of a team in relation to a given competition and season.

It is possible to add the `date` parameter to calculate statistics from the beginning of the season to the given date. By default the API returns the statistics of all games played by the team for the competition and the season.

Update Frequency : This endpoint is updated twice a day.

Recommended Calls : 1 call per day for the teams who have at least one fixture during the day otherwise 1 call per week.

Here is an example of what can be achieved



	LIVERPOOL			MANCHESTER UNITED		
	HOME	AWAY	ALL	HOME	AWAY	ALL
Games played	11	10	21	11	11	22
Wins	11	9	20	6	3	9
Draws	0	1	1	4	3	7
Loss	0	0	0	1	5	6
GOALS						
Goals For	29	21	50	24	12	36
Goals Against	9	5	14	10	15	25
GOALS AVERAGE						
Goals For	2.6	2.1	2.4	2.2	1.1	1.6
Goals Against	0.8	0.5	0.7	0.9	1.4	1.1

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league (YYYY)
- **team** (*integer*) – The id of the team
- **date** (*string*) – The limit date (YYYY-MM-DD)

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

timezone(***kwargs*)

Get the list of available timezone to be used in the fixtures endpoint.

This endpoint does not require any parameters.

Update Frequency : This endpoint contains all the existing timezone, it is not updated.

Recommended Calls : 1 call when you need.

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

transfers(***kwargs*)

Get all available transfers for players and teams

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **player** (*integer*) – The id of the player
- **team** (*integer*) – The id of the team

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

trophies(***kwargs*)

Get all available trophies for a player or a coach.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **player** (*integer*) – The id of the player
- **coach** (*integer*) – The id of the coach

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

venues(***kwargs*)

Get the list of available venues.

The venue `id` are **unique** in the API.

To get the image of a venue you have to call the following url: `https://media.api-sports.io/football/venues/{venue_id}.png`

Examples available in Request samples "Use Cases".

All the parameters of this endpoint can be used together.

This endpoint requires at least one parameter.

Update Frequency : This endpoint is updated several times a week.

Recommended Calls : 1 call per day.

Parameters

- **id** (*integer*) – The id of the venue
- **name** (*string*) – The name of the venue
- **city** (*string*) – The city of the venue
- **country** (*string*) – The country name of the venue
- **search** (*string*) – The name, city or the country of the venue

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

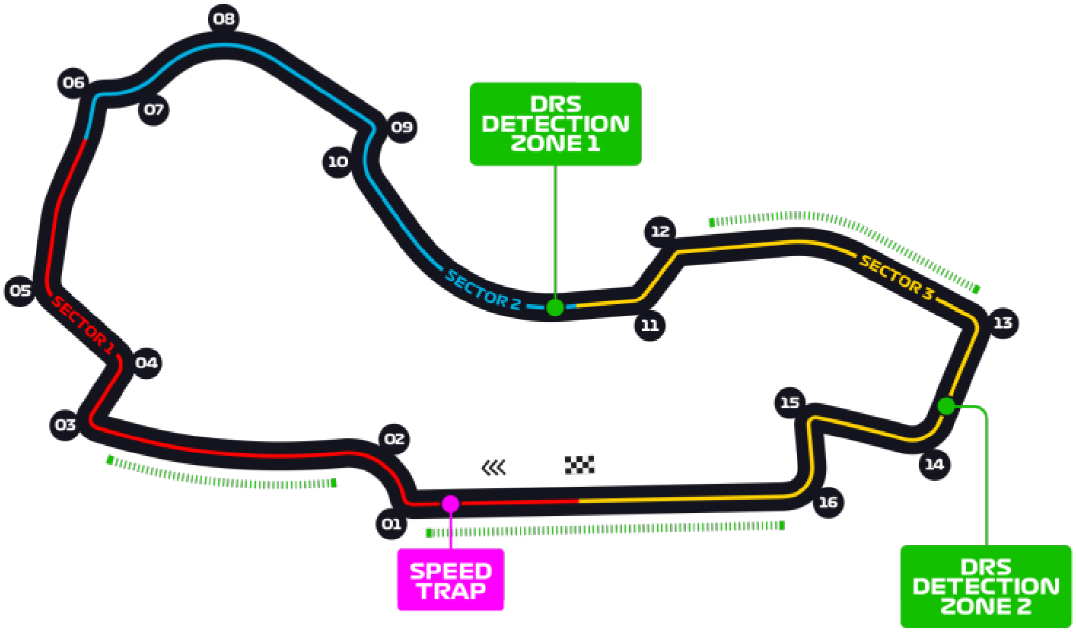
2.1.4 Formula1

class apisports.**Formula1**(*host=None*, *api_key=None*, *session=None*)

circuits(***kwargs*)
Get the list of available circuits.

The circuit `id` are **unique** in the API and circuits keep it across all `seasons`

Sample ``image`` of a circuit :



All the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the circuit
- **competition** (*integer*) – The id of the competition
- **name** (*string*) – The name of the circuit
- **search** (*string*) – Allow to search for a circuit name

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

competitions(***kwargs*)
Get the list of available competitions.

The competition `id` are **unique** in the API and competitions keep it across all `seasons`

All the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the competition
- **name** (*string*) – The name of the competition
- **country** (*string*) – The name of the country

- **city** (*string*) – The name of the city
- **search** (*string*) – Allow to search for a competition name

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

drivers (***kwargs*)

Get the list of available drivers.

The driver `id` are **unique** in the API and drivers keep it across all `seasons`

Sample ``image`` of a driver :

All the parameters of this endpoint can be used together.

This endpoint require at least one parameter.

Parameters

- **id** (*integer*) – The id of the driver
- **name** (*string*) – The name of the driver
- **search** (*string*) – Allow to search for a driver name

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

races (***kwargs*)

Get the list of available races for a competition.

For all requests to races you can add the query parameter `timezone` to your request in order to retrieve the list of races in the time zone of your choice like *"Europe/London"*

To know the list of available time zones you have to use the endpoint timezone

Available Status

- Completed
- Cancelled
- Postponed
- Scheduled

Available Types

- Race
- 1st Practice
- 2nd Practice
- 3rd Practice
- 1st Qualifying
- 2nd Qualifying
- 3rd Qualifying

> This endpoint requires at least parameters `id`, `date`, `next`, `last` or `competition` and `season`.

All the parameters of this endpoint can be used together.

Parameters

- **`id`** (*`integer`*) – The id of the race
- **`date`** (*`string`*) – A valid date (YYYY-MM-DD)
- **`next`** (*`integer`*) – The x next races
- **`last`** (*`integer`*) – The x last races
- **`competition`** (*`integer`*) – The id of the competition
- **`season`** (*`integer`*) – The season of the race
- **`type`** (*`string`*) – The type of the race
- **`timezone`** (*`string`*) – A valid timezone

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

`rankings_drivers`(***kwargs*)

Get the teams rankings for a season.

All the parameters of this endpoint can be used together.

Parameters

- **`season`** (*`integer`*) – The season
- **`driver`** (*`integer`*) – The id of the driver

- **team** (*integer*) – The id of the team

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

rankings_races (***kwargs*)

Get the rankings for a race.

All the parameters of this endpoint can be used together.

Parameters

- **race** (*integer*) – The id of the race
- **team** (*integer*) – The id of the team
- **driver** (*integer*) – The id of the driver

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

rankings_teams (***kwargs*)

Get the teams rankings for a season.

All the parameters of this endpoint can be used together.

Parameters

- **season** (*string*) – The season
- **team** (*integer*) – The id of the team

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

seasons (***kwargs*)

Get all seasons available.

All seasons are only **4-digit keys**. All results can be used in other endpoints as filters.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams (***kwargs*)

Get the list of available competitions.

The team `id` are **unique** in the API and teams keep it across all `seasons`

Sample ``logo`` of a team :

All the parameters of this endpoint can be used together.

Parameters

- **`id`** (*`integer`*) – The id of the team
- **`name`** (*`string`*) – The name of the team
- **`search`** (*`string`*) – Allow to search for a team name

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

`timezone`(***kwargs*)

Get the list of available timezone to be used in the races endpoint.

This endpoint does not require any parameters.

Returns *`AbstractResponse`* object

Return type *apisports.response.AbstractResponse*

2.1.5 Rugby

`class` `apisports.`**`Rugby`**(*host=None*, *api_key=None*, *session=None*)

`countries`(***kwargs*)

Get the list of available countries.

The `name`, `id` and `code` fields can be used in other endpoints as filters.

All the parameters of this endpoint can be used together.

Parameters

- **`id`** (*`integer`*) – The id of the country

- **name** (*string*) – The name of the country
- **code** (*string*) – The code of the country
- **search** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

games (***kwargs*)

For all requests to games you can add the query parameter `timezone` to your request in order to retrieve the list of games in the time zone of your choice like *"Europe/London"*

To know the list of available time zones you have to use the endpoint timezone.

Available status

- NS : Not Started
- AW : Awarded
- POST : Postponed
- CANC : Cancelled
- INTR : Interrupted
- ABD : Abandoned
- AET : After Extra Time
- FT : Finished

Games are updated every 15 seconds

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the game
- **date** (*string*) – A valid date
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **timezone** (*string*) – A valid timezone

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

games_h2h (***kwargs*)

Get heads to heads between two teams.

Parameters

- **h2h** (*string*) – The ids of the teams (id-id)
- **date** (*string*) – A valid date (YYYY-MM-DD)
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league

- **timezone** (*string*) – A valid timezone

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

leagues (***kwargs*)

Get the list of available leagues and cups.

The league `id` are **unique** in the API and leagues keep it across all `seasons`

Most of the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the league
- **name** (*string*) – The name of the league
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country
- **type** (*string*) – The type of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

odds (***kwargs*)

Get odds from games or leagues.

We provide pre-match odds between 1 and 7 days before the game.

We keep a 7-day history *(The availability of odds may vary according to the leagues, seasons, games and bookmakers)*

Odds are updated once a day

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **game** (*integer*) – The id of the game
- **bookmaker** (*integer*) – The id of the bookmaker
- **bet** (*integer*) – The id of the bet

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

odds_bets (***kwargs*)

Get all available bets.

All bets `id` can be used in endpoint odds as filters

Parameters

- **id** (*integer*) – The id of the bet

- **search** (*string*) – The name of the bet

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

odds_bookmakers(***kwargs*)

Get all available bookmakers.

All bookmakers `id` can be used in endpoint odds as filters.

Parameters

- **id** (*integer*) – The id of the bookmaker
- **search** (*string*) – The name of the bookmaker

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

seasons(***kwargs*)

All seasons are only **4-digit keys**, so for a league whose season is `2018-2019` the season in the API will be `2018`.

All `seasons` can be used in other endpoints as filters.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings(***kwargs*)

Get the standings for a league.

Return a table of one or more rankings according to the league / cup. Some competitions have several rankings in a year, regular season, pre season etc. . .

To know the list of available stages or groups you have to use the endpoint standings/stages or standings/groups

Standings are updated every hours

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **stage** (*string*) – A valid stage
- **group** (*string*) – A valid group

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_groups(***kwargs*)

Get the list of available groups for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league

- **season** (*integer*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_stages (***kwargs*)

Get the list of available stages for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams (***kwargs*)

Get data about teams.

The team `id` are **unique** in the API and teams keep it among all the leagues/cups in which they participate.

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the team
- **name** (*string*) – The name of the team
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams_statistics (***kwargs*)

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league
- **team** (*integer*) – The id of the team
- **date** (*string*) – A date limit (YYYY-MM-DD)

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

timezone (***kwargs*)

Get the list of available timezone to be used in the games endpoint.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

2.1.6 Hockey

class apisports.**Hockey**(*host=None*, *api_key=None*, *session=None*)

countries(***kwargs*)

Get the list of available countries.

The `name`, `id` and `code` fields can be used in other endpoints as filters.

All the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the country
- **name** (*string*) – The name of the country
- **code** (*string*) – The code of the country
- **search** (*string*) –

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

games(***kwargs*)

For all requests to games you can add the query parameter `timezone` to your request in order to retrieve the list of games in the time zone of your choice like *"Europe/London"*

To know the list of available time zones you have to use the endpoint timezone.

The `events` field indicates if events are available for this game. It is a Boolen type *"True|False"*.

Available status

- NS : Not Started
- P1 : First Period
- P2 : Second Period
- P3 : Third Period
- OT : Over Time
- PT : Penalties Time
- BT : Break Time
- AW : Awarded
- POST : Postponed
- CANC : Cancelled
- INTR : Interrupted
- ABD : Abandoned
- AOT : After Over Time
- AP : After Penalties

- FT : Finished

Games are updated every 15 seconds

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the game
- **date** (*string*) – A valid date
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **timezone** (*string*) – A valid timezone

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

games_events (***kwargs*)

Get the available events for a game.

Parameters game (*integer*) – The id of the game

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

games_h2h (***kwargs*)

Get heads to heads between two teams.

Parameters

- **h2h** (*string*) – The ids of the teams (id-id)
- **date** (*string*) – A valid date (YYYY-MM-DD)
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **timezone** (*string*) – A valid timezone

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

leagues (***kwargs*)

Get the list of available leagues and cups.

The league `id` are **unique** in the API and leagues keep it across all `seasons`

Most of the parameters of this endpoint can be used together.

Parameters

- **id** (*integer*) – The id of the league
- **name** (*string*) – The name of the league
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country

- **type** (*string*) – The type of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns AbstractResponse object

Return type *apisports.response.AbstractResponse*

odds(***kwargs*)

Get odds from games or leagues.

We provide pre-match odds between 1 and 7 days before the game.

We keep a 7-day history *(The availability of odds may vary according to the leagues, seasons, games and bookmakers)*

Odds are updated once a day

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **game** (*integer*) – The id of the game
- **bookmaker** (*integer*) – The id of the bookmaker
- **bet** (*integer*) – The id of the bet

Returns AbstractResponse object

Return type *apisports.response.AbstractResponse*

odds_bets(***kwargs*)

Get all available bets.

All bets `id` can be used in endpoint odds as filters

Parameters

- **id** (*integer*) – The id of the bet
- **search** (*string*) – The name of the bet

Returns AbstractResponse object

Return type *apisports.response.AbstractResponse*

odds_bookmakers(***kwargs*)

Get all available bookmakers.

All bookmakers `id` can be used in endpoint odds as filters.

Parameters

- **id** (*integer*) – The id of the bookmaker
- **search** (*string*) – The name of the bookmaker

Returns AbstractResponse object

Return type *apisports.response.AbstractResponse*

seasons(***kwargs*)

All seasons are only **4-digit keys**, so for a league whose season is `2018-2019` the season in the API will be `2018`.

All `seasons` can be used in other endpoints as filters.

This endpoint does not require any parameters.

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings(***kwargs*)

Get the standings for a league.

Return a table of one or more rankings according to the league / cup. Some competitions have several rankings in a year, regular season, pre season etc…

To know the list of available stages or grou^ you have to use the endpoint standings/stages or standings/groups

Standings are updated every hours

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **team** (*integer*) – The id of the team
- **stage** (*string*) – A valid stage
- **group** (*string*) – A valid group

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_groups(***kwargs*)

Get the list of available groups for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

standings_stages(***kwargs*)

Get the list of available stages for a league to be used in the standings endpoint.

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league

Returns *AbstractResponse* object

Return type *apisports.response.AbstractResponse*

teams (***kwargs*)
Get data about teams.

The team `id` are **unique** in the API and teams keep it among all the leagues/cups in which they participate.

This endpoint requires at least one parameter.

Parameters

- **id** (*integer*) – The id of the team
- **name** (*string*) – The name of the team
- **country_id** (*integer*) – The id of the country
- **country** (*string*) – The name of the country
- **league** (*integer*) – The id of the league
- **season** (*integer*) – The season of the league
- **search** (*string*) –

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

teams_statistics (***kwargs*)

Parameters

- **league** (*integer*) – The id of the league
- **season** (*string*) – The season of the league
- **team** (*integer*) – The id of the team
- **date** (*string*) – A date limit (YYYY-MM-DD)

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

timezone (***kwargs*)
Get the list of available timezone to be used in the games endpoint.

This endpoint does not require any parameters.

Returns `AbstractResponse` object

Return type *apisports.response.AbstractResponse*

2.2 Response

When calling one of the *Sports Classes* methods, you get an `AbstractResponse` object as result. Usually you only need the `ok` property to check if the request succeeded, and the `errors` and `error_description` property if there was an error. If the request succeeded, you can iterate directly over the `AbstractResponse` object. In effect, the iteration is delegated to a `AbstractData` object, which you can also access directly by calling the `data` property.

class apisports.response.**AbstractResponse** (*client*, *response*, *data=None*)
Bases: `object`

Generic response object.

Parameters

- **client** (*apisports._client.Client*) – Client object
- **response** (*requests.Response*) – Response object
- **data** (*Union[None, dict]*) – The data object returned by the API call

__iter__()

Delegates iteration to the *AbstractData* class.

__len__()

Delegates len() to the *AbstractData* class.

static create(*client*, *response*)

AbstractResponse factory method.

Parameters

- **client** (*apisports._client.Client*) – Client object
- **response** (*requests.Response*) – Response object

Returns *AbstractResponse* object

Return type *AbstractResponse*

property data

Get the *AbstractData* object.

Returns *AbstractData* object

Return type *AbstractData*

property error_description

Get a string representation of the errors, or "Success" on success...

Returns Error string

Return type str

property errors

Get the errors.

Returns Dict of errors

Return type dict

property headers

Get response headers

Returns *Headers* object

Return type *Headers*

property ok

Whether the request has completed without errors.

Type bool

property raw

Get raw Response object.

Returns Response object.

Return type *requests.Response*

property text
Get raw response text.

Returns response body as string

Return type str

class `apisports.response.`**`ErrorResponse`**(*client*, *response*, *data=None*)
Bases: *`apisports.response.AbstractResponse`*

class `apisports.response.`**`Headers`**(*headers*)
Bases: `object`

Response Headers details class.

Parameters **`headers`** (*`requests.structures.CaseInsensitiveDict`*) – `CaseInsensitiveDict` object

property rate_limit
Maximum number of API calls per minute.

Return type str

property rate_limit_remaining
Number of API calls remaining before reaching the limit per minute.

Return type str

property raw
Get raw headers.

Returns `CaseInsensitiveDict` object

Return type CaseInsensitiveDict

property requests_limit
The number of requests allocated per day according to your subscription

Return type str

property requests_remaining
The number of remaining requests per day according to your subscription.

Return type str

property server
Get the current version of the API proxy used by APISports/RapidAPI.

Return type str

class `apisports.response.`**`HttpErrorResponse`**(*client*, *response*, *data=None*)
Bases: *`apisports.response.ErrorResponse`*

property error_description
Get a string representation of the errors, or “Success” on success…

Returns Error string

Return type str

property errors
Get the errors.

Returns Dict of errors

Return type dict

class apisports.response.**SuccessResponse**(*client*, *response*, *data=None*)

Bases: *apisports.response.AbstractResponse*

property data

Get the *AbstractData* object.

Returns *AbstractData* object

Return type *AbstractData*

property ok

Whether the request has completed without errors.

Type bool

2.3 Data

class apisports.data.**AbstractData**

__iter__()

__len__()

static create(*client*, *data*)

Factory method that creates the appropriate *AbstractData* object for the given data

Returns *AbstractData* object

Return type *AbstractData*

class apisports.data.**PagedData**(*client*, *data*)

Automagically fetches next page for multi-page data.

__iter__()

__len__()

This is just an estimate (maximum) until after all data has been fetched

exception apisports.data.**PagedDataError**(*message*, *error_description=''*)

Raised when *PagedData* encounters an error when fetching the next page

class apisports.data.**SimpleData**(*data*)

Provides a simple iterator for non-paged data.

__iter__()

__len__()

class apisports.data.**SingleData**(*data*)

Adds an extra method to easily access data that only contains a single item.

__iter__()

__len__()

item()

apisports.data.**NoneData = <apisports.data.NoneData object>**

Singleton Null AbstractData implementation. To represent no data available.

2.4 License

MIT License

Copyright (c) 2021 MikeSmithEU

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CHAPTER THREE

EXTRA

- genindex

PYTHON MODULE INDEX

INDEX