

# 제 17 강 GUI Automation

# 학습 목차

- 마우스 인터랙션
- 키보드 인터랙션
- 화면 이미지 분석
- 네이버 로그인

# PyAutoGui - <https://pyautogui.readthedocs.io/en/latest/>

- 키보드 및 마우스 입력 자동화 도구.
- 단일 기본 스크린에서만 작동 - 멀티스크린 서브화면 지원하지 않음.
- 테스트 및 디버깅을 위한 설정 작업 필수 !

```
import pyautogui  
pyautogui.PAUSE = 1  
pyautogui.FAILSAFE = True
```

# 마우스 이동 - 절대값 이동

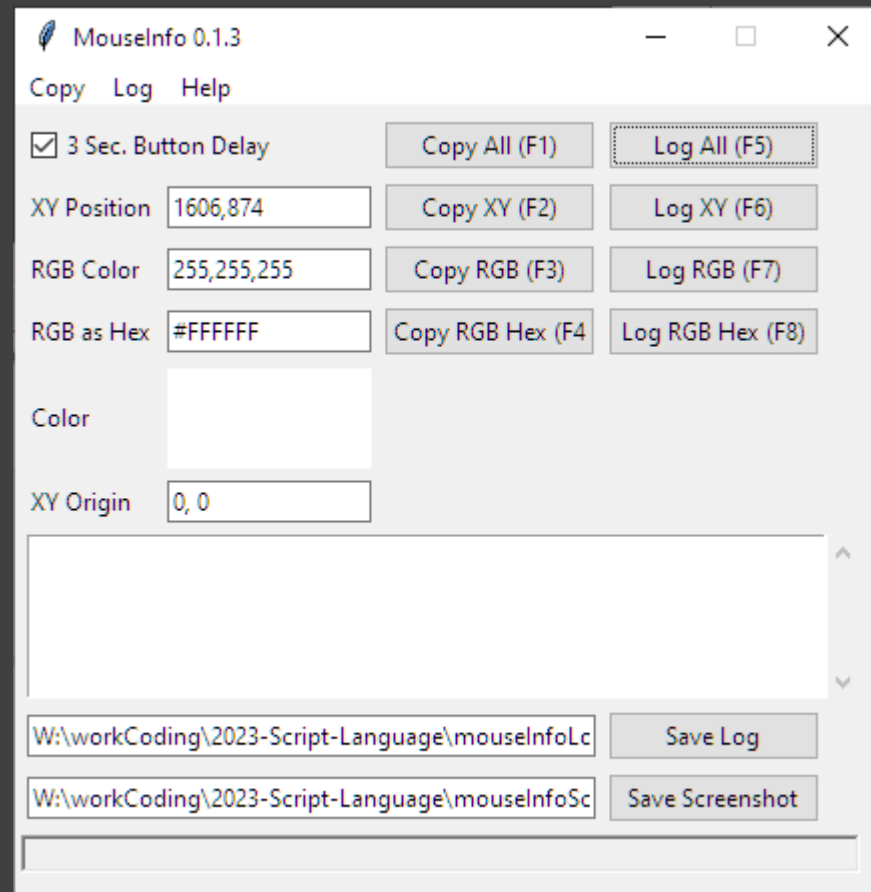
```
w, h = pyautogui.size()

for i in range(10):
    pyautogui.moveTo(100, 100, duration=0.25)
    pyautogui.moveTo(200, 100, duration=0.25)
    pyautogui.moveTo(200, 200, duration=0.25)
    pyautogui.moveTo(100, 200, duration=0.25)
```

# 마우스 이동 - 상대값 이동

```
for i in range(10):  
    pyautogui.move(100, 0, duration=1)  
    pyautogui.move(0, 100, duration=1)  
    pyautogui.move(-100, 0, duration=1)  
    pyautogui.move(0, -100, duration=1)
```

# pyautogui.mouseInfo()



# 마우스 커서 위치 모니터링 - MouseNow.py

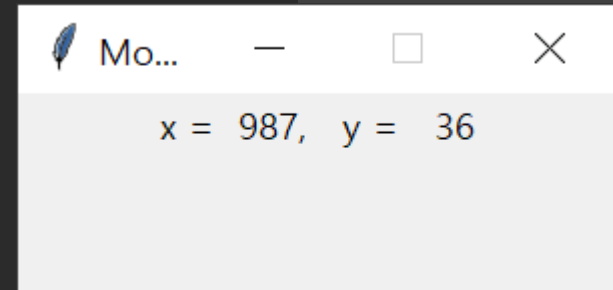
```
from tkinter import *
from tkinter.ttk import *
import pyautogui

window = Tk()
window.title("Mouse Now")
window.geometry("300x100")
window.resizable(False, False)

x, y = pyautogui.position()
loc_text = f"x = {x:4},    y = {y:4}"
label = Label(text=loc_text)
label.pack()

def update_mouse_position():
    global label
    x, y = pyautogui.position()
    loc_text = f"x = {x:4},    y = {y:4}"
    label.configure(text=loc_text)
    window.after(10, update_mouse_position)

window.attributes('-topmost', True)
window.after(10, update_mouse_position)
window.mainloop()
```



# Mouse click

```
pyautogui.moveTo(53, 2042, duration=0.25)  
pyautogui.click(1305, 1961)
```

```
pyautogui.rightClick(1413, 1952)  
pyautogui.middleClick(1413, 1952)  
pyautogui.doubleClick(1413, 1952, 0.2)
```



# Scroll

```
pyautogui.click(300, 300, duration=0.25)  
pyautogui.scroll(5000)
```

# Keyboard write & hot key – 카카오톡 메시지 전송

```
pyautogui.click(552,2050, duration=1)
pyautogui.typewrite('Sending kitty.\n', 0.25)
pyautogui.click(552,2050, duration=1)
pyautogui.typewrite('Sending kitty.\n')

pyautogui.write(['a', 'b', 'left', 'left', 'X', 'Y'])

import pyperclip
pyautogui.click(552,2050, duration=1)
pyperclip.copy('한글은 이렇게 보냅니다.\n')
pyautogui.hotkey('ctrl', 'v')
pyautogui.typewrite('\n')
```

# Keyboard key mapping

Keyboard key string	Meaning
'a', 'b', 'c', 'A', 'B', 'C', '1', '2', '3', '!', '@', '#', and so on	The keys for single characters
'enter' (or 'return' or '\n')	The ENTER key
'esc'	The ESC key
'shiftright', 'shiftright'	The left and right SHIFT keys
'altleft', 'altright'	The left and right ALT keys
'ctrlleft', 'ctrlright'	The left and right CTRL keys
'tab' (or '\t')	The TAB key
'backspace', 'delete'	The BACKSPACE and DELETE keys
'pageup', 'pagedown'	The PAGE UP and PAGE DOWN keys
'home', 'end'	The HOME and END keys
'up', 'down', 'left', 'right'	The up, down, left, and right arrow keys
'f1', 'f2', 'f3', and so on	The F1 to F12 keys
'volumemute', 'volumedown', 'volumeup'	The mute, volume down, and volume up keys (some keyboards do not have these keys, but your operating system will still be able to understand these simulated keypresses)
'pause'	The PAUSE key
'capslock', 'numlock', 'scrolllock'	The CAPS LOCK, NUM LOCK, and SCROLL LOCK keys
'insert'	The INS or INSERT key
'printscreen'	The PRN or PRINT SCREEN key
'winleft', 'winright'	The left and right WIN keys (on Windows)
'command'	The Command (⌘) key (on OS X)
'option'	The OPTION key (on OS X)

# Key press & release

- `pyautogui.keyDown()`
- `pyautogui.keyUp()`
- `pyautogui.press()`

# 드래그 - 카카오톡 파일 전송

```
pyautogui.click(1413, 1952) # click file  
pyautogui.dragTo(552, 2050, duration=1) # drag  
pyautogui.click(495, 903) # click confirm window  
pyautogui.press('enter') # send confirm
```

# 스크린샷

```
# screen shot

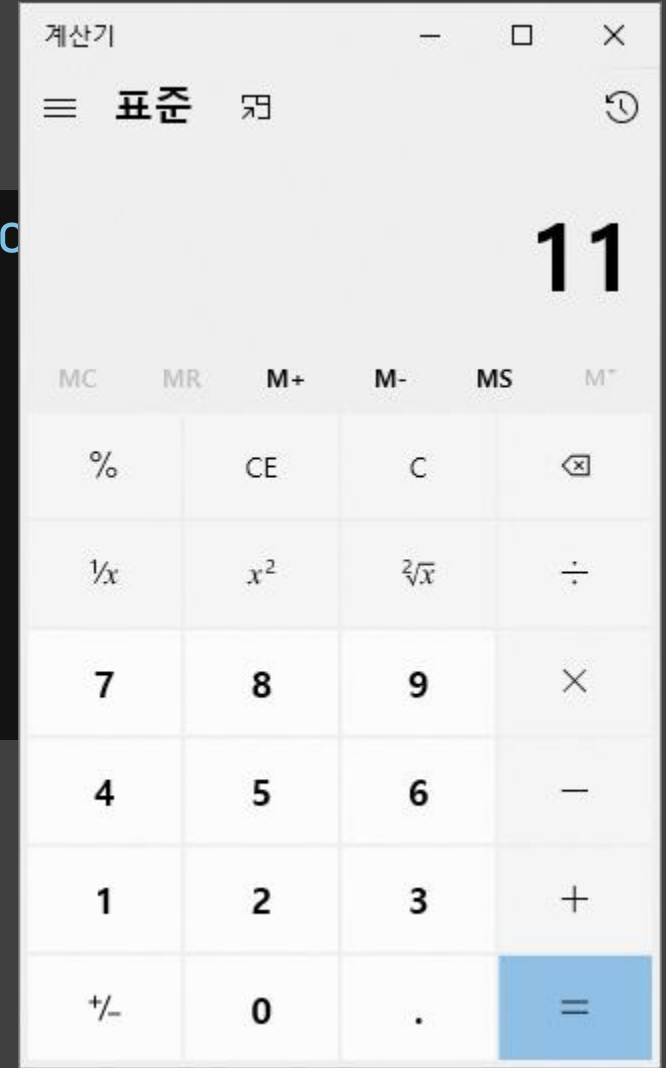
im = pyautogui.screenshot() # PIL image
im.getpixel((100,100))

im = pyautogui.screenshot('screen.png') # save to file
im = pyautogui.screenshot(region=(0, 0, 100, 100))
im.show()
```

# 화면 상 이미지 인식 및 좌표 추출

```
rect = pyautogui.locateOnScreen('1.png') # return rect  
p = pyautogui.locateCenterOnScreen('1.png') # return  
pyautogui.click(p[0], p[1])
```

```
pyautogui.click('1.png')  
all_rects = pyautogui.locateAllOnScreen('1.png')
```



# 간단한 윈도우 관리

```
# windows management

fw = pyautogui.getActiveWindow()
str(fw)
fw.size
fw.topleft
type(fw)

windows = pyautogui.getAllWindows()
dx, dy = 0, 0
for w in windows:
    if w.title:
        print(w)
        w.topleft = 0+dx, 0+dy
        w.size = 1920, 1080
        dx += 20
        dy += 20
```



```
windows = pyautogui.getAllWindows()
w1 = windows[0]
w1.size
print(w1)
w1.size = 800, 800
w1.topleft = 0, 1920
```