

Code examples

In this page you will find examples for request our endpoints in different programming languages:

cURL

```
curl --location --request GET 'https://{{sport}}.sportdevs.com/{{endpoint}}' \
--header 'Authorization: Bearer {API-KEY}'
```



C# - RestSharp

```
var client = new RestClient("https://{{sport}}.sportdevs.com/{{endpoint}}");
client.Timeout = -1;
var request = new RestRequest(Method.GET);
request.AddHeader("Authorization", "Bearer {API-KEY}");
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);
```

Dart - http

```
var headers = {
  'Authorization': 'Bearer {API-KEY}'
};
var request = http.Request('GET', Uri.parse('https://{{sport}}.sportdevs.com/{{endpoint}}'));

request.headers.addAll(headers);

http.StreamedResponse response = await request.send();

if (response.statusCode == 200) {
  print(await response.stream.bytesToString());
}
else {
  print(response.reasonPhrase);
}
```

Go - Native

```
package main

import (
    "fmt"
    "net/http"
    "io/ioutil"
)

func main() {

    url := "https://{{sport}}.sportdevs.com/{{endpoint}}"
    method := "GET"

    client := &http.Client {
    }
    req, err := http.NewRequest(method, url, nil)

    if err != nil {
        fmt.Println(err)
        return
    }
    req.Header.Add("Authorization", "Bearer {API-KEY}")
}
```

HTTP

```
GET /{{endpoint}} HTTP/1.1
Host: {{sport}}.sportdevs.com
Authorization: Bearer {API-KEY}
```

Java

```
Unirest.setTimeouts(0, 0);
HttpResponse<String> response = Unirest.get("https://{{sport}}.sportdevs.com/{{endpoint}}")
    .header("Authorization", "Bearer {API-KEY}")
    .asString();
```

JavaScript - Fetch

```
var myHeaders = new Headers();
myHeaders.append("Authorization", "Bearer {API-KEY}");

var requestOptions = {
    method: 'GET',
    headers: myHeaders,
    redirect: 'follow'
```

```
};

fetch("https://{{sport}}.sportdevs.com/{{endpoint}}", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error))
```

JavaScript - jQuery

```
var settings = {
  "url": "https://{{sport}}.sportdevs.com/{{endpoint}}",
  "method": "GET",
  "timeout": 0,
  "headers": {
    "Authorization": "Bearer {API-KEY}"
  },
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

JavaScript - XHR

```
// WARNING: For GET requests, body is set to null by browsers.

var xhr = new XMLHttpRequest();
xhr.withCredentials = true;

xhr.addEventListener("readystatechange", function() {
  if(this.readyState === 4) {
    console.log(this.responseText);
  }
});

xhr.open("GET", "https://{{sport}}.sportdevs.com/{{endpoint}}");
xhr.setRequestHeader("Authorization", "Bearer {API-KEY}");

xhr.send();
```

C - libcurl

```
CURL *curl;
CURLcode res;
curl = curl_easy_init();
```

```
if(curl) {
    curl_easy_setopt(curl, CURLOPT_CUSTOMREQUEST, "GET");
    curl_easy_setopt(curl, CURLOPT_URL, "https://{{sport}}.sportdevs.com/{{endpoint}}");
    curl_easy_setopt(curl, CURLOPT_FOLLOWLOCATION, 1L);
    curl_easy_setopt(curl, CURLOPT_DEFAULT_PROTOCOL, "https");
    struct curl_slist *headers = NULL;
    headers = curl_slist_append(headers, "Authorization: Bearer {API-KEY}");
    curl_easy_setopt(curl, CURLOPT_HTTPHEADER, headers);
    res = curl_easy_perform(curl);
}
curl_easy_cleanup(curl);
```

NodeJS - Axios

```
var axios = require('axios');

var config = {
  method: 'get',
  url: 'https://{{sport}}.sportdevs.com/{{endpoint}}',
  headers: {
    'Authorization': 'Bearer {API-KEY}'
  }
};

axios(config)
.then(function (response) {
  console.log(JSON.stringify(response.data));
})
.catch(function (error) {
  console.log(error);
});
```

NodeJS - Native

```
var https = require('follow-redirects').https;
var fs = require('fs');

var options = {
  'method': 'GET',
  'hostname': '{{sport}}.sportdevs.com',
  'path': '/{{endpoint}}',
  'headers': {
    'Authorization': 'Bearer {API-KEY}'
  },
  'maxRedirects': 20
};
```

```
var req = https.request(options, function (res) {
  var chunks = [];

  res.on("data", function (chunk) {
    chunks.push(chunk);
  });

  res.on("end", function (chunk) {
    var body = Buffer.concat(chunks);
    console.log(body.toString());
  });

  res.on("error", function (error) {
    console.error(error);
  });
});

req.end();
```

NodeJS - Request

```
var request = require('request');
var options = {
  'method': 'GET',
  'url': 'https://{{sport}}.sportdevs.com/{{endpoint}}',
  'headers': {
    'Authorization': 'Bearer {API-KEY}'
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
  console.log(response.body);
});
```

NodeJS - Unirest

```
var unirest = require('unirest');
var req = unirest('GET', 'https://{{sport}}.sportdevs.com/{{endpoint}}')
  .headers({
    'Authorization': 'Bearer {API-KEY}'
  })
  .end(function (res) {
    if (res.error) throw new Error(res.error);
    console.log(res.raw_body);
});
```

Objective-C - NSURLConnection

```
#import <Foundation/Foundation.h>

dispatch_semaphore_t sema = dispatch_semaphore_create(0);

NSMutableURLRequest *request = [NSMutableURLRequest requestWithURL:[NSURL
URLWithString:@"https://{{sport}}.sportdevs.com/{{endpoint}}"]
cachePolicy:NSURLRequestUseProtocolCachePolicy
timeoutInterval:10.0];
NSDictionary *headers = @{
    @"Authorization": @"Bearer {{API-KEY}}"
};

[request setAllHTTPHeaderFields:headers];

[request setHTTPMethod:@"GET"];

NSURLSession *session = [NSURLSession sharedSession];
NSURLSessionDataTask *dataTask = [session dataTaskWithRequest:request
completionHandler:^(NSData *data, NSURLResponse *response, NSError *error) {
    if (error) {
        NSLog(@"%@", error);
        dispatch_semaphore_signal(sema);
    } else {
        NSHTTPURLResponse *httpResponse = (NSHTTPURLResponse *) response;
        NSError *parseError = nil;
        NSDictionary *responseDictionary = [NSJSONSerialization JSONObjectWithData:data options:0
error:&parseError];
        NSLog(@"%@", responseDictionary);
        dispatch_semaphore_signal(sema);
    }
}];

[dataTask resume];
dispatch_semaphore_wait(sema, DISPATCH_TIME_FOREVER);
```

OCaml - Cohttp

```
open Lwt
open Cohttp
open Cohttp_lwt_unix

let reqBody =
  let uri = Uri.of_string "https://{{sport}}.sportdevs.com/{{endpoint}}" in
  let headers = Header.init () in
  Header.add h "Authorization" "Bearer {{API-KEY}}"
in
```

```
Client.call ~headers `GET uri >>= fun (_resp, body) ->
body |> Cohttp_lwt.Body.to_string >|= fun body -> body
```

```
let () =
let respBody = Lwt_main.run reqBody in
print_endline (respBody)
```

PHP - cURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
    CURLOPT_URL => 'https://{$sport}.sportdevs.com/{$endpoint}',
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_ENCODING => '',
    CURLOPT_MAXREDIRS => 10,
    CURLOPT_TIMEOUT => 0,
    CURLOPT_FOLLOWLOCATION => true,
    CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
    CURLOPT_CUSTOMREQUEST => 'GET',
    CURLOPT_HTTPHEADER => array(
        'Authorization: Bearer {API-KEY}'
    ),
));
$response = curl_exec($curl);

curl_close($curl);
echo $response;
```

PHP - Guzzle

```
<?php
$client = new Client();
$headers = [
    'Authorization' => 'Bearer {API-KEY}'
];
$request = new Request('GET', 'https://{$sport}.sportdevs.com/{$endpoint}', $headers);
$res = $client->sendAsync($request)->wait();
echo $res->getBody();
```

PHP - HTTP_Request2

```
<?php
require_once 'HTTP/Request2.php';
$request = new HTTP_Request2();
$request->setUrl('https://{{sport}}.sportdevs.com/{{endpoint}}');
$request->setMethod(HTTP_Request2::METHOD_GET);
$request->setConfig(array(
    'follow_redirects' => TRUE
));
$request->setHeader(array(
    'Authorization' => 'Bearer {API-KEY}'
));
try {
    $response = $request->send();
    if ($response->getStatus() == 200) {
        echo $response->getBody();
    }
    else {
        echo 'Unexpected HTTP status: ' . $response->getStatus() . ' ' .
            $response->getReasonPhrase();
    }
}
catch(HTTP_Request2_Exception $e) {
    echo 'Error: ' . $e->getMessage();
}
```

PHP - pecl_http

```
<?php
$client = new http\Client;
$request = new http\Client\Request;
$request->setRequestUrl('https://{{sport}}.sportdevs.com/{{endpoint}}');
$request->setRequestMethod('GET');
$request->setOptions(array());
$request->setHeaders(array(
    'Authorization' => 'Bearer {API-KEY}'
));
$client->enqueue($request)->send();
$response = $client->getResponse();
echo $response->getBody();
```

PowerShell - RestMethod

```
$headers = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$headers.Add("Authorization", "Bearer {API-KEY}")

$response = Invoke-RestMethod 'https://{{sport}}.sportdevs.com/{{endpoint}}' -Method 'GET' -Headers
```

```
$headers  
$response | ConvertTo-Json
```

Python - http.client

```
import http.client

conn = http.client.HTTPSConnection("{sport}.sportdevs.com")
payload = ""
headers = {
    'Authorization': 'Bearer {API-KEY}'
}
conn.request("GET", "/{endpoint}", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

Python - Requests

```
import requests

url = "https://{sport}.sportdevs.com/{endpoint}"

payload={}
headers = {
    'Authorization': 'Bearer {API-KEY}'
}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.text)
```

R - httr

```
library(httr)

headers = c(
  'Authorization' = 'Bearer {API-KEY}'
)

res <- VERB("GET", url = "https://{sport}.sportdevs.com/{endpoint}", add_headers(headers))

cat(content(res, 'text'))
```

R - RCurl

```
library(RCurl)
headers = c(
  "Authorization" = "Bearer {API-KEY}"
)
res <- getURL("https://{sport}.sportdevs.com/{endpoint}", .opts=list(httpheader = headers,
followlocation = TRUE))
cat(res)
```

Ruby - Net::HTTP

```
require "uri"
require "net/http"

url = URI("https://{sport}.sportdevs.com/{endpoint}")

https = Net::HTTP.new(url.host, url.port)
https.use_ssl = true

request = Net::HTTP::Get.new(url)
request["Authorization"] = "Bearer {API-KEY}"

response = https.request(request)
puts response.read_body
```

Shell - Httpie

```
http --follow --timeout 3600 GET 'https://{sport}.sportdevs.com/{endpoint}' \
Authorization:'Bearer {API-KEY}'
```

Shell - wget

```
wget --no-check-certificate --quiet \
--method GET \
--timeout=0 \
--header 'Authorization: Bearer {API-KEY}' \
'https://{sport}.sportdevs.com/{endpoint}'
```

Swift - URLSession

```
import Foundation
#if canImport(FoundationNetworking)
import FoundationNetworking
#endif

var semaphore = DispatchSemaphore (value: 0)

var request = URLRequest(url: URL(string: "https://{{sport}}.sportdevs.com/{{endpoint}}")!,timeoutInterval:
Double.infinity)
request.addValue("Bearer {{API-KEY}}", forHTTPHeaderField: "Authorization")

request.httpMethod = "GET"

let task = URLSession.shared.dataTask(with: request) { data, response, error in
    guard let data = data else {
        print(String(describing: error))
        semaphore.signal()
        return
    }
    print(String(data: data, encoding: .utf8)!)
    semaphore.signal()
}

task.resume()
semaphore.wait()
```

WebSockets

WebSockets allow real-time updates by enabling a two-way communication channel. To access WebSockets, ensure you have added the WebSockets add-on to your subscription plan during the subscription setup process.

Available WebSocket Data Types

We offer WebSocket connections for the following data types, each providing specific updates in real-time:

1. Score Updates - ws://football.sportdevs.com/live/score?token={api-key}

```
{  
  "sport": "",  
  "match_id": "",  
  "score": "",  
  "period_1": "",  
  "period_2": "",  
  "overtime": "",  
  "penalties": "",  
  "series": "",  
  "aggregated": "",  
  "extra1": "",  
  "extra2": "",  
  "team": ""  
}
```

2. Commentary Updates - ws://football.sportdevs.com/live/commentary?token={api-key}

```
{  
  "sport": "",  
  "match_id": "",  
  "minute": "",  
  "text": "",  
  "player_id": "",  
  "player_name": "",  
  "player_hash_image": "",  
  "player_in_id": ""  
}
```

```

"player_in_name": "",
"player_in_hash_image": "",
"player_out_id": "",
"player_out_name": "",
"player_out_hash_image": "",
"assist1_id": "",
"assist1_name": "",
"assist1_hash_image": "",
"match_clock": ""
}

```

3. Time Updates - ws://football.sportdevs.com/live/time?token={api-key}

```

{
  "sport": "",
  "match_id": "",
  "specific_start_time": "",
  "period_1": "",
  "period_2": "",
  "period_3": "",
  "period_4": ""
}

```

4. Graph Updates - ws://football.sportdevs.com/live/graph?token={api-key}

```

{
  "sport": "",
  "match_id": "",
  "minute": "",
  "value": ""
}

```

5. Incident Updates - ws://football.sportdevs.com/live/incident?token={api-key}

```

{
  "sport": "",
  "match_id": "",
  "type": "",
  "text": "",
  "home_score": "",
  "away_score": "",
  "is_live": ""
}

```

```

"added_time": "",
"length": "",
"player_id": "",
"player_name": "",
"player_hash_image": "",
"reason": "",
"is_home": "",
"class": "",
"player_in_id": "",
"player_in_hash_image": "",
"player_out_id": "",
"player_out_name": "",
"player_out_hash_image": "",
"assist1_id": "",
"assist1_name": "",
"assist1_hash_image": "",
"assist2_id": "",
"assist2_name": "",
"assist2_hash_image": "",
"reversed_period_time": "",
"reversed_period_time_seconds": ""
}

```

6. Match Status Updates - ws://football.sportdevs.com/live/status?token={api-key}

```

{
  "sport": "",
  "match_id": "",
  "type": "",
  "reason": ""
}

```

Using WebSockets for Real-Time Updates

Example: Subscribing to Updates by Data Type

To receive updates for a specific data type, use the following code:

WebSockets Example:

```

const ws = new WebSocket('wss://football.sportdevs.com/live/{type}?token={api-key}', ['updates']);
ws.addEventListener('message', ev => console.log(ev.data));

```

Server-Sent Events Example:

```
const es = new EventSource('https://football.sportdevs.com/live/{type}?token={api-key}');
es.addEventListener('message', ev => console.log(ev.data));
```

Example: Subscribing to Updates for a Specific Event ID

To track changes for a particular event ID, use the following code:

WebSockets Example:

```
const ws = new WebSocket('wss://football.sportdevs.com/live/{type}/1?token={api-key}', ['updates']);
ws.addEventListener('message', ev => console.log(ev.data));
```

Server-Sent Events Example:

```
const es = new EventSource('https://football.sportdevs.com/live/{type}/1?token={api-key}');
es.addEventListener('message', ev => console.log(ev.data));
```

Additional Information

For more details about working with WebSockets and Server-Sent Events, visit the [MDN Web Docs on EventSource](#).



Logos and Images

All logos and images are completely free to use.

Accessing Logos and Images

To retrieve logos or images, use the following URL format:

```
https://images.sportdevs.com/{hash}.png
```

Replace `{hash}` with the corresponding image hash provided in the API responses.

Available Image Types

SportDevs offers images for the following categories:

-  Classes
-  Arenas
-  Leagues
-  Teams
-  Players
-  Referees
-  Coaches

Feel free to use these assets in your application to enrich the user experience.

Classes

This endpoint allows you to retrieve the classes. You can use the `alpha` parameter to get a specific class as a country.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times per day.

Recommended Calls: Limit calls to 1 per day.

You can get the image of the category by calling the following url:

https://images.sportdevs.com/{hash_image}.png

Use Cases

Get classes

<https://football.sportdevs.com/classes>

Get class based on the `id`

<https://football.sportdevs.com/classes?id=eq.{id}>

Get classes based on the `alpha`

<https://football.sportdevs.com/classes?alpha=eq.{alpha}>

Query Parameters

`id` integer

Default value: `eq.121`

The id of the class

`alpha` character varying

Default value: `eq.NX`

The alpha name of the country

`offset` string

Default value: `0`

Limiting and Pagination

— **limit** string

Default value: 50

Limiting and Pagination

— **lang** string

Default value: en

Multi-language

Responses

200

OK

application/json

Schema

Example (from schema)

▼ Schema array

— **id** integer

The id of the class

Note: This is a Primary Key.

— **name** text

The name of the class

— **flag** text

The flag of the class

— **alpha** character varying

Possible values: <= 2 characters

The alpha name of the country

— **importance** integer

The importance of the class specified with a number

— **hash_image** character varying

Possible values: <= 64 characters

The name of the image

GET /classes

Request

Base URL

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/classes' \
-H 'Accept: application/json'
```

Countries

This endpoint allows you to retrieve the countries. You can use the `alpha` field to get the country.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times per day.

Recommended Calls: Limit calls to 1 per week.

You can get the image of the country by calling the following url:

`https://images.sportdevs.com/{hash_image}.png`

Use Cases

Get countries

`https://football.sportdevs.com/countries`

Get country based on the `id`

`https://football.sportdevs.com/countries?id=eq.{id}`

Get countries based on the `alpha`

`https://football.sportdevs.com/countries?alpha=eq.{alpha}`

▼ Query Parameters

`id` integer

Default value: `eq.1`

The id of the country

`alpha` character varying

Default value: `eq.CA`

The alpha name of the country

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: 50

Limiting and Pagination

lang string

Default value: en

Multi-language

Responses

200

OK

application/json

Schema

Example (from schema)

▼ Schema array

id integer

The id of the country

Note: This is a Primary Key.

name text

The name of the country

alpha character varying

Possible values: <= 2 characters

The alpha name of the country

hash_image character varying

Possible values: <= 64 characters

The name of the image

GET /countries

Request

Base URL

<https://football.sportdevs.com>

+ Show optional parameters

```
curl -L -X GET 'https://football.sportdevs.com/countries' \
-H 'Accept: application/json'
```

Cup Bracket

This endpoint allows you to retrieve the cup knock outs. You can use `id` from the `seasons` or `leagues` endpoint to get the wanted cup knock out tree.

Fields in rounds:

Field	Explanation
<code>order</code>	The order of the round
<code>description</code>	The description of the round
<code>blocks</code>	The blocks of the round
<code>finished</code>	If the match is finished
<code>order</code>	The order of the block
<code>home_team_score</code>	The score of the home team
<code>away_team_score</code>	The score of the away team
<code>has_next_round_link</code>	If the match has next round
<code>match_in_progress</code>	If the match is in play
<code>series_start_date_timestamp</code>	The start date of the match
<code>automatic_progression</code>	If the match has automatic progression
<code>participants</code>	The participants of the block (match)
<code>team_id</code>	The id of the team
<code>winner</code>	If the team is winner of the block

Field	Explanation
order	The order of the participants

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated after every finished match.

Recommended Calls: Limit calls to 1 per finished match.

Use Cases

Get cup knock out based on the `id`

`https://football.sportdevs.com/cup-bracket?id=eq.{id}`

Get cups based on the `season_id`

`https://football.sportdevs.com/cup-bracket?season_id=eq.{season_id}`

Get cups based on the `league_id`

`https://football.sportdevs.com/cup-bracket?league_id=eq.{league_id}`

Query Parameters

`id` integer

Default value: `eq.15`

The id of the cup bracket

`season_id` integer

Default value: `eq.1185`

The id of the season

`league_id` integer

Default value: `eq.106`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

lang string

Default value: `en`

Multi-language

Responses

200

OK

application/json

Schema

Example (from schema)

▼ Schema array

id integer

The id of the cup bracket

Note: This is a Primary Key.

name text

The name of the cup bracket

current_round integer

The last or the current round in the cup bracket

type integer

The type of the cup bracket

season_id integer

The id of the season

Note: This is a Foreign Key to `seasons.id`.

season_name text

The name of the season

league_id integer

The id of the league

Note: This is a Foreign Key to `leagues.id`.

league_name text

The name of the league

league_hash_image character varying

Possible values: `<= 64 characters`

The name image of the league

rounds jsonb

The rounds of the cup bracket

GET /cup-bracket

Request

Base URL

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/cup-bracket' \
-H 'Accept: application/json'
```

Injuries

This endpoint allows you to retrieve the injuries and missing matches.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every minute for live matches and every 20 minutes for ongoing matches.

Recommended Calls: Limit calls to 1 per minute for live matches and every 20 minutes for ongoing matches.

Use Cases

Get the injuries from a specific `player_id`

`https://football.sportdevs.com/injuries?player_id=eq.{player_id}`

Get the injuries from a specific `match_id`

`https://football.sportdevs.com/injuries?match_id=eq.{match_id}`

Get the injuries from a specific `season_id`

`https://football.sportdevs.com/injuries?season_id=eq.{season_id}`

Get the injuries from a specific `tournament_id`

`https://football.sportdevs.com/injuries?tournament_id=eq.{tournament_id}`

Query Parameters

`player_id` integer

Default value: `eq.8364`

The id of the player

`match_id` integer

Default value: `eq.1737`

The id of the match

`season_id` integer

Default value: `eq.67`

The id of the season

tournament_id integer
Default value: eq.13023

The id of the tournament

offset string
Default value: 0

Limiting and Pagination

limit string
Default value: 50

Limiting and Pagination

lang string
Default value: en

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

player_id integer
The id of the player
Note: This is a Foreign Key to [players.id](#).

player_name text
The name of the player

player_hash_image character varying
Possible values: <= 64 characters
The hash image of the player

type text
The type

reason text
The reason

match_id integer
The id of the match
Note: This is a Primary Key.

— **season_id** integer

The id of the season

Note: This is a Foreign Key to `seasons.id`.

— **tournament_id** integer

The id of the tournament

Note: This is a Foreign Key to `seasons-by-tournament.tournament_id`.

— **start_timestamp** timestamp with time zone

The start timestamp of the match

`GET /injuries`

Request

`Base URL`

`https://football.sportdevs.com`

`+ Show optional parameters`

```
curl -L -X GET 'https://football.sportdevs.com/injuries' \
-H 'Accept: application/json'
```

Leagues

This endpoint allows you to retrieve the leagues and the cups.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times a day.

Recommended Calls: Limit calls to 1 every day.

You can get the image of the league by calling the following url:

`https://images.sportdevs.com/{hash_image}.png`

Use Cases

Get leagues

`https://football.sportdevs.com/leagues`

Search the leagues by the `name`

`https://football.sportdevs.com/leagues?name=like.*Championship*`

Get leagues based on the `id`

`https://football.sportdevs.com/leagues?id=eq.{id}`

Get leagues from a specific `class_id`

`https://football.sportdevs.com/leagues?class_id=eq.{class_id}`

Query Parameters

`id` integer

Default value: `eq.1`

The id of the league

`class_id` integer

Default value: `eq.130`

The id of the class

`offset` string

Default value: 0

Limiting and Pagination

limit string

Default value: 50

Limiting and Pagination

lang string

Default value: en

Multi-language

Responses

200

OK

application/json

Schema

Example (from schema)

▼ Schema array

id integer

The id of the league

Note: This is a Primary Key.

name text

The name of the league

importance integer

The importance level of the league

level integer

The level of the league

current_champion_team_id integer

The id of the team that hold the title

current_champion_team_name text

The name of the team that hold the title

current_champion_team_hash_image character varying

Possible values: <= 64 characters

The name image of the team that hold the title

current_champion_team_num_titles integer

The number of the title that the team is holding

teams_most_titles jsonb

The ids of the teams with the most titles
most_titles integer
The number of the most titles accomplished by one team
primary_color character varying
Possible values: <= 7 characters
The primary color of the league
secondary_color character varying
Possible values: <= 7 characters
The secondary color of the league
connected_leagues jsonb
The ids of connected leagues
higher_leagues jsonb
The ids of higher leagues
lower_leagues jsonb
The ids of lower leagues
start_league timestamp with time zone
The date of starting the league
end_league timestamp with time zone
The date of ending the league
hash_image character varying
Possible values: <= 64 characters
The name of the image
class_id integer
The id of the class
Note: This is a Foreign Key to classes.id .
class_name text
The name of the class
class_hash_image character varying
Possible values: <= 64 characters
The name image of the class

GET /leagues

Request

Base URL

<https://football.sportdevs.com>

+ Show optional parameters

```
curl -L -X GET 'https://football.sportdevs.com/leagues' \
-H 'Accept: application/json'
```

Leagues By Class

This endpoint allows you to retrieve the leagues from a specific class.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get leagues from a specific `class_id`

```
https://football.sportdevs.com/leagues-by-category?class_id=eq.{class_id}
```

▼ Query Parameters

`class_id` integer

Default value: `eq.121`

The id of the class

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **class_id** integer

The id of the class

Note: This is a Foreign Key to [classes.id](#).

— **class_name** text

The name of the class

— **class_hash_image** character varying

Possible values: <= 64 characters

The hash image of the class

— **leagues** jsonb

The leagues

[GET](#) /leagues-by-class

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/leagues-by-class' \
-H 'Accept: application/json'
```

Leagues By Date

This endpoint allows you to retrieve the leagues from a specific date. The date has a format of YYYY-DD-MM.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get leagues from a specific date

`https://football.sportdevs.com/leagues-by-date?date=eq.{date}`

Query Parameters

`date` text

Default value: `eq.1930-07-26`

The date

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— date text

The date

— leagues jsonb

The leagues

GET /leagues-by-date

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/leagues-by-date' \
-H 'Accept: application/json'
```

Leagues Info

This endpoint allows you to retrieve the info from leagues and cups.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times a day.

Recommended Calls: Limit calls to 1 every day.

Use Cases

Get league info from a specific `league_id`

```
https://football.sportdevs.com/leagues-info?league_id=eq.{league_id}
```

▼ Query Parameters

`league_id` integer

Default value: `eq.1`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **league_id** integer

The id of the league

Note: This is a Primary Key.

— **promoted_teams** jsonb

The promoted teams

— **most_title_names** jsonb

The teams with most titles

— **number_of_teams** integer

The number of the teams in the league

— **league_grade** text

The grade of the league

— **age_category** text

The age class of the league

— **first_season_winner** text

The first season winner of the league

— **total_teams_count** integer

The count of the teams in the league

— **total_tables_count** integer

The count of the tables in the league

— **promoting_teams_count** text

The count of the promoting teams in the league

— **relegating_teams_count** integer

The count of the relegating teams in the league

— **total_games_count** integer

The count of the games in the league

— **game_frequency** integer

The frequency of the league

— **total_rounds_count** integer

The count of the rounds in the league

— **competition_type** text

The type of the competition

— **league_gender** text

- The gender people playing in the league
- **first_season_start_year** text
First season start year
- **most_popular_teams** jsonb
The name of the most popular teams in the league
- **playoff_teams_count** integer
The number of the teams playing in the playoffs
- **main_competition** text
The main competition
- **games_per_group** text
The number of the games per group
- **playoff_qualifying_teams_count** integer
The count of the teams that are qualified to play in the playoffs
- **number_of_promoted_teams** integer
The number of promoted teams
- **total_number_of_rounds** integer
The number of rounds
- **largest_stadium** jsonb
The largest stadium
- **last_season_magnitude** jsonb
The last season magnitude
- **tv_partners** jsonb
The tv partners of the league
- **official_organization** jsonb
The official organisation of the league
- **promotions** jsonb
The promotions
- **last_season_top_scorers** jsonb
The last season top scorers in the league

GET /leagues-info

Request

Base URL

<https://football.sportdevs.com>

+ Show optional parameters

```
curl -L -X GET 'https://football.sportdevs.com/leagues-info' \
-H 'Accept: application/json'
```

Media Leagues

This endpoint allows you to retrieve the media from leagues.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times a day.

Recommended Calls: Limit calls to 1 per hour.

Use Cases

Get media from a specific `league_id`

```
https://football.sportdevs.com/media-leagues?league_id=eq.{league_id}
```

▼ Query Parameters

`league_id` integer

Default value: `eq.21`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **id** integer

The id of the media

Note: This is a Primary Key.

— **league_id** integer

The id of the league

— **title** text

The title of the media

— **subtitle** text

The subtitle of the media

— **url** text

The url of the media

— **thumbnail_url** text

The thumbnail url of the media

— **date_published** timestamp with time zone

The date of publishing the media

— **channel_url** text

The source url of the media

[GET](#) /media-leagues

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/media-leagues' \
-H 'Accept: application/json'
```


Media Players

This endpoint allows you to retrieve the media from players.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times a day.

Recommended Calls: Limit calls to 1 per hour.

Use Cases

Get media from a specific `player_id`

https://football.sportdevs.com/media-players?player_id=eq.{player_id}

▼ Query Parameters

`player_id` integer

Default value: `eq.328523`

The id of the player

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **id** integer

The id of the media

Note: This is a Primary Key.

— **player_id** integer

The id of the player

— **title** text

The title of the media

— **subtitle** text

The subtitle of the media

— **url** text

The url of the media

— **thumbnail_url** text

The thumbnail url of the media

— **date_published** timestamp with time zone

The date of publishing the media

— **channel_url** text

The source url of the media

[GET](#) /media-players

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/media-players' \
-H 'Accept: application/json'
```


Media Teams

This endpoint allows you to retrieve the media from teams.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated several times a day.

Recommended Calls: Limit calls to 1 per hour.

Use Cases

Get media from a specific `team_id`

`https://football.sportdevs.com/media-teams?team_id=eq.{team_id}`

▼ Query Parameters

`team_id` integer

Default value: `eq.14`

The id of the team

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **id** integer

The id of the media

Note: This is a Primary Key.

— **team_id** integer

The id of the team

— **title** text

The title of the media

— **subtitle** text

The subtitle of the media

— **url** text

The url of the media

— **thumbnail_url** text

The thumbnail url of the media

— **date_published** timestamp with time zone

The date of publishing the media

— **channel_url** text

The source url of the media

[GET](#) /media-teams

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/media-teams' \
-H 'Accept: application/json'
```


Seasons

This endpoint allows you to retrieve the seasons.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get seasons

<https://football.sportdevs.com/seasons>

Get seasons based on the `id`

<https://football.sportdevs.com/seasons?id=eq.{id}>

Get seasons from a specific `league_id`

https://football.sportdevs.com/seasons?league_id=eq.{league_id}

Query Parameters

`id` integer

Default value: `eq.25`

The id of the season

`league_id` integer

Default value: `eq.2`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: en

Multi-language

Responses

200

OK

application/json

Schema**Example (from schema)**

▼ Schema array

id integer

The id of the season

Note: This is a Primary Key.

name text

The name of the season

year text

The year of the season

start_time text

The start time of the season

end_time text

The end time of the season

league_id integer

The id of the league

Note: This is a Foreign Key to **leagues.id**.**league_name** text

The name of the league

league_hash_image text

The name image of the league

GET /seasons

Request

Base URL

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/seasons' \
-H 'Accept: application/json'
```

Seasons By League

This endpoint allows you to retrieve the seasons from a specific league.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get seasons from a specific `league_id`

`https://football.sportdevs.com/seasons-by-league?league_id=eq.{league_id}`

▼ Query Parameters

`league_id` integer

Default value: `eq.1`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

league_id integer

The id of the league

Note: This is a Foreign Key to [leagues.id](#).

league_name text

The name of the league

league_hash_image text

The hash image of the league

seasons jsonb

The seasons

[GET](#) /seasons-by-league

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/seasons-by-league' \
-H 'Accept: application/json'
```

Seasons By Tournament

This endpoint allows you to retrieve the seasons from a specific tournament.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get seasons from a specific `tournament_id`

`https://football.sportdevs.com/seasons-by-tournament?tournament_id=eq.{tournament_id}`

▼ Query Parameters

`tournament_id` integer

Default value: `eq.1`

The id of the tournament

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **tournament_id** integer

The id of the tournament

Note: This is a Primary Key.

— **tournament_name** text

The name of the tournament

— **seasons** jsonb

The seasons

[GET](#) /seasons-by-tournament

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/seasons-by-tournament' \
-H 'Accept: application/json'
```

Seasons Groups

This endpoint allows you to retrieve the groups from seasons.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get groups from a specific `season_id`

`https://football.sportdevs.com/seasons-groups?season_id=eq.{season_id}`

▼ Query Parameters

`season_id` integer

Default value: `eq.13`

The id of the season

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **season_id** integer

The id of the season

Note: This is a Primary Key.

— **season_name** text

The name of the season

— **groups** jsonb

The groups of the season

GET /seasons-groups

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/seasons-groups' \
-H 'Accept: application/json'
```

Seasons Info

This endpoint allows you to retrieve the info from seasons.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get info from a specific `season_id`

`https://football.sportdevs.com/seasons-info?season_id=eq.{season_id}`

▼ Query Parameters

`season_id` integer

Default value: `eq.1`

The id of the season

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **season_id** integer

The id of the season

Note: This is a Primary Key.

— **season_name** text

The name of the season

— **goals** integer

The number of goals by the season

— **home_wins** integer

The number of home team wins by the season

— **away_wins** integer

The number of away team wins by the season

— **draws** integer

The number of draws by the season

— **yellow_cards** integer

The number of yellow cards by the season

— **red_cards** integer

The number of red cards by the season

— **rounds** integer

The number of rounds by the season

— **competitors** integer

The number of competitors by the season

— **total_prize_money** integer

The number of total prize money by the season

— **total_prize_money_currency** text

The currency of the total prize money by the season

— **city** text

The host city of the season

— **countries** string[]

The host countries of the season

— **higher_tournament_teams** jsonb

The number of newcomers higher tournaments by the season

— **lower_tournament_teams** jsonb

The number of newcomers lower tournaments by the season

tournament_other_teams jsonb

The number of newcomers other tournaments by the season

GET /seasons-info

Request

Base URL

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/seasons-info' \
-H 'Accept: application/json'
```

Seasons Rounds

This endpoint allows you to retrieve the rounds from seasons.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get rounds from a specific `season_id`

`https://football.sportdevs.com/seasons-rounds?season_id=eq.{season_id}`

▼ Query Parameters

`season_id` integer

Default value: `eq.1`

The id of the season

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **season_id** integer

The id of the season

Note: This is a Primary Key.

— **season_name** text

The name of the season

— **rounds** jsonb

The rounds of the season

GET /seasons-rounds

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/seasons-rounds' \
-H 'Accept: application/json'
```

Standings

This endpoint allows you to retrieve the standings. For the type argument you can search by: **home**, **away** and **total**.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated after every finished match.

Recommended Calls: Limit calls to 1 per finished match.

Use Cases

Get standings

`https://football.sportdevs.com/standings`

Get standing based on the **id**

`https://football.sportdevs.com/standings?id=eq.{id}`

Get standing from a specific **league_id**

`https://football.sportdevs.com/standings?league_id=eq.{league_id}`

Get standing from a specific **league_id** and **season_id**

`https://football.sportdevs.com/standings?league_id=eq.{league_id}&season_id=eq.{season_id}`

Get standing from a specific **league_id**, **season_id** and **type**

`https://football.sportdevs.com/standings?league_id=eq.{league_id}&season_id=eq.{season_id}&type=eq.{type}`

Query Parameters

id integer

Default value: `eq.1`

The id of the standing

type text

Default value: `eq.total`

The type of the standing

season_id integer

Default value: eq.2

The id of the season

league_id integer

Default value: eq.29

The id of the league

offset string

Default value: 0

Limiting and Pagination

limit string

Default value: 50

Limiting and Pagination

lang string

Default value: en

Multi-language

Responses

200

OK

application/json

Schema

Example (from schema)

▼ Schema array

id integer

The id of the standing

Note: This is a Primary Key.

tournament_id integer

The id of the tournament

Note: This is a Foreign Key to `seasons-by-tournament.tournament_id`.

tournament_name text

The name of the tournament

type text

The type of the standing

name text

The name of the standing

season_id integer

The id of the season

Note: This is a Foreign Key to `seasons.id`.

season_name text

The name of the season

league_id integer

The id of the league

Note: This is a Foreign Key to `leagues.id`.

league_name text

The name of the league

league_hash_image text

The name image of the league

competitors jsonb

The competitors of the standing

`GET /standings`

Request

`Base URL`

<https://football.sportdevs.com>

`+ Show optional parameters`

```
curl -L -X GET 'https://football.sportdevs.com/standings' \
-H 'Accept: application/json'
```

Teams

This endpoint allows you to retrieve the teams.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

You can get the image of the team by calling the following url:

`https://images.sportdevs.com/{hash_image}.png`

Use Cases

Get teams

`https://football.sportdevs.com/teams`

Get team based on the **name**

`https://football.sportdevs.com/teams?name=like.*Real*`

Get team based on the **id**

`https://football.sportdevs.com/teams?id=eq.{id}`

Get teams from a specific **country_id**

`https://football.sportdevs.com/teams?country_id=eq.{country_id}`

Get teams from a specific **category_id**

`https://football.sportdevs.com/teams?category_id=eq.{category_id}`

Get teams from a specific **arena_id**

`https://football.sportdevs.com/teams?arena_id=eq.{arena_id}`

Get teams from a specific **coach_id**

`https://football.sportdevs.com/teams?coach_id=eq.{coach_id}`

Get team from a specific primary_league_id

https://football.sportdevs.com/teams?primary_league_id=eq.{primary_league_id}

Query Parameters

id integer

Default value: eq.384685

The id of the team

country_id integer

Default value: eq.99

The id of the country

class_id integer

Default value: eq.782

The id of the class

tournament_id integer

Default value: eq.91926

The id of the tournament

primary_league_id integer

Default value: eq.22346

The id of the primary league

arena_id integer

The id of the arena

coach_id integer

The id of the coach

offset string

Default value: 0

Limiting and Pagination

limit string

Default value: 50

Limiting and Pagination

lang string

Default value: en

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

- **id** integer
The id of the team
Note: This is a Primary Key.
- **name** text
The name of the team
- **short_name** text
The short name of the team
- **full_name** text
The full name of the team
- **gender** character
Possible values: `<= 1 characters`
The gender of the team
- **name_code** text
The name code of the team
- **national** boolean
If the team is national
- **type** integer
The type of the team
- **color_primary** character varying
Possible values: `<= 7 characters`
The primary color of the team
- **color_secondary** character varying
Possible values: `<= 7 characters`
The secondary color of the team
- **color_text** character varying
Possible values: `<= 7 characters`
The text color of the team
- **foundation_date** timestamp with time zone
The foundation date of the team
- **hash_image** character varying
Possible values: `<= 64 characters`
The name of the image
- **country_id** integer
The id of the country

Note: This is a Foreign Key to [countries.id](#).

country_name text

The name of the country

country_hash_image character varying

Possible values: [=< 64 characters](#)

The name image of the country

class_id integer

The id of the class

Note: This is a Foreign Key to [classes.id](#).

class_name text

The name of the class

class_hash_image character varying

Possible values: [=< 64 characters](#)

The name image of the class

tournament_id integer

The id of the tournament

Note: This is a Foreign Key to [seasons-by-tournament.tournament_id](#).

tournament_name text

The name of the tournament

primary_league_id integer

The id of the primary league

Note: This is a Foreign Key to [leagues.id](#).

primary_league_name text

The name of the primary league

primary_league_hash_image character varying

Possible values: [=< 64 characters](#)

The name image of the primary league

arena_id integer

The id of the arena

Note: This is a Foreign Key to [arenas.id](#).

arena_name text

The name of the arena

arena_hash_image character varying

Possible values: [=< 64 characters](#)

The name image of the arena

coach_id integer

The id of the coach

Note: This is a Foreign Key to [coaches.id](#).

coach_name text

The name of the coach

coach_hash_image character varying

Possible values: <= 64 characters

The name image of the coach

parent_team_id integer

The id of the parent team

Note: This is a Foreign Key to [teams.id](#).

parent_team_name text

The name of the parent team

parent_team_hash_image character varying

Possible values: <= 64 characters

The name image of the parent team

GET /teams

Request

Base URL

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/teams' \
-H 'Accept: application/json'
```

Teams By League

This endpoint allows you to retrieve the teams from a specific league.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get teams from a specific `league_id`

```
https://football.sportdevs.com/teams-by-league?league_id=eq.{league_id}
```

▼ Query Parameters

`league_id` integer

Default value: `eq.1`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

league_id integer

The id of the league

Note: This is a Foreign Key to [leagues.id](#).

league_name text

The name of the league

league_hash_image character varying

Possible values: `<= 64 characters`

The hash image of the league

teams jsonb

The teams

`GET /teams-by-league`

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/teams-by-league' \
-H 'Accept: application/json'
```

Teams By Season

This endpoint allows you to retrieve the teams from a specific season.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get teams from a specific `season_id`

`https://football.sportdevs.com/teams-by-season?season_id=eq.{season_id}`

▼ Query Parameters

`season_id` integer

Default value: `eq.1`

The id of the season

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **season_id** integer

The id of the season

Note: This is a Primary Key.

— **season_name** text

The name of the season

— **teams** jsonb

The teams

[GET](#) /teams-by-season

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/teams-by-season' \
-H 'Accept: application/json'
```

Teams By Tournament

This endpoint allows you to retrieve the teams from a specific tournament.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get teams from a specific `tournament_id`

`https://football.sportdevs.com/teams-by-tournament?tournament_id=eq.{tournament_id}`

▼ Query Parameters

`tournament_id` integer

Default value: `eq.1`

The id of the tournament

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **tournament_id** integer

The id of the tournament

Note: This is a Primary Key.

— **tournament_name** text

The name of the tournament

— **teams** jsonb

The teams

[GET](#) /teams-by-tournament

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/teams-by-tournament' \
-H 'Accept: application/json'
```

Teams Seasons

This endpoint allows you to retrieve the seasons from teams.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get seasons from a specific `team_id`

https://football.sportdevs.com/teams-seasons?team_id=eq.{team_id}

▼ Query Parameters

`team_id` integer

Default value: `eq.1`

The id of the team

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

team_id integer

The id of the team

Note: This is a Primary Key.

team_name text

The name of the team

team_hash_image character varying

Possible values: <= 64 characters

The hash image of the team

tournaments jsonb

The tournaments of the team playing

GET /teams-seasons

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/teams-seasons' \
-H 'Accept: application/json'
```

Tournaments

This endpoint allows you to retrieve the tournaments.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get tournaments

<https://football.sportdevs.com/tournaments>

Get tournaments based on the `id`

<https://football.sportdevs.com/tournaments?id=eq.{id}>

Get tournaments from a specific `league_id`

https://football.sportdevs.com/tournaments?league_id=eq.{league_id}

Get tournaments from a specific `class_id`

https://football.sportdevs.com/tournaments?class_id=eq.{class_id}

Query Parameters

`id` integer

Default value: `eq.1`

The id of the tournament

`league_id` integer

Default value: `eq.1`

The id of the league

`class_id` integer

Default value: `eq.130`

The id of the class

`offset` string

Default value: 0

Limiting and Pagination

limit string

Default value: 50

Limiting and Pagination

lang string

Default value: en

Multi-language

Responses

200

OK

application/json

Schema

Example (from schema)

▼ Schema array

id integer

The id of the tournament

Note: This is a Primary Key.

name text

The name of the tournament

importance integer

The importance level of the tournament

league_id integer

The id of the league

Note: This is a Foreign Key to [leagues.id](#).

league_name text

The name of the league

league_hash_image character varying

Possible values: <= 64 characters

The name image of the league

class_id integer

The id of the class

Note: This is a Foreign Key to [classes.id](#).

class_name text

The name of the class

class_hash_image character varying

Possible values: <= 64 characters

The name image of the class

GET /tournaments

Request

Base URL

<https://football.sportdevs.com>

+ Show optional parameters

```
curl -L -X GET 'https://football.sportdevs.com/tournaments' \
-H 'Accept: application/json'
```

Tournaments By Class

This endpoint allows you to retrieve the tournaments from a specific class.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get tournaments from a specific `class_id`

`https://football.sportdevs.com/tournaments-by-class?class_id=eq.{class_id}`

▼ Query Parameters

`class_id` integer

Default value: `eq.121`

The id of the class

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

— **class_id** integer

The id of the class

Note: This is a Foreign Key to `classes.id`.

— **class_name** text

The name of the class

— **class_hash_image** character varying

Possible values: <= 64 characters

The hash image of the class

— **tournaments** jsonb

The tournaments

GET /tournaments-by-class

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/tournaments-by-class' \
-H 'Accept: application/json'
```

Tournaments By League

This endpoint allows you to retrieve the tournaments from a specific league.

Multi-language: This feature allows the user to manually change the language of the API response by adding value to the `lang` attribute. Example: `lang=en`.

Pagination: The endpoint uses an offset and limit system. Specify the offset and limit parameters to fetch the desired range of results. The maximum and default number of results per request is 50.

Update Period: The endpoint is updated every day.

Recommended Calls: Limit calls to 1 per day.

Use Cases

Get tournaments from a specific `league_id`

```
https://football.sportdevs.com/tournaments-by-league?league_id=eq.{league_id}
```

▼ Query Parameters

`league_id` integer

Default value: `eq.1`

The id of the league

`offset` string

Default value: `0`

Limiting and Pagination

`limit` string

Default value: `50`

Limiting and Pagination

`lang` string

Default value: `en`

Multi-language

Responses

200

OK

application/json

[Schema](#)[Example \(from schema\)](#)

▼ Schema array

league_id integer

The id of the league

Note: This is a Foreign Key to [leagues.id](#).

league_name text

The name of the league

league_hash_image character varying

Possible values: `<= 64 characters`

The hash image of the league

tournaments jsonb

The tournaments

[GET](#) /tournaments-by-league

Request

[Base URL](#)

<https://football.sportdevs.com>

[+ Show optional parameters](#)

```
curl -L -X GET 'https://football.sportdevs.com/tournaments-by-league' \
-H 'Accept: application/json'
```