SteamQuery Documentation

Release stable

Mar 11, 2022

Contents

SteamQuery allows you to gather information about a steam server and return it in a dictionary format

CHAPTER 1

Installing

Installing via pip

```
$ pip install steamquery
```

CHAPTER 2

How to use

Standard use

```
>>> from steam import SteamQuery
>>> server_obj = SteamQuery("serverip", port)
>>> return_dictionary = server_obj.return_last_data()  # This will store the last␣
↪results so you dont need to query again
# OR
>>> return_dictionary = server_obj.query_game_server() # New results, also saved and␣
↪can be retrieved via the return_last_data method
>>> return_dictionary
>>> {'online': True, 'ip': 'ip', 'port': port, 'name': 'name', 'map': 'map', 'game':
↪'game', 'description': 'server desc', 'players': players, 'max_players': slots,
↪'bots': bots, 'password_required': bool, 'vac_secure': bool, 'server-type': 'type',
↪'os': 'os'}
>>> return_dictionary["players"]
>>> 10  # Example as the dictionary above just has 'players'
```

If the server is offline

```
>>> from steam import SteamQuery
>>> server_obj = SteamQuery("serverip", port)
>>> return_dictionary = server_obj.return_last_data()
>>> return_dictionary
>>> {'online': False, 'error': 'Request timed out'}
```

Timeout has a default value of 1 second, however a different integer can be passed

```
>>> from steam import SteamQuery
>>> server_obj = SteamQuery("serverip", port, 2)  # 2 seconds
```

CHAPTER 3

What it returns

- online: Boolean
- ip: String
- port: Integer
- name: String
- map: String
- game: String
- description: String
- players: Integer
- max_players: Integer
- bots: Integer
- password_required: Boolean
- vac_secure: Boolean
- server_type: String (Dedicated/Non-Dedicated/SourceTV)
- os: String (Windows/Linux/Mac)

CHAPTER 4

Note

This was made for python 3 and has no support for python 2