Python 键盘/鼠标监听及控制

dawsonenjoy 关注
0.685 2018.09.24 09:52:43 字数 424 阅读 57,111

键盘监听

需要用到`keyboard`模块（`pip install keyboard`）

按键字符

```
#字符
'1'
'a'
...
#控制
'ctrl'
'alt'
'shift'
'enter'
'esc'
'f1'
...
#方向键
'up'
'down'
'left'
'right'
#组合按键
'ctrl'+'alt'+'a'
...
```

常用方法

wait()

监听按键，如果没设置按键，将会一直监听这句之前的按键；如果设置了按键，那么在按下该按键后就会停止监听，并执行后面的语句，举例：

```
import keyboard

print(0)
keyboard.wait('a')
#在按下a之前后面的语句都不会执行，下面同理
print(1)
keyboard.wait('b')
print(2)
keyboard.wait('c')
print(3)
keyboard.wait()

结果:
0
1
2
3
#继续监听
#只有按顺序按下abc（中间过程随便按不干扰）才能输出0123，但因为最后一个没设置按键，所以会一直监听下去
```

add_hotkey()

设置热键，一般和`wait()`组合使用，在wait监听下，当按下热键时会执行对应函数，举例：

```
import keyboard  #监听键盘

def test_a():
    print('aaa')

def test(x):
    print(x)

if __name__ == '__main__':
    keyboard.add_hotkey('f1', test_a)
    #按f1输出aaa
    keyboard.add_hotkey('ctrl+alt', test, args=('b',))
    #按ctrl+alt输出b
    keyboard.wait()
    #wait里也可以设置按键，说明当按到该键时结束
```

record()

记录键盘事件，如果加上until参数，可以设置当按下某按键时结束监听，和wait方法有点像，举例：

```
import keyboard

keyboard.add_hotkey('ctrl', print, args=('aaa',))
keyboard.add_hotkey('alt', print, args=('bbb',))

recorded = keyboard.record(until='esc')
#当按下esc时结束按键监听，并输出所有按键事件
print(recorded)
结果为:
aaa
aaa
bbb
[KeyboardEvent(ctrl down), KeyboardEvent(ctrl up), KeyboardEvent(ctrl down),
KeyboardEvent(ctrl up), KeyboardEvent(alt down), KeyboardEvent(alt up),
KeyboardEvent(esc down)]
```

hook()

绑定所有按键事件，当只要有按键按下/松开时就会触发的回调函数，举例：

```
import keyboard


def abc(x):
    print(x)
    print("111")

keyboard.hook(abc)
#按下任何按键时，都会调用abc，其中一定会传一个值，就是键盘事件
keyboard.wait()

结果:
KeyboardEvent(w down)
111
KeyboardEvent(w up)
111
KeyboardEvent(space down)
111
KeyboardEvent(space up)
111
KeyboardEvent(tab down)
111
KeyboardEvent(tab up)
111
KeyboardEvent(ctrl down)
111
...
```

KeyboardEvent()

一个按键事件，里面有3个常用参数：`event_type`、`scan_code`、`name`，分别代表按键类型（`down`/`up`）、按键号（每个键都有对应的）和按键名，举例：

```
import keyboard

def abc(x):
    a = keyboard.KeyboardEvent('down', 28, 'enter')
    #按键事件a为按下enter键，第二个参数如果不知道每个按键的值就随便写，
    #如果想知道按键的值可以用hook绑定所有事件后，输出x.scan_code即可
    if x.event_type == 'down' and x.name == a.name:
        print("你按下了enter键")
    #当监听的事件为enter键，且是按下的时候

keyboard.hook(abc)
# keyboard.hook_key('enter', bcd)
# recorded = keyboard.record(until='esc')
keyboard.wait()

结果:

你按下了enter键

你按下了enter键
```

更多参考

https://pypi.org/project/keyboard/

鼠标、键盘监听控制

这里介绍两个模块，一个是`pyautogui`模块（`pip install pyautogui`），另一个是`pynput`（`pip install pynput`）

pyautogui基本使用

常用方法

position()

获取鼠标位置，举例：

```
import pyautogui as pag    #监听鼠标

x1, y1 = pag.position()
print(x1, y1)
#输出鼠标当前位置
```

实例-结合鼠标键盘截图

（按下两次alt+ctrl来确定图片左上角和右上角，然后截图保存）

```
import keyboard  #监听键盘
import pyautogui as pag    #监听鼠标
from PIL import  ImageGrab    #截图、读取图片、保存图片

if keyboard.wait(hotkey='ctrl+alt') == None:
    x1, y1 = pag.position()
    if keyboard.wait(hotkey='ctrl+alt') == None:
        x2, y2 = pag.position()
        image = ImageGrab.grab((x1, y1, x2, y2))
        image.save("screen.png")
```

click()

鼠标点击控制，举例：

```
import pyautogui

pyautogui.click(button='right')
#点击鼠标右键
pyautogui.click(100, 100)
#要在指定位置点击左键
```

更多参考

pyautogui自动化控制鼠标键盘：https://blog.csdn.net/weixin_43430036/article/details/84650938
pyautogui控制键盘输入：https://blog.csdn.net/qq_40379759/article/details/80427235

pynput基本使用

参考：https://blog.csdn.net/longgb123/article/details/79090559

dawsonenjoy 关注
总资产71 共写了32.1W字 获得276个赞 共156个粉丝