Chess.com Wrapper

Release 1.2

Artur Saradzhyan

Jan 13, 2022

CONTENTS:

CHAPTER

ONE

DESCRIPTION

“A full Python Wrapper around Chess.com API which provides public data from the Chess.com website. All endpoints provided by Chess.com’s API are available in the respectively named methods. The package allows for simple interaction with the API, eliminating the need for repetitive code and testing.”

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

CHAPTER

THREE

INSTALLATION

The package requires Python 3.7 or higher.

Install from PyPI: `pip install chess.com`

CHAPTER

FOUR

RETRIEVING DATA

All the functions return a *ChessDotComResponse* object. The data can be accessed in dictionary format or via attributes. All functions can be made asynchronous. The package uses aiohttp to send requests to the API.

4.1 Synchronous

```
from chessdotcom import get_player_profile

response = get_player_profile("fabianocaruana")

player_name = response.json['player']['name']
#or
player_name = response.player.name
```

4.2 Asynchronous

```
from asyncio import gather

from chessdotcom.aio import get_player_profile, Client
#or
from chessdotcom import get_player_profile, Client
Client.aio = True

usernames = ["fabianocaruana", "GMHikaruOnTwitch", "MagnusCarlsen", "GarryKasparov"]
cors = [get_player_profile(name) for name in usernames]
responses = Client.loop.run_until_complete(gather(*cors))
```

CHAPTER FIVE

CONFIGURING THE CLIENT OBJECT

The project uses *requests* package to interact with the API. Headers and proxies can be set through the *Client* object. Official Chess.com documentation recommends adding a *User-Agent* header.

```
#optional
from chessdotcom import Client

Client.headers["User-Agent"] = (
   "My Python Application. "
   "Contact me at email@example.com"
)
```

All the methods from the module will now include the header when making a request to the API.

CHAPTER

SIX

API REFERENCE

6.1 chessdotcom.types

exception chessdotcom.types.**ChessDotComError**(*status_code: int*, *response_text: str*, *headers: multidict._multidict.CIMultiDictProxy*)

Custom Exception object.

Variables

- **status_code** – Contains the status code of the API's response.
- **json** – Dictionary representation of the API's response.
- **text** – API's raw response decoded into a string.

class chessdotcom.types.**ChessDotComResponse**(*response_text: str*, *top_level_attr: Optional[str] = None*, *no_json=False*)

Custom object for holding the API's response.

Variables

- **json** – Dictionary representation of the API's response.
- **{nested_object}** – Object representation of the API's response.
- **text** – API's raw response decoded into a string.

class chessdotcom.types.**Collection**(***kwargs*)

6.2 chessdotcom.client

class chessdotcom.client.**Client**

Client for Chess.com Public API. The client is only responsible for making calls.

Variables

- **config** – Dictionary containing extra keyword arguments for requests to the API (headers, proxy, etc).
- **loop** – Asyncio event loop.
- **aio** – Determines if the functions behave asynchronously.

chessdotcom.client.**get_club_details**(*url_id: str*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters **url_id** – URL for the club's web page on www.chess.com.

Returns `ChessDotComResponse` object containing additional details about a club.

`chessdotcom.client.`**`get_club_matches`**`(`*url_id: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`url_id`** – URL for the club's web page on www.chess.com.

Returns `ChessDotComResponse` object containing a list of daily and club matches.

`chessdotcom.client.`**`get_club_members`**`(`*url_id: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`url_id`** – URL for the club's web page on www.chess.com.

Returns `ChessDotComResponse` object containing a list of club members.

`chessdotcom.client.`**`get_country_clubs`**`(`*iso: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`iso`** – country's 2-character ISO 3166 code.

Returns `ChessDotComResponse` object containing a list of URLs for clubs identified as being in or associated with this country.

`chessdotcom.client.`**`get_country_details`**`(`*iso: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`iso`** – country's 2-character ISO 3166 code.

Returns `ChessDotComResponse` object containing additional details about a country.

`chessdotcom.client.`**`get_country_players`**`(`*iso: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`iso`** – country's 2-character ISO 3166 code.

Returns `ChessDotComResponse` object containing a list of usernames for players who identify themselves as being in this country.

`chessdotcom.client.`**`get_current_daily_puzzle`**`(`***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Returns `ChessDotComResponse` object containing information about the daily puzzle found in www.chess.com.

`chessdotcom.client.`**`get_leaderboards`**`(`***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Returns `ChessDotComResponse` object containing information about top 50 player for daily and live games, tactics and lessons.

`chessdotcom.client.`**`get_player_clubs`**`(`*username: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing a list of clubs the player is a member of.

`chessdotcom.client.`**`get_player_current_games`**`(`*username: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing a list of Daily Chess games that a player is currently playing.

`chessdotcom.client.`**`get_player_current_games_to_move`**(*username: str*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing a list of Daily Chess games where it is the player's turn to act.

`chessdotcom.client.`**`get_player_game_archives`**(*username: str*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing a list of monthly archives available for this player.

`chessdotcom.client.`**`get_player_games_by_month`**(*username: str*, *year: Optional[Union[str, int]] = None*, *month: Optional[Union[str, int]] = None*, *datetime_obj: Optional[datetime.datetime] = None*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters

- **`username`** – username of the player.
- **`year`** – the year (yyyy).
- **`month`** – the month (mm).
- **`date`** – datetime.datetime of the month. Can be passed in instead of month and year parameters.

Returns `ChessDotComResponse` object containing a list of live and daily Chess games that a player has finished.

`chessdotcom.client.`**`get_player_games_by_month_pgn`**(*username: str*, *year: Optional[Union[str, int]] = None*, *month: Optional[Union[str, int]] = None*, *datetime_obj: Optional[datetime.datetime] = None*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters

- **`username`** – username of the player.
- **`year`** – the year (yyyy).
- **`month`** – the month (mm).
- **`date`** – datetime.datetime of the month. Can be passed in instead of month and year parameters.

Returns `ChessDotComResponse` object containing standard multi-game format PGN containing all games for a month.

`chessdotcom.client.`**`get_player_profile`**`(`*username: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing information about the player's profile.

`chessdotcom.client.`**`get_player_stats`**`(`*username: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing information about the plyers's ratings, win/loss, and other stats.

`chessdotcom.client.`**`get_player_team_matches`**`(`*username: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing a list of team matches the player has attended, is participating or is currently registered.

`chessdotcom.client.`**`get_player_tournaments`**`(`*username: str*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`username`** – username of the player.

Returns `ChessDotComResponse` object containing a list of tournaments the player is registered, is attending or has attended in the past.

`chessdotcom.client.`**`get_random_daily_puzzle`**`(`***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Returns `ChessDotComResponse` object containing information about a randomly picked daily puzzle.

`chessdotcom.client.`**`get_streamers`**`(`***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Returns `ChessDotComResponse` object containing information about Chess.com streamers.

`chessdotcom.client.`**`get_team_match`**`(`*match_id: int*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters **`match_id`** – the id of the match.

Returns `ChessDotComResponse` object containing details about a team match and players playing that match.

`chessdotcom.client.`**`get_team_match_board`**`(`*match_id: int*, *board_num: int*, ***kwargs*`)` → *chessdotcom.types.ChessDotComResponse*

Parameters

- **`match_id`** – the id of the match.
- **`board_num`** – the number of the board.

Returns `ChessDotComResponse` object containing details about a team match board.

`chessdotcom.client.`**`get_team_match_live`**(*match_id: int*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters **`match_id`** – the id of the match.

Returns `ChessDotComResponse` object containing details about a team match and players playing that match.

`chessdotcom.client.`**`get_team_match_live_board`**(*match_id: int*, *board_num: int*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters

- **`match_id`** – the id of the match.
- **`board_num`** – the number of the board.

Returns `ChessDotComResponse` object containing details about a team match board.

`chessdotcom.client.`**`get_titled_players`**(*title_abbrev: str*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters **`title_abbrev`** – abbreviation of chess title.

Returns `ChessDotComResponse` object containing a list of usernames.

`chessdotcom.client.`**`get_tournament_details`**(*url_id: str*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters **`url_id`** – URL for the club's web page on www.chess.com.

Returns `ChessDotComResponse` object containing details about a daily, live and arena tournament.

`chessdotcom.client.`**`get_tournament_round`**(*url_id: str*, *round_num: int*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters

- **`url_id`** – URL for the club's web page on www.chess.com.
- **`round_num`** – the round of the tournament.

Returns `ChessDotComResponse` object containing details about a tournament's round.

`chessdotcom.client.`**`get_tournament_round_group_details`**(*url_id: str*, *round_num: int*, *group_num: int*, ***kwargs*) → *chessdotcom.types.ChessDotComResponse*

Parameters

- **`url_id`** – URL for the club's web page on www.chess.com.
- **`round_num`** – the round of the tournament.
- **`group_num`** – the group in the tournament.

Returns `ChessDotComResponse` object containing details about a tournament's round group.

PYTHON MODULE INDEX

INDEX