
PUBGy

Release 1.1.6

Jul 28, 2021

Contents

1	Installation	3
2	Documentation	5
2.1	Reference	5
2.1.1	Client	5
2.1.2	Objects	7
	Python Module Index	9
	Index	11

PUBGy is a beginner-friendly, fully implemented, asynchronous API wrapper for PLAYERUNKNOWN'S BATTLE-
GROUNDS (PUBG).

Before starting, ensure that you've first created an account and have acquired an API token via <https://developer.pubg.com/>

CHAPTER 1

Installation

To install using PyPI:

```
python -m pip install -U pubgy
```

Building from source (warning: this may be an unstable version of PUBGy, ensure you know what you are doing):

```
git clone https://github.com/Discord-ian/pubgy.git
cd pubgy
python setup.py install
```


2.1 Reference

2.1.1 Client

Copyright (c) 2018-2021 Discordian

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

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

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

class `pubgy.client.Pubgy` (*auth_token*, *defaultshard=None*)

Represents the core of PUBGy. Start by initializing an instance of it, using

```
import pubgy
client = pubgy.Pubgy("your auth token")
```

Parameters

- **auth_token** (*str*) – The API Authentication token
- **defaultshard** – A Shard from the list of shards in `constants.SHARD_LIST`

Returns A Pubgy object to do requests from.

get_leaderboard (*platform, season, gamemode*)

Gets the leaderboard corresponding to the arguments

Parameters

- **platform** –
- **season** –
- **gamemode** –

Returns

get_player (*plyname, shard=None*)

This function is a coroutine.

Gets a player's stats by using either their player name or account id.

If given a list of player names/ids, they all must be the same type.

Parameters **plyname** (*str or list*) – A Players name/ID

Returns `objects.Player`

get_samples (*shard=None, amount=1*)

This function is a coroutine.

Gets sample matches from the /samples endpoint

You can comfortably ignore this if you do not know what it does

Parameters

- **shard** (*str or None*) – Defaults to shard passed on client initialization
- **amount** (*int*) – Defaults to 1, only returns the amount of match objects equal to length

Returns A list of `objects.Match`

matches (*id, shard=None, sorts=None, filter=None*)

This function is a coroutine.

Gets specific match info depending on the parameters supplied.

Parameters

- **id** (*str*) – Required.
- **shard** (*str or None*) – Defaults to `Query.shard`
- **amount** (*int*) – Defaults to 5, only returns the amount of match objects equal to length
- **offset** (*int*) – Defaults to 0, where to start parsing the stats from.

Returns `objects.Match`

shard

Returns the shard the client was initiated with. This is used as the default shard for all commands, unless another one is passed

Returns `str`

solve (*telemetry*)

This function is a coroutine.

Puts a Telemetry object into a useful set of data.

Parameters **telemetry** (*A Telemetry object with only telemetry.url filled.*) – A telemetry object that has just been received from a match.

stats (*player*, *id=None*)

This function is a coroutine. :param player: A player name :type player: `objects.Player` :param id: A player id (with account) :type id: `str` :return: A `objects.Player` with a filled `objects.Stats` property.

2.1.2 Objects

These are all the objects that PUBGy uses, which are both used internally and returned to the end user. Not recommended to build these yourself, unless you know what you are doing

p

`pubgy.client`, 5
`pubgy.objects`, 7

G

`get_leaderboard()` (*pubgy.client.Pubgy method*), 5
`get_player()` (*pubgy.client.Pubgy method*), 6
`get_samples()` (*pubgy.client.Pubgy method*), 6

M

`matches()` (*pubgy.client.Pubgy method*), 6

P

`Pubgy` (*class in pubgy.client*), 5
`pubgy.client` (*module*), 5
`pubgy.objects` (*module*), 7

S

`shard` (*pubgy.client.Pubgy attribute*), 6
`solve()` (*pubgy.client.Pubgy method*), 6
`stats()` (*pubgy.client.Pubgy method*), 6