



오빠두엑셀 기초강좌 E-BOOK

1

Free Tutorial

2

Business Excel

3

Basic to Expert

한국어

#엑셀 VBA3-5

>> KeyPressAPI

키보드입력

인식하기



1. User32.dll 라이브러리 사용하기

User32.dll 라이브러리는 마이크로소프트의 운영체제인 윈도우에 기본적으로 설치되는 라이브러리 파일입니다. 라이브러리 파일은 '단독'으로 실행될 수 없는 파일이며 주로 다른 프로그램에서 해당 라이브러리 파일을 참조하여 동작하게 됩니다.

그 중 User32.dll 라이브러리는 실행중인 윈도우의 GUI (Graphic User Interface)와 밀접한 연관이 있는 라이브러리로서, 실행중인 프로그램과의 소통(예, 키보드/마우스 클릭 이벤트, 알림창 출력 등)을 주로 합니다. 기본적으로 윈도우 시스템에 큰 영향을 끼치지 않는 안전한 라이브러리이며 윈도우 운영체제에서 프로그램을 제작하기 위해서 반드시 숙지해야 할 라이브러리 중 하나입니다.

[링크] [마이크로소프트 User32.dll 라이브러리 설명 바로가기](https://docs.microsoft.com/en-us/windows/desktop/api/winuser/)

<https://docs.microsoft.com/en-us/windows/desktop/api/winuser/>

이번 강의에서는 크게 아래의 3 가지 함수를 사용합니다.

1. FindWindowA 함수

<https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-findwindowa>

: 윈도우에서 실행중인 프로그램 중 하나를 선택하여 Top Level 에서 실행되는 정보를 받아옵니다.

```

'// TOP LEVEL 윈도우 정보를 받습니다.
'// 더 자세한 내용은 아래 링크를 참고하세요
'// For more details, please refer to below links at MSDN
'// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-findwindowa
Private Declare PtrSafe Function FindWindow Lib "user32" Alias "FindWindowA" _
    (ByVal lpClassName As String, _
    ByVal lpWindowName As String) As LongPtr
  
```

2. PeekMessageA 함수

<https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-peekmessagea>

: FindWindowA 함수를 통해 받아온 윈도우(프로그램)에서 입력된 메시지를 받아옵니다.

```

'// 입력된 키정보를 받습니다.
'// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-peekmessagea
Private Declare PtrSafe Function PeekMessage Lib "user32" Alias "PeekMessageA" _
    (ByRef lpMsg As msgKeyPress, ByVal hwnd As LongPtr, _
    ByVal wParamFilterMin As LongPtr, _
    ByVal wParamFilterMax As LongPtr, _
    ByVal wRemoveMsg As LongPtr) As LongPtr
  
```

3. PostMessageA 함수

<https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-postmessagea>

: PeekMessage 함수와는 반대로, 선택된 윈도우에 원하는 키 값을 입력합니다.

```

'// 키 값을 TOP LEVEL WINDOW에 Return 합니다.
'// https://msdn.microsoft.com/en-us/library/ms010658.aspx
Private Declare PtrSafe Function PostMessage Lib "user32" Alias "PostMessageA" _
    (ByVal hwnd As LongPtr, _
    ByVal wParam As LongPtr, _
    ByVal lParam As LongPtr, _
    lParam As Any) As LongPtr
  
```

2. 64 비트 VBA 호환성 문제 해결

User32.dll 라이브러리를 사용할 경우 64 비트 엑셀 VBA 와의 호환성 문제를 고려해야 합니다. User32.dll 에는 수많은 종류의 함수가 있습니다. 각 함수마다 사용되는 인수가 모두 다르기때문에, “이렇게 하면 모든 호환성 문제가 해결된다”라는 명확한 정답은 없습니다.

마이크로소프트에서 제시하는 호환성문제 해결방안은 크게 2 가지입니다.

자세한 설명 보러가기 ::

<https://docs.microsoft.com/en-us/office/vba/language/concepts/getting-started/64-bit-visual-basic-for-applications-overview>

1. User32.dll 라이브러리를 참조하여 함수를 선언할 경우, **PtrSafe** 를 사용하여 Private Declare **PtrSafe** Function 으로 선언합니다.

```

'/// TOP LEVEL 윈도우 정보를 받습니다.
'/// 더 자세한 내용은 아래 링크를 참고하세요
'/// For more details, please refer to below links at MSDN
'/// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-findwindowa
Private Declare PtrSafe Function FindWindow Lib "user32" Alias "FindWindowA" _
    (ByVal lpClassName As String, _
    ByVal lpWindowName As String) As LongPtr
'/// 자세한 내용은 위 링크에 보시면 됩니다.

```

2. Long 데이터 타입 대신, 64 비트에서도 호환 가능한 **LongPtr** 데이터타입을 사용합니다.

```

'/// 입력된 키정보를 받습니다.
'/// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-peekmessagea
Private Declare PtrSafe Function PeekMessage Lib "user32" Alias "PeekMessageA" _
    (ByRef lpMsg As msgKeyPress, ByVal hwnd As LongPtr, _
    ByVal wParamFilterMin As LongPtr, _
    ByVal wParamFilterMax As LongPtr, _
    ByVal wRemoveMsg As LongPtr) As LongPtr
'/// 자세한 내용은 위 링크에 보시면 됩니다.

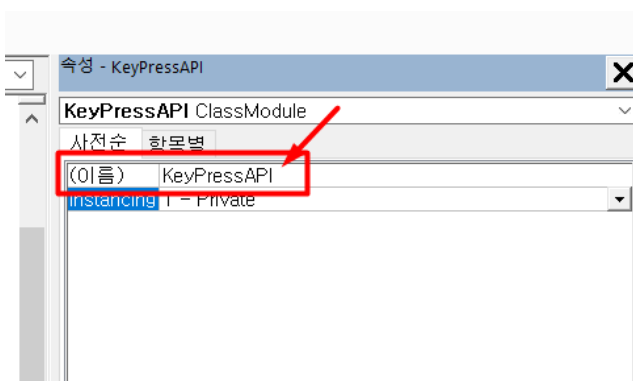
```

하지만 앞서 말씀드린 바와 같이, 함수마다 사용되는 인수는 각각 다르기 때문에 상황에 따라 LongPtr 데이터타입을 적용할 필요가 있습니다.

3. KeyPressAPI 클래스 모듈 전체 명령문

엑셀에서 키보드 입력을 인식받기위해 새로운 클래스모듈 개체를 생성합니다. 이번 강의에서는 64 비트를 기준으로 작성할 것이므로 32 비트/64 비트 엑셀 사용자 모두 사용가능합니다.

클래스 모듈을 생성한 뒤, 속성창에서 클래스모듈의 이름을 KeyPressAPI 로 변경해줍니다.



우선 첫번째로 아래 코드를 복사하여 붙여넣기 해줍니다. (각 타입과 변수 및 User32 라이브러리에서 불러온 사용자 지정함수를 선언합니다.) 자세한 내용은 **영상강의를 참고**해주세요.

```
'#####
#####
'# 해당 명령문에 대한 저작권은 오빠두엑셀 (http://oppadu.com) 에 있습니다.
#
'# 모든 정보는 Cretive Commns License 에 의해 저작권을 보호받습니다.
#
'# 영리를 목적으로 하지 않는 사용 및 공유는 허용됩니다. 반드시 저작자,
오빠두엑셀 (http://oppadu.com) 을 명시해야합니다. #
'# This VBA Code is protected by Creative Commons License.
#
'# All information can be posted, uploaded, shared at online for NON-commercial
use only. #
'# The Author, '오빠두엑셀 (http://oppadu.com) ' have to be mentioned when you post
this code. #
'#####
#####

Option Explicit

'// 마우스 포인트 위치 (x,y) 를 받아옵니다.
Private Type POINTAPI
    x As Long
    y As Long
End Type

'// TOP LEVEL WINDOW 의 키정보, 마우스위치정보를 받아올 메세지 Type 을 설정합니다.
(윈도우 Default 값)
Private Type msgKeyPress
    hwnd As LongPtr
    Message As LongPtr
    wParam As LongPtr
    lParam As LongPtr
    time As Long
    pt As POINTAPI
End Type

'// TOP LEVEL 윈도우 정보를 받습니다.
'// 더 자세한 내용은 아래 링크를 참고하세요
'// For more details, please refer to below links at MSDN
'// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-findwindowa
Private Declare PtrSafe Function FindWindow Lib "user32" Alias "FindWindowA" _
    (ByVal lpClassName As String, _
    ByVal lpWindowName As String) As LongPtr

'// 실행중인 Thread 에서 키입력을 받기위해 대기합니다.
'// 더 자세한 내용은 아래 링크를 참고하세요
'// For more details, please refer to below links at MSDN
'// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-waitmessage
Private Declare PtrSafe Function WaitMessage Lib "user32" () As LongPtr

'// 입력된 키정보를 받습니다.
```

```

'// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-peekmessagea
Private Declare PtrSafe Function PeekMessage Lib "user32" Alias "PeekMessageA"
-
    (ByRef lpMsg As msgKeyPress, ByVal hwnd As LongPtr, _
    ByVal wParam As LongPtr, _
    ByVal lParam As LongPtr, _
    ByVal wRemoveMsg As LongPtr) As LongPtr

'// 키 값을 TOP LEVEL WINDOW 에 Return 합니다.
'// https://msdn.microsoft.com/en-us/library/ms910658.aspx
Private Declare PtrSafe Function PostMessage Lib "user32" Alias "PostMessageA"
-
    (ByVal hwnd As LongPtr, _
    ByVal wParam As LongPtr, _
    ByVal lParam As LongPtr, _
    lParam As Any) As LongPtr

'// 받아온 키 정보를 처리가능한 데이터로 변환합니다.
'// https://docs.microsoft.com/en-us/windows/desktop/api/winuser/nf-winuser-translatemessage
Private Declare PtrSafe Function TranslateMessage Lib "user32" _
    (ByRef lpMsg As msgKeyPress) As LongPtr

Private Const WM_KEYDOWN As LongPtr = &H100
Private Const PM_REMOVE As LongPtr = &H1
Private Const WM_CHAR As LongPtr = &H102
Private blnExit As Boolean

'// 키입력 이벤트가 일어났을때 발생시킵니다.
Public Event KeyPressed(ByVal KeyAscii As LongPtr, _
    ByVal KeyCode As LongPtr, _
    ByVal Target As Range, _
    ByRef Cancel As Boolean)

```

두번째로 클래스모듈의 2 개의 프로시저 명령문을 작성합니다. 마찬가지로 복사해서 생성한 클래스모듈에 붙여넣기 해주세요. (*프로시저 구문은 위에서 붙여넣기 한 명령문 아래로 이어서 붙여넣기 해주어야 합니다.)

```

#####
#####
'# 해당 명령문에 대한 저작권은 오빠두엑셀 (http://oppadu.com) 에 있습니다.
#
'# 모든 정보는 Cretive Commns License 에 의해 저작권을 보호받습니다.
#
'# 영리를 목적으로 하지 않는 사용 및 공유는 허용됩니다. 반드시 저작자,
오빠두엑셀 (http://oppadu.com) 을 명시해야합니다. #
'# This VBA Code is protected by Creative Commons License.
#
'# All information can be posted, uploaded, shared at online for NON-commercial
use only. #
'# The Author, '오빠두엑셀 (http://oppadu.com) ' have to be mentioned when you post
this code. #
#####
#####
Public Sub StartKeyPress()

```

```

Dim msgMessage As msgKeyPress
Dim blnCancel As Boolean
Dim iMessage As LongPtr
Dim iKeyCode As LongPtr
Dim lXLhwnd As LongPtr

On Error GoTo errHandler
Application.EnableCancelKey = xlErrorHandler

'// blnExit 이 TRUE 가 되면 키입력인식을 종료합니다.
blnExit = False

'// 현재 실행중인 TOP LEVEL WINDOW (프로그램 창)을 인식하여 longptr (64 비트
호환)로 받아옵니다.
lXLhwnd = FindWindow("XLMAIN", Application.Caption)

Do
    '// 실행중인 Thread 에서 키입력을 받기위해 대기합니다.
    WaitMessage

    '// 오류 방지를 위해 blnExit 를 Cross-Check 합니다.
    If blnExit Then Exit Do

    '// 키 입력을 받아옵니다.
    '// PeekMessage 값이 존재하는지 (>0) Cross-Check 합니다.
    If PeekMessage(msgMessage, lXLhwnd, wmKeyDown, wmKeyDown, PM_REMOVE)
Then
        'Store the virtual key code for later use.
        iMessage = msgMessage.Message
        iKeyCode = msgMessage.WParam    '// <- 입력된 키보드 Input 입니다.

        '받아온 msgKeyPress(=Virtual Message)를 실제 처리가능한 정보로
        변환합니다.
        TranslateMessage msgMessage
        PeekMessage msgMessage, lXLhwnd, wmChar, wmChar, PM_REMOVE

        '// 오류 방지를 위해 blnCancel 를 Cross-Check 합니다.
        blnCancel = False

        'KeyPressed 이벤트를 실행합니다.
        RaiseEvent KeyPressed(msgMessage.WParam, iKeyCode, Selection,
blnCancel)

        'TOP LEVEL WINDOW 에 입력값 Return 합니다.
        If Not blnCancel Then
            PostMessage lXLhwnd, iMessage, iKeyCode, 0
        End If
    End If
errHandler:
    'StopKeyPress 까지 DoEvents 로 Loop 합니다.
    DoEvents
    Loop Until blnExit
End Sub

Public Sub StopKeyPress()
    'Set this boolean flag to exit the above loop.
    blnExit = True
End Sub

```

4. 원하는 시트에 KeyPressAPI 클래스 모듈 적용하기

이제 원하는 시트에 키보드 입력이 들어왔을 때, 방금 작성한 KeyPressAPI 클래스 모듈을 사용하여 시트에 입력된 키보드 정보를 받아보도록 하겠습니다.

이번 강의에서는 예제파일의 “Sheet1”에 키보드 입력이 들어왔을 때, 해당 키의 Ascii(아스키)코드를 받아서 메시지박스로 출력하는 명령문을 작성하겠습니다.

[아스키코드 자세한 설명 바로가기\(나무위키\)](https://namu.wiki/w/%EC%95%84%EC%8A%A4%ED%82%A4%20%EC%BD%94%EB%93%9C)

<https://namu.wiki/w/%EC%95%84%EC%8A%A4%ED%82%A4%20%EC%BD%94%EB%93%9C>

VBA 편집창에서 Sheet1 을 더블클릭 하신 뒤, 아래 명령문을 복사하여 붙여넣기합니다.

```
'#####
'#####
'# 해당 명령문에 대한 저작권은 오빠두엑셀(http://oppadu.com)에 있습니다.
'#
'# 모든 정보는 Cretive Commns License 에 의해 저작권을 보호받습니다.
'#
'# 영리를 목적으로 하지 않는 사용 및 공유는 허용됩니다. 반드시 저작자,
오빠두엑셀(http://oppadu.com)을 명시해야합니다. #
'# This VBA Code is protected by Creative Commons License.
'#
'# All information can be posted, uploaded, shared at online for NON-commercial
use only. #
'# The Author, '오빠두엑셀(http://oppadu.com)' have to be mentioned when you post
this code. #
'#####
'#####
Option Explicit

Dim WithEvents KeyPressWatcher As KeyPressAPI

Sub Start_KeyPress()

If KeyPressWatcher Is Nothing Then
    Set KeyPressWatcher = New KeyPressAPI
End If

KeyPressWatcher.StartKeyPress

End Sub

Sub End_KeyPress()

If KeyPressWatcher Is Nothing Then Exit Sub
KeyPressWatcher.StopKeyPress

End Sub

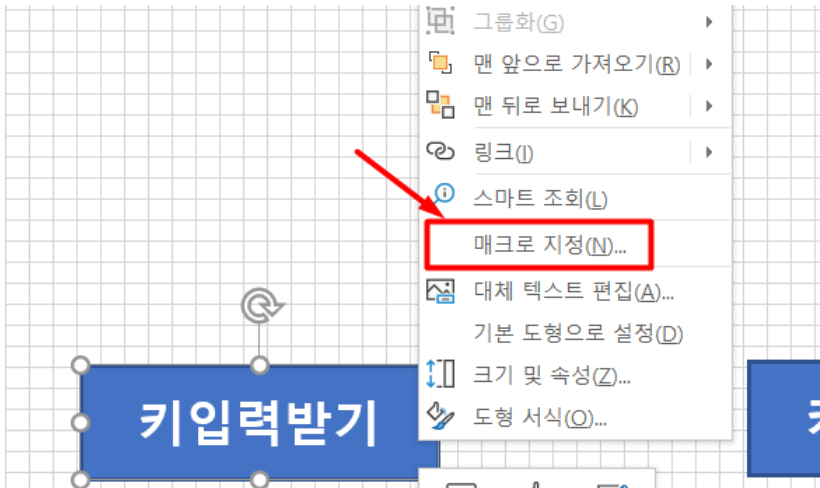
Private Sub KeyPressWatcher_KeyPressed(ByVal KeyAscii As LongPtr, ByVal KeyCode
As LongPtr, ByVal Target As Range, Cancel As Boolean)

MsgBox "[[알림]]" & vbNewLine & _
    "현재 입력한 키는 " & Chr(CLng(KeyAscii)) & " 입니다." & vbNewLine & _
    "키의 아스키코드는 " & CLng(KeyAscii) & " 입니다."

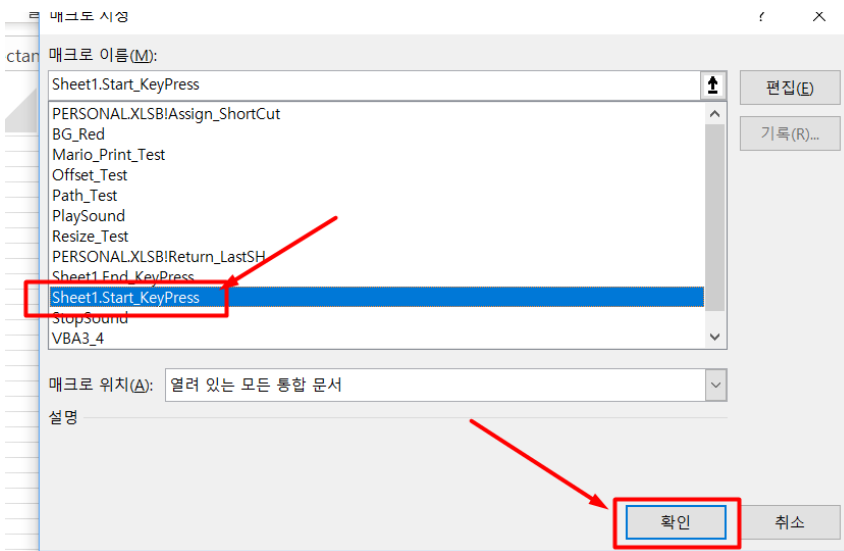
End Sub
```

5. 해당 시트에 매크로 실행을 위한 버튼 만들기

이제 모든 명령문과 모듈 작성이 끝났습니다. VBA 편집창을 닫고 엑셀 시트로 돌아옵니다. Sheet1 을 보시면 2 개의 버튼이 있습니다. 각 버튼을 클릭한 뒤 [매크로지정]에 들어갑니다.



[키입력받기]버튼에는 Start_KeyPress 명령문을, [키입력중단]버튼에는 End_KeyPress 명령문을 입력합니다.



이제 모든 단계가 완료되었습니다. [키입력받기]버튼을 클릭한 뒤 아무 키나 입력하시면 실행중인 엑셀 프로그램에서 입력된 키보드정보를 받아와 아래처럼 메시지박스를 출력합니다.

