Screenshot Functions

Testing

Roadmap

pyautogui

Read the Docs

Keyboard Control Functions

The write() Function %

Docs » Keyboard Control Functions

The primary keyboard function is write(). This function will type the characters in the string that is passed. To add a delay interval in between pressing each character key, pass an int or float for the interval keyword argument.

G Edit on GitHub

For example:

```
>>> pyautogui.write('Hello world!')  # prints out "Hello world!" instantly
>>> pyautogui.write('Hello world!', interval=0.25) # prints out "Hello world!" with a quarter second de
```

You can only press single-character keys with write(), so you can't press the Shift or F1 keys, for example.

The press(), keyDown(), and keyUp() Functions

To press these keys, call the press() function and pass it a string from the pyautogui.KEYBOARD_KEYSsuch as enter, esc, f1. See KEYBOARD_KEYS.

For example:

v: latest ▼

```
>>> pyautogui.press('enter') # press the Enter key
>>> pyautogui.press('f1') # press the F1 key
>>> pyautogui.press('left') # press the left arrow key
```

The press() function is really just a wrapper for the keyDown() and keyUp() functions, which simulate pressing a key down and then releasing it up. These functions can be called by themselves. For example, to press the left arrow key three times while holding down the Shift key, call the following:

```
>>> pyautogui.keyDown('shift') # hold down the shift key
>>> pyautogui.press('left') # press the left arrow key
>>> pyautogui.press('left') # press the left arrow key
>>> pyautogui.press('left') # press the left arrow key
>>> pyautogui.keyUp('shift') # release the shift key
```

To press multiple keys similar to what write() does, pass a list of strings to press(). For example:

```
>>> pyautogui.press(['left', 'left'])
```

Or you can set how many presses <a>left:

```
>>> pyautogui.press('left', presses=3)
```

To add a delay interval in between each press, pass an int or float for the interval keyword argument.

The hotkey() Function

To make pressing hotkeys or keyboard shortcuts convenient, the hotkey() can be passed several key strings which will be pressed down in order, and then released in reverse order. This code:

```
>>> pyautogui.hotkey('ctrl', 'shift', 'esc')
```

...is equivalent to this code:

```
>>> pyautogui.keyDown('ctrl')
>>> pyautogui.keyDown('shift')
>>> pyautogui.keyDown('esc')
>>> pyautogui.keyUp('esc')
>>> pyautogui.keyUp('shift')
>>> pyautogui.keyUp('ctrl')
```

To add a delay interval in between each press, pass an int or float for the interval keyword argument.

KEYBOARD_KEYS

The following are the valid strings to pass to the press(), keyDown(), keyUp(), and hotkey() functions:

```
')', '*', '+', ',', '-', '.', '/', '0', '1', '2', '3', '4', '5', '6', '7',
'8', '9', ':', ';', '<', '=', '>', '?', '@', '[', '\\', ']', '^', '_', '`,
'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o',
'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', '{', '|', '}', '~',
'accept', 'add', 'alt', 'altleft', 'altright', 'apps', 'backspace',
'browserback', 'browserfavorites', 'browserforward', 'browserhome',
'browserrefresh', 'browsersearch', 'browserstop', 'capslock', 'clear',
'convert', 'ctrl', 'ctrlleft', 'ctrlright', 'decimal', 'del', 'delete',
'divide', 'down', 'end', 'enter', 'esc', 'escape', 'execute', 'f1', 'f10',
'f11', 'f12', 'f13', 'f14', 'f15', 'f16', 'f17', 'f18', 'f19', 'f2', 'f20',
'f21', 'f22', 'f23', 'f24', 'f3', 'f4', 'f5', 'f6', 'f7', 'f8', 'f9',
'final', 'fn', 'hanguel', 'hangul', 'hanja', 'help', 'home', 'insert', 'junja',
'kana', 'kanji', 'launchapp1', 'launchapp2', 'launchmail',
'launchmediaselect', 'left', 'modechange', 'multiply', 'nexttrack',
'nonconvert', 'num0', 'num1', 'num2', 'num3', 'num4', 'num5', 'num6',
'num7', 'num8', 'num9', 'numlock', 'pagedown', 'pageup', 'pause', 'pgdn',
'pgup', 'playpause', 'prevtrack', 'print', 'printscreen', 'prntscrn',
'prtsc', 'prtscr', 'return', 'right', 'scrolllock', 'select', 'separator',
'shift', 'shiftleft', 'shiftright', 'sleep', 'space', 'stop', 'subtract', 'tab',
'up', 'volumedown', 'volumemute', 'volumeup', 'win', 'winleft', 'winright', 'yen',
'command', 'option', 'optionleft', 'optionright']
```

G Previous

Next **②**

© Copyright 2019, Al Sweigart Revision ae441d85.

Built with Sphinx using a theme provided by Read the Docs.