



College Basketball Data API 1.22.4 OAS 3.0

This is an API for query various college basketball datasets and analytics. API keys can be acquired by registering on the CollegeBasketballData.com website.

[Contact the developer](#)

MIT

Servers

<https://api.collegebasketballdata.com/>

[Authorize](#)



games Games scores and statistics



[GET /games](#)



Returns information on the first 3000 games that match the provided filters, ordered by start date.

Parameters

[Try it out](#)

Name	Description
startDateRange <small>string(\$date-time) (query)</small>	Optional start timestamp in ISO 8601 format <input type="text" value="startDateRange"/>
endDateRange <small>string(\$date-time) (query)</small>	Optional end timestamp in ISO 8601 format <input type="text" value="endDateRange"/>
team <small>string (query)</small>	Optional team name filter <input type="text" value="team"/>
conference <small>string (query)</small>	Optional conference abbreviation filter <input type="text" value="conference"/>
season <small>integer(\$int32) (query)</small>	Optional season filter <input type="text" value="season"/>
seasonType <small>string (query)</small>	Optional season type filter <i>Available values : postseason, preseason, regular</i> <input type="text" value="--"/>
status <small>string (query)</small>	Optional game status filter <i>Available values : scheduled, in_progress, final, postponed, cancelled</i> <input type="text" value="--"/>
tournament <small>string (query)</small>	Optional tournament filter (e.g. NCAA, NIT, etc) <input type="text" value="tournament"/>

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": 0, "sourceId": "string", "seasonLabel": "string", "season": 0, "seasonType": "postseason", "startDate": "2026-01-25T03:53:24.578Z", "startTimeTbd": true, "neutralSite": true, "conferenceGame": true, "gameType": "string", "tournament": "string", "gameNotes": "string", "status": "scheduled", "attendance": 0, "homeTeamId": 0, "homeTeam": "string", "homeConferenceId": 0, "homeConference": "string", "homeSeed": 0, "homePoints": 0, "homePeriodPoints": [0] }]</pre>	<i>No links</i>



GET /games/media

Returns broadcast information on the first 3000 games that match the provided filters, ordered by start date.

Parameters

[Try it out](#)

Name	Description
startDateRange <small>string(\$date-time) (query)</small>	Optional start timestamp in ISO 8601 format <input type="text" value="startDateRange"/>
endDateRange <small>string(\$date-time) (query)</small>	Optional end timestamp in ISO 8601 format <input type="text" value="endDateRange"/>
team <small>string (query)</small>	Optional team name filter <input type="text" value="team"/>
conference <small>string (query)</small>	Optional conference abbreviation filter <input type="text" value="conference"/>
season <small>number(\$double) (query)</small>	Optional season filter <input type="text" value="season"/>
seasonType <small>string (query)</small>	Optional season type filter <i>Available values : postseason, preseason, regular</i> <input type="text" value="--"/>
tournament	Optional tournament filter (e.g. NCAA, NIT, etc)

Name	Description
string (query)	tournament

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"id": 1, "gameId": 0, "season": 0, "seasonLabel": "string", "seasonType": "postseason", "tournament": "string", "startDate": "2026-01-25T03:53:24.528Z", "startTimeTbd": true, "homeTeamId": 0, "homeTeam": "string", "homeConference": "string", "awayTeamId": 0, "awayTeam": "string", "awayConference": "string", "neutralSite": true, "conferenceGame": true, "gameType": "string", "notes": "string", "broadcasts": [{"id": 1, "name": "TV", "order": 1}, {"id": 2, "name": "Radio", "order": 2}, {"id": 3, "name": "Online", "order": 3}]}]</pre>	No links

Code

Description

```
{  
    "broadcastName": "string",  
    "broadcastType": "string"  
}
```

GET /games/teams



Returns team box score statistics and metrics on the first 3000 games that match the provided filters, ordered by start date.

Parameters

[Try it out](#)

Name

Description

startDateRange Optional start timestamp in ISO 8601 format

`string($date-time)
(query)`

endDateRange Optional end timestamp in ISO 8601 format

`string($date-time)
(query)`

team Optional team name filter

`string
(query)`

conference Optional conference abbreviation filter

`string
(query)`

season Optional season filter

Name	Description
number(\$double) (query)	season
seasonType string (query)	Optional season type filter <i>Available values : postseason, preseason, regular</i> --
tournament string (query)	Optional tournament filter (e.g. NCAA, NIT, etc) tournament

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "gameId": 0, "season": 0, "seasonLabel": "string", "seasonType": "postseason", "tournament": "string", "startDate": "2026-01-25T03:53:24.530Z", "startTimeTbd": true, "teamId": 0, "teamLabel": "string" }]</pre>	No links

[Links](#)**Code**

```
"team": "string",
"conference": "string",
"teamSeed": 0,
"opponentId": 0,
"opponent": "string",
"opponentConference": "string",
"opponentSeed": 0,
"neutralSite": true,
"isHome": true,
"conferenceGame": true,
"gameType": "string",
"notes": "string",
"gameMinutes": 0,
"pace": 0,
```

GET /games/players

Returns player box score statistics and metrics on the first 1000 games that match the provided filters, ordered by start date.

Parameters[Try it out](#)**Name****Description**

startDateRange Optional start timestamp in ISO 8601 format

string(\$date-time)
(query)

startDateRange

endDateRange Optional end timestamp in ISO 8601 format

string(\$date-time)
(query)

endDateRange

team Optional team name filter

Name	Description
string (query)	team
conference	Optional conference abbreviation filter
string (query)	conference
season	Optional season filter
number(\$double) (query)	season
seasonType	Optional season type filter
string (query)	<i>Available values : postseason, preseason, regular</i> --
tournament	Optional tournament filter (e.g. NCAA, NIT, etc)
string (query)	tournament

Responses

Code	Description	Links
200	Ok	No links

Media type

application/json

Controls Accept header.

Example Value Schema

Code	Description	Links
	<pre>[{ "gameId": 0, "season": 0, "seasonLabel": "string", "seasonType": "postseason", "tournament": "string", "startDate": "2026-01-25T03:53:24.535Z", "startTimeTbd": true, "teamId": 0, "team": "string", "conference": "string", "teamSeed": 0, "opponentId": 0, "opponent": "string", "opponentConference": "string", "opponentSeed": 0, "neutralSite": true, "isHome": true, "conferenceGame": true, "gameType": "string", "notes": "string", "gameMinutes": 0, "gamePace": 0, "nPlayers": 0 }]</pre>	

plays Play by play data



GET /substitutions/game/{gameId}



Returns all player substitutions for a given game

Parameters

Try it out

Name	Description
gameId * required integer(\$int32) (path)	Game id filter <input type="text" value="gameId"/>

Responses

Code	Description	Links
200	Ok Media type application/json Controls Accept header. Example Value Schema <pre>[{ "gameId": 0, "startDate": "2026-01-25T03:53:24.537Z", "teamId": 0, "team": "string", "conference": "string", "athleteId": 0, "athlete": "string", "position": "string", "opponentId": 0, "opponent": "string", "opponentConference": "string", "subIn": { "opponentPoints": 0, "teamPoints": 0, "secondsRemaining": 0, } }]</pre>	<i>No links</i>

[Links](#)**Code**

```
        "period": 0
    },
    "subOut": {
        "opponentPoints": 0,
        "teamPoints": 0,
        "secondsRemaining": 0,
        "period": 0
    }
}
```

GET /substitutions/player/{playerId}

Retrieve all player substitutions for a given player and season

Parameters[Try it out](#)**Name** **Description****playerId** * required
Required player id filterinteger(\$int32)
(path)**season** * required
Required season filterinteger(\$int32)
(query)**Responses**

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"gameId": 0, "startDate": "2026-01-25T03:53:24.538Z", "teamId": 0, "team": "string", "conference": "string", "athleteId": 0, "athlete": "string", "position": "string", "opponentId": 0, "opponent": "string", "opponentConference": "string", "subIn": {"opponentPoints": 0, "teamPoints": 0, "secondsRemaining": 0, "period": 0}, "subOut": {"opponentPoints": 0, "teamPoints": 0, "secondsRemaining": 0, "period": 0}}</pre>	<i>No links</i>

GET /substitutions/team



Retrieve all player substitutions for a given team and season

Parameters

[Try it out](#)

Name	Description
------	-------------

season * required Required season filter

integer(\$int32)
(query)

team * required Required team filter

string
(query)

Responses

Code	Description	Links
------	-------------	-------

200 Ok

No links

Media type

Controls Accept header.

Example Value Schema

```
[  
  {  
    "gameId": 0,  
    "startDate": "2026-01-25T03:53:24.539Z",  
    "substitution": {  
      "playerId": 1,  
      "playerName": "John Doe",  
      "position": "Guard",  
      "status": "Active"  
    }  
  }  
]
```

[Links](#)**Code**

```

"teamId": 0,
"team": "string",
"conference": "string",
"athleteId": 0,
"athlete": "string",
"position": "string",
"opponentId": 0,
"opponent": "string",
"opponentConference": "string",
"subIn": {
    "opponentPoints": 0,
    "teamPoints": 0,
    "secondsRemaining": 0,
    "period": 0
},
"subOut": {
    "opponentPoints": 0,
    "teamPoints": 0,
    "secondsRemaining": 0,
    "period": 0
}

```

GET /plays/game/{gameId}



Returns all plays for a given game

Parameters[Try it out](#)

Name	Description
gameId * required	Game id filter
integer(\$int32) (path)	<input type="text" value="gameId"/>
shootingPlaysOnly	Optional filter to only return shooting plays
boolean	

Name	Description	
(query)	--	
<h3>Responses</h3>		
Code	Description	Links
200	Ok	<i>No links</i>
<p>Media type</p>		
application/json		
<p>Controls Accept header.</p>		
<p>Example Value Schema</p>		
<pre>[{ "id": 0, "sourceId": "string", "gameId": 0, "gameSourceId": "string", "gameStartDate": "2026-01-25T03:53:24.540Z", "season": 0, "seasonType": "postseason", "gameType": "string", "tournament": "string", "playType": "string", "isHomeTeam": true, "teamId": 0, "team": "string", "conference": "string", "teamSeed": 0, "opponentId": 0, "opponent": "string", "opponentConference": "string", "opponentSeed": 0, "oppo</pre>		

[Links](#)**Code**

```
"period": 0,  
"clock": "string",  
"secondsRemaining": 0,
```

GET /plays/player/{playerId}

Retrieve all plays for a given player and season

Parameters[Try it out](#)**Name** **Description****playerId** * requiredinteger(\$int32)
(path)

Required player id filter

season * requiredinteger(\$int32)
(query)

Required season filter

shootingPlaysOnlyboolean
(query)

Optional filter to only return shooting plays

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"id": 0, "sourceId": "string", "gameId": 0, "gameSourceId": "string", "gameStartDate": "2026-01-25T03:53:24.542Z", "season": 0, "seasonType": "postseason", "gameType": "string", "tournament": "string", "playType": "string", "isHomeTeam": true, "teamId": 0, "team": "string", "conference": "string", "teamSeed": 0, "opponentId": 0, "opponent": "string", "opponentConference": "string", "opponentSeed": 0, "period": 0, "clock": "string", "secondsRemaining": 0, "homeScore": 0}</pre>	<i>No links</i>

Retrieve all plays for a given team and season

Parameters

[Try it out](#)

Name	Description
------	-------------

season * required Required season filter

integer(\$int32)
(query)

team * required Required team filter

string
(query)

shootingPlaysOnly Optional filter to only return shooting plays

boolean
(query)

Responses

Code	Description	Links
------	-------------	-------

200 Ok

No links

Media type

Controls Accept header.

Example Value Schema

[Links](#)[Code](#)

```
[
  {
    "id": 0,
    "sourceId": "string",
    "gameId": 0,
    "gameSourceId": "string",
    "gameStartDate": "2026-01-25T03:53:24.544Z",
    "season": 0,
    "seasonType": "postseason",
    "gameType": "string",
    "tournament": "string",
    "playType": "string",
    "isHomeTeam": true,
    "teamId": 0,
    "team": "string",
    "conference": "string",
    "teamSeed": 0,
    "opponentId": 0,
    "opponent": "string",
    "opponentConference": "string",
    "opponentSeed": 0,
    "period": 0,
    "clock": "string",
    "secondsRemaining": 0,
    "homeScore": 0
  }
]
```

GET /plays/date



Retrieve all plays for a given UTC date

[Parameters](#)[Try it out](#)

Name	Description
date * required	Required date filter in ISO 8601 format (YYYY-MM-DD)
string(\$date-time)	

date * required	Required date filter in ISO 8601 format (YYYY-MM-DD)
string(\$date-time)	

Name	Description
(query)	date
shootingPlaysOnly	Optional filter to only return shooting plays
boolean	--
(query)	
utcOffset	Optional UTC offset in hours to adjust the date range
number(\$double)	
(query)	utcOffset

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": 0, "sourceId": "string", "gameId": 0, "gameSourceId": "string", "gameStartDate": "2026-01-25T03:53:24.546Z", "season": 0, "seasonType": "postseason", "gameType": "string", "status": "string" }]</pre>	No links

Code	Description	Links
	<pre>"tournament": "string", "playType": "string", "isHomeTeam": true, "teamId": 0, "team": "string", "conference": "string", "teamSeed": 0, "opponentId": 0, "opponent": "string", "opponentConference": "string", "opponentSeed": 0, "period": 0, "clock": "string", "secondsRemaining": 0.</pre>	

GET /plays/tournament 🔒 ⌂

Retrieve all plays for a given tournament and season

Parameters **Try it out**

Name	Description
tournament * required string (query)	Required tournament filter (e.g. NCAA, NIT, etc) <input type="text" value="tournament"/>
season * required number(\$double) (query)	Required season filter <input type="text" value="season"/>

Name	Description
shootingPlaysOnly boolean (query)	Optional filter to only return shooting plays --

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": 0, "sourceId": "string", "gameId": 0, "gameSourceId": "string", "gameStartDate": "2026-01-25T03:53:24.548Z", "season": 0, "seasonType": "postseason", "gameType": "string", "tournament": "string", "playType": "string", "isHomeTeam": true, "teamId": 0, "team": "string", "conference": "string", "teamSeed": 0, "opponentId": 0, "oppo }]</pre>	No links

[Links](#)**Code****Description**

```
"opponent": "string",
"opponentConference": "string",
"opponentSeed": 0,
"period": 0,
"clock": "string",
"secondsRemaining": 0,
```

GET**/plays/types**

Retrieve list of play types

Parameters[Try it out](#)

No parameters

Responses**Code****Description**[Links](#)

200

Ok

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
[  
  {  
    "id": 0,  
    "playType": "string",  
    "clock": "string",  
    "secondsRemaining": 0,  
    "opponent": "string",  
    "opponentConference": "string",  
    "opponentSeed": 0,  
    "period": 0  
  }  
]
```

[Links](#)**Code**

```
        "name": "string"  
    }  
]
```

teams Team information

**GET** /teams

Retrieves historical team information

Parameters

[Try it out](#)

Name	Description
------	-------------

conference	Optional conference filter
------------	----------------------------

string (query)	<input type="text" value="conference"/>
--------------------------	---

season	Optional season filter
--------	------------------------

integer(\$int32) (query)	<input type="text" value="season"/>
------------------------------------	-------------------------------------

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"id": 0, "sourceId": "string", "school": "string", "mascot": "string", "abbreviation": "string", "displayName": "string", "shortDisplayName": "string", "primaryColor": "string", "secondaryColor": "string", "currentVenueId": 0, "currentVenue": "string", "currentCity": "string", "currentState": "string", "conferenceId": 0, "conference": "string"}]</pre>	<i>No links</i>

GET /teams/roster



Retrieves team roster information

Parameters

Try it out

Name	Description
season * required integer(\$int32) (query)	Season filter <input type="text" value="season"/>
team string (query)	Optional team filter <input type="text" value="team"/>

Responses

Code	Description	Links
200	Ok Media type <input type="text" value="application/json"/> Controls Accept header. Example Value Schema <pre>[{ "teamId": 0, "teamSourceId": "string", "team": "string", "conference": "string", "season": 0, "players": [{ "id": 0, "sourceId": "string", "name": "string", "team": "string" }] }]</pre>	No links

Code	Description	Links
	<pre>"firstName": "string", "lastName": "string", "jersey": "string", "position": "string", "height": 0, "weight": 0, "hometown": { "countyFips": "string", "longitude": 0, "latitude": 0, "country": "string", "state": "string", "city": "string" }</pre>	

conferences Conference information



GET	/conferences	
Retrieves list of available conferences		
Parameters		
No parameters		Try it out
Responses		
Code	Description	Links
200	Ok	<i>No links</i>

[Links](#)[Code](#)[Description](#)

Media type

application/json

Controls Accept header.

[Example Value](#)[Schema](#)

```
[  
  {  
    "id": 0,  
    "sourceId": "string",  
    "name": "string",  
    "abbreviation": "string",  
    "shortName": "string"  
  }  
]
```

[GET](#)[/conferences/history](#)

Retrieves historical conference membership information

[Parameters](#)[Try it out](#)

Name	Description
------	-------------

conference	Optional conference abbreviation filter
------------	---

string <i>(query)</i>	
---------------------------------	--

conference

[Responses](#)

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"id": 0, "sourceId": "string", "name": "string", "abbreviation": "string", "shortName": "string", "teams": [{"endSeason": 0, "startSeason": 0, "mascot": "string", "school": "string", "sourceId": "string", "id": 0}]}]</pre>	<i>No links</i>

venues Information about venues



GET /venues



Retrieves list of available venues

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	Ok	<i>No links</i>

Media type

application/json

Controls Accept header.

Example Value Schema

```
[  
  {  
    "id": 0,  
    "sourceId": "string",  
    "name": "string",  
    "city": "string",  
    "state": "string",  
    "country": "string"  
  }  
]
```

stats Statistical data



GET /stats/team/season



Returns team season statistics by year or team

Parameters

[Try it out](#)

Name	Description
season <small>number(\$double) (query)</small>	Optional season filter, required if team is not provided <input type="text" value="season"/>
seasonType <small>string (query)</small>	Optional season type filter <i>Available values : postseason, preseason, regular</i> <input type="text" value="--"/>
team <small>string (query)</small>	Optional team name filter, required if season is not provided <input type="text" value="team"/>
conference <small>string (query)</small>	Optional conference abbreviation filter <input type="text" value="conference"/>
startDateRange <small>string(\$date-time) (query)</small>	<input type="text" value="startDateRange"/>

Name	Description
endDateRange string(\$date-time) (query)	endDateRange

Responses

Code	Description	Links
200	Ok	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
[{"id": 1, "team": "Team A", "conference": "Big Ten", "totalMinutes": 100, "pace": 80, "wins": 10, "losses": 5, "teamStats": {"fieldGoals": {"pct": 0.45, "attempted": 100, "made": 45}, "threePointers": {"pct": 0.35, "attempted": 50, "made": 18}, "freeThrows": {"pct": 0.85, "attempted": 30, "made": 26}}}, {"id": 2, "team": "Team B", "conference": "SEC", "totalMinutes": 100, "pace": 80, "wins": 12, "losses": 8, "teamStats": {"fieldGoals": {"pct": 0.48, "attempted": 110, "made": 52}, "threePointers": {"pct": 0.38, "attempted": 60, "made": 23}, "freeThrows": {"pct": 0.88, "attempted": 35, "made": 31}}}, {"id": 3, "team": "Team C", "conference": "ACC", "totalMinutes": 100, "pace": 80, "wins": 11, "losses": 7, "teamStats": {"fieldGoals": {"pct": 0.47, "attempted": 105, "made": 50}, "threePointers": {"pct": 0.37, "attempted": 55, "made": 21}, "freeThrows": {"pct": 0.87, "attempted": 32, "made": 28}}}, {"id": 4, "team": "Team D", "conference": "Big 12", "totalMinutes": 100, "pace": 80, "wins": 9, "losses": 9, "teamStats": {"fieldGoals": {"pct": 0.44, "attempted": 100, "made": 44}, "threePointers": {"pct": 0.34, "attempted": 50, "made": 17}, "freeThrows": {"pct": 0.84, "attempted": 30, "made": 25}}}, {"id": 5, "team": "Team E", "conference": "Pac-12", "totalMinutes": 100, "pace": 80, "wins": 8, "losses": 10, "teamStats": {"fieldGoals": {"pct": 0.43, "attempted": 95, "made": 42}, "threePointers": {"pct": 0.33, "attempted": 45, "made": 15}, "freeThrows": {"pct": 0.83, "attempted": 28, "made": 23}}}, {"id": 6, "team": "Team F", "conference": "Midwest", "totalMinutes": 100, "pace": 80, "wins": 7, "losses": 11, "teamStats": {"fieldGoals": {"pct": 0.42, "attempted": 90, "made": 38}, "threePointers": {"pct": 0.32, "attempted": 40, "made": 13}, "freeThrows": {"pct": 0.82, "attempted": 25, "made": 20}}}, {"id": 7, "team": "Team G", "conference": "Mountain West", "totalMinutes": 100, "pace": 80, "wins": 6, "losses": 12, "teamStats": {"fieldGoals": {"pct": 0.41, "attempted": 85, "made": 35}, "threePointers": {"pct": 0.31, "attempted": 35, "made": 11}, "freeThrows": {"pct": 0.81, "attempted": 22, "made": 18}}}, {"id": 8, "team": "Team H", "conference": "American", "totalMinutes": 100, "pace": 80, "wins": 5, "losses": 13, "teamStats": {"fieldGoals": {"pct": 0.4, "attempted": 80, "made": 32}, "threePointers": {"pct": 0.3, "attempted": 30, "made": 9}, "freeThrows": {"pct": 0.8, "attempted": 20, "made": 16}}}, {"id": 9, "team": "Team I", "conference": "Colonial", "totalMinutes": 100, "pace": 80, "wins": 4, "losses": 14, "teamStats": {"fieldGoals": {"pct": 0.39, "attempted": 75, "made": 29}, "threePointers": {"pct": 0.29, "attempted": 25, "made": 8}, "freeThrows": {"pct": 0.79, "attempted": 18, "made": 14}}}, {"id": 10, "team": "Team J", "conference": "Horizon", "totalMinutes": 100, "pace": 80, "wins": 3, "losses": 15, "teamStats": {"fieldGoals": {"pct": 0.38, "attempted": 70, "made": 26}, "threePointers": {"pct": 0.28, "attempted": 20, "made": 5}, "freeThrows": {"pct": 0.78, "attempted": 15, "made": 12}}}, {"id": 11, "team": "Team K", "conference": "SoCon", "totalMinutes": 100, "pace": 80, "wins": 2, "losses": 16, "teamStats": {"fieldGoals": {"pct": 0.37, "attempted": 65, "made": 23}, "threePointers": {"pct": 0.27, "attempted": 15, "made": 4}, "freeThrows": {"pct": 0.77, "attempted": 10, "made": 8}}}, {"id": 12, "team": "Team L", "conference": "AAC", "totalMinutes": 100, "pace": 80, "wins": 1, "losses": 17, "teamStats": {"fieldGoals": {"pct": 0.36, "attempted": 60, "made": 21}, "threePointers": {"pct": 0.26, "attempted": 10, "made": 3}, "freeThrows": {"pct": 0.76, "attempted": 5, "made": 4}}}, {"id": 13, "team": "Team M", "conference": "CUSA", "totalMinutes": 100, "pace": 80, "wins": 0, "losses": 18, "teamStats": {"fieldGoals": {"pct": 0.35, "attempted": 55, "made": 19}, "threePointers": {"pct": 0.25, "attempted": 5, "made": 1}, "freeThrows": {"pct": 0.75, "attempted": 3, "made": 2}}}]
```

[Links](#)[Code](#)[Description](#)

```
},
  "twoPointFieldGoals": {
    "pct": 0,
    "attempted": 0,
    "made": 0
  },
  "threePointFieldGoals": {
```

GET [/stats/team/shooting/season](#)



Retrieves team season shooting statistics

Parameters

[Try it out](#)

Name	Description
season * required	Required season filter
integer(\$int32) (query)	<input type="text" value="season"/>
seasonType	Optional season type filter
string (query)	<input type="text" value="Available values : postseason, preseason, regular"/> --
team	Team filter, required if conference is not provided
string (query)	<input type="text" value="team"/>
conference	Conference abbreviation filter, required if team is not provided
string (query)	<input type="text" value="conference"/>
startDateRange	Optional start date range filter

Name	Description
<code>string(\$date-time) (query)</code>	<code>startDateRange</code>
<code>endDateRange</code> <code>string(\$date-time) (query)</code>	Optional end date range filter <code>endDateRange</code>

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p><code>application/json</code></p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "season": 0, "teamId": 0, "team": "string", "conference": "string", "trackedShots": 0, "assistedPct": 0, "freeThrowRate": 0, "dunks": { "made": 0, "attempted": 0, "pct": 0, "order": 0 } }]</pre>	<i>No links</i>

[Links](#)**Code**

```

        "assistedPct": 0,
        "assisted": 0
    },
    "layups": {
        "made": 0,
        "attempted": 0,
        "pct": 0,
        "assistedPct": 0,
        "assisted": 0
    },
    "tipIns": {

```

GET /stats/player/season



Returns player statistics by season

Parameters[Try it out](#)**Name****Description****season** * required

Required season filter

number(\$double)
(query)

season

seasonType

Optional season type filter

string
(query)

Available values : postseason, preseason, regular

--

team

Optional team name filter

string
(query)

team

conference

Optional conference abbreviation filter

Name	Description
string (query)	conference
startDateRange string(\$date- time) (query)	startDateRange
endDateRange string(\$date- time) (query)	endDateRange

Responses

Code	Description	Links
200	Ok	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
[  
  {  
    "season": 0,  
    "seasonLabel": "string",  
    "teamId": 0,  
    "team": "string",  
    "conference": "string",  
    "athleteId": 0,
```

[Links](#)**Code****Description**

```
"athleteSourceId": "string",
"name": "string",
"position": "string",
"games": 0,
"starts": 0,
"minutes": 0,
"points": 0,
"turnovers": 0,
"fouls": 0,
"assists": 0,
"steals": 0,
"blocks": 0,
"usage": 0,
"offensiveRating": 0,
"defensiveRating": 0,
"netRating": 0
```

GET /stats/player/shooting/season



Retrieves player season shooting statistics

Parameters[Try it out](#)**Name****Description****season** * required

Required season filter

integer(\$int32)
*(query)***seasonType**

Optional season type filter

string
*(query)**Available values :* postseason, preseason, regular**team**

Team filter, required if conference is not provided

Name	Description
string (query)	team
conference	Conference abbreviation filter, required if team is not provided
string (query)	conference
startDateRange	Optional start date range filter
string(\$date- time) (query)	startDateRange
endDateRange	Optional end date range filter
string(\$date- time) (query)	endDateRange

Responses

Code	Description	Links
200	Ok	No links

Media type

application/json

Controls Accept header.

Example Value Schema

Code

```
        "attempted": 0,  
        "pct": 0,  
        "assistedPct": 0,  
        "assisted": 0  
    },  
    "freeThrows": {  
        "made": 0,  
        "attempted": 0,  
        "pct": 0  
    },  
    "attemptsBreakdown": {  
        "threePointJumpers": 0,  
        "twoPointJumpers": 0,  
        "tipIns": 0,  
        "layups": 0,  
        "dunks": 0  
    },  
    "athleteId": 0,  
    "athleteName": "string"  
}  
]
```

Links

rankings Historical polls and rankings



GET /rankings



ratings Historical ratings systems



GET /ratings/srs



Retrieves SRS ratings for the provided season, team, or conference.

Parameters

[Try it out](#)

Name	Description
season <code>integer(\$int32) (query)</code>	Optional season filter season
team <code>string (query)</code>	Optional team filter team
conference <code>string (query)</code>	Optional conference abbreviation filter conference

Responses

Code	Description	Links
200	Ok	<i>No links</i>

Media type

`application/json`

Controls Accept header.

Example Value Schema

```
[  
  {  
    "season": 0,  
    "teamId": 0,  
    "team": "string",  
  }]
```

[Links](#)**Code**

```
        "conference": "string",
        "rating": 0
    }
]
```

GET**/ratings/adjusted**

Retrieves adjusted efficiency ratings for the provided season, team, or conference.

Parameters[Try it out](#)**Name****Description**

season Optional season filter

integer(\$int32)
(query) season

team Optional team filter

string
(query) team

conference Optional conference abbreviation filter

string
(query) conference

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"season": 0, "teamId": 0, "team": "string", "conference": "string", "offensiveRating": 0, "defensiveRating": 0, "netRating": 0, "rankings": {"net": 0, "defense": 0, "offense": 0}}]</pre>	No links

lines Historical spreads and betting data ^

GET	/lines	🔒	▼
GET	/lines/providers	🔒	▼

lineups Statistics and data around specific lineups



GET /lineups/team



Queries lineup statistics for a given team and season

Parameters

[Try it out](#)

Name	Description
season * required integer(\$int32) (query)	Required season filter <input type="text" value="season"/>
team * required string (query)	Required team filter <input type="text" value="team"/>
startDateRange string(\$date-time) (query)	Optional start date range filter <input type="text" value="startDateRange"/>
endDateRange string(\$date-time) (query)	Optional end date range filter <input type="text" value="endDateRange"/>

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "lineups": [{ "id": 1, "gameId": 1, "team": "Team A", "players": [{ "id": 1, "name": "Player 1" }, { "id": 2, "name": "Player 2" }, { "id": 3, "name": "Player 3" }, { "id": 4, "name": "Player 4" }, { "id": 5, "name": "Player 5" }], "statistics": [{ "type": "twoPointers", "made": 0, "attempted": 0, "pct": 0 }, { "type": "threePointers", "made": 0, "attempted": 0, "pct": 0 }, { "type": "freeThrows", "made": 0, "attempted": 0, "pct": 0 }, { "type": "fourFactors", "freeThrowRate": 0, "offensiveReboundPct": 0, "turnoverRatio": 0, "effectiveFieldGoalPct": 0 }] }] }</pre>	No links

GET /lineups/game/{gameId}



Queries lineup statistics for a specific game

Parameters

Try it out

Name	Description
------	-------------

gameId * required
Required game id filter
integer(\$int32)
(path)

gameId

Responses

Code	Description	Links
------	-------------	-------

200 Ok No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
[  
  {  
    "teamId": 0,  
    "team": "string",  
    "conference": "string",  
    "idHash": "string",  
    "athletes": [  
      {  
        "name": "string",  
        "id": "string",  
        "position": "string",  
        "height": "string",  
        "weight": "string",  
        "age": "string",  
        "college": "string",  
        "gradYear": "string",  
        "experience": "string",  
        "status": "string",  
        "order": 0  
      }  
    ]  
  }  
]
```

Code	Description	Links
	<pre> "id": 0 },], "totalSeconds": 0, "pace": 0, "offenseRating": 0, "defenseRating": 0, "netRating": 0, "teamStats": { "possessions": 0, "points": 0, "blocks": 0, "assists": 0, "steals": 0, "turnovers": 0, "defensiveRebounds": 0 } } }</pre>	

draft Historical NBA draft data



GET /draft/teams



Retrieves list of NBA teams

Parameters

[Try it out](#)

No parameters

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"id": 0, "sourceId": 0, "location": "string", "name": "string", "displayName": "string", "abbreviation": "string"}]</pre>	<i>No links</i>

GET /draft/positions



Retrieves list of position names for NBA draft prospects

Parameters

[Try it out](#)

No parameters

Responses

Code	Description	Links
200	<p>Ok</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{"name": "string", "abbreviation": "string"}]</pre>	No links

GET /draft/picks 🔒 ⌂

Retrieves historical NBA draft picks

Parameters Try it out

Name	Description
year <code>integer(\$int32) (query)</code>	Optional draft year filter <input type="text" value="year"/>
draftTeam <code>string (query)</code>	Optional NBA team filter <input type="text" value="draftTeam"/>

Name	Description
sourceTeam <small>string (query)</small>	Optional source team (e.g. NCAA) filter sourceTeam
position <small>string (query)</small>	Optional player position abbreviation filter position

Responses

Code	Description	Links
200	Ok Media type application/json Controls Accept header. Example Value Schema [{ "athleteId": 0, "sourceTeamId": 0, "sourceTeamLocation": "string", "sourceTeamName": "string", "sourceTeamLeagueAffiliation": "string", "sourceTeamCollegeId": 0, "draftTeamId": 0, "draftTeam": "string", "year": 0, "overall": 0, "round": 0, }]	No links

[Links](#)**Code**

```
"pick": 0,  
"name": "string",  
"overallRank": 0,  
"positionRank": 0,  
"height": 0,  
"weight": 0  
}  
]
```

recruiting Historical recruiting data

**GET /recruiting/players**

Retrieves historical composite player recruiting ranking and ratings

Parameters[Try it out](#)

Name	Description
year <code>integer(\$int32) (query)</code>	Optional year filter <input type="text" value="year"/>
team <code>string (query)</code>	Optional college team filter <input type="text" value="team"/>
conference <code>string (query)</code>	Optional college conference filter <input type="text" value="conference"/>

Name	Description
position string (query)	Optional position filter position

Responses

Code	Description	Links
200	Ok	<i>No links</i>

Media type

application/json

Controls Accept header.

Example Value Schema

```
[{"id": 0, "sourceId": "string", "position": "string", "schoolId": 0, "school": "string", "hometown": {"countyFips": "string", "longitude": 0, "latitude": 0, "country": "string", "state": "string", "city": "string"}, "committedTo": {"conference": "string", "name": "string", "id": 0}}
```

Links**Code**

```
        "id": 0
    },
    "athleteId": 0,
    "year": 0,
    "name": "string",
    "heightInches": 0,
```

Schemas



```
VenueInfo {
    id*                   [...]
    sourceId*             [...]
    name*                 [...]
    city*                 [...]
    state*                [...]
    country*              [...]
}
```

```
TeamInfo {  
    id* [...]  
    sourceId* [...]  
    school* [...]  
    mascot* [...]  
    abbreviation* [...]  
    displayName* [...]  
    shortDisplayName* [...]  
    primaryColor* [...]  
    secondaryColor* [...]  
    currentVenueId* [...]  
    currentVenue* [...]  
    currentCity* [...]  
    currentState* [...]  
    conferenceId* [...]  
    conference* [...]  
}
```

```
TeamRosterPlayer {  
    id* [...]  
    sourceId* [...]  
    name* [...]  
    firstName* [...]  
    lastName* [...]  
    jersey* [...]  
    position* [...]  
    height* [...]  
    weight* [...]  
    hometown* {...}  
    dateOfBirth* [...]  
    startSeason* [...]  
    endSeason* [...]  
}
```

```
TeamRoster {  
    teamId* [...]  
    teamSourceId* [...]  
    team* [...]  
    conference* [...]  
    season* [...]  
    players* [...]  
}
```

```
PlayerSubstitution {  
    gameId*           [...]  
    startDate*        [...]  
    teamId*           [...]  
    team*              [...]  
    conference*       [...]  
    athleteId*         [...]  
    athlete*           [...]  
    position*          [...]  
    opponentId*        [...]  
    opponent*           [...]  
    opponentConference* [...]  
    subIn*              {...}  
    subOut*             {...}  
}
```

```
TeamSeasonUnitStats {  
    fieldGoals*           {...}  
    twoPointFieldGoals*  {...}  
    threePointFieldGoals* {...}  
    freeThrows*          {...}  
    rebounds*            {...}  
    turnovers*           {...}  
    fouls*               {...}  
    points*              {...}  
    fourFactors*         {...}  
    assists*             [...]  
    blocks*              [...]  
    steals*              [...]  
    possessions*         [...]  
    rating*              [...]  
    trueShooting*        [...]  
}
```

```
TeamSeasonStats {  
    season* [...]  
    seasonLabel* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    games* [...]  
    wins* [...]  
    losses* [...]  
    totalMinutes* [...]  
    pace* [...]  
    teamStats* TeamSeasonUnitStats {...}  
    opponentStats* TeamSeasonUnitStats {...}  
}  
}
```

```
SeasonType string  
Enum:  
    Array [ 3 ]
```

```
ShotTypeBreakdown {  
    made* [...]  
    attempted* [...]  
    pct* [...]  
}
```

```
SeasonShootingStats {  
    season* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    trackedShots* [...]  
    assistedPct* [...]  
    freeThrowRate* [...]  
    dunks* {...}  
    layups* {...}  
    tipIns* ShotTypeBreakdown {...}  
    twoPointJumpers* {...}  
    threePointJumpers* {...}  
    freeThrows* ShotTypeBreakdown {...}  
    attemptsBreakdown* {...}  
}
```

```
PlayerSeasonStats {  
    season* [...]  
    seasonLabel* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    athleteId* [...]  
    athleteSourceId* [...]  
    name* [...]  
    position* [...]  
    games* [...]  
    starts* [...]  
    minutes* [...]  
    points* [...]  
    turnovers* [...]  
    fouls* [...]  
    assists* [...]  
    steals* [...]  
    blocks* [...]  
    usage* [...]  
    offensiveRating* [...]  
    defensiveRating* [...]  
    netRating* [...]  
    PORPAG* [...]  
    effectiveFieldGoalPct* [...]  
    trueShootingPct* [...]  
    assistsTurnoverRatio* [...]  
    freeThrowRate* [...]  
    offensiveReboundPct* [...]  
    fieldGoals* {...}  
    twoPointFieldGoals* {...}  
    threePointFieldGoals* {...}
```

```
    "freeThrows": {....},  
    "rebounds": {....},  
    "winShares": {....}  
}
```

```
PlayerSeasonShootingStats {  
    season*: [...]  
    teamId*: [...]  
    team*: [...]  
    conference*: [...]  
    trackedShots*: [...]  
    assistedPct*: [...]  
    freeThrowRate*: [...]  
    dunks*: {...}  
    layups*: {...}  
    tipIns*: ShotTypeBreakdown {...}  
    twoPointJumpers*: {...}  
    threePointJumpers*: {...}  
    freeThrows*: ShotTypeBreakdown {...}  
    attemptsBreakdown*: {...}  
    athleteId*: [...]  
    athleteName*: [...]  
}
```

```
Recruit  {  
    id*                  [...]  
    sourceId*             [...]  
    position*              [...]  
    schoolId*             [...]  
    school*                [...]  
    hometown*              {...}  
                           nullable: true  
    committedTo*           {...}  
                           nullable: true  
    athleteId*              [...]  
    year*                  [...]  
    name*                  [...]  
    heightInches*            [...]  
    weightPounds*            [...]  
    stars*                  [...]  
    rating*                  [...]  
    ranking*                  [...]  
}
```

```
SrsInfo  {  
    season*                 [...]  
    teamId*                  [...]  
    team*                      [...]  
    conference*                [...]  
    rating*                      [...]  
}
```

```
AdjustedEfficiencyInfo {  
    season* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    offensiveRating* [...]  
    defensiveRating* [...]  
    netRating* [...]  
    rankings* {...}  
}
```

```
PollTeamInfo {  
    season* [...]  
    seasonType* SeasonType [...]  
    week* [...]  
    pollDate* [...]  
    pollType* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    ranking* [...]  
    points* [...]  
    firstPlaceVotes* [...]  
}
```

```
ShotInfo {  
    shooter*          {...}  
    made*             [...]  
    range*            [...]  
    assisted*         [...]  
    assistedBy*       {...}  
    location*         {...}  
}
```

```
PlayInfo  {
    id*                      [...]
    sourceId*                 [...]
    gameId*                  [...]
    gameSourceId*             [...]
    gameStartDate*            [...]
    season*                  [...]
    seasonType*               SeasonType  [...]
    gameType*                 [...]
    tournament*               [...]
    playType*                 [...]
    isHomeTeam*               [...]
    teamId*                  [...]
    team*                     [...]
    conference*              [...]
    teamSeed*                 [...]
    opponentId*               [...]
    opponent*                 [...]
    opponentConference*       [...]
    opponentSeed*              [...]
    period*                  [...]
    clock*                   [...]
    secondsRemaining*         [...]
    homeScore*                [...]
    awayScore*                [...]
    homeWinProbability*       [...]
    scoringPlay*              [...]
    shootingPlay*              [...]
    scoreValue*               [...]
    wallclock*                [...]
    playText*                 [...]
    participation*           [...]
```

```
participants*      [...]
onFloor*          [...]
shotInfo*         {...}
    nullable: true
}
```

```
PlayTypeInfo  {
    id*            [...]
    name*          [...]
}
```

```
ShootingStats  {
    made*          [...]
    attempted*     [...]
    pct*           [...]
}
```

```
LineupUnitStats {  
    possessions*      [...]  
    points*          [...]  
    blocks*          [...]  
    assists*          [...]  
    steals*          [...]  
    turnovers*        [...]  
    defensiveRebounds* [...]  
    offensiveRebounds* [...]  
    trueShooting*     [...]  
    fieldGoals*       ShootingStats {...}  
    twoPointers*      {...}  
    threePointers*    ShootingStats {...}  
    freeThrows*       ShootingStats {...}  
    fourFactors*      {...}  
}
```

```
LineupStats {  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    idHash* [...]  
    athletes* [...]  
    totalSeconds* [...]  
    pace* [...]  
    offenseRating* [...]  
    defenseRating* [...]  
    netRating* [...]  
    teamStats* LineupUnitStats {...}  
    opponentStats* LineupUnitStats {...}  
}
```

```
GameLineInfo {  
    provider* [...]  
    spread* [...]  
    overUnder* [...]  
    homeMoneyline* [...]  
    awayMoneyline* [...]  
    spreadOpen* [...]  
    overUnderOpen* [...]  
}
```

```
GameLines  {
    gameId*          [...]
    season*          [...]
    seasonType*      SeasonType  [...]
    startDate*       [...]
    homeTeamId*      [...]
    homeTeam*         [...]
    homeConference*  [...]
    homeScore*        [...]
    awayTeamId*       [...]
    awayTeam*         [...]
    awayConference*  [...]
    awayScore*        [...]
    lines*           [...]
}
```

```
LineProviderInfo  {
    id*              [...]
    name*             [...]
}
```

```
GameStatus   string
Enum:
    Array [ 5 ]
```

```
GameInfo {  
    id* [...]  
    sourceId* [...]  
    seasonLabel* [...]  
    season* [...]  
    seasonType* SeasonType [...]  
    startDate* [...]  
    startTimeTbd* [...]  
    neutralSite* [...]  
    conferenceGame* [...]  
    gameType* [...]  
    tournament* [...]  
    gameNotes* [...]  
    status* GameStatus [...]  
    attendance* [...]  
    homeTeamId* [...]  
    homeTeam* [...]  
    homeConferenceId* [...]  
    homeConference* [...]  
    homeSeed* [...]  
    homePoints* [...]  
    homePeriodPoints* [...]  
    homeWinner* [...]  
    awayTeamId* [...]  
    awayTeam* [...]  
    awayConferenceId* [...]  
    awayConference* [...]  
    awaySeed* [...]  
    awayPoints* [...]  
    awayPeriodPoints* [...]  
    awayWinner* [...]  
    ...  
}
```

```
    exclusionList*          [...]
    venueId*              [...]
    venue*                [...]
    city*                 [...]
    state*                [...]
}
```

```
GameMediaInfo    {
    gameId*              [...]
    season*              [...]
    seasonLabel*          [...]
    seasonType*          SeasonType  [...]
    tournament*          [...]
    startDate*           [...]
    startTimeTbd*         [...]
    homeTeamId*          [...]
    homeTeam*             [...]
    homeConference*       [...]
    awayTeamId*           [...]
    awayTeam*             [...]
    awayConference*        [...]
    neutralSite*          [...]
    conferenceGame*        [...]
    gameType*             [...]
    notes*                [...]
    broadcasts*          [...]
}
```

```
GameBoxScoreTeamStats {  
    fieldGoals*           {...}  
    twoPointFieldGoals*  {...}  
    threePointFieldGoals* {...}  
    freeThrows*          {...}  
    rebounds*            {...}  
    turnovers*           {...}  
    fouls*               {...}  
    points*              {...}  
    fourFactors*         {...}  
    assists*             [...]  
    blocks*              [...]  
    steals*              [...]  
    possessions*         [...]  
    rating*              [...]  
    trueShooting*        [...]  
    gameScore*            [...]  
}
```

```
GameBoxScoreTeam {  
    gameId* [...]  
    season* [...]  
    seasonLabel* [...]  
    seasonType* SeasonType [...]  
    tournament* [...]  
    startDate* [...]  
    startTimeTbd* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    teamSeed* [...]  
    opponentId* [...]  
    opponent* [...]  
    opponentConference* [...]  
    opponentSeed* [...]  
    neutralSite* [...]  
    isHome* [...]  
    conferenceGame* [...]  
    gameType* [...]  
    notes* [...]  
    gameMinutes* [...]  
    pace* [...]  
    teamStats* GameBoxScoreTeamStats {...}  
    opponentStats* GameBoxScoreTeamStats {...}  
}
```

```
GameBoxScorePlayers {  
    gameId* [...]  
    season* [...]  
    seasonLabel* [...]  
    seasonType* SeasonType [...]  
    tournament* [...]  
    startDate* [...]  
    startTimeTbd* [...]  
    teamId* [...]  
    team* [...]  
    conference* [...]  
    teamSeed* [...]  
    opponentId* [...]  
    opponent* [...]  
    opponentConference* [...]  
    opponentSeed* [...]  
    neutralSite* [...]  
    isHome* [...]  
    conferenceGame* [...]  
    gameType* [...]  
    notes* [...]  
    gameMinutes* [...]  
    gamePace* [...]  
    players* [...]  
}
```

```
DraftTeam {  
    id* [...]  
    sourceId* [...]  
    location* [...]  
    name* [...]  
    displayName* [...]  
    abbreviation* [...]  
}
```

```
DraftPosition {  
    name* [...]  
    abbreviation* [...]  
}
```

```
DraftPick {  
    athleteId* [...]  
    sourceTeamId* [...]  
    sourceTeamLocation* [...]  
    sourceTeamName* [...]  
    sourceTeamLeagueAffiliation* [...]  
    sourceTeamCollegeId* [...]  
    draftTeamId* [...]  
    draftTeam* [...]  
    year* [...]  
    overall* [...]  
    round* [...]  
    pick* [...]  
    name* [...]  
    overallRank* [...]  
    positionRank* [...]  
    height* [...]  
    weight* [...]  
}
```

```
ConferenceInfo {  
    id* [...]  
    sourceId* [...]  
    name* [...]  
    abbreviation* [...]  
    shortName* [...]  
}
```

```
ConferenceHistory {  
    id* [...]  
    sourceId* [...]  
    name* [...]  
    abbreviation* [...]  
    shortName* [...]  
    teams* [...]  
}
```