

---

# YesCaptcha

发布 *0.1.0*

**Germey**

2022 年 07 月 02 日



---

## Contents

---

<b>1</b>	<b>介绍</b>	<b>1</b>
<b>2</b>	<b>安装</b>	<b>3</b>
2.1	准备工作 . . . . .	3
2.2	pip3 安装 . . . . .	3
2.3	验证安装 . . . . .	3
<b>3</b>	<b>使用</b>	<b>5</b>
3.1	前提条件 . . . . .	5
3.2	全部任务类型 . . . . .	6
3.3	基本使用 . . . . .	6
3.4	其他任务 . . . . .	7
3.5	异步任务 . . . . .	8
3.6	非任务接口 . . . . .	8
3.7	节点设置 . . . . .	9
<b>4</b>	<b>作者</b>	<b>11</b>
<b>5</b>	<b>协议</b>	<b>13</b>



本文档用于介绍 YesCaptcha 服务的 Python SDK - yescaptcha-python 的使用方法。

该 SDK 提供了 YesCaptcha 所有服务接口的封装，使得我们可以更方便地使用 YesCaptcha 的相关服务。

本文档具体的内容包括：

- 安装方式
- 基本初始化使用
- 各任务的使用



本文介绍如何安装 YesCaptcha Python SDK。

### 2.1 准备工作

在此之前请确保已经正确安装好 Python3 环境并能正常使用 pip3 命令，详情请查看：[安装说明](#)。

### 2.2 pip3 安装

安装好 Python3 环境之后，我们就可以使用 pip3 来安装 yescaptcha-python 了，命令如下：

```
pip3 install yescaptcha
```

### 2.3 验证安装

安装完毕之后，尝试使用 Python 运行如下代码：

```
from yescaptcha.client import Client
```

如果没有发生报错，则证明相关 SDK 已经安装成功。



本文档介绍 YesCaptcha SDK 的基本使用方法。

### 3.1 前提条件

在使用之前，请到 <https://yescaptcha.com/> 官网注册账号，并获取账户密钥 ClientKey，如图所示：



## 3.2 全部任务类型

YesCaptcha 提供了多种验证码识别服务，每种服务对应一种任务，所有的任务类型参见[官方文档](#)。

此 Python SDK 提供了对各种任务的封装，使得开发者可以更方便地对接 YesCaptcha 服务，而不用从 0 开始使用 requests 等库来进行开发。

## 3.3 基本使用

这里拿 GoogleCaptchaV2 验证码来做样例，样例地址为<https://www.google.com/recaptcha/api2/demo>，具体的使用说明参见 NoCaptchaTaskProxyless : reCaptcha V2 协议接口。

样例代码如下：

```
from yescaptcha.task import NoCaptchaTaskProxyless
from yescaptcha.client import Client

CLIENT_KEY = <YOUR_CLIENT_KEY>
website_url = 'https://www.google.com/recaptcha/api2/demo'
website_key = '6Le-wvkSAAAAAPBMRTvw0Q4Muxeq9bi0DJwx_mJ-'

client = Client(client_key=CLIENT_KEY, debug=True)
task = NoCaptchaTaskProxyless(website_key=website_key, website_url=website_url)
job = client.create_task(task)
print('result', job.get_solution())
```

首先，我们需要引入 Client 类，用于初始化一个 YesCaptcha 服务的操作对象，这时候我们需要传入 client\_key，即前文所描述的账户密钥 ClientKey，另外我们可以传入 debug 参数，设置调试模式，这样在运行过程中可以输出更多调试信息。

另外还需要引入 NoCaptchaTaskProxyless 类，类的名称等同于文档中所提及的 type，如图所示：

### 请求示例

```
1 {
2   "clientKey": "cc9c18d3e263515c2c072b36a7125eccc078618f",
3   "task": {
4     "websiteURL": "https://www.google.com/recaptcha/api2/demo",
5     "websiteKey": "6Le-wvkSAAAAAPBMRTvw0Q4Muxeq9bi0DJwx_mJ-",
6     "type": "NoCaptchaTaskProxyless"
7   }
8 }
```



另外，阅读文档我们可以发现此类别的 Task 需要两个参数 websiteURL 和 websiteKey，如图所示：

## 对象结构

属性	类型	必须	说明
type	string	是	<b>NoCaptchaTaskProxyless</b> 15 POINTS <small>插件默认</small> <b>RecaptchaV2TaskProxyless</b> 20 POINTS
websiteURL	string	是	ReCaptcha 网页地址, 一般固定值。 <a href="#">如何找到websiteURL值</a>
websiteKey	String	是	ReCaptcha 网站密钥, 固定值。 <a href="#">如何找到websiteKey值</a>

所以, 在初始化 `NoCaptchaTaskProxyless` 类的时候我们需要传入对应的两个参数, 参数需要转成蛇形命名法, 单词与单词之间需要以下划线分隔, 即转为 `website_url` 和 `website_key`。

接着, 我们可以使用 `client` 对象的 `create_task` 方法来创建一个任务, 其返回值命名为 `job`, 最后, 我们调用 `get_solution` 方法就可以获取对应的结果了。

运行结果如下:

```
2022-07-02 14:30:38.124 | DEBUG | yescaptcha.client:create_task:61 - construct_
↳ result {'clientKey': '50a07aa563cf4688270a6a968086c80b4ef23baf78', 'task': {'type':
↳ 'NoCaptchaTaskProxyless', 'websiteURL': 'https://www.google.com/recaptcha/api2/demo
↳ ', 'websiteKey': '6Le-wvkSAAAAAPBMRTvw0Q4MueXq9bi0DJwx_mJ-'}}
2022-07-02 14:30:39.435 | DEBUG | yescaptcha.job:_update:34 - try to get result of_
↳ task 7cf17d74-f9d0-11ec-b008-ae8dd81878b9
2022-07-02 14:30:40.527 | DEBUG | yescaptcha.job:_update:36 - task result {'errorId
↳ ': 0, 'errorCode': '', 'errorDescription': None, 'status': 'processing', 'solution':
↳ None}
2022-07-02 14:30:41.528 | DEBUG | yescaptcha.job:_update:34 - try to get result of_
↳ task 7cf17d74-f9d0-11ec-b008-ae8dd81878b9

2022-07-02 14:31:36.359 | DEBUG | yescaptcha.job:_update:36 - task result {'errorId
↳ ': 0, 'errorCode': None, 'errorDescription': None, 'solution': {'gRecaptchaResponse
↳ ': '03AGdBq27gCf7UkBF07Ns150qK6kiL911UYPYsyrbbYh9jQfvzp1qrsehQ62UKgJ...
↳ 2SNtNtWfJJ8TkXJs4AHIceSmE-JR8BSs_m--Qf5MtySwRPZnYewbKBOSOSHBCQarMPBFUj-
↳ dscgbI8rlfIpbG0hbgrF-MDJ3aHI6ZdgrahHt1RpB6PwDdWCXX1S1w'}
result_
↳ 03AGdBq27gCf7UkBF07Ns150qK6kiL911UYPYsyrbbYh9jQfvzp1qrsehQ62UKgJB9s9x5bKk2pEg392DYy3HVTo1KbdBywk7U
↳ ..HBCQarMPBFUj-dscgbI8rlfIpbG0hbgrF-MDJ3aHI6ZdgrahHt1RpB6PwDdWCXX1S1w
```

## 3.4 其他任务

对于其他任务来说, 使用方法类似, 我们只需要参照文档, 引入对应的 `Task` 对象即可, 比如对于 `ReCaptchaV2Classification`: reCaptcha V2 图像识别, 我们需要引入的就是:

```
from yescaptcha.task import ReCaptchaV2Classification
```

在创建 `Task` 的时候同样参考文档的参数说明:

```
task = ReCaptchaV2Classification(image='<image payload>', question='<question>')
```

这里的 image 就是 Base64 编码的图片, 不要包含 “data:image/\*\*\*;base64,” , question 参数就是问题 ID, 请查表, 以 /m/ 开头。

最后调用方式是一样的:

```
job = client.create_task(task)
print('result', job.get_solution())
```

### 3.5 异步任务

对于某些任务来说, 其验证码的识别结果不会立马返回, 而是需要有一个处理过程, 比如 NoCaptchaTaskProxyless、RecaptchaV3TaskProxyless 等任务。而某些图像识别的任务, 其识别过程则会同步的, 如 ReCaptchaV2Classification、HCaptchaClassification 等任务。

为了方便开发者开发, 对于异步任务, 本 SDK 做了自动等待处理, 也就是会自动轮询等待一直到识别结果返回, 原因是因为在初始化 Client 对象的时候, 有一个默认参数 auto\_join 为 True。

所以, 如果你不想开启自动等待处理, 可以手动处理这个过程, 这时候需要在初始化 Client 对象的时候将 auto\_join 设置为 False, 然后在合适的时候调用 job 对象的 join 方法, 写法如下:

```
from yescaptcha.task import NoCaptchaTaskProxyless
from yescaptcha.client import Client

CLIENT_KEY = <YOUR_CLIENT_KEY>
website_url = 'https://www.google.com/recaptcha/api2/demo'
website_key = '6Le-wvkSAAAAAPBMRTvw0Q4Muxq9bi0DJwx_mJ-'

client = Client(client_key=CLIENT_KEY, auto_join=False)
task = NoCaptchaTaskProxyless(website_key=website_key, website_url=website_url)
job = client.create_task(task)
job.join()
print('result', job.get_solution())
```

### 3.6 非任务接口

另外, 本 SDK 还提供了一些非验证码识别的接口, 比如获取账户信息、获取开发者 ID 等, 列举如下:

- get\_balance: 获取账户余额
- get\_soft\_id: 获取开发者 ID

比如对于 get\_soft\_id, 用法如下:

```
from yescaptcha.client import Client

CLIENT_KEY = <YOUR_CLIENT_KEY>
client = Client(client_key=CLIENT_KEY)
soft_id = client.get_soft_id()
print('result', soft_id)
```

## 3.7 节点设置

YesCaptcha 服务背后有两个服务节点，一个是国际节点（默认），但在中国区访问可能会出现問題，所以对于中国区的用户，可以专门设置一个中国节点。

设置的时候只需要更改 Client 对象的初始化参数 region 即可：

```
from yescaptcha.client import Client, Region

CLIENT_KEY = <YOUR_CLIENT_KEY>
client = Client(client_key=CLIENT_KEY, region=Region.CHINA)
```



## CHAPTER 4

---

作者

---

- YesCaptcha
  - 主页: <https://yescaptcha.com/>



---

### 协议

---

本项目基于 MIT 协议。

版权所有 (c) 2022 - 2030, YesCaptcha

特此向任何得到本软件副本或相关文档的人授权：被授权人有权使用、复制、修改、合并、出版、发布、散布、再授权和/或贩售软件及软件的副本，及授予被供应人同等权利，只需服从以下义务：

在软件和软件的所有副本中都必须包含以上版权声明和本许可声明。

该软件是”按原样”提供的，没有任何形式的明示或暗示，包括但不限于为特定目的和不侵权的适销性和适用性的保证担保。在任何情况下，作者或版权持有人，都无权要求任何索赔，或有关损害赔偿的其他责任。无论在本软件的使用上或其他买卖交易中，是否涉及合同，侵权或其他行为。