**模拟登录**

[云打码官网](https://zhuce.jfbym.com/demo/)

**云打码自定义包**

import json

import time

import requests

import base64

class YdmVerify(object):

\_custom\_url = "https://www.jfbym.com/api/YmServer/customApi"

\_token = "$"#用户中心密钥(自己的)

\_headers = {

'Content-Type': 'application/json'

}

def common\_verify(self, image, verify\_type="10110"):

# 数英汉字类型

# 通用数英1-4位 10110

# 通用数英5-8位 10111

# 通用数英9~11位 10112

# 通用数英12位及以上 10113

# 通用数英1~6位plus 10103

# 定制-数英5位~qcs 9001

# 定制-纯数字4位 193

# 中文类型

# 通用中文字符1~2位 10114

# 通用中文字符 3~5位 10115

# 通用中文字符6~8位 10116

# 通用中文字符9位及以上 10117

# 中文字符 1~4位 plus 10118

# 定制-XX西游苦行中文字符 10107

# 计算类型

# 通用数字计算题 50100

# 通用中文计算题 50101

# 定制-计算题 cni 452

payload = {

"image": base64.b64encode(image).decode(),

"token": self.\_token,

"type": verify\_type

}

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def slide\_verify(self, slide\_image, background\_image, verify\_type="20101"):

# 滑块类型

# 通用双图滑块 20111

payload = {

"slide\_image": base64.b64encode(slide\_image).decode(),

"background\_image": base64.b64encode(background\_image).decode(),

"token": self.\_token,

"type": verify\_type

}

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def sin\_slide\_verify(self, image, verify\_type="20110"):

# 通用单图滑块(截图) 20110

payload = {

"image": base64.b64encode(image).decode(),

"token": self.\_token,

"type": verify\_type

}

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def traffic\_slide\_verify(self, seed, data, href, verify\_type="900010"):

# 定制-滑块协议slide\_traffic 900010

payload = {

"seed": seed,

"data": data,

"href": href,

"token": self.\_token,

"type": verify\_type

}

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def click\_verify(self, image, extra=None, verify\_type="30100"):

# 通用任意点选1~4个坐标 30009

# 通用文字点选1(通用,xd;extra,点选文字逗号隔开,原图) 30100

# 定制-文字点选2(xy3,extra="click",原图) 30103

# 定制-单图文字点选(xd) 30102

# 定制-图标点选1(xd,原图) 30104

# 定制-图标点选2(xy3,原图,extra="icon") 30105

# 定制-语序点选1(xy3,原图,extra="phrase") 30106

# 定制-语序点选2(xd,原图) 30107

# 定制-空间推理点选1(xd,原图,extra="请点击xxx") 30109

# 定制-空间推理点选1(xy3,原图,extra="请\_点击\_小尺寸绿色物体。") 30110

# 定制-tx空间点选(extra="请点击侧对着你的字母") 50009

# 定制-tt\_空间点选 30101

# 定制-推理拼图1(xd,原图,extra="交换2个图块") 30108

# 定制-xy4九宫格点选(label\_image,image) 30008

# 点选二字TX 30111

# 定制-文字点选3(extra="je4\_click") 30112

# 定制-图标点选3(extra="je4\_icon") 30113

# 定制-语序点选3(extra="je4\_phrase") 30114

payload = {

"image": base64.b64encode(image).decode(),

"token": self.\_token,

"type": verify\_type

}

if extra:

payload['extra'] = extra

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def rotate(self, image):

# 定制-X度单图旋转 90007

payload = {

"image": base64.b64encode(image).decode(),

"token": self.\_token,

"type": "90007"

}

# 定制-Tt双图旋转,2张图,内圈图,外圈图 90004

# payload = {

# "out\_ring\_image": base64.b64encode(image).decode(),

# "inner\_circle\_image": base64.b64encode(image).decode(),

# "token": self.\_token,

# "type": "90004"

# }

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def google\_verify(self, googlekey, pageurl, invisible=1, data\_s=""):

\_headers = {

'Content-Type': 'application/json'

}

"""

第一步，创建验证码任务

:param

:return taskId : string 创建成功的任务ID

"""

url = "https://www.jfbym.com/api/YmServer/funnelApi"

payload = json.dumps({

"token": self.\_token,

"type": "40010", ## v2

# "type": "40011", ## v3

"googlekey": googlekey,

"enterprise": 0, ## 是否为企业版

"pageurl": pageurl,

"invisible": invisible,

"data-s": data\_s, ## V2+企业如果能找到，找不到传空字符串

# 'action':"" #V3必传

# 'min\_score':"" #V3才支持的可选参数

})

# 发送JSON格式的数据

result = requests.request("POST", url, headers=\_headers, data=payload).json()

print(result)

# {'msg': '识别成功', 'code': 10000, 'data': {'code': 0, 'captchaId': '51436618130', 'recordId': '74892'}}

captcha\_id = result.get('data').get("captchaId")

record\_id = result.get('data').get("recordId")

times = 0

is\_solved = 0

while times < 150:

try:

url = f"https://www.jfbym.com/api/YmServer/funnelApiResult"

data = {

"token": self.\_token,

"captchaId": captcha\_id,

"recordId": record\_id

}

result = requests.post(url, headers=\_headers, json=data).json()

print(result)

# {'msg': '结果准备中，请稍后再试', 'code': 10009, 'data': []}

if result['msg'] == "结果准备中，请稍后再试":

continue

if result['msg'] == '请求成功' and result['code'] == 10001:

is\_solved = 1

return result['data']['data']

# {'msg': '请求成功', 'code': 10001, 'data': {'data': '03AGdBq2611GTOgA2v9HUpMMEUE70p6dwOtYyHJQK4xhdKF0Y8ouSGsFZt647SpJvZ22qinYrm6MYBJGFQxMUIApFfSBN6WTGspk6DmFdQAoWxynObRGV7qNMQOjZ\_m4w3\_6iRu8SJ3vSUXH\_HHuA7wXARJbKEpU4J4R921NfpKdahgeFD8rK1CFYAqLd5fz4l-8\_VRmRE83dRSfkgyTN338evQ1doWKJRipZbk4ie-89Ud0KGdOsP4QzG3stRZgj2oaEoMDSAP62vxKGYqtDEqTcwtlgo-ot3rF5SmntaoKGwcKPo0NrekWA5gtj0vqKLU6lY2GcnSci\_tgBzBwuH40uvyR1PFu02VK\_E44mopJ7FOO4cUukNaLGqypU2YCA8QuaaebOIoCMU7RGqGs\_41RYNCG1GSdthiwcwk2hHFbi-TXuICXSwh4Er5mgVW9A3t\_9Ndp0eJcyr3HtuJrcA7BtlcgruuQxK5h4Ew4ert4KPH\_aQGN9ww5VsUtbSManzUDnUOs7aEdvFk1DOOPmLys-aX20ZFN2CcQcZZSO-7HZpZZt3EDeWWE5S02HFDY8gl3\_0xqIts8774Tr4GMVJaddG0NR6pcBFC11FqNcK2a18gM3gaKDy3\_2ZMeSU4nj4NWwoAhPjQN2BS8JxX4kKVpX4rD959kc93vczVD3TYD6\_4GJahGSpBvM7Y5\_GGIdLL8imXde1R35mZnEcFYXQ40zcy3DdJFkk\_gzGTVOEb1Q1IZpjMxzCxyGgwjgL9dtDIgst5H5CSZoerX\_Lz-DmsBvYIYZdpbPLEMROx9MODImaEw8Cp6M8Xj7\_foijiGE9hh-pzJSTlKl3HytiSUyJJ7r1BssrX5C\_TFWxl0IXNg8azP8H-ZIOWwnYlMWCS1w9piHdoLg5zACiYIN3Txdlsvi61MuPmzJggJd1\_dlyMdAlzb5\_zdfweqj0\_Ko1ODP378YT7sV7LECgRj5QJU6sF5nlf4m2g5sFypBw9GFAkEE-OaWGYxRJOy2ioU41ggAJIkcza2B\_N5AL2KLROtm0-c2MxplM4ZzHxrUv9A24zlgzo3Pz4NONwU\_gaOcDB7j1dZKXD8UaoIrZv0BTd8JeojYowm9Usdg7Rt4Fpo\_vDLJdrEUfbxVlXieDD9Fr1fu72-d4AduT\_J3n-rIhyX4gFav-KfP-qOxqOZsmjXZirsBxZs7042NYeirRYnLv35cxIAJARz03FJmeKViUivwC5mCWw64hjRad9XyyBOP2n8KFOrTXhPskC-WwEfksGtfLxi6VW76FHGvRdwHXzMwVfNqe3P5H\_WZUc-vxeTAsTnqZz3WA97lM4MLrX0nTZYgXxCEiS6raSOiEMqcx\_Nv7Zxre-abj4LZRbFpH8nx1SEiaOV2Dm-a1iPFEmCs0L4kDtt6VImSVIQaTOAd3KFSo7W\_XTvRPsQJOtblrcKyuagztX\_Yr0lT0YqN9I9MZAARo7M5OfwSLJW16rdmp4NuRefEvNPNHO2cVh1Xha1qNGuF\_QDvWFFmWG0Y6IbRqLmF-Dv8BY4TWyOeVnADJftGQw2QSr8RmbCHryA'}}

except Exception as e:

print(e)

finally:

if is\_solved:

break

print("sleep 5s...")

time.sleep(5)

times += 5

def hcaptcha\_verify(self, site\_key, site\_url, verify\_type="50001"):

# 定制类接口-Hcaptcha

payload = {

"site\_key": site\_key,

"site\_url": site\_url,

"token": self.\_token,

"type": verify\_type

}

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

def fun\_captcha\_verify(self, publickey, pageurl, verify\_type="40007"):

# 定制类接口-Hcaptcha

payload = {

"publickey": publickey,

"pageurl": pageurl,

"token": self.\_token,

"type": verify\_type

}

resp = requests.post(self.\_custom\_url, headers=self.\_headers, data=json.dumps(payload))

print(resp.text)

return resp.json()['data']['data']

if \_\_name\_\_ == '\_\_main\_\_':

Y = YdmVerify()

with open('2.png', 'rb') as f:

s = f.read()

Y.click\_verify(image=s, extra="家,炉,私")