

Day 29. 거북이 마우스로 조정하기

어제는 키보드를 이용해 거북이를 조정해봤습니다. 오늘은 마우스로 거북이를 조정해 보도록 하겠습니다. 마우스로 클릭하는 곳으로 거북이를 이동하려면 어떻게 할까요?

마우스가 클릭하는 좌표를 알아낸 후, 앞에서 배운 goto문을 이용해 이동시키면 되지 않을까? 하는 생각이 드네요.

mouse01.py 를 다음과 같이 작성해 실행해보세요.

mouse01.py

```
import turtle

turtle.setup(500, 500)
turtle.shape("turtle")

def goto_mouse_click_position(x, y):
    print("x :" + str(x) + ", y :" + str(y))
    turtle.goto(x, y)

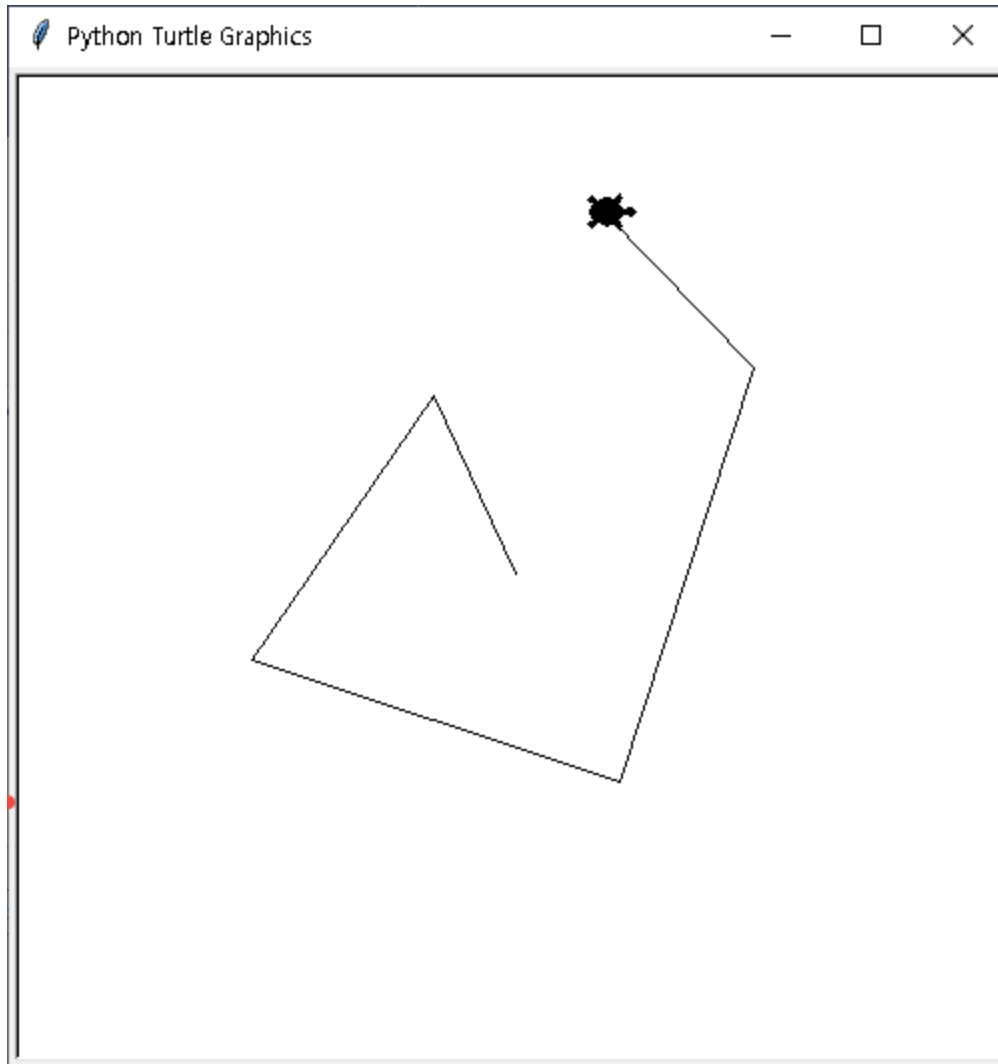
def mouse_right_button_click(x, y):
    print("right button click - x :" + str(x) + ", y :" + str(y))

def mouse_wheel_push(x, y):
    print("wheel push - x :" + str(x) + ", y :" + str(y))

turtle.onscreenclick(goto_mouse_click_position)
turtle.onscreenclick(mouse_wheel_push, btn = 2)
turtle.onscreenclick(mouse_right_button_click, btn = 3)

turtle.done()
```

실행된 결과는 다음과 같습니다.



```
def goto_mouse_click_position(x, y):  
    print("x :" + str(x) + ", y :" + str(y))  
    turtle.goto(x, y)
```

x와 y 2개의 매개변수를 받아들이는 `goto_mouse_click_position()` 함수를 선언했습니다. 해당 함수가 실행되면 x와 y의 값을 출력하고 (x,y)좌표로 거북이를 이동시킵니다.

```
def mouse_right_button_click(x, y):  
    print("right button click - x :" + str(x) + ", y :" + str(y))
```

x와 y값을 출력하는 `mouse_right_button_click()` 함수를 선언했습니다.

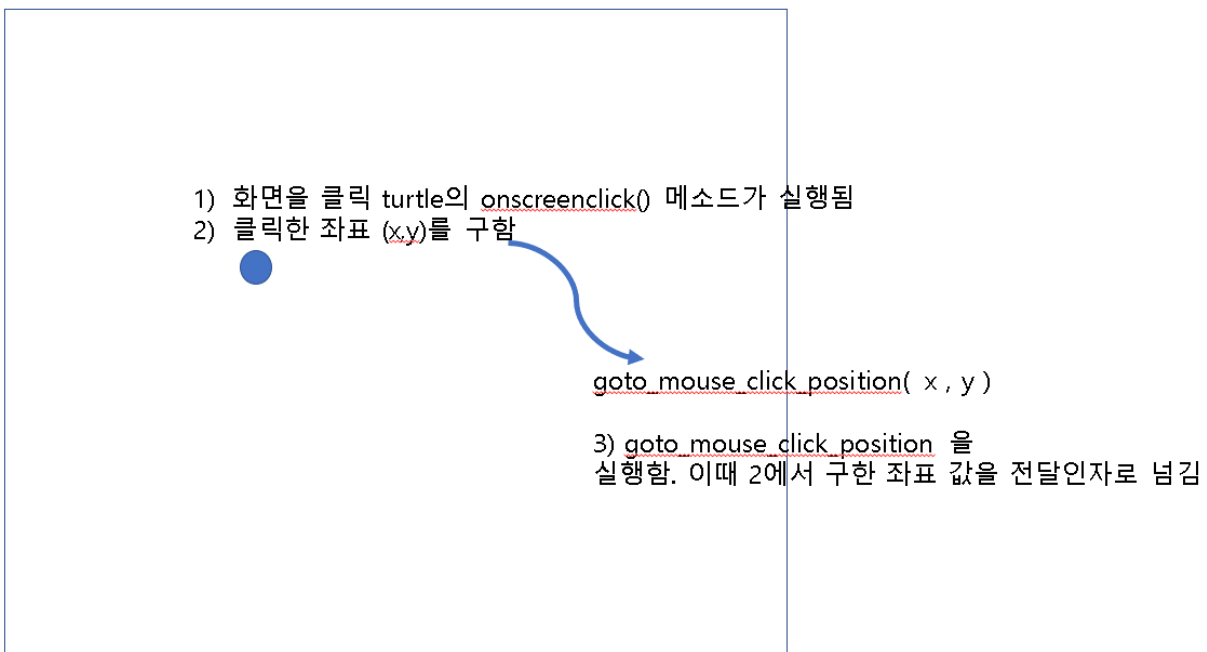
```
def mouse_wheel_push(x, y):  
    print("wheel push - x :" + str(x) + ", y :" + str(y))
```

x와 y값을 출력하는 mouse_wheel_push() 함수를 선언했습니다.

```
turtle.onscreenclick(goto_mouse_click_position)
```

turtle이 가지고 있는 onscreenlick() 메소드는 여러분이 만든 함수를 전달인자로 받습니다. 여러분이 화면을 클릭했을 때 전달인자로 받은 함수를 대신 실행하는 역할을 수행합니다.

아래 그림과 같이 화면을 클릭할 때 turtle의 onscreenlick()메소드가 실행되고, 해당 메소드 안에서는 클릭한 x,y좌표를 구하게 됩니다. 그리고 여러분이 전달인자로 받은 함수를 onscreenlick()안에서 실행하게 됩니다. 이때 x,y좌표를 전달인자로 받은 함수에게 전달하게 됩니다.



```
turtle.onscreenclick(mouse_wheel_push, btn = 2)
```

마우스 휠 버튼을 클릭할 경우 mouse_wheel_push함수를 호출합니다.

```
turtle.onscreenclick(mouse_right_button_click, btn = 3)
```

마우스 오른쪽 버튼을 클릭할 경우 `mouse_right_button_click` 함수를 호출합니다.

숙제

1. 마우스 휠 버튼을 누르면 거북이를 들어 올리세요. 마우스 오른쪽 버튼을 누르면 거북이를 내려놓으세요.
2. 윈도우 그림판처럼 색깔을 선택해서 거북이로 그림을 그리고 싶습니다. 어떻게 프로그래밍 하면 될까요? 생각을 말해보세요.