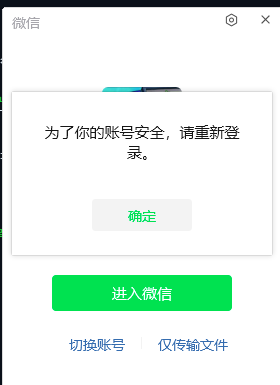
**Pywechat操作手册**

## pywechat架构图

**图1 pywechat结构**

# 声明

1在使用pywechat进行微信自动化操作时，若操作过于频繁，微信会自动退出登录，此为微信正常检测机制，只要不违反微信使用协议(高频率大量发送非法信息)，不会造成封号。此类情况无法避免，请用户自行注意操作频率,对此，你可以将各项函数内的delay参数设置为合适的数值,通常为1s即可。



2. pywechat未经作者同意不得用于任何商业用途或非法活动，否则后果自负。

3.使用pywechat前，请确保你已经了解并遵守微信的相关规定和政策。

**目录**

[声明 1](#_Toc6015)

[获取方式 5](#_Toc16446)

[模块导入 5](#_Toc17437)

[WechatAuto 6](#_Toc16477)

[1 Messages类 6](#_Toc28014)

[1.1给单个好友发送单条消息 send\_message\_to\_friend() 6](#_Toc5240)

[1.2 给单个好友发送多条消息send\_messages\_to\_friend() 7](#_Toc6550)

[1.3 给多个好友分别发送单条消息send\_message\_to\_friends() 8](#_Toc19643)

[1.4 给多个好友分别发送多条消息send\_messages\_to\_friends() 9](#_Toc6204)

[1.5 转发消息forward\_message() 10](#_Toc16797)

[1.6 检查(监听)新消息check\_new\_message() 11](#_Toc12736)

[2 Files类 12](#_Toc11443)

[2.1 给单个好友发送单个文件send\_file\_to\_friend() 12](#_Toc26932)

[2.2 给单个好友发送多个文件send\_files\_to\_friend() 13](#_Toc15701)

[2.3 给多个好友分别发送单个文件send\_file\_to\_friends() 15](#_Toc12018)

[2.4给多个好友分别发送多个文件send\_files\_to\_friends() 16](#_Toc12113)

[2.5转发文件forward\_file() 17](#_Toc4353)

[3 Contacts类 18](#_Toc23607)

[3.1获取所有好友的备注与名称get\_friends\_names() 18](#_Toc8118)

[3.2获取所有好友昵称备注与微信号get\_friends\_info() 19](#_Toc17413)

[3.3获取所有好友详细信息get\_friends\_detail() 20](#_Toc12328)

[3.4获取所有群聊信息get\_groups\_info() 20](#_Toc12349)

[3.5获取群内所有群成员群昵称get\_groupmembers\_info() 21](#_Toc13169)

[3.6获取所有企业微信好友信息get\_wecomfriends\_info() 22](#_Toc27822)

[4 FriendSettings类 23](#_Toc20694)

[4.1获取好友聊天记录 get\_chat\_history() 23](#_Toc11124)

[4.2 添加好友至黑名单 add\_friend\_to\_blacklist() 24](#_Toc8775)

[4.3 添加新朋友 add\_new\_friend() 25](#_Toc5624)

[4.4 好友会话内置顶 pin\_friend() 26](#_Toc11740)

[4.5 好友置顶 sticky\_friend\_on\_top() 26](#_Toc23288)

[4.6 将好友设置为星标朋友 set\_friend\_as\_star\_friend() 27](#_Toc19513)

[4.7 删除好友 delete\_friend() 28](#_Toc18920)

[4.8 清空好友聊天记录 clear\_friend\_chat\_history() 29](#_Toc27384)

[4.9 好友消息免打扰 mute\_friend\_notifications() 30](#_Toc20000)

[4.10 拍一拍好友 tickle\_friend() 30](#_Toc21392)

[4.11 推荐好友给朋友 share\_contact 31](#_Toc28853)

[4.12 修改好友备注和标签 change\_friend\_remark\_and\_tag() 32](#_Toc8790)

[4.13获取单个好友微信号get\_friend\_wechatnumber 33](#_Toc32246)

[4.14获取多个好友微信号get\_friends\_wechatnumbers 34](#_Toc13951)

[4.15 修改好友权限 change\_friend\_privacy() 35](#_Toc5082)

[5 GroupSettings类 36](#_Toc27478)

[5.1获取群聊聊天记录get\_chat\_history() 36](#_Toc18666)

[5.2 保存群聊至通讯录 save\_group\_to\_contacts() 37](#_Toc11268)

[5.3从群聊中添加新朋友add\_new\_friend\_from\_group() 38](#_Toc22377)

[5.4 群聊会话内置顶 pin\_group() 39](#_Toc16016)

[5.5 群聊置顶 sticky\_group\_on\_top() 39](#_Toc22215)

[5.6 修改群聊名称 change\_group\_name()#管理员权限 40](#_Toc18447)

[5.7 修改群聊备注change\_group\_remark() 41](#_Toc7586)

[5.8 修改我的群昵称 change\_my\_nickname\_in\_group() 42](#_Toc23318)

[5.9 发布群公告 edit\_group\_notice()#管理员权限 43](#_Toc30949)

[5.10 退出群聊 quit\_group() 44](#_Toc10982)

[5.11 清空群聊聊天记录 clear\_group\_chat\_history() 44](#_Toc2244)

[5.12 群聊消息免打扰 mute\_group\_notifications() 45](#_Toc4344)

[5.13 邀请好友进入群聊 invite\_others\_to\_group() 46](#_Toc23201)

[5.14 创建新群聊 create\_new\_group() 47](#_Toc16328)

[5.15 踢出群成员 remove\_friend\_from\_group()#管理员权限 48](#_Toc20259)

[5.16 显示群成员昵称 show\_group\_memebers\_nickname() 48](#_Toc5390)

[6 AutoReply类 49](#_Toc5642)

[6.1自动回复指定好友消息 auto\_reply\_to\_friend() 49](#_Toc20176)

[6.2 自动回复消息 auto\_reply\_messages() 50](#_Toc5499)

[6.3 自动接听电话 auto\_answer\_call() 51](#_Toc16881)

[6.4 接入AI自动回复消息 AI\_auto\_reply\_to\_friend() 52](#_Toc221)

[7 WechatSettings类 53](#_Toc26141)

[7.1 适配系统缩放比例Adapt\_to\_PC\_display\_scalling() 53](#_Toc17284)

[7.2 聊天中的语音消息自动转成文字 Auto\_convert\_voice\_messages\_to\_text() 54](#_Toc27211)

[7.3 有更新时自动更新微信 Auto\_update\_wechat() 55](#_Toc17844)

[7.4 修改截取屏幕快捷键 Change\_capture\_screen\_shortcut() 55](#_Toc24112)

[7.5 修改锁定微信快捷键 Change\_lock\_wechat\_shortcut() 56](#_Toc1734)

[7.6 修改打开微信快捷键 Change\_open\_wechat\_shortcut() 57](#_Toc8273)

[7.7 修改发送消息快捷键 Change\_send\_message\_shortcut() 58](#_Toc7546)

[7.8 清空所有聊天记录clear\_chat\_history() 58](#_Toc32149)

[7.9 关闭PC微信自动登录Close\_auto\_log\_in() 59](#_Toc30639)

[7.10 PC微信退出登录Log\_out() 60](#_Toc5773)

[7.11 快捷键恢复默认设置Shortcut\_default() 60](#_Toc15650)

[7.12视频号通知标记 Channel\_notification\_flag() 61](#_Toc32710)

[7.13 看一看通知标记 Topstories\_notification\_flag() 62](#_Toc5347)

[7.14 朋友圈通知标记 Moments\_notification\_flag() 62](#_Toc32225)

[7.15 小程序通知标记 Miniprogram\_notification\_flag() 63](#_Toc19352)

[7.16 新消息通知声音 New\_message\_alert\_sound() 64](#_Toc844)

[7.17 语音和视频通话通知声音 Voice\_and\_video\_calls\_alert\_sound() 65](#_Toc2259)

[7.18 保留聊天记录Save\_chat\_history() 65](#_Toc1225)

[7.19 显示网络搜索历史Show\_web\_search\_history() 66](#_Toc29688)

[7.20 使用系统默认浏览器打开网页Using\_default\_browser() 67](#_Toc8999)

[8 Call类 68](#_Toc28049)

[8.1与好友语音通话voice\_call() 68](#_Toc23520)

[8.2 与好友视频通话video\_call() 68](#_Toc26623)

[8.3 群聊内发起通话voice\_call\_in\_group() 69](#_Toc16240)

[WechatTools 70](#_Toc739)

[Tools类 70](#_Toc25510)

[1.1 查找微信路径 find\_wechat\_path() 70](#_Toc26303)

[1.2 添加微信路径至windows用户环境变量set\_wechat\_as\_environ\_path() 71](#_Toc14653)

[1.3 打开微信open\_wechat() 71](#_Toc2710)

[1.4 打开微信设置界面open\_settings() 72](#_Toc26496)

[1.5 打开通讯录open\_contacts() 73](#_Toc7062)

[1.6 打开通讯录设置界面open\_contacts\_settings() 73](#_Toc9669)

[1.7 打开收藏界面open\_collections() 74](#_Toc16148)

[1.8 打开朋友圈界面open\_wechat\_moments() 74](#_Toc2560)

[1.9 打开聊天文件界面open\_chat\_files() 75](#_Toc26976)

[1.10 打开搜一搜界面open\_search() 76](#_Toc15557)

[1.11 打开看一看界面open\_top\_stories() 76](#_Toc6201)

[1.12 打开视频号界面open\_channels() 77](#_Toc18023)

[1.13 打开小程序面板open\_program\_pane() 78](#_Toc30188)

[1.14 打开与好友或群聊的聊天界面open\_dialog\_window() 79](#_Toc1570)

[1.15 打开好友设置界面open\_friend\_settings() 79](#_Toc21091)

[1.16 打开好友设置界面菜单open\_friend\_settings\_menu() 80](#_Toc11112)

[1.17 打开好友个人简介面板open\_friend\_profile() 81](#_Toc2477)

[1.18 打开群聊设置界面open\_group\_settings() 81](#_Toc2767)

[API类 82](#_Toc27592)

[1.1 打开指定微信小程序open\_wechat\_miniprogram() 82](#_Toc13103)

[1.2 打开指定微信公众号open\_wechat\_official\_account() 83](#_Toc6066)

[1.3 打开视频号并搜索指定内容search\_channels() 84](#_Toc3313)

[Clock 84](#_Toc3085)

[Schedule 84](#_Toc23714)

[Schtasks 86](#_Toc472)

[1.1 创建定时任务 schtasks.create\_task() 86](#_Toc2881)

[1.2 取消创建过的定时任务 schtasks.cancel\_task() 87](#_Toc2580)

[1.3 修改创建过的定时任务 schtasks.change\_task() 87](#_Toc27111)

# 获取方式

下载最新版本

pip install pywechat127==1.8.7

更新为最新版本

pip install *--*upgrade pywchat127

# 模块导入

Pywechat可以用来自动化微信的模块只有WechatTools WechatAuto

因此在使用文档中我们主要介绍这两个模块,同时考虑到有用户可能需要用到定时微信自动化的功能,对此我在pywechat内增加了Clock模块用来实现定时操作,该模块也将在本文档中介绍到。

直接从pywechat导入模块内的各个类或函数

from pywechat import auto\_reply\_messages

auto\_reply\_messages(duration='1min',content='你好我不在，我的微信由我的微信机器人接管,请您稍后再试！')

直接导入pywechat结构图中的5个主要模块，使用**模块名.函数名**方式使用各项函数

import pywechat.WechatAuto as wechat

wechat.auto\_reply\_messages(duration='1min',content='你好我不在，我的微信由我的微信机器人接管,请您稍后再试！')

导入模块内的类，使用**类名.函数名方式**使用各项方法

from pywechat.WechatAuto import AutoReply

AutoReply.auto\_reply\_messages(duration='1min',content='你好我不在，我的微信由我的微信机器人接管,请您稍后再试！')

# WechatAuto

WechatAuto是pywechat自动化操作微信的主模块,其内部共计包含7个类与74个函数用来实现微信内置功能。本模块内所有函数与方法同名。

## 1 Messages类

### 1.1给单个好友发送单条消息 send\_message\_to\_friend()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friend | 待发送消息者备注 | str |
| message | 发送的消息 | str |
| delay | 输入消息后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | bool默认值:False |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面进行后续自动化操作 | Int 默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Messages

Messages.send\_message\_to\_friend(friend='文件传输助手',message='你好',delay=0.2,tickle=False,search\_pages=0)

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_message\_to\_friend(friend='文件传输助手',message='你好',delay=0.2,tickle=False,search\_pages=0)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手的聊天界面，向其发送message指定内容。**

### 1.2 给单个好友发送多条消息send\_messages\_to\_friend()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friend | 待发送消息者备注 | str |
| messages | 所有要发送的消息列表 | List[str] |
| delay | 输入消息后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | bool默认值:False |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面给好友发消息 | Int 默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Messages

Messages.send\_messages\_to\_friend(friend='文件传输助手',messages=['你好','干嘛呢？'],delay=0.2,tickle=False,search\_pages=0)

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_messages\_to\_friend(friend='文件传输助手',messages=['你好','干嘛呢？'],delay=0.2,tickle=False,search\_pages=0)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手的聊天界面，向其发送messages指定内容。**

### 1.3 给多个好友分别发送单条消息send\_message\_to\_friends()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friends | 所有待发送消息者备注列表 | List[str] |
| message | 给各个发送者分别发送的单条消息 | List[str] |
| delay | 输入消息后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | List[boo]l默认值:List[] 默认值状态下不会拍一拍好友 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Messages

Messages.send\_message\_to\_friends(friends=['文件传输助手','测试ing'],message=['给文件传输助手发消息','给测试ing发消息'],tickle=[False,True],delay=0.2)

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_message\_to\_friends(friends=['文件传输助手','测试ing'],message=['给文件传输助手发消息','给测试ing发消息'],tickle=[False,True],delay=0.2)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手和其他好友的聊天界面，向其发送message内指定内容,并且拍一拍第二个好友。**

### 1.4 给多个好友分别发送多条消息send\_messages\_to\_friends()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friends | 所有待发送消息者的备注列表 | List[str] |
| messages | 给各个发送者分别发送的单条消息 | List[[str]] |
| delay | 输入消息后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | List[boo]l默认值:List=[]，默认值状态下不会拍一拍好友 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Messages

Messages.send\_messages\_to\_friends(friends=['文件传输助手','测试ing'],messages=[['你好','干嘛呢?'],['你好','干嘛呢?']],delay=0.2)

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_messages\_to\_friends(friends=['文件传输助手','测试ing'],messages=[['你好','干嘛呢?'],['你好','干嘛呢?']],delay=0.2)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手和其他好友的聊天界面，向其发送messages内指定内容。**

### 1.5 转发消息forward\_message()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friends | 所有接受转发消息者的备注 | List[str] |
| message | 待转发消息 | Str |
| delay | 输入消息后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| Search\_pages | 打开第一个好友的聊天界面时在消息列表中查找页数一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索第一个好友备注后打开其聊天界面进行后续操作 | Int  默认值:5 |
| tickle | 是否拍一拍好友 | List[boo]l默认值:List=[] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Messages

Messages.forward\_message(friends=['文件传输助手','测试ing'],message='通知:测试转发消息，望周知，收到请回复',search\_pages=0)

**或者**

import pywechat.WechatAuto as wechat

wechat.forward\_message(friends=['文件传输助手','测试ing'],message='通知:测试转发消息，望周知，收到请回复',search\_pages=0)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手的聊天界面，向其发送message内指定内容后右键该消息，点击转发后在选择联系人面板搜索其余好友分别发送待转发消息。**

### 1.6 检查(监听)新消息check\_new\_message()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| duration | 监听消息列表内新消持续时间,单位:’s’,‘min’,’h’ | Str默认值None  若不传入duration只检查一遍，传入后在指定时间内一直检查新消息，可以实现监听新消息功能。 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import check\_new\_message

print(check\_new\_message(duration='1min'))

**或者**

iimport pywechat.WechatAuto as wechat

print(wechat.check\_new\_message(duration='1min'))

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输住手的聊天界面(为了避免当前好友的聊天窗口的新消息无法记录),若你指定了duration，那么会在指定的duration内一直遍历消息列表,每遍历一次会停顿一小段时间,结束后会打开所有右上角头顶有红色新消息标记的好友聊天界面,依次记录新消息内容。若你未指定duration,只遍历一次消息列表,检查一遍是否有新消息,然后退出。**

## 2 Files类

### 2.1 给单个好友发送单个文件send\_file\_to\_friend()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friend | 待发送消息者备注 | str |
| file\_path | 发送的文件路径 | str |
| with\_messages | 是否顺便发送消息 | bool 默认值:False |
| messages | 待发送的消息 | List[str] |
| messages\_first | 是否先发送消息 | bool默认值:False |
| delay | 输入消息或文件后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | bool默认值:False |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作 | Int 默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Files

Files.send\_file\_to\_friend(friend='文件传输助手',file\_path="E:\OneDrive\Desktop\文件发送测试.jpg",delay=0.2)

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_file\_to\_friend(friend='文件传输助手',file\_path="E:\OneDrive\Desktop\文件发送测试.jpg",delay=0.2)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手的聊天界面，向其发送file\_path下的文件.**

### 2.2 给单个好友发送多个文件send\_files\_to\_friend()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friend | 待发送消息者备注 | str |
| folder\_path | 发送的所有文件所在文件夹路径。为了方便,再给好友发多个文件时,程序会遍历该文件夹，将文件夹下所有文件发送给指定好友。 | str |
| with\_messages | 是否顺便发送消息 | bool 默认值:False |
| messages | 待发送的消息列表 | List[str] |
| messages\_first | 是否先发送消息 | bool默认值:False |
| delay | 输入消息或文件后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | bool默认值:False |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面进行后续操作 | Int 默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Files

Files.send\_files\_to\_friend(friend='文件传输助手',folder\_path=r"E:\OneDrive\Desktop\文件发送测试",delay=0.2)

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_files\_to\_friend(friend='文件传输助手',folder\_path=r"E:\OneDrive\Desktop\文件发送测试",delay=0.2)

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手的聊天界面，并向其发送folder\_path第一级目录下的所有文件.**

### 2.3 给多个好友分别发送单个文件send\_file\_to\_friends()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friends | 待发送消息者备注 | str |
| file\_paths | 给所有待发送好友发送的文件路径列表 | List[str] |
| with\_messages | 是否顺便发送消息 | bool 默认值:False |
| messages | 待发送的消息 | List[str] |
| messages\_first | 是否先发送消息 | bool默认值:False |
| delay | 输入消息或文件后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | List[boo]l默认值:List=[],默认值状态下不会拍一拍好友 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Files

Files.send\_file\_to\_friends(friends=['文件传输助手','测试ing'],file\_paths=["E:\OneDrive\Desktop\文件发送测试1.jpg","E:\OneDrive\Desktop\文件发送测试2.jpg"])

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_file\_to\_friends(friends=['文件传输助手','测试ing'],file\_paths=["E:\OneDrive\Desktop\文件发送测试1.jpg","E:\OneDrive\Desktop\文件发送测试2.jpg"])

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手和其他好友的聊天界面，向其分别发送file\_paths列表内的对应文件。**

### 2.4给多个好友分别发送多个文件send\_files\_to\_friends()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friends | 待发送消息者备注 | str |
| folder\_paths | 给每个好友发送的多个文件的文件夹路径列表 | List[str] |
| with\_messages | 是否顺便发送消息 | bool 默认值:False |
| messages | 待发送的消息 | List[List[str]] |
| messages\_first | 是否先发送消息 | bool默认值:False |
| delay | 输入消息或文件后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | bool默认值:False |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Files

Files.send\_files\_to\_friends(friends=['文件传输助手','测试ing'],folder\_paths=["E:\OneDrive\Desktop\文件发送测试1","E:\OneDrive\Desktop\文件发送测试2"])

**或者**

import pywechat.WechatAuto as wechat

wechat.send\_files\_to\_friends(friends=['文件传输助手','测试ing'],folder\_paths=["E:\OneDrive\Desktop\文件发送测试1","E:\OneDrive\Desktop\文件发送测试2"])

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手和其他好友的聊天界面，分别给这些好友发送folder\_paths列表内各个文件夹下的多个文件。**

### 2.5转发文件forward\_file()

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **参数类型及默认值** |
| friends | 待转发消息者备注列表 | List[str] |
| file\_path | 发送的文件路径 | str |
| delay | 输入消息或文件后停顿delay秒后发送(避免太快被检测到自动退出微信) | float默认值：1  注意:delay≥0.2 |
| tickle | 是否拍一拍好友 | bool默认值:False |
| search\_pages | 在消息列表中查找第一个转发文件好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friends[0]，然后打开聊天界面给该好友发消息 | Int 默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Files

Files.forward\_file(friends=['文件传输助手','测试ing'],file\_path="E:\OneDrive\Desktop\文件发送测试.jpg")

**或者**

import pywechat.WechatAuto as wechat

wechat.forward\_file(friends=['文件传输助手','测试ing'],file\_path="E:\OneDrive\Desktop\文件发送测试.jpg")

**说明:运行上述代码后，你将会看到微信被自动打开至文件传输助手的聊天界面，向其发送file\_path下指定文件后右键该文件，点击转发后在选择联系人面板搜索其余好友分别发送待转发文件。**

## 3 Contacts类

### **3.1**获取所有好友的备注与名称get\_friends\_names()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Contacts

print(Contacts.get\_friends\_names())

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_friends\_names())

**说明:运行上述代码后，你将会看到微信被自动打开至通讯录管理界面，并滚动好友列表页面记录信息，速率约为 6个好友/s。**

### **3.2**获取所有好友昵称备注与微信号get\_friends\_info()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Contacts

print(Contacts.get\_friends\_info())

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_friends\_info())

**说明:运行上述代码后，你将会看到微信被自动打开至通讯录界面，并滚动好友列表页面记录好友的备注与昵称以及微信号，速率约为 20个好友/s。**

### 3.3获取所有好友详细信息get\_friends\_detail()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Contacts

print(Contacts.get\_friends\_detail())

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_friends\_detail())

**说明:运行上述代码后，你将会看到微信被自动打开至通讯录界面，并滚动好友列表页面记录好友面板内的所有信息，速率约为 10~20个好友/s。**

### 3.4获取所有群聊信息get\_groups\_info()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Contacts

print(Contacts.get\_groups\_info())

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_groups\_info())

**说明:运行上述代码后，你将会看到微信被自动打开至通讯录管理界面，点击最近群聊并滚动群聊列表页面记录群聊的名称与人数，速率约为 10个群聊/s。**

### 3.5获取群内所有群成员群昵称get\_groupmembers\_info()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊名称 | str |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开群聊聊天界面进行后续操作 | Int  默认值5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Contacts

print(Contacts.get\_groupmembers\_info(group\_name='一家人'))

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_groupmembers\_info(group\_name='一家人'))

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面，点击保存群聊至通讯录然后再在通讯录中找到群聊记录该群内所有群成员昵称s。**

### 3.6获取所有企业微信好友信息get\_wecomfriends\_info()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import Contacts

print(Contacts.get\_wecom\_friends\_info())

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_wecom\_friends\_info())

**说明:运行上述代码后，你将会看到微信被自动打开至群聊界面，在通讯录中遍历查找企业微信好友。**

## 4 FriendSettings类

### 4.1获取好友聊天记录 get\_chat\_history()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| number | 获取聊天记录条数 | Int 默认值:10 |
| capture\_screen | 聊天记录是否截图 | bool 默认值:False |
| folder\_path | 保存截图文件夹路径，若不指定将保存至当前工作区文件夹 | str 默认值:None |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.get\_chat\_history(friend='文件传输助手',number=100,capture\_screen=True,folder\_path="E:\OneDrive\Desktop\保存聊天记录")

**或者**

import pywechat.WechatAuto as wechat

wechat.get\_chat\_history(friend='文件传输助手',number=100,capture\_screen=True,folder\_path="E:\OneDrive\Desktop\保存聊天记录")

**说明:运行上述代码后，你将会看到微信被自动打开至好友聊天记录界面，在聊天记录中快速遍历获取聊天记录。返回结果为json**

### 4.2 添加好友至黑名单 add\_friend\_to\_blacklist()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.add\_friend\_to\_blacklist(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.add\_friend\_to\_blacklist(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至制定好友设置界面，并点击添加好友至黑名单,若好友在黑名单中,调用此函数会触发相应的错误。**

### 4.3 添加新朋友 add\_new\_friend()

|  |  |  |
| --- | --- | --- |
| Phone\_number | 待添加好友手机号 | str |
| Wechat\_number | 待添加好友微信号 注意:微信号与手机号只需有一个即可 | str |
| request\_content | 添加好友时发送的请求信息 | Str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.add\_new\_friend(phone\_number='12125262220',request\_content='你好,我是....')

**或者**

import pywechat.WechatAuto as wechat

wechat.add\_new\_friend(phone\_number='12125262220',request\_content='你好，我是....')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面，并点击将好友移出黑名单，若好友不在黑名单中使用此函数会产生相应的错误。**

### 4.4 好友会话内置顶 pin\_friend()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时好友会话内置顶,当state为’close’时  取消好友会话内置顶 | str  默认值open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.pin\_friend(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.pin\_friend(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友聊天界面，并在聊天界面右上角点击置顶按钮。**

### 4.5 好友置顶 sticky\_friend\_on\_top()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时好友聊天置顶,当state为’close’时  取消好友聊天置顶 | str  默认值open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.sticky\_friend\_on\_top(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.sticky\_friend\_on\_top(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面，并在设置界面内点击聊天置顶按钮,点击后好友将位于会话列表最顶部。**

### 4.6 将好友设置为星标朋友 set\_friend\_as\_star\_friend()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时好友设置为星标朋友.当state为’close’时关闭设置好友为星标朋友 | str  默认值open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.set\_friend\_as\_star\_friend(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.set\_friend\_as\_star\_friend(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友详情界面，并在该界面内点击设为星标好友按钮,点击后好友将被设置为星标朋友。**

### 4.7 删除好友 delete\_friend()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.delete\_friend(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.delete\_friend(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面，并在该界面内点击设删除好友按钮,点击后好友将被删除。**

### 4.8 清空好友聊天记录 clear\_friend\_chat\_history()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.clear\_friend\_chat\_history(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.clear\_friend\_chat\_history(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面，并在该界面内点击清空聊天记录按钮,点击后与好友的聊天记录将被删除。**

### 4.9 好友消息免打扰 mute\_friend\_notifications()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时开启好友消息免打扰.当state为’close’时关闭好友消息免打扰 | str  默认值open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.mute\_friend\_notifications(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.mute\_friend\_notifications(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面，并在该界面内点击消息免打扰按钮,点击后好友将被设置为免打扰发送的新消息不再提示。**

### 4.10 拍一拍好友 tickle\_friend()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| times | 拍一拍次数  注意,连续拍一拍好友需要间隔10s。若设置为多次，那么总耗时times\*10 s | Int  默认值1 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.tickle\_friend(friend='测试ing')

**或者**

from pywechat.WechatAuto import FriendSettings

FriendSettings.tickle\_friend(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友聊天记录界面,找到最近一条好友的聊天记录后,右键好友头像，点击拍一拍**

### 4.11 推荐好友给朋友 share\_contact

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| others | 待推荐人备注列表 | List[str] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.share\_contact(friend='测试ing',others=['文件传输助手','路人甲','路人乙'])

**或者**

import pywechat.WechatAuto as wechat

wechat.share\_contact(friend='测试ing',others=['文件传输助手','路人甲','路人乙'])

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面,点击推荐给朋友按钮，然后在选择联系人面板内选择others内所有待推荐人备注选择后点击分别发送按钮。**

### 4.12 修改好友备注和标签 change\_friend\_remark\_and\_tag()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| remark | 好友新的备注 | str |
| tag | 好友标签,可以不传入该参数，此时调用此函数只是修改好友备注 | Str  默认值None |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.change\_friend\_remark\_and\_tag(friend='测试ing',remark='起个破名想半天了')

**或者**

import pywechat.WechatAuto as wechat

wechat.change\_friend\_remark\_and\_tag(friend='测试ing',remark='起个破名想半天了')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面,并点击设置备注和标签按钮，然后点击编辑框,将好友的备注设置为remark。**

### 4.13获取单个好友微信号get\_friend\_wechatnumber

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开好友聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import FriendSettings

print(FriendSettings.get\_friend\_wechat\_number(friend='测试ing'))

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_friend\_wechat\_number(friend='测试ing'))

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友详情界面，并在此界面内查询好友微信号。**

### 4.14获取多个好友微信号get\_friends\_wechatnumbers

|  |  |  |
| --- | --- | --- |
| friends | 好友备注列表 | List[st]r |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import FriendSettings

print(FriendSettings.get\_friends\_wechat\_numbers(friends=['路人甲','路人乙']))

**或者**

import pywechat.WechatAuto as wechat

print(wechat.get\_friends\_wechat\_numbers(friends=['路人甲','路人乙']))

**说明:运行上述代码后，你将会看到微信被自动打开,并搜索打开friends内所有好友的聊天界面内的详情界面，在此界面内查询各个好友微信号。**

### 4.15 修改好友权限 change\_friend\_privacy()

|  |  |  |
| --- | --- | --- |
| friends | 好友备注列表 | List[st]r |
| privacy | 好友权限,共有：'仅聊天',"聊天、朋友圈、微信运动等",'不让他（她）看',"不看他（她）"四种 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import FriendSettings

FriendSettings.change\_friend\_privacy(friend='测试ing',privacy='仅聊天')

**或者**

import pywechat.WechatAuto as wechat

wechat.change\_friend\_privacy(friend='测试ing',privacy='仅聊天')

**说明:运行上述代码后，你将会看到微信被自动打开,并搜索打开friends内所有好友的聊天界面内的详情界面，在此界面内查询各个好友微信号。**

## 5 GroupSettings类

### 5.1获取群聊聊天记录get\_chat\_history()

|  |  |  |
| --- | --- | --- |
| friend | 群聊备注 | str |
| number | 获取聊天记录条数 | Int 默认值:10 |
| capture\_screen | 聊天记录是否截图 | bool 默认值:False |
| folder\_path | 保存截图文件夹路径 | str 默认值:None |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的grouip\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.get\_chat\_history(friend='测试群',number=100,capture\_screen=True,folder\_path="E:\OneDrive\Desktop\保存聊天记录")

**或者**

import pywechat.WechatAuto as wechat

wechat.get\_chat\_history(friend='测试群',number=100,capture\_screen=True,folder\_path="E:\OneDrive\Desktop\保存聊天记录")

**说明:运行上述代码后，你将会看到微信被自动打开群聊的聊天记录界面，在聊天记录中快速遍历获取聊天记录。返回结果为json**

### 5.2 保存群聊至通讯录 save\_group\_to\_contacts()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时保存群聊至通讯录,当state为’close’时不再保存群聊至通讯录 | str  默认值open |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int  默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.save\_group\_to\_contacts(group\_name='测试群',state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.save\_group\_to\_contacts(group\_name='测试群',state='open')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面，并点击保存群聊至通讯录按钮。**

### 5.3从群聊中添加新朋友add\_new\_friend\_from\_group()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| friend | 待添加好友群昵称 | str |
| request\_content | 添加好友时发送的请求信息 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int  默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.add\_friend\_from\_group(group\_name='测试群',friend='路人甲',request\_content='你好，我是...')

**或者**

import pywechat.WechatAuto as wechat

wechat.add\_friend\_from\_group(group\_name='测试群',friend='路人甲',request\_content='你好，我是...')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊,并在群聊内添加群昵称为friend的群好友.**

### 5.4 群聊会话内置顶 pin\_group()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时开启群聊会话内聊天置顶,当state为’close’时取消群聊聊天在会话内置顶 | str  默认值open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.pin\_group(group\_name='测试群',state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.pin\_group(group\_name='测试群',state='open')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊聊天界面,并点击顶部置顶按钮。**

**群聊将会在会话列表内处于置顶的位置。注意，该方法不能将群聊置于会话列表最顶端,需要使用stiky\_group\_on\_top()**

### 5.5 群聊置顶 sticky\_group\_on\_top()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时开启群聊聊天置顶,当state为’close’时取消群聊聊天置顶 | str  默认值open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.sticky\_group\_on\_top(group\_name='测试群',state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.sticky\_group\_on\_top(group\_name='测试群',state='open')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面，并点击置顶该群聊按钮。群聊将被置于会话列表最顶部。**

### 5.6 修改群聊名称 change\_group\_name()#管理员权限

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| change\_name | 修改后的群聊名称 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.change\_group\_name(group\_name='测试群',change\_name='测试群1')

**或者**

import pywechat.WechatAuto as wechat

wechat.change\_group\_name(group\_name='测试群',change\_name='测试群1')

**说明:运行上述代码后，你将会看到微信被自动打开至指定群聊设置界面，并在设置界面内点击群聊名称按钮,接着在编辑框内输入change\_name。**

### 5.7 修改群聊备注change\_group\_remark()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| group\_remark | 待修改的群聊备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的grouip\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.change\_group\_remark(group\_name='测试群',group\_remark='测试群1')

**或者**

import pywechat.WechatAuto as wechat

wechat.change\_group\_remark(group\_name='测试群',group\_remark='测试群1')

**说明:运行上述代码后，你将会看到微信被自动打开至指定好友设置界面，并在设置界面内点击取消聊天置顶按钮,点击后好友不再将位于会话列表最顶部。**

### 5.8 修改我的群昵称 change\_my\_nickname\_in\_group()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| my\_nickname | 待修改的我在群内昵称 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.change\_my\_nickname\_in\_group(group\_name='测试群',my\_nickname='你好')

**或者**

import pywechat.WechatAuto as wechat

wechat.change\_my\_nickname\_in\_group(group\_name='测试群',my\_nickname='你好')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面,并点击我的群昵称按钮后重新编辑，修改我在群内的昵称为传入的my\_nickname。**

### 5.9 发布群公告 edit\_group\_notice()#管理员权限

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| content | 群公告内容 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.edit\_group\_notice(group\_name='测试群',content='通知:所有人下午来7421开会')

**或者**

import pywechat.WechatAuto as wechat

wechat.edit\_group\_notice(group\_name='测试群',content='通知:所有人下午来7421开会')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面,并点击群公告,在群公告面板内编辑传入的content并发布。**

### 5.10 退出群聊 quit\_group()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.quit\_group\_chat(group\_name='测试群')

**或者**

import pywechat.WechatAuto as wechat

wechat.quit\_group\_chat(group\_name='测试群')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面,并点击退出该群聊按钮后确定。**

### 5.11 清空群聊聊天记录 clear\_group\_chat\_history()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

FriendSettings.clear\_group\_chat\_history(group\_name='测试群')

**或者**

import pywechat.WechatAuto as wechat

wechat.clear\_group\_chat\_history(group\_name='测试群')

**说明:运行上述代码后，你将会看到微信被自动打开群聊设置界面，并在该界面内点击清空聊天记录按钮,点击后该群的聊天记录将被删除。**

### 5.12 群聊消息免打扰 mute\_group\_notifications()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| state | 打开或关闭某项操作,取值为:’open’,’close’  当state为‘open’时开启群聊的消息免打扰,当state为’close’时取消群聊的消息免打扰。 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

FriendSettings.mute\_group\_notifications(friend='测试群')

**或者**

import pywechat.WechatAuto as wechat

wechat.mute\_group\_notifications(friend='测试群')

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面，并在该界面内点击消息免打扰按钮,点击后群聊将被设置为免打扰发送的新消息不再提示。**

### 5.13 邀请好友进入群聊 invite\_others\_to\_group()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊备注 | str |
| friends | 待邀请人列表 | List[str] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.invite\_others\_to\_group(group\_name='测试群',friends=['路人甲','路人乙'])

**或者**

import pywechat.WechatAuto as wechat

wechat.invite\_others\_to\_group(group\_name='测试群',friends=['路人甲','路人乙'])

**说明:运行上述代码后，你将会看到微信被自动打开至指定群聊设置界面,点击邀请好友群聊后，在选择联系人面板内选择friends内所有邀请人备注，选择后点击确定按钮。**

### 5.14 创建新群聊 create\_new\_group()

|  |  |  |
| --- | --- | --- |
| friends | 新群聊所有成员备注 | str |
| group\_name | 新群聊群名 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.create\_group\_chat(friends=['路人甲','路人乙','路人丙'],group\_name='测试群',messages=['大家好,欢迎来到...','接下来...'])

**或者**

import pywechat.WechatAuto as wechat

wechat.create\_group\_chat(friends=['路人甲','路人乙','路人丙'],group\_name='测试群',messages=['大家好,欢迎来到...','接下来...'])

**说明:运行上述代码后，你将会看到微信被自动打开并点击搜索栏旁边的新建群聊按钮，然后再联系人选择面板内选择所有friends内好友,创建一个名为group\_name的群聊,若你传入了messages那么还会再创建新群聊后再群内发送指定消息。**

### 5.15 踢出群成员 remove\_friend\_from\_group()#管理员权限

|  |  |  |
| --- | --- | --- |
| Group\_name | 群聊备注。 | Str |
| friends | 待踢出群聊群成员的群昵称 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊的页数，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开该群聊聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.remove\_friend\_from\_group(group\_name='测试群',friends=['路人甲','路人乙','路人丙'])

**或者**

import pywechat.WechatAuto as wechat

wechat.remove\_friend\_from\_group(group\_name='测试群',friends=['路人甲','路人乙','路人丙'])

**说明:运行上述代码后，你将会看到微信被自动打开至群聊设置界面,并点击删除按钮,将friends内的群成员全部踢出群聊。**

### 5.16 显示群成员昵称 show\_group\_memebers\_nickname()

|  |  |  |
| --- | --- | --- |
| friend | 好友备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开聊天界面给该好友发消息 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import GroupSettings

GroupSettings.show\_group\_members\_nickname(group\_name='测试群',state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.show\_group\_members\_nickname(group\_name='测试群',state='open')

**说明:运行上述代码后，你将会看到微信被自动打开至指定群聊设置界面，并在此界面内点击显示群成员昵称按钮。**

## 6 AutoReply类

### **6.1自动回复指定好友消息 auto\_reply\_to\_friend()**

|  |  |  |
| --- | --- | --- |
| friend | 待自动回复好友的备注 | str |
| duration | 自动回复指定好友消息持续时间,单位:’s’,‘min’,’h’ | str |
| content | 自动回复内容 | str |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面给该好友发消息 | Int  默认值:5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| save\_chat\_history | 是否保留聊天记录 | bool  默认值  False |
| capture\_screen | 是否截屏 | bool  默认值  False |
| folder\_path | 聊天截图存放路径 | str |

**示例**

from pywechat.WechatAuto import AutoReply

chat\_history=AutoReply.auto\_reply\_to\_friend(friend='测试ing',duration='5min',content='你好我不在请稍后联系',save\_chat\_history=True,capture\_screen=True,folder\_path="E:\OneDrive\Desktop\测试专用")

print(chat\_history)

**或者**

import pywechat.WechatAuto as wechat

chat\_history=wechat.auto\_reply\_to\_friend(friend='测试ing',duration='5min',content='你好我不在请稍后联系',save\_chat\_history=True,capture\_screen=True,folder\_path="E:\OneDrive\Desktop\测试专用")

print(chat\_history)

**说明:运行上述代码后，你将会看到微信被自动打开至通讯录管理界面好友聊天界面,在指定时间内只要好友向你发送消息，便会自动回复content指定内容。**

### **6.2 自动回复消息 auto\_reply\_messages()**

|  |  |  |
| --- | --- | --- |
| friend | 待自动回复好友的备注 | str |
| duration | 自动回复消息持续时间,单位:’s’,‘min’,’h’ | str |
| content | 自动回复内容 | str |
| max\_pages | 遍历会话列表页数,一页为8~10人,设定持续时间后,将持续在max\_pages内循环遍历查找是否有新消息\n | Int  默认值5 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | str  默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| never\_reply | 不自动回复的好友或群聊备注列表 | List[str] |

**示例**

from pywechat.WechatAuto import AutoReply

AutoReply.auto\_reply\_messages(duration='5min',content='你好我不在请稍后联系')

**或者**

import pywechat.WechatAuto as wechat

wechat.auto\_reply\_to\_friend(duration='5min',content='你好我不在请稍后联系')

**说明:运行上述代码后，你将会看到微信被自动打开至聊天界面,自动回复前10个会话列表中含有红色消息提示的好友。**

### 6.3 自动接听电话 auto\_answer\_call()

|  |  |  |
| --- | --- | --- |
| duration | 自动接听功能持续时间,单位:’s’,‘min’,’h | str |
| broadcast\_content | Windows API语音播报内容 | str |
| message | 接听电话预约播报挂断电话后发送的消息 | str |
| times | Widnow API语音播报的次数 | Int  默认值1 |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatAuto import AutoReply

AutoReply.auto\_answer\_call(duration='5min',broadcast\_content='你好我不在,请稍后联系',message='你好我不在请稍后联系',times=3)

**或者**

import pywechat.WechatAuto as wechat

wechat.auto\_answer\_call(duration='5min',broadcast\_content='你好我不在,请稍后联系',message='你好我不在请稍后联系',times=3)

**说明:运行上述代码后，和没运行一样,但是只要有好友向你拨打语音或视频电话，便会自动点击接听,然后使用windows 朗读文本的api音量最大播放broadcast\_content内容,如果你传入了message在挂断电话后还会打开好友聊天界面,向好友发送message。**

### 6.4 接入AI自动回复消息 AI\_auto\_reply\_to\_friend()

|  |  |  |
| --- | --- | --- |
| friend | 自动回复好友的备注 | str |
| duration | 自动接听功能持续时间,单位:’s’,‘min’,’h | str |
| AI\_engine | AI大模型API函数 | func |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值  None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面给该好友发消息 | Int  默认值:5 |

**示例**

from pywechat.WechatAuto import AutoReply

def chatgpt(content):

    return content

AutoReply.AI\_auto\_reply\_to\_friend(friend='测试ing',duration='5min',content='你好我不在,请稍后联系',AI\_engine=chatgpt)

**或者**

import pywechat.WechatAuto as wechat

def chatgpt(content):

    return content

wechat.AI\_auto\_reply\_to\_friend(friend='测试ing',duration='5min',content='你好我不在,请稍后联系',AI\_engine=chatgpt,message='你好我不在请稍后联系')

**说明:运行上述代码后，你将会看到微信被自动打开至好友聊天界面,好友发送消息后,会点击该条消息并引用,等待一段时间后AI\_engine传入的AI大模型函数返回结果后会直接发送出去。**

## 7 WechatSettings类

### 7.1 适配系统缩放比例Adapt\_to\_PC\_display\_scalling()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Adapt\_to\_PC\_display\_scalling(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Adapt\_to\_PC\_display\_scalling(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并点击适配系统缩放比例选项。**

### 7.2 聊天中的语音消息自动转成文字 Auto\_convert\_voice\_messages\_to\_text()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Auto\_convert\_voice\_messages\_to\_text(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Auto\_convert\_voice\_messages\_to\_text(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并点击聊天中的语音消息自动转成文字复选框。**

### 7.3 有更新时自动更新微信 Auto\_update\_wechat()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Auto\_update\_wechat(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Auto\_update\_wechat(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并点击有更新时自动更新微信复选框。**

### 7.4 修改截取屏幕快捷键 Change\_capture\_screen\_shortcut()

|  |  |  |
| --- | --- | --- |
| shrotcuts | 快捷键名称列表 | List[str] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool  默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Change\_capture\_screen\_shortcut(shortcuts=['ctrl','s'])

**或者**

import pywechat.WechatAuto as wechat

wechat.Change\_capture\_screen\_shortcut(shortcuts=['ctrl','s'])

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,点击快捷键,并修改截取屏幕的快捷键。**

### 7.5 修改锁定微信快捷键 Change\_lock\_wechat\_shortcut()

|  |  |  |
| --- | --- | --- |
| shrotcuts | 快捷键名称列表 | List[str] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool  默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool  默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Change\_lock\_wechat\_shortcut(shortcuts=['ctrl','L'])

**或者**

import pywechat.WechatAuto as wechat

wechat.Change\_lock\_wechat\_shortcut(shortcuts=['ctrl','L'])

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,点击快捷键,并修改锁定微信的快捷键。**

### 7.6 修改打开微信快捷键 Change\_open\_wechat\_shortcut()

|  |  |  |
| --- | --- | --- |
| shrotcuts | 快捷键名称列表 | List[str] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Change\_open\_wechat\_shortcut(shortcuts=['shift','w'])

**或者**

import pywechat.WechatAuto as wechat

wechat.Change\_open\_wechat\_shortcut(shortcuts=['shift','w'])

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,点击快捷键,并修改打开微信的快捷键。**

### 7.7 修改发送消息快捷键 Change\_send\_message\_shortcut()

|  |  |  |
| --- | --- | --- |
| shrotcuts | 快捷键键位名称微信发送消息快捷键只有’Enter’,’ctrl+enter’两个选项 | Str  默认值;’Enter’ |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Change\_send\_message\_shortcut(shortcuts='Enter')

**或者**

import pywechat.WechatAuto as wechat

wechat.Change\_send\_message\_shortcut(shortcuts='Enter')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,点击快捷键,并修改发送消息快捷键。**

### 7.8 清空所有聊天记录clear\_chat\_history()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Clear\_chat\_history()

**或者**

import pywechat.WechatAuto as wechat

wechat.Clear\_chat\_history()

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并点击清空聊天记录按钮。点击确定后,微信内所有的聊天记录将被删除。**

### 7.9 关闭PC微信自动登录Close\_auto\_log\_in()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Close\_auto\_log\_in()

**或者**

import pywechat.WechatAuto as wechat

wechat.Close\_auto\_log\_in()

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并点击关闭自动登录按钮,注意,开启自动登录需要下次手机扫码登录后,在手机端设置。**

### 7.10 PC微信退出登录Log\_out()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Log\_out()

**或者**

import pywechat.WechatAuto as wechat

wechat.Log\_out()

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并点击退出登录按钮,然后PC微信便会处于退出登录状态。**

### 7.11 快捷键恢复默认设置Shortcut\_default()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Shortcut\_default()

**或者**

import pywechat.WechatAuto as wechat

wechat.Shortcut\_default()

### 7.12视频号通知标记 Channel\_notification\_flag()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Channel\_notification\_flag(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Channel\_notification\_flag(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击视频号复选框。**

### 7.13 看一看通知标记 Topstories\_notification\_flag()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Topstories\_notification\_flag(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Topstories\_notification\_flag(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击看一看复选框。**

### 7.14 朋友圈通知标记 Moments\_notification\_flag()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Moments\_notification\_flag(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Moments\_notification\_flag(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击朋友圈复选框。**

### 7.15 小程序通知标记 Miniprogram\_notification\_flag()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Miniprogram\_notification\_flag(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Miniprogram\_notification\_flag(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击朋友圈复选框。**

### 7.16 新消息通知声音 New\_message\_alert\_sound()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.New\_message\_alert\_sound(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.New\_message\_alert\_sound(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击新消息通知声音复选框。**

### 7.17 语音和视频通话通知声音 Voice\_and\_video\_calls\_alert\_sound()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Voice\_and\_video\_calls\_alert\_sound(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Voice\_and\_video\_calls\_alert\_sound(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击语音和视频通话通知声音声音复选框。**

### 7.18 保留聊天记录Save\_chat\_history()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Save\_chat\_history(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Save\_chat\_history(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击语音和视频通话通知声音声音复选框。**

### 7.19 显示网络搜索历史Show\_web\_search\_history()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Show\_web\_search\_history(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Show\_web\_search\_history(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击显示网络搜索历史复选框。**

### 7.20 使用系统默认浏览器打开网页Using\_default\_browser()

|  |  |  |
| --- | --- | --- |
| state | 打开或关闭某项操作,取值为:’open’,’close’。 | Str  默认值:open |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| close\_setings\_window | 任务结束后关闭设置界面 | bool默认值：True |

**示例**

from pywechat.WechatAuto import WechatSettings

WechatSettings.Using\_default\_browser(state='open')

**或者**

import pywechat.WechatAuto as wechat

wechat.Using\_default\_browser(state='open')

**说明:运行上述代码后，你将会看到微信被自动打开设置界面,并在消息通知内点击显示网络搜索历史复选框。**

## 8 Call类

### 8.1与好友语音通话voice\_call()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊名称 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面给该好友发消息 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import Call

Call.voice\_call(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.voice\_call(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至好友聊天界面，并点击语音通话按钮与好朋友语音通话。**

### 8.2 与好友视频通话video\_call()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊名称 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面给该好友发消息 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import Call

Call.video\_call(friend='测试ing')

**或者**

import pywechat.WechatAuto as wechat

wechat.video\_call(friend='测试ing')

**说明:运行上述代码后，你将会看到微信被自动打开至好友聊天界面，并点击视屏通话按钮与好朋友视频通话。**

### 8.3 群聊内发起通话voice\_call\_in\_group()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊名称 | str |
| friends | 待呼叫的群成员 | List[str] |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatAuto import Call

Call.voice\_call\_in\_group(group\_name='测试群',friends=['路人甲','路人乙'])

**或者**

import pywechat.WechatAuto as wechat

wechat.voice\_call\_in\_group(group\_name='测试群',friends=['路人甲','路人乙'])

**说明:运行上述代码后，你将会看到微信被自动打开至群聊聊天界面,并点击语音聊天按钮,在群内与指定的群成员进行语音通话。**

# WechatTools

wechatTools是用来辅助wechatAuto模块进行自动化操作的工具,内置了Tools与API两个类,本模块内所有函数与方法同名。

## Tools类

### 1.1 查找微信路径 find\_wechat\_path()

|  |  |  |
| --- | --- | --- |
| copy\_to\_clipboard | 是否将路径复制到剪贴板 | bool:True |

**示例**

from pywechat.WechatTools import Tools

Tools.find\_wechat\_path()

**或者**

import pywechat.WechatTools as wechat

wechat.find\_wechat\_path()

**说明:只有在微信运行的条件下,该函数才能正确查找到微信的路径,否则会引发错误。**

### 1.2 添加微信路径至windows用户环境变量set\_wechat\_as\_environ\_path()

**示例**

from pywechat.WechatTools import Tools

Tools.set\_wechat\_as\_environ\_path()

**或者**

import pywechat.WechatTools as wechat

wechat.set\_wechat\_as\_environ\_path()

**说明:只有在微信运行的条件下,上述代码才能正确查找到微信的路径,否则会引发错误。**

### 1.3 打开微信open\_wechat()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_wechat()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_wechat()

**说明:运行上述代码后后无论微信处于什么状态，都可以迅速打开微信。**

### 1.4 打开微信设置界面open\_settings()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 微信是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_settings()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_settings()

**说明:运行上述代码后,微信将被自动打开至设置界面。**

### 1.5 打开通讯录open\_contacts()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_contacts()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_contacts()

**说明:运行上述代码后,微信将被自动打开至通讯录界面。**

### 1.6 打开通讯录设置界面open\_contacts\_settings()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_contacts\_settings()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_contacts\_settings()

**说明:运行上述代码后,微信将被自动打开至通讯录管理界面。**

### 1.7 打开收藏界面open\_collections()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_collections()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_collections()

**说明:运行上述代码后,微信将被自动打开至收藏界面。**

### 1.8 打开朋友圈界面open\_wechat\_moments()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| Is\_maximize | 是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_wechat\_moments()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_wechat\_moments()

**说明：运行上述代码后微信将被自动打开至朋友圈界面。**

### 1.9 打开聊天文件界面open\_chat\_files()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| wechat\_maximize | 微信界面是否全屏 | bool 默认值:True |
| Is\_maximize | 聊天文件界面是否全屏,默认为True | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_chat\_files()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_chat\_files()

**说明：运行上述代码后微信将被自动打开至聊天文件界面。**

### 1.10 打开搜一搜界面open\_search()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| wechat\_maximize | 微信界面是否全屏 | bool 默认值:True |
| is\_maximize | 搜一搜界面是否全屏 | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_search()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_search()

**说明：运行上述代码后微信将被自动打开至搜一搜界面。**

### 1.11 打开看一看界面open\_top\_stories()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| wechat\_maximize | 微信界面全屏 | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 看一看界面全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_top\_stories()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_top\_stories()

**说明：运行上述代码后微信将被自动打开至看一看界面。**

### 1.12 打开视频号界面open\_channels()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| wechat\_maximize | 微信界面是否全屏 | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 视频号界面是否全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_channels()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_channels()

**说明：运行上述代码后微信将被自动打开至视频号界面。**

### 1.13 打开小程序面板open\_program\_pane()

|  |  |  |
| --- | --- | --- |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| wechat\_maximize | 微信界面是否全屏 | bool 默认值:True |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 小程序面板界面是否全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_program\_pane()

**或者**

import pywechat.WechatTools as wechat

wechat.open\_program\_pane()

**说明：运行上述代码后微信将被自动打开至小程序面板界面。**

### 1.14 打开与好友或群聊的聊天界面open\_dialog\_window()

|  |  |  |
| --- | --- | --- |
| friend | 好友或群聊的备注 |  |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_dialog\_window(friend='文件传输助手')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_dialog\_window('文件传输助手')

**说明：运行上述代码后微信将被自动打开至文件传输助手的聊天界面。**

### 1.15 打开好友设置界面open\_friend\_settings()

|  |  |  |
| --- | --- | --- |
| friend | 好友的备注 |  |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_friend\_settings(friend='测试ing')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_friend\_settings('测试ing')

**说明：运行上述代码后微信将被自动打开至文件传输助手的聊天界面。**

### 1.16 打开好友设置界面菜单open\_friend\_settings\_menu()

|  |  |  |
| --- | --- | --- |
| friend | 好友的备注 |  |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_friend\_settings\_menu(friend='测试ing')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_friend\_settings\_menu('测试ing')

**说明：运行上述代码后微信将被自动打开至文件传输助手的聊天界面。**

### 1.17 打开好友个人简介面板open\_friend\_profile()

|  |  |  |
| --- | --- | --- |
| friend | 好友的备注 |  |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |
| search\_pages | 在消息列表中查找好友，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的friend，然后打开聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_friend\_profile(friend='测试ing')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_friend\_profile('测试ing')

**说明：运行上述代码后微信将被自动打开至指定好友的个人简介面板。**

### 1.18 打开群聊设置界面open\_group\_settings()

|  |  |  |
| --- | --- | --- |
| group\_name | 群聊的备注 | str |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |
| search\_pages | 在消息列表中查找群聊，一页为5-12人，默认为5，当search\_pages设置为0时，直接从顶部搜索栏中搜索传入的group\_name，然后打开聊天界面进行后续操作。 | Int 默认值:5 |

**示例**

from pywechat.WechatTools import Tools

Tools.open\_group\_settings(friend='测试群')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_group\_settings('测试群')

**说明：运行上述代码后微信将被自动打开至指定群聊的设置界面。**

## API类

### 1.1 打开指定微信小程序open\_wechat\_miniprogram()

|  |  |  |
| --- | --- | --- |
| name | 微信小程序名称 | str |
| load\_delay | 加载时长 | Float  默认值  3.5s |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import API

API.open\_wechat\_miniprogram(name='问卷星')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_wechat\_miniprogram(name='问卷星')

**说明：运行上述代码后微信将被自动打开至指定的小程序。**

### 1.2 打开指定微信公众号open\_wechat\_official\_account()

|  |  |  |
| --- | --- | --- |
| name | 微信公众号名称 | str |
| load\_delay | 加载时长 | Float  默认值  3.5s |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import API

API.open\_wechat\_official\_account(name='顺丰快递')

**或者**

import pywechat.WechatTools as wechat

wechat.open\_wechat\_official\_account(name='顺丰快递')

**说明：运行上述代码后微信将被自动打开至指定的公众号。**

### 1.3 打开视频号并搜索指定内容search\_channels()

|  |  |  |
| --- | --- | --- |
| search\_content | 视频号内搜索内容 | str |
| load\_delay | 加载时长 | Float  默认值  3.5s |
| wechat\_path | 微信路径，你的微信已经登录的情况下，不需要传入该参数，微信未登录时你可以将其添加到环境变量中，这样即使微信未登录也可以自动点击进入微信登录微信后进行自动化操作 | 默认值:None |
| close\_wechat | 任务结束后关闭微信 | bool 默认值:True |
| Is\_maximize | 微信主界面是否全屏 | bool 默认值:True |

**示例**

from pywechat.WechatTools import API

API.search\_channels(search\_content='微信自动化')

**或者**

import pywechat.WechatTools as wechat

wechat.search\_channels(search\_content='微信自动化')

**说明：运行上述代码后微信将被自动打开至视频号界面并在内部搜索指定的内容。**

# Clock

Clock是pywechat内的一个实现定时任务的模块,他主要包括两个类，schedule与schtasks

这两个类分别从构建同步协程池与windows系统下schtasks定时任务命令来实现定时操作。

## Schedule

Schedule模块主要是通过构建协程池实现定时任务，即计算出每个函数的目标时间与当前时间点的时间间隔后等待asyncio.sleep秒，然后开始执行对应的任务,因此代码编辑器不能关闭。

|  |  |  |
| --- | --- | --- |
| funcs | 待定时执行的所有函数组成的列表,注意，只需要函数名即可 | List[func] |
| Times | 与funcs一一对应的各个函数定时执行时间 | List[str] |
| parameters | funcs内各个函数的参数的字典形式 | List[dict] |
| waitPeriods | 与funcs一一对应的各个函数定时间隔时间后执行,注意Times与waitPeriods二者有一个即可 | List[str] |

**示例**



**代码**

from pywechat.Clock import schedule

import pywechat.WechatAuto as wechat

funcs\_to\_be\_executed=[wechat.send\_message\_to\_friend,wechat.voice\_call]

parameters=[{'friend':'文件传输助手','message':'你好,你将在16:04看到这条消息'},{'friend':'测试ing'}]

exectue\_time=['16:04','16:10']

task=schedule(funcs=funcs\_to\_be\_executed,Times=exectue\_time,parameters=parameters)

task.execute()

**说明:schedule初始化后,使用execute方法来开始定时任务。并且在运行完上述代码后,一定不要关闭代码编辑器,你可以将代码编辑器最小化然后做自己的事情,到达指定时间后微信会自动打开执行设定的任务。**

## Schtasks

schtasks实现定时任务的思路是通过subprocess.run()函数在windows cmd终端中运行windows系统下的schtasks命令,该命令的内容是在指定的时间下在cmd中使用sys.executable的本机python路径运行指定路径的python文件，在指定路径的python文件内你可以写一些pywechat的脚本。

### 1.1 创建定时任务 schtasks.create\_task()

|  |  |  |
| --- | --- | --- |
| taskname | 创建的任务名 | str |
| start\_time | 任务开始执行时间 | str |
| Pyfile\_path | 待执行的py文件路径 | str |

**示例**

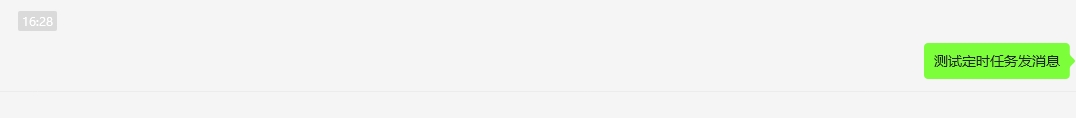
from pywechat.Clock import schtasks

start\_time='16:28'

pyfile\_path=r'D:**\a**utowechat**\t**o\_be\_executed.py'

schtasks.create\_task(taskname='测试schtasks',start\_time=start\_time,pyfile\_path=pyfile\_path)

**微信截图_20250212162640**



### 1.2 取消创建过的定时任务 schtasks.cancel\_task()

|  |  |  |
| --- | --- | --- |
| taskname | 已经创建过的任务名 | str |

**示例**

from pywechat.Clock import schtasks

schtasks.cancel\_task(taskname='测试schtasks')

**微信截图_20250212164006**

### 1.3 修改创建过的定时任务 schtasks.change\_task()

|  |  |  |
| --- | --- | --- |
| taskname | 创建过的任务名 | str |
| start\_time | 更新后的任务执行时间 | str |
| Pyfile\_path | 更新后的待执行的py文件路径 | str |

**示例**

from pywechat.Clock import schtasks

new\_start\_time='17:25'

new\_pyfile\_path=r'E:**\O**neDrive\Desktop**\t**est.py'

schtasks.change\_task(taskname='测试schtasks',start\_time=new\_start\_time,pyfile\_path=new\_pyfile\_path)