

# NEW GRAPHICS를 이용한 EVENT

## NEW GRAPHICS와 EVENT

EVENT는 GUI 프로그램에서 유저의 입력에 따라서 프로그램이 반응하는 것을 뜻합니다. 특히 GUI중에서 WIDGET\_WINDOW객체의 EVENT는 NEW GRAPHICS의 객체에서도 동일하게 사용이 가능합니다.

## WINDOW에서 받는 EVENT

WINDOW는 그래프나 그림이 뜨는 창이기 때문에, 버튼을 누르는 EVENT가 없습니다. 대신 WINDOW에 마우스를 누르고 땄 때 발생하는 EVENT와 WINDOW 안에서 마우스가 이동하고 있을 때 발생하는 EVENT가 있습니다. 이러한 EVENT는 WINDOW함수에서 (EVENT이름)\_HANDLER속성으로 지정할 수 있습니다. 예를 들어 MOUSE\_DOWN\_HANDLER속성에 EVENT의 함수를 등록하면, 마우스를 누를 때 그 함수에서 정의된 작업을 시행합니다.

## 개별 EVENT 함수가 받는 변수들

Handler Type	Syntax
Mouse_Down	Result = FUNCTION_NAME(Window, X, Y, Button, KeyMods, Clicks)
Mouse_Motion	Result = FUNCTION_NAME(Window, X, Y, KeyMods)
Mouse_Up	Result = FUNCTION_NAME(Window, X, Y, Button)
Mouse_Wheel	Result = FUNCTION_NAME(Window, X, Y, Delta, KeyMods)

각각의 EVENT는 몇 가지의 변수를 받습니다. 공통적으로 등장하는 변수는 Window, X, Y가 있고, 그 외 Button, Keymods, Clicks, Delta는 몇 가지의 EVENT에만 존재하는 변수들입니다. 여기서 각각의 변수는 다음과 같은 의미를 가지고 있습니다.

- Window는 EVENT가 일어난 Window의 ID입니다. 대부분의 EVENT는 발생하면 이것을 변화시키게 될 것입니다.
- X, Y는 커서가 있는 곳의 좌표입니다. 이것은 시스템 좌표로 그 위치의 픽셀 값을 주게 됩니다.
- Button은 어느 버튼으로 클릭을 했는지 알려주는 값입니다. 1은 왼쪽, 2는 가운데(휠), 4는 오른쪽입니다. 두개의 버튼을 이용할 경우 두 값의 합이 들어오게 됩니다. 예를 들어 왼쪽과 오른쪽을 누르면 5가 나옵니다.

- Keymods는 키보드의 Shift, Control, Caps lock, Alt 버튼을 누른 상태에서 EVENT가 발생한 것을 알려주는 변수입니다. Shift는 1, Control은 2, Caps lock은 4, Alt는 8의 값을 가지고 있습니다.

- Delta는 휠이 얼마나 움직였는지 알려주는 값입니다. 위로 올리면 +, 아래로 내리면 -의 값을 줍니다. 또한 더 많이 움직이면 더 큰 값을 줍니다.

- Clicks는 마우스를 몇 번 클릭했는지 알려주는 값입니다. 클릭 횟수를 그대로 값으로 줍니다.

## EVENT를 구현할 함수를 만들기

이제는 실제로 EVENT를 구현할 함수를 만들어 보겠습니다. 이번에 만들 EVENT는 매우 간단한 것으로, 마우스 커서가 닿은 지점의 X, Y값을 Print로 출력하는 함수가 될 것입니다. 함수는 다음과 같습니다.

```
FUNCTION DIM_MOTION_EVENT, WIN, X, Y, KeyMods
  IF KeyMods EQ 1 THEN print, X, Y
  return,0
END
```

매우 간단한 함수이지만, Mouse\_Motion 타입의 함수가 가져야할 문법적인 부분은 다 가지고 있습니다. WIN에는 window의 ID가 들어올 것이고, X, Y에 좌표가 들어오고 Keymods에 누르고 있는 키의 값이 들어올 것입니다. 또한 조건문을 통해서 Shift를 누를 때만 동작하도록 했습니다. 이것을 dim\_motion\_event.pro의 파일로 저장을 하고, 컴파일을 합니다.

## WINDOW에 EVENT 함수를 등록하기

이제 간단히 작성한 EVENT함수를 WINDOW에 등록하도록 하겠습니다. 우선 윈도우를 만들어서, 그 안에다가 간단한 Plot을 그린 후에 EVENT를 등록하는 과정입니다.

```
WIN=WINDOW()
P=PLOT(/TEST,/CURRENT)
WIN.MOUSE_MOTION_HANDLER='DIM_MOTION_EVENT'
```

위와 같이 하면 새로운 NEW GRAPHICS의 윈도우가 뜨면서 /TEST의 속성에 기본으로 내장된 그래프를 작성해줍니다. 마지막으로 WIN에 있는 MOUSE\_MOTION\_HANDLER 속성에 함수의 이름을 넣으면, EVENT가 등록이 됩니다. 출력된 그래픽 창에 마우스를 움직여보다가 Shift를 누르고 움직여보면 위의 경우 IDLDE의 스크립트에 숫자들이 출력되는 것을 알

수 있습니다. 지금 나타나는 값은 마우스가 현재 있는 위치의 픽셀좌표 입니다.

## UVALUE의 이용

EVENT를 더 잘 이용하기 위해서는 더 많은 정보가 EVENT에 있어야 할 필요가 있습니다. 이런 정보를 옮기는 데 편의를 주기 위해서 WINDOW에는 UVALUE라는 속성이 있습니다. 이것을 통해서 WINDOW에 자신이 원하는 종류의 데이터를 WINDOW에 넣고 다닐 수 있습니다. 우선은 UVALUE에 PLOT 객체인 P와 TEXT 객체인 T를 넣고 이것을 통해서 픽셀 값을 좌표 값으로 바꿔서 출력하는 EVENT로 바꿔보도록 하겠습니다. 우선 윈도우에 TEXT를 생성하고 P와 T를 구조체로 UVALUE에 넣도록 하겠습니다.

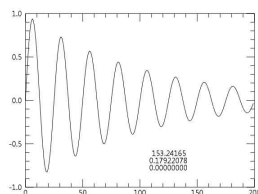
```
T = text(0.5,0.2,'TEXT',/CURRENT)
WIN.UVALUE = {P:P , T:T}
```

이제 dim\_motion\_event.pro파일을 다음과 같이 바꿔줍니다.

```
FUNCTION DIM_MOTION_EVENT,WIN, X, Y, KeyMods
  IF KeyMods EQ 1 THEN BEGIN
    P = WIN.UVALUE.P
    T = WIN.UVALUE.T
    XY = P.ConvertCoord(X,Y,/DEVICE, /TO_DATA)
    T.String = String(XY)
  ENDIF
  return,0
END
```

이 함수에서 ConvertCoord는 Plot객체의 내부 함수로 시스템 좌표를 Graph의 좌표로 변경시켜주는 역할을 하게 됩니다.

주어진 윈도우를 새로 그릴 필요 없이 바로 컴파일 시키면, 바로 EVENT가 변경되어 적용됩니다. 다시 Shift를 누르고 마우스를 움직여보면 우측의 그림과 같이 PLOT의 좌표에 맞춰서 값이 출력되는 것을 볼 수 있을 것입니다.



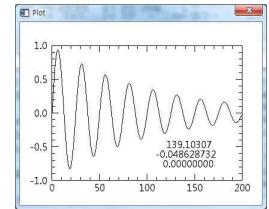
## WINDOW와 WIDGET\_WINDOW

지금까지 했던 모든 WINDOW에서의 작업은 GUI에서 사용이 가능한 NEW GRAPHICS전용 위젯인 WIDGET\_WINDOW에서도 사용이 가능한 것들입니다. 간단하게 다음과 같은 GUI 코드를 작성해봅시다.

```
PRO WINDOWWIDGET
TLB = WIDGET_BASE(/column, title='WINDOW
```

```
EXAMPLE',xs=405,ys=305,tlb_frame_attr=1)
WND = WIDGET_WINDOW(TLB, XSIZE=400, YSIZE=300,
mouse_motion_handler='DIM_MOTION_EVENT')
WIDGET_CONTROL, TLB, /REALIZE
WIDGET_CONTROL, WND, GET_VALUE=WIN
WIN.SELECT
P = PLOT(/TEST,/CURRENT)
T = TEXT(0.5,0.2,'TEXT')
WIN.UVALUE = {P:P , T:T}
XMANAGER, 'IONOSONDE_GUI', TLB
END
```

작성이 완료되어 실행을 하고 Shift를 누르고 마우스를 움직이면 오른쪽처럼 아래쪽의 TEXT객체의 숫자가 마우스의 움직임에 따라서 바뀌는 것을 볼 수 있습니다. 이러한 과정을 통해서 EVENT를 WIDGET\_WINDOW에 적용시킬 수 있습니다.



## 추가적으로 알아야 할 것들

- HANDLER를 만드는 데 있어서 HANDLER함수의 이름이나, 변수의 이름은 자유롭게 정할 수 있습니다. 그러나 헛갈리지 않기 위해서는 일관된 이름을 붙여주는 것이 좋습니다.
- EVENT를 구현하는 HANDLER속성은 문자열의 형태로 EVENT의 함수 이름이 들어가게 됩니다. 또한 이러한 속성을 지정하는 것은 WINDOW객체가 생성된 뒤에 하는 방법 외에도 객체를 생성하면서 지정하는 방법도 있습니다.

```
WIN2=WINDOW(MOUSE_MOTION_HANDLER='DIM_MOTION_EVENT')
P=PLOT(/TEST, /CURRENT)
```

위 코드는 'WINDOW에 EVENT 함수를 등록하기'에 있는 코드와 같은 역할을 합니다.

- 먼저 PLOT등의 GRAPHIC을 만들었을 경우 이러한 그래픽들은 WINDOW의 속성을 가지고 있습니다. 이 속성은 WINDOW 객체의 ID이므로, 여기다가 EVENT를 구현시킬 수 있습니다.

```
P=PLOT(/TEST)
WIN3=P.WINDOW
WIN3.MOUSE_MOTION_HANDLER='DIM_MOTION_EVENT'
```

위의 코드를 통해서 EVENT함수를 등록시키는 것도 가능합니다.

김종현 / SELab IT Solution Section