
proxy_py Documentation

Release 1.0.2

DevAlone

Apr 09, 2018

Contents

1 proxy_py API	3
1.1 Possible keys	3
2 Indices and tables	7

Contents:

proxy_py expects HTTP POST requests with JSON as a body, so you need to add header `Content-Type: application/json` and send correct JSON document.

Example of correct request:

```
{  
  "model": "proxy",  
  "method": "get"  
}
```

Response is also HTTP with JSON and status code depending on whether error happened or not.

- 200 if there wasn't error
- 400 if you sent bad request
- 500 if there was an error during execution your request or in some other cases

status_code is also duplicated in JSON body.

1.1 Possible keys

- model - specifies what you will work with. Now it's only supported to work with proxy model.
- method - what you're gonna do with it
 - get - get model items as json objects. Detailed description is below
 - count - count how many items there are. Detailed description is below

1.1.1 get method

get method supports following keys:

- order_by (string) - specifies ordering fields as comma separated value.

Examples:

"uptime" just sorts proxies by uptime field ascending.

Note: uptime is the timestamp from which proxy is working, NOT proxy's working time

To sort descending use - before field name.

"-response_time" returns proxies with maximum response_time first (in microseconds)

It's possible to sort using multiple fields

"number_of_bad_checks, response_time" returns proxies with minimum number_of_bad_checks first, if there are proxies with the same number_of_bad_checks, sorts them by response_time

- limit (integer) - specifying how many proxies to return
- offset (integer) - specifying how many proxies to skip

Example of get request:

```
{
  "model": "proxy",
  "method": "get",
  "order_by": "number_of_bad_checks, response_time",
  "limit": 100,
  "offset": 200
}
```

Response

```
{
  "count": 6569,
  "data": [
    {
      "address": "socks5://localhost:9999",
      "auth_data": "",
      "bad_proxy": false,
      "domain": "localhost",
      "last_check_time": 1517089048,
      "number_of_bad_checks": 0,
      "port": 9999,
      "protocol": "socks5",
      "response_time": 1819186,
      "uptime": 1517072132
    },
    ...
  ],
  "has_more": true,
  "status": "ok",
  "status_code": 200
}
```

- count (integer) - total number of proxies for that request
- data (array) - list of proxies
- has_more (boolean) - value indicating whether you can increase offset to get more proxies or not
- status (string) - "error" if error happened, "ok" otherwise

Example of error:

Request:

```
{
  "model": "user",
  "method": "get",
  "order_by": "number_of_bad_checks, response_time",
  "limit": 100,
  "offset": 200
}
```

Response:

```
{
  "error_message": "Model \"user\" doesn't exist or isn't allowed",
  "status": "error",
  "status_code": 400
}
```

1.1.2 count method

Same as get, but not returns data

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`