UIAutomation学习笔记

因个人水平有限，部分函数功能可能有误，欢迎指正.

**一．环境搭建：**

方法一：

下载UIAutomation源码，下载地址：<https://github.com/yinkaisheng/Python-Automation-for-Windows>

解压源码，将源码中的uiautomation.py和dll文件放在Python的site-packages目录下。

方法二：

通过命令pip install uiautomation 在线安装。

**二．获取/查看窗口控件信息**

以QQ登录做例子：

1.先打开运行QQ，保持在QQ登录界面；

2.运行md命令，进入解压的源码目录，输入命令automation.py -t3，然后3秒内切换到qq的登录界面

cmd窗口中会显示qq窗口中的控件信息，同时automation模块会把显示的信息写入到文件@AutomaitonLog.txt,方便查看。

**三．导入uiautomation库，开始使用**

import subprocess

from uiautomation import \*

subprocess.Popen(‘arg’) #用Popen来创建进程，并与进程进行复杂的交互。

time.sleep(2)

#查找qq窗口，searchDepth = 1，设置查找深度为1，查找Desktop的第一层子窗口就能很快找到QQ

qqWindow = WindowControl(searchDepth = 1, ClassName = 'TXGuiFoundation', Name = 'QQ')

#查找QQ帐号Edit，设置searchFromControl = qqWindow，从qqWindow开始查找子控件

#foundIndex = 1,表示查找第一个符合要求的控件，子窗口中的第一个Edit

edit = EditControl(searchFromControl = qqWindow, foundIndex = 1)

edit.Click()

Win32API.SendKeys('{Ctrl}A')

Logger.Write('Current QQ is ')

其他函数参考附录

**四．经验总结**

有人说有的软件用UIAutomation获取不到软件界面内的控件信息，这是非常正常的。

因为这些程序大多是用DirectUI开发的，不是用微软提供的GUI库开发的。

大多数的DirectUI都不支持UIAutomation，只有很少量的DirectUI支持UIAutomation，比如QQ，Firefox，Qt ...

如果要让DirectUI程序支持UIAutomation，DirectUI程序的作者必须实现UI Automation Provider，DirectUI程序实现了UI Automation Provider，才能支持UIAutomation。

**附录**

uiautomation函数大全(VERSION = '1.0.9')

Logger.SetLogFile(path) 设置输出路径

Logger.Write(log, consoleColor = -1, writeToFile = True, printToStdout = True) 打印输出，consoleColor的值在ConsoleColor类里

Logger.WriteLine(log, consoleColor = -1, writeToFile = True, printToStdout = True) 打印输出并换行，consoleColor的值在ConsoleColor类里

Logger.ColorfulWrite(log, consoleColor = -1, writeToFile = True, printToStdout = True) 指定某些字体颜色输出

Logger.ColorfulWriteLine(log, consoleColor = -1, writeToFile = True, printToStdout = True) 指定某些字体颜色输出并换行

Logger.log(log = '', consoleColor = -1, writeToFile = True, printToStdout = True) 打印输出当前时间+调用的方法+给定内容，consoleColor的值在ConsoleColor类里

Logger.Colorfullog(log = '', consoleColor = -1, writeToFile = True, printToStdout = True) 打印输出当前时间+调用的方法+给定内容（指定某些字体颜色），consoleColor的值在ConsoleColor类里

Logger.DeleteLog() 删除输出的日志信息

Win32API.GetClipboardText() 获取剪切板文本内容

Win32API.SetClipboardText(text) 设置剪切板文本内容

Win32API.SetConsoleColor(color) 设置控制窗口文本颜色

Win32API.ResetConsoleColor() 重置控制窗口文本颜色

Win32API.WindowFromPoint(x, y) 根据坐标，返回窗口句柄

Win32API.GetCursorPos() 获取当前坐标

Win32API.SetCursorPos(x, y) 设置当前坐标

Win32API.GetDoubleClickTime() 双击鼠标

Win32API.mouse\_event(dwFlags, dx, dy, dwData, dwExtraInfo) Call API mouse\_event from user32.dll

Win32API.keybd\_event(bVk, bScan, dwFlags, dwExtraInfo) Call API keybd\_event from user32.dll

Win32API.PostMessage(handle, msg, wparam, lparam) Call API PostMessageW from user32.dll

Win32API.SendMessage(handle, msg, wparam, lparam) Call API SendMessageW from user32.dll

Win32API.MouseClick(x, y, waitTime = OPERATION\_WAIT\_TIME) 鼠标左键点击(x,y)坐标

Win32API.MouseMiddleClick(x, y, waitTime = OPERATION\_WAIT\_TIME) 鼠标中键点击(x,y)坐标

Win32API.MouseRightClick(x, y, waitTime = OPERATION\_WAIT\_TIME) 鼠标右键点击(x,y)坐标

Win32API.MouseMoveTo(x, y, moveSpeed = 1, waitTime = OPERATION\_WAIT\_TIME) 鼠标从当前坐标移动到(x,y)坐标

Win32API.MouseDragDrop(x1, y1, x2, y2, moveSpeed = 1, waitTime = OPERATION\_WAIT\_TIME) 鼠标从(x1,y1)拖动到(x2,y2)

Win32API.GetScreenSize() 获取屏幕大小

Win32API.GetPixelColor(x, y, handle = 0) 根据窗口句柄，获取(x,y)坐标的像素颜色

Win32API.MessageBox(content, title, flags = MB.OK) Call API MessageBox from user32.dll

Win32API.SetForegroundWindow(hWnd) Set a window to foreground

Win32API.SetWindowTopmost(hWnd, isTopmost) Set a window to Topmost

Win32API.ShowWindow(hWnd, cmdShow) 显示窗口

Win32API.MoveWindow(hWnd, x, y, width, height, repaint = 1) Call API MoveWindow from user32.dll

Win32API.SetWindowPos(hWnd, hWndInsertAfter, x, y, width, height, flags) Call API SetWindowPos from user32.dll

Win32API.GetWindowText(hWnd) 获取窗口内容

Win32API.SetWindowText(hWnd, text) 设置窗口内容

Win32API.GetConsoleOriginalTitle() GetConsoleOriginalTitle

Win32API.GetConsoleTitle() GetConsoleTitle

Win32API.SetConsoleTitle(text) SetConsoleTitle

Win32API.GetForegroundWindow() Get a window to foreground

Win32API.IsDesktopLocked() 获取桌面锁屏状态（是否锁屏）

Win32API.PlayWaveFile(filePath, isAsync = True) play wave file

Win32API.GetProcessCommandLine(processId) may not work

Win32API.GetParentProcessId(processId = -1)

Win32API.IsProcess64Bit(processId) 进程是64bit返回True，32bit返回False，其他返回None

Win32API.TerminateProcess(processId) 终止给定id的进程

Win32API.TerminateProcessByName(processName) 终止给定名称的进程

Win32API.EnumProcess() 返回进程列表

Win32API.SendKey(key, waitTime = OPERATION\_WAIT\_TIME) 模拟按键

Win32API.PressKey(key) 模拟按下按键

Win32API.ReleaseKey(key) 模拟松开按键

Win32API.IsKeyPressed(key) 检查按键是否被按下，True /False

Win32API.VKtoSC(key) 返回scanCode

Win32API.SendKeys(text, interval = 0.01, waitTime = OPERATION\_WAIT\_TIME, debug = False) 发送多个按键或者文字内容

Bitmap.Release()

Bitmap.Width()

Bitmap.Height()

Bitmap.FromHandle(hwnd, left = 0, top = 0, right = 0, bottom = 0) 对窗口截图

Bitmap.FromControl(control, x = 0, y = 0, width = 0, height = 0) 对控件截图

Bitmap.FromFile(filePath) 加载图片

Bitmap.ToFile(savePath) 指定保存路径，名称要加上后缀.bmp, .jpg, .jpeg, .png, .gif, .tif, .tiff

Bitmap.GetPixelColor( x, y) 获取图片坐标的像素颜色

Bitmap.SetPixelColor( x, y, argb) 设置图片坐标的像素颜色

Bitmap.GetPixelColorsHorizontally(x, y, count) get list of argb form x,y horizontally

Bitmap.GetPixelColorsVertically(x, y, count) get list of argb form x,y vertically

Bitmap.GetPixelColorsOfRow(y) return list of argb of y row

Bitmap.GetPixelColorsOfColumn(x) return list of argb of x column

Bitmap.GetPixelColorsOfRect(x, y, width, height) return list of argb of rect

Bitmap.GetPixelColorsOfRects(rects) return list of argb of rects

Bitmap.GetAllPixelColors() return all argb of all pixels horizontally from 0,0

Bitmap.SetPixelColorsHorizontally(x, y, colors) set colors form x,y horizontally

Bitmap.SetPixelColorsVertically(x, y, colors) set colors form x,y vertically

QTPLikeSyntaxSupport.ButtonControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.CalendarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.CheckBoxControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ComboBoxControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.CustomControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.DataGridControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.DataItemControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.DocumentControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.EditControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.GroupControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.HeaderControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.HeaderItemControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.HyperlinkControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ImageControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ListControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ListItemControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.MenuControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.MenuBarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.MenuItemControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.PaneControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ProgressBarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.RadioButtonControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ScrollBarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.SemanticZoomControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.SeparatorControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.SliderControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.SpinnerControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.SplitButtonControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.StatusBarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TabControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TabItemControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TableControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TextControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ThumbControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TitleBarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ToolBarControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.ToolTipControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TreeControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.TreeItemControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

QTPLikeSyntaxSupport.WindowControl(self, element = 0, searchDepth = 0xFFFFFFFF, searchWaitTime = SEARCH\_INTERVAL, foundIndex = 1, \*\*searchPorpertyDict)

Control.\_\_del\_\_

Control.SetSearchFromControl(searchFromControl)

Control.SetSearchDepth(searchDepth)

Control.AddSearchProperty(\*\*searchPorpertyDict)

Control.RemoveSearchProperty(\*\*searchPorpertyDict)

Control.Exists(maxSearchSeconds = 5, searchIntervalSeconds = SEARCH\_INTERVAL) Find control every searchIntervalSeconds seconds in maxSearchSeconds seconds, if found, return True else False

Control.Refind(maxSearchSeconds = TIME\_OUT\_SECOND, searchIntervalSeconds = SEARCH\_INTERVAL, raiseException = True) Refind the control every searchIntervalSeconds seconds in maxSearchSeconds seconds, raise an LookupError if timed out

Control.Element() Return value of control's IUIAutomationElement

Control.Name() Return unicode Nam

Control.ControlType() Return an integer in class ControlType

Control.ControlTypeName() Return str ControlTypeName

Control.LocalizedControlType() Return unicode LocalizedControlType name

Control.ClassName() Return unicode ClassName

Control.AutomationId() Return unicode AutomationId

Control.ProcessId() Return process id

Control.IsEnabled() Return bool

Control.HasKeyboardFocus() Return boo

Control.IsKeyboardFocusable() Return bool

Control.IsOffScreen() Return bool

Control.BoundingRectangle() Return tuple (left, top, right, bottom)

Control.Handle() Return control's handle

Control.SetFocus() Make the control have focus

Control.MoveCursor(ratioX = 0.5, ratioY = 0.5, simulateMove = True) Move cursor to control's rect, default to center

Control.MoveCursorToMyCenter(simulateMove = True) Move cursor to control's center

Control.Click(ratioX = 0.5, ratioY = 0.5, simulateMove = True, waitTime =

OPERATION\_WAIT\_TIME)

Control.MiddleClick(ratioX = 0.5, ratioY = 0.5, simulateMove = True, waitTime =

OPERATION\_WAIT\_TIME)

Control.RightClick(ratioX = 0.5, ratioY = 0.5, simulateMove = True, waitTime =

OPERATION\_WAIT\_TIME)

Control.DoubleClick(ratioX = 0.5, ratioY = 0.5, simulateMove = True, waitTime =

OPERATION\_WAIT\_TIME)

Control.GetParentControl()

Control.GetTopWindow()

Control.GetFirstChildControl()

Control.GetLastChildControl()

Control.GetNextSiblingControl()

Control.GetPreviousSiblingControl()

Control.GetChildren()

Control.ShowWindow(cmdShow)

Control.Show()

Control.Hide()

Control.MoveWindow(x, y, width, height, repaint = 1)

Control.GetWindowText()

Control.SetWindowText(text)

Control.SendKey(key, waitTime = OPERATION\_WAIT\_TIME)

Control.SendKeys(keys, interval = 0.01, waitTime = OPERATION\_WAIT\_TIME)

Control.GetPixelColor(x, y)

Control.ToBitmap(x = 0, y = 0, width = 0, height = 0)

Control.CaptureToImage(savePath, x = 0, y = 0, width = 0, height = 0)

Control.Convert()

Control.CreateControlFromElement(element)

Control.CreateControlFromControl(control)

DockPattern.IsDockPatternAvailable()

ExpandCollapsePattern.IsExpandCollapsePatternAvailable()

ExpandCollapsePattern.Expand(waitTime = OPERATION\_WAIT\_TIME)

ExpandCollapsePattern.Collapse(waitTime = OPERATION\_WAIT\_TIME)

ExpandCollapsePattern.CurrentExpandCollapseState()

GridItemPattern.IsGridItemPatternAvailable()

GridItemPattern.CurrentContainingGrid()

GridItemPattern.CurrentRow()

GridItemPattern.CurrentColumn()

GridItemPattern.CurrentRowSpan()

GridItemPattern.CurrentColumnSpan()

GridPattern.IsGridPatternAvailable()

GridPattern.GetItem(row, column)

GridPattern.CurrentRowCount()

GridPattern.CurrentColumnCount()

InvokePattern.IsInvokePatternAvailable()

InvokePattern.Invoke(waitTime = OPERATION\_WAIT\_TIME)

MultipleViewPattern.IsMultipleViewPatternAvailable()

ScrollItemPattern.IsScrollItemPatternAvailable()

ScrollItemPattern.ScrollIntoView()

ScrollPattern.IsScrollPatternAvailable(self)

ScrollPattern.CurrentHorizontallyScrollable()

ScrollPattern.CurrentHorizontalViewSize()

ScrollPattern.CurrentHorizontalScrollPercent()

ScrollPattern.CurrentVerticallyScrollable()

ScrollPattern.CurrentVerticalViewSize()

ScrollPattern.CurrentVerticalScrollPercent()

ScrollPattern.SetScrollPercent(horizontalPercent, verticalPercent)

SelectionItemPattern.IsSelectionItemPatternAvailable()

SelectionItemPattern.Select()

SelectionItemPattern.AddToSelection()

SelectionItemPattern.RemoveFromSelection()

SelectionItemPattern.CurrentIsSelected()

SelectionPattern.IsSelectionPatternAvailable()

SelectionPattern.GetCurrentSelection()

RangeValuePattern.IsRangeValuePatternAvailable()

RangeValuePattern.RangeValuePatternCurrentValue()

RangeValuePattern.RangeValuePatternSetValue(value)

RangeValuePattern.CurrentMaximum()

RangeValuePattern.CurrentMinimum()

TableItemPattern.IsTableItemPatternAvailable()

TableItemPattern.CurrentRowHeaderItems()

TableItemPattern.CurrentColumnHeaderItems()

TablePattern.IsTablePatternAvailable()

TablePattern.CurrentRowHeaders()

TablePattern.CurrentColumnHeaders()

TablePattern.CurrentRowOrColumnMajor()

TextPattern.IsTextPatternAvailable()

TogglePattern.IsTogglePatternAvailable()

TogglePattern.Toggle(waitTime = OPERATION\_WAIT\_TIME)

TogglePattern.CurrentToggleState()

TransformPattern.IsTransformPatternAvailable()

TransformPattern.Move(x, y)

TransformPattern.Resize(width, height)

TransformPattern.Rotate(degrees)

TransformPattern2.IsTransformPattern2Available()

ValuePattern.IsValuePatternAvailable()

ValuePattern.CurrentValue()

ValuePattern.SetValue(value, waitTime = OPERATION\_WAIT\_TIME)

ValuePattern.CurrentIsReadOnly()

WindowPattern.IsWindowPatternAvailable()

WindowPattern.CurrentWindowVisualState()

WindowPattern.SetWindowVisualState(value, waitTime = OPERATION\_WAIT\_TIME)

WindowPattern.CurrentCanMaximize()

WindowPattern.Maximize(waitTime = OPERATION\_WAIT\_TIME)

WindowPattern.CurrentCanMinimize()

WindowPattern.Minimize(waitTime = OPERATION\_WAIT\_TIME)

WindowPattern.Normal(waitTime = OPERATION\_WAIT\_TIME)

WindowPattern.IsMaximize()

WindowPattern.IsMinimize()

WindowPattern.CurrentIsModal()

WindowPattern.CurrentIsTopmost()

WindowPattern.Close(waitTime = OPERATION\_WAIT\_TIME)

AppBarControl

ButtonControl

CalendarControl

CheckBoxControl

ComboBoxControl.Select(name, waitTime = OPERATION\_WAIT\_TIME)

CustomControl

DataGridControl

DataItemControl

DocumentControl

EditControl

GroupControl

HeaderControl

HeaderItemControl

HyperlinkControl

ImageControl

ListControl.GetSelectedItems()

ListItemControl

MenuControl

MenuBarControl

MenuItemControl

PaneControl

ProgressBarControl

RadioButtonControl

ScrollBarControl

SemanticZoomControl

SeparatorControl

SliderControl

SpinnerControl

SplitButtonControl

StatusBarControl

TabControl

TabItemControl

TableControl

TextControl

ThumbControl

TitleBarControl

ToolBarControl

ToolTipControl

TreeControl

TreeItemControl

WindowControl.SetTopmost(isTopmost = True)

WindowControl.MoveToCenter()

WindowControl.MetroClose(waitTime = OPERATION\_WAIT\_TIME)

WindowControl.SetActive(waitTime = OPERATION\_WAIT\_TIME)

SetGlobalSearchTimeOut(seconds)

GetClipboardText()

SetClipboardText(text)

Click(x, y, waitTime = OPERATION\_WAIT\_TIME)

RightClick(x, y, waitTime = OPERATION\_WAIT\_TIME)

MiddleClick(x, y, waitTime = OPERATION\_WAIT\_TIME)

MoveTo(x, y, waitTime = OPERATION\_WAIT\_TIME)

DragDrop(x1, y1, x2, y2, waitTime = OPERATION\_WAIT\_TIME)

KeyDown(key, waitTime = OPERATION\_WAIT\_TIME)

KeyUp(key, waitTime = OPERATION\_WAIT\_TIME)

SendKey(key, waitTime = OPERATION\_WAIT\_TIME)

SendKeys(keys, interval=0.01, waitTime = OPERATION\_WAIT\_TIME, debug=False)

WaitForExist(control, timeout)

WaitForDisappear(control, timeout)

WalkTree(top, getChildrenFunc = None, getFirstChildFunc = None, getNextSiblingFunc = None, includeTop = False, maxDepth = 0xFFFFFFFF)

ControlsAreSame(control1, control2)

GetRootControl()

GetFocusedControl()

GetForegroundControl()

GetConsoleWindow()

ControlFromPoint(x, y)

ControlFromPoint2(x, y)

ControlFromCursor()

ControlFromCursor2()

ControlFromHandle(handle)

WalkControl(control, includeTop = False, maxDepth = 0xFFFFFFFF)

LogControl(control, depth = 0, showAllName = True, showMore = False)

EnumAndLogControlAncestors(control, showAllName = True, showMore = False)

EnumAndLogControl(control, maxDepth = 0xFFFFFFFF, showAllName = True, showMore = False)

FindControl(control, compareFunc, maxDepth = 0xFFFFFFFF, findFromSelf = False, foundIndex = 1)

ShowDesktop()

RunWithHotKey(keyFunctionDict, stopHotKey = None)

usage() 显示命令参数使用方法

main() 主函数，默认3s内获取窗体控件信息，并输出相关信息到txt