

---

# **python-anticaptcha Documentation**

*Release 0.2.0*

**Adam Dobrawy**

**Jul 17, 2020**



---

## Contents:

---

<b>1 Usage</b>	<b>3</b>
1.1 Solve recaptcha . . . . .	3
1.2 Solve text captcha . . . . .	4
1.3 Solve funcaptcha . . . . .	4
1.4 Report incorrect image . . . . .	4
1.5 Custom tasks . . . . .	5
1.6 Setup proxy . . . . .	5
1.7 Error handling . . . . .	5
<b>2 Custom fields</b>	<b>7</b>
<b>3 API</b>	<b>9</b>
3.1 Base . . . . .	9
3.2 Exceptions . . . . .	10
3.3 Tasks . . . . .	10
3.4 Fields . . . . .	13
<b>4 Changelog</b>	<b>15</b>
4.1 0.7.1 - 2020-07-17 . . . . .	15
4.2 0.7.0 - 2020-06-08 . . . . .	15
4.3 0.6.0 - 2020-04-13 . . . . .	16
4.4 0.5.1 - 2020-03-31 . . . . .	16
4.5 0.5.0 - 2020-03-30 . . . . .	16
4.6 0.4.2 - 2019-10-27 . . . . .	16
4.7 0.4.1 - 2019-07-09 . . . . .	17
4.8 0.4.0 - 2019-06-28 . . . . .	17
4.9 0.3.2 - 2018-10-17 . . . . .	17
4.10 0.3.1 - 2018-03-18 . . . . .	17
<b>5 Development</b>	<b>19</b>
5.1 New release . . . . .	19
<b>6 Indices and tables</b>	<b>21</b>
<b>Python Module Index</b>	<b>23</b>
<b>Index</b>	<b>25</b>



Client library for solve captchas with [Anticaptcha.com](#) support. The library supports both Python 2.7 and Python 3.

The library is cyclically and automatically tested for proper operation. We are constantly making the best efforts for its effective operation.

In case of any problems with integration - [read the documentation](#), [create an issue](#), use [Gitter](#) or contact privately.



To use this library do you need [Anticaptcha.com](https://www.anticaptcha.com) API key.

## 1.1 Solve recaptcha

Example snippet for Recaptcha:

```
from python_anticaptcha import AnticaptchaClient, NoCaptchaTaskProxylessTask

api_key = '174faff8fbc769e94a5862391ecfd010'
site_key = '6Le-wvkSAAAAAPBMRTvw0Q4Muexq9bi0DJwx_mJ-' # grab from site
url = 'https://www.google.com/recaptcha/api2/demo'

client = AnticaptchaClient(api_key)
task = NoCaptchaTaskProxylessTask(url, site_key)
job = client.createTask(task)
job.join()
print job.get_solution_response()
```

The full integration example is available in file `examples/recaptcha.py`.

If you only process few page many times to increase reliability, you can specify whether the captcha is visible or not. This parameter is not required, as is the system detects invisible sitekeys automatically, and needs several recursive measures for automated training and analysis. For provide that pass `is_invisible` parameter to `NoCaptchaTaskProxylessTask` or `NoCaptchaTask` eg.:

```
from python_anticaptcha import AnticaptchaClient, NoCaptchaTaskProxylessTask

api_key = '174faff8fbc769e94a5862391ecfd010'
site_key = '6Lc-0DYUAAAAAOPM3RGobCfKjIE5STmzvZfHbbNx' # grab from site
url = 'https://losangeles.craigslist.org/lac/kid/d/housekeeper-sitting-pet-care/
↳6720136191.html'
```

(continues on next page)

(continued from previous page)

```
client = AnticaptchaClient(api_key)
task = NoCaptchaTaskProxylessTask(url, site_key, is_invisible=True)
job = client.createTask(task)
job.join()
print job.get_solution_response()
```

## 1.2 Solve text captcha

Example snippet for text captcha:

```
from python_anticaptcha import AnticaptchaClient, ImageToTextTask

api_key = '174faff8fbc769e94a5862391ecfd010'
captcha_fp = open('examples/captcha_ms.jpeg', 'rb')
client = AnticaptchaClient(api_key)
task = ImageToTextTask(captcha_fp)
job = client.createTask(task)
job.join()
print job.get_captcha_text()
```

## 1.3 Solve funcaptcha

Example snippet for funcaptcha:

```
from python_anticaptcha import AnticaptchaClient, FunCaptchaTask, Proxy
UA = 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_1) AppleWebKit/537.36 ' \
     '(KHTML, like Gecko) Chrome/39.0.2171.95 Safari/537.36'

api_key = '174faff8fbc769e94a5862391ecfd010'
site_key = 'DE0B0BB7-1EE4-4D70-1853-31B835D4506B' # grab from site
url = 'https://www.google.com/recaptcha/api2/demo'
proxy = Proxy.parse_url("socks5://login:password@123.123.123.123")

client = AnticaptchaClient(api_key)
task = FunCaptchaTask(url, site_key, proxy=proxy, user_agent=user_agent)
job = client.createTask(task)
job.join()
print job.get_token_response()
```

## 1.4 Report incorrect image

Example snippet for reporting an incorrect image task:

```
from python_anticaptcha import AnticaptchaClient, ImageToTextTask

api_key = '174faff8fbc769e94a5862391ecfd010'
captcha_fp = open('examples/captcha_ms.jpeg', 'rb')
client = AnticaptchaClient(api_key)
task = ImageToTextTask(captcha_fp)
```

(continues on next page)



(continued from previous page)

```
job = client.createTask(task)
job.join()
print job.get_captcha_text()
job.report_incorrect()
```

## 1.5 Custom tasks

There is support for your own (captcha) forms. It allows you to analyze any data in various ways, eg. classify offensive image, count elements on the image, etc. The scope of the data, the form to describe them, you specify yourself.

For details, go to 'Custom fields' section in the documentation.

## 1.6 Setup proxy

The library is not responsible for managing the proxy server. However, we point to the possibility of simply launching such a server by:

```
pip install mitmproxy
mitmweb -p 9190 -b 0.0.0.0 --ignore '.' --socks
```

Next to in your application use something like:

```
proxy = Proxy.parse_url("socks5://123.123.123.123:9190")
```

We recommend entering IP-based access control for incoming addresses to proxy. IP address required by [Anti-captcha.com](#) is:

```
69.65.41.21
209.212.146.168
```

## 1.7 Error handling

In the event of an application error, the `AnticaptchaException` exception is thrown. To handle the exception, do the following:

```
from python_anticaptcha import AnticaptchaException, ImageToTextTask

try:
    # any actions
except AnticaptchaException as e:
    if e.error_code == 'ERROR_ZERO_BALANCE':
        notify_about_no_funds(e.error_id, e.error_code, e.error_description)
    else:
        raise
```



## CHAPTER 2

---

### Custom fields

---

Python-Anticaptcha since 0.3.0 supports task type `CustomCaptchaTask`. It allows collect arbitrary description of image.

This type of tasks is suitable for use when you need to describe what is on an image and you need workers to fill a custom form about this.

Examples:

- Build ML dataset
- Read letters and numbers from a car license plate
- Find a phone number on a commercial
- Reading data from the invoice scan
- Complete task like “count monkeys on a picture”
- etc.

If your case does not fit in such a flow, and yet you need a human factor for many repetitive tasks - [create an issue](#), use [Gitter](#) or contact privately.

At the beginning you need to initialize the [Anti-captcha.com](#) client:

```
from python_anticaptcha import AnticaptchaClient
api_key = 'xxxx'
client = AnticaptchaClient(api_key)
```

Then you build the form as dict of fields:

```
from collections import OrderedDict
from python_anticaptcha.fields import TextInput, Select

form = OrderedDict()
form['dot_count'] = TextInput(label="Dot count",
                             labelHint="Enter the number of dots.")
form['dot_color'] = Select(label="Dot color",
```

(continues on next page)

(continued from previous page)

```
labelHint="Select the color of dots.",
choices=[('red', "Red"),
          ('blue', "Blue"),
          ('green', "Green")])
```

The form field key name specifies the name of the answer key.

There is multiple types of form fields, see *Fields*.

If they do not meet your expectations - use the JSON structure directly. You can use the following to build a proper structure:

- use [FormBuilder](#) tool in Anti-Captcha.com clients area.
- code it manually using [Anti-captcha.com documentation](#).

Next to send form & image URL to [Anti-captcha.com](#) and receive answer:

```
url = "https://s.jawne.info.pl/dot_img"
task = CustomCaptchaTask(imageUrl=url,
                          assignment="Count the dots and indicate their color.",
                          form=form)

job = client.createTask(task)
job.join()
answer = job.get_answers()
print(answer['dot_count'], answer['dot_color'])
```

### 3.1 Base

```
class python_anticaptcha.base.AnticaptchaClient(client_key, lan-  
                                               guage_pool='en',  
                                               host='api.anti-  
                                               captcha.com',  
                                               use_ssl=True)  
  
    APP_STAT_URL = '/getAppStats'  
    BALANCE_URL = '/getBalance'  
    CREATE_TASK_URL = '/createTask'  
    REPORT_IMAGE_URL = '/reportIncorrectImageCaptcha'  
    REPORT_RECAPTCHA_URL = '/reportIncorrectRecaptcha'  
    SOFT_ID = 847  
    TASK_RESULT_URL = '/getTaskResult'  
    client_ip  
    client_key = None  
    createTask(task)  
    createTaskSmee(task, timeout=300)  
        Beta method to stream response from smee.io  
    getAppStats(soft_id, mode)  
    getBalance()  
    getTaskResult(task_id)  
    language_pool = 'en'
```

```
    reportIncorrectImage (task_id)
    reportIncorrectRecaptcha (task_id)
    response_timeout = 5
class python_anticaptcha.base.Job (client, task_id)

    check_is_ready ()
    client = None
    get_answers ()
    get_captcha_text ()
    get_cells_numbers ()
    get_solution_response ()
    get_token_response ()
    join (maximum_time=None)
    report_incorrect ()
    report_incorrect_image ()
    report_incorrect_recaptcha ()
    task_id = None
```

## 3.2 Exceptions

```
exception python_anticaptcha.exceptions.AnticaptchaException (error_id,
                                                                er-
                                                                ror_code,
                                                                er-
                                                                ror_description,
                                                                *args)

python_anticaptcha.exceptions.AnticatpchaException
    alias of python_anticaptcha.exceptions.AnticaptchaException

exception python_anticaptcha.exceptions.InvalidWidthException (width)
exception python_anticaptcha.exceptions.MissingNameException (cls)
```

## 3.3 Tasks

```
class python_anticaptcha.tasks.BaseTask

    serialize (**result)

class python_anticaptcha.tasks.CustomCaptchaTask (imageUrl, form=None,
                                                  assignment=None)

    assignment = None
    form = None
```

```

    imageUrl = None
    serialize()
    type = 'CustomCaptchaTask'
class python_anticaptcha.tasks.FunCaptchaProxylessTask (website_url,
                                                         website_key,
                                                         *args,
                                                         **kwargs)

    serialize(**result)
    type = 'FunCaptchaTaskProxyless'
    websiteKey = None
    websiteURL = None
class python_anticaptcha.tasks.FunCaptchaTask (*args, **kwargs)

    type = 'FunCaptchaTask'
class python_anticaptcha.tasks.HCaptchaTask (*args, **kwargs)

    type = 'HCaptchaTask'
class python_anticaptcha.tasks.HCaptchaTaskProxyless (website_url, web-
                                                         site_key, *args,
                                                         **kwargs)

    serialize(**result)
    type = 'HCaptchaTaskProxyless'
    websiteKey = None
    websiteURL = None
class python_anticaptcha.tasks.ImageToTextTask (fp, phrase=None,
                                                  case=None, nu-
                                                  meric=None, math=None,
                                                  min_length=None,
                                                  max_length=None)

    case = None
    fp = None
    math = None
    maxLength = None
    minLength = None
    numeric = None
    phrase = None
    serialize()
    type = 'ImageToTextTask'
class python_anticaptcha.tasks.NoCaptchaTask (*args, **kwargs)

```

```
    type = 'NoCaptchaTask'  
  
class python_anticaptcha.tasks.NoCaptchaTaskProxylessTask(website_url,  
                                                         web-  
                                                         site_key,  
                                                         web-  
                                                         site_s_token=None,  
                                                         is_invisible=None,  
                                                         re-  
                                                         captcha_data_s_value=None)  
  
    recaptchaDataSValue = None  
    serialize()  
    type = 'NoCaptchaTaskProxyless'  
    websiteKey = None  
    websiteSToken = None  
    websiteURL = None  
  
class python_anticaptcha.tasks.ProxyMixin(*args, **kwargs)  
  
    serialize(**result)  
  
class python_anticaptcha.tasks.RecaptchaV3TaskProxyless(website_url,  
                                                         website_key,  
                                                         min_score,  
                                                         page_action)  
  
    minScore = None  
    pageAction = None  
    serialize()  
    type = 'RecaptchaV3TaskProxyless'  
    websiteKey = None  
    websiteURL = None  
  
class python_anticaptcha.tasks.SquareNetTask(fp, objectName, rowsCount,  
                                              columnsCount)  
  
    columnsCount = None  
    fp = None  
    objectName = None  
    rowsCount = None  
    serialize()  
    type = 'SquareNetTask'
```



## 3.4 Fields

```

class python_anticaptcha.fields.BaseField

    label = None
    labelHint = None
    serialize(name=None)

class python_anticaptcha.fields.Checkbox(text, label=None, labelHint=None)

    serialize(name=None)

class python_anticaptcha.fields.Image(imageUrl, label=None, labelHint=None)

    contentType = 'image'
    serialize(name=None)

class python_anticaptcha.fields.ImageUpload(label=None, labelHint=None)

    serialize(name=None)

class python_anticaptcha.fields.NameBaseField

    name = None
    serialize(name=None)

class python_anticaptcha.fields.Radio(label=None, choices=None, label-
                                     Hint=None)

    type = 'radio'

class python_anticaptcha.fields.Select(label=None, choices=None, label-
                                     Hint=None)

    get_choices()
    serialize(name=None)
    type = 'select'

class python_anticaptcha.fields.SimpleText(content, label=None, label-
                                     Hint=None, width=None)

    contentType = 'text'
    serialize(name=None)

class python_anticaptcha.fields.TextInput(placeholder=None, label=None,
                                     labelHint=None, width=None)

    serialize(name=None)

class python_anticaptcha.fields.Textarea(placeholder=None, rows=None,
                                     label=None, width=None, label-
                                     Hint=None)

```

```
    serialize (name=None)  
class python_anticaptcha.fields.WebLink (linkText, linkUrl, label=None, label-  
                                           Hint=None, width=None)  
  
    contentType = 'link'  
    serialize (name=None)
```

### 4.1 0.7.1 - 2020-07-17

#### 4.1.1 Added

- Added examples for proxy mode including *hcaptcha\_request\_proxy*

#### 4.1.2 Changed

- Fix inheritance of *FunCaptchaTask*
- Added *FunCaptchaTask* to e2e tests

### 4.2 0.7.0 - 2020-06-08

#### 4.2.1 Added

- Added parameter *recaptchaDataSValue* in *NoCaptchaTask\**

Thanks to this change added support for additional “data-s” used by some custom ReCaptcha deployments, which is in fact a one-time token and must be grabbed every time you want to solve a Recaptcha. `<div class="g-recaptcha" data-sitekey="some sitekey" data-s="THIS_TOKEN"></div>`

#### 4.2.2 Changed

- Fixed deprecated method *report\_incorrect*. You should currently use *report\_incorrect\_image* instead already.

## 4.3 0.6.0 - 2020-04-13

### 4.3.1 Added

- Added custom timeout for `createTaskSmee`. Use as `client.createTaskSmee(task, timeout=5*60)`. Default timeout is 5 minutes.
- Added `squarenet_validator` for usage with thread pool for concurrent execution

### 4.3.2 Changed

- Default 5 seconds timeout apply to all API request.

## 4.4 0.5.1 - 2020-03-31

### 4.4.1 Changed

- Fix import of package

## 4.5 0.5.0 - 2020-03-30

### 4.5.1 Added

- Added `HCaptchaTaskProxyless` and `HCaptchaTask` for support `hCaptcha`. See `examples/hcaptcha_request.py` for detailed usage example.
- Added `SquareNetTask`. See `examples/squarenet.py` for detailed usage example.
- Added `Job.report_incorrect_recaptcha` and `Job.report_incorrect_image`.

### 4.5.2 Changed

- Exposed `FunCaptchaProxylessTask` as `python_anticaptcha.FunCaptchaProxylessTask`
- Exposed `CustomCaptchaTask` as `python_anticaptcha.CustomCaptchaTask`
- Formated code via Black
- Move constant monitoring to GitHub Actions
- Deprecated `Job.report_incorrect`. Use `report_incorrect_image` instead.

## 4.6 0.4.2 - 2019-10-27

### 4.6.1 Added

- Added example `remote_image.py`

## 4.6.2 Changed

- Switch CI from TravisCI to GitHub Actions
- Automate PyPI releases
- Use `use_scm_version` for versioning
- Drop `use_2to3` in `setup.py`

## 4.7 0.4.1 - 2019-07-09

### 4.7.1 Added

- Added `python_anticaptcha.__version__` to provide version signature (see PEP396)

### 4.7.2 Changed

- `python_anticaptcha.AnticaptchaClient.createTaskSmee` use shared session & keep connection.

## 4.8 0.4.0 - 2019-06-28

### 4.8.1 Added

- **Added `python_anticaptcha.AnticaptchaClient.createTaskSmee` to receive responses without polling**  
The method, which is based on the callback result of `captcha / task` factory to Smee.io, which immediately transfers it to library. Allows to significantly shorten the waiting time for a response and to reduce load the network connection. The method is in beta and the way it works may change. All comments are welcome.
- Recaptcha V3 is now officially supported by Anti-Captcha. Added `python_anticaptcha.RecaptchaV3TaskProxyless`.

## 4.9 0.3.2 - 2018-10-17

### 4.9.1 Added

- Added support for `IsInvisible` flag in `NoCaptchaTaskProxylessTask` and `NoCaptchaTask`

## 4.10 0.3.1 - 2018-03-18

### 4.10.1 Changed

- Replaced `python_anticaptcha.AnticatpchaException` to `python_anticaptcha.AnticaptchaException` due typo

### 4.10.2 Added

- Added `python_anticaptcha.exceptions.AnticatpchaException`
- Added docs about error handling

### 4.10.3 Removed

- Deprecated `python_anticaptcha.exceptions.AnticatpchaException`

The project is open-source.

Changes are managed through GitHub. Pull requests are particularly welcome.

All changes are automatically tested using TravisCI.

## 5.1 New release

Follow these steps to publish the new release:

- update changelog - use any text editor
- bump version - use ``bumpversion {major,minor,patch}``
- build package - use ``python setup.py sdist bdist_wheel --universal``
- upload release to PyPI - use ``twine upload dist/*``
- push changes to GitHub - ``git push origin && git push --tags``





## CHAPTER 6

---

### Indices and tables

---

- [genindex](#)
- [modindex](#)
- [search](#)



**p**

`python_anticaptcha.base`, 9  
`python_anticaptcha.exceptions`, 10  
`python_anticaptcha.fields`, 13  
`python_anticaptcha.tasks`, 10



## A

AnticaptchaClient (class in *python\_anticaptcha.base*), 9  
 AnticaptchaException, 10  
 AnticaptchaException (in module *python\_anticaptcha.exceptions*), 10  
 APP\_STAT\_URL (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9  
 assignment (*python\_anticaptcha.tasks.CustomCaptchaTask* attribute), 10

## B

BALANCE\_URL (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9  
 BaseField (class in *python\_anticaptcha.fields*), 13  
 BaseTask (class in *python\_anticaptcha.tasks*), 10

## C

case (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11  
 check\_is\_ready() (*python\_anticaptcha.base.Job* method), 10  
 Checkbox (class in *python\_anticaptcha.fields*), 13  
 client (*python\_anticaptcha.base.Job* attribute), 10  
 client\_ip (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9  
 client\_key (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9  
 columnsCount (*python\_anticaptcha.tasks.SquareNetTask* attribute), 12  
 contentType (*python\_anticaptcha.fields.Image* attribute), 13  
 contentType (*python\_anticaptcha.fields.SimpleText* attribute), 13  
 contentType (*python\_anticaptcha.fields.WebLink* attribute), 14  
 CREATE\_TASK\_URL (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9

## F

form (*python\_anticaptcha.tasks.CustomCaptchaTask* attribute), 10  
 fp (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11  
 fp (*python\_anticaptcha.tasks.SquareNetTask* attribute), 12  
 FunCaptchaProxylessTask (class in *python\_anticaptcha.tasks*), 11  
 FunCaptchaTask (class in *python\_anticaptcha.tasks*), 11

## G

get\_answers() (*python\_anticaptcha.base.Job* method), 10  
 get\_captcha\_text() (*python\_anticaptcha.base.Job* method), 10  
 get\_cells\_numbers() (*python\_anticaptcha.base.Job* method), 10  
 get\_choices() (*python\_anticaptcha.fields.Select* method), 13  
 get\_solution\_response() (*python\_anticaptcha.base.Job* method), 10  
 get\_token\_response() (*python\_anticaptcha.base.Job* method), 10  
 getAppStats() (*python\_anticaptcha.base.AnticaptchaClient* method), 9  
 getBalance() (*python\_anticaptcha.base.AnticaptchaClient* method), 9  
 getTaskResult() (*python\_anticaptcha.base.AnticaptchaClient* method), 9

## H

HCaptchaTask (class in *python\_anticaptcha.tasks*), 11  
 HCaptchaTaskProxyless (class in *python\_anticaptcha.tasks*), 11

## I

Image (class in *python\_anticaptcha.fields*), 13  
 ImageToTextTask (class in *python\_anticaptcha.tasks*), 11  
 ImageUpload (class in *python\_anticaptcha.fields*), 13  
 imageUrl (*python\_anticaptcha.tasks.CustomCaptchaTask* attribute), 10  
 InvalidWidthException, 10

## J

Job (class in *python\_anticaptcha.base*), 10  
 join() (*python\_anticaptcha.base.Job* method), 10

## L

label (*python\_anticaptcha.fields.BaseField* attribute), 13  
 labelHint (*python\_anticaptcha.fields.BaseField* attribute), 13  
 language\_pool (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9

## M

math (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11  
 maxLength (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11  
 minLength (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11  
 minScore (*python\_anticaptcha.tasks.RecaptchaV3TaskProxyless* attribute), 12  
 MissingNameException, 10

## N

name (*python\_anticaptcha.fields.NameBaseField* attribute), 13  
 NameBaseField (class in *python\_anticaptcha.fields*), 13  
 NoCaptchaTask (class in *python\_anticaptcha.tasks*), 11  
 NoCaptchaTaskProxylessTask (class in *python\_anticaptcha.tasks*), 12  
 numeric (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11

## O

objectName (*python\_anticaptcha.tasks.SquareNetTask* attribute), 12

## P

pageAction (*python\_anticaptcha.tasks.RecaptchaV3TaskProxyless* attribute), 12  
 phrase (*python\_anticaptcha.tasks.ImageToTextTask* attribute), 11  
 ProxyMixin (class in *python\_anticaptcha.tasks*), 12  
 python\_anticaptcha.base (module), 9  
 python\_anticaptcha.exceptions (module), 10  
 python\_anticaptcha.fields (module), 13  
 python\_anticaptcha.tasks (module), 10

## R

Radio (class in *python\_anticaptcha.fields*), 13  
 recaptchaDataSValue (*python\_anticaptcha.tasks.NoCaptchaTaskProxylessTask* attribute), 12  
 RecaptchaV3TaskProxyless (class in *python\_anticaptcha.tasks*), 12  
 REPORT\_IMAGE\_URL (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9  
 report\_incorrect() (*python\_anticaptcha.base.Job* method), 10  
 report\_incorrect\_image() (*python\_anticaptcha.base.Job* method), 10  
 report\_incorrect\_recaptcha() (*python\_anticaptcha.base.Job* method), 10  
 REPORT\_RECAPTCHA\_URL (*python\_anticaptcha.base.AnticaptchaClient* attribute), 9  
 reportIncorrectImage() (*python\_anticaptcha.base.AnticaptchaClient* method), 9  
 reportIncorrectRecaptcha() (*python\_anticaptcha.base.AnticaptchaClient* method), 10  
 response\_timeout (*python\_anticaptcha.base.AnticaptchaClient* attribute), 10  
 rowCount (*python\_anticaptcha.tasks.SquareNetTask* attribute), 12

## S

Select (class in *python\_anticaptcha.fields*), 13  
 serialize() (*python\_anticaptcha.fields.BaseField* method), 13  
 serialize() (*python\_anticaptcha.fields.Checkbox* method), 13  
 serialize() (*python\_anticaptcha.fields.Image* method), 13  
 serialize() (*python\_anticaptcha.fields.ImageUpload* method), 13  
 serialize() (*python\_anticaptcha.fields.NameBaseField* method), 13  
 serialize() (*python\_anticaptcha.fields.Select* method), 13

*serialize()* (*python\_anticaptcha.fields.SimpleText method*), 13  
*serialize()* (*python\_anticaptcha.fields.Textarea method*), 13  
*serialize()* (*python\_anticaptcha.fields.TextInput method*), 13  
*serialize()* (*python\_anticaptcha.fields.WebLink method*), 14  
*serialize()* (*python\_anticaptcha.tasks.BaseTask method*), 10  
*serialize()* (*python\_anticaptcha.tasks.CustomCaptchaTask method*), 11  
*serialize()* (*python\_anticaptcha.tasks.FunCaptchaProxylessTask method*), 11  
*serialize()* (*python\_anticaptcha.tasks.HCaptchaTaskProxyless method*), 12  
*serialize()* (*python\_anticaptcha.tasks.HCaptchaTaskProxyless method*), 11  
*serialize()* (*python\_anticaptcha.tasks.ImageToTextTask method*), 11  
*serialize()* (*python\_anticaptcha.tasks.NoCaptchaTaskProxylessTask method*), 12  
*serialize()* (*python\_anticaptcha.tasks.ProxyMixin method*), 12  
*serialize()* (*python\_anticaptcha.tasks.RecaptchaV3TaskProxyless method*), 12  
*serialize()* (*python\_anticaptcha.tasks.SquareNetTask method*), 12  
*SimpleText* (*class in python\_anticaptcha.fields*), 13  
*SOFT\_ID* (*python\_anticaptcha.base.AnticaptchaClient attribute*), 9  
*SquareNetTask* (*class in python\_anticaptcha.tasks*), 12

## T

*task\_id* (*python\_anticaptcha.base.Job attribute*), 10  
*TASK\_RESULT\_URL* (*python\_anticaptcha.base.AnticaptchaClient attribute*), 9  
*Textarea* (*class in python\_anticaptcha.fields*), 13  
*TextInput* (*class in python\_anticaptcha.fields*), 13  
*type* (*python\_anticaptcha.fields.Radio attribute*), 13  
*type* (*python\_anticaptcha.fields.Select attribute*), 13  
*type* (*python\_anticaptcha.tasks.CustomCaptchaTask attribute*), 11  
*type* (*python\_anticaptcha.tasks.FunCaptchaProxylessTask attribute*), 11  
*type* (*python\_anticaptcha.tasks.FunCaptchaTask attribute*), 11  
*type* (*python\_anticaptcha.tasks.HCaptchaTask attribute*), 11  
*type* (*python\_anticaptcha.tasks.HCaptchaTaskProxyless attribute*), 11  
*type* (*python\_anticaptcha.tasks.ImageToTextTask attribute*), 11  
*type* (*python\_anticaptcha.tasks.NoCaptchaTask attribute*), 11

## W

*WebLink* (*class in python\_anticaptcha.fields*), 14  
*websiteKey* (*python\_anticaptcha.tasks.FunCaptchaProxylessTask attribute*), 11  
*websiteKey* (*python\_anticaptcha.tasks.HCaptchaTaskProxyless attribute*), 11  
*websiteKey* (*python\_anticaptcha.tasks.NoCaptchaTaskProxylessTask attribute*), 12  
*websiteKey* (*python\_anticaptcha.tasks.RecaptchaV3TaskProxyless attribute*), 12  
*websiteSToken* (*python\_anticaptcha.tasks.NoCaptchaTaskProxylessTask attribute*), 12  
*websiteURL* (*python\_anticaptcha.tasks.FunCaptchaProxylessTask attribute*), 11  
*websiteURL* (*python\_anticaptcha.tasks.HCaptchaTaskProxyless attribute*), 11  
*websiteURL* (*python\_anticaptcha.tasks.NoCaptchaTaskProxylessTask attribute*), 12  
*websiteURL* (*python\_anticaptcha.tasks.RecaptchaV3TaskProxyless attribute*), 12