

### 1. 사용자 정의 폼 추가하기

사용자가 원하는 파일을 선택할 수 있도록 **사용자정의폼을 추가**합니다. 예제파일을 다운받은 후 단축키 ALT + F11 을 눌러 매크로편집기를 실행합니다.

상단의 **[삽입] - [사용자 정의 폼]**을 클릭하여 아래와 같이 사용자정의폼을 생성합니다.

속성창이 안 보일 경우, 키보드 F4 키를 누르거나 또는 상단의 **[보기] - [속성 창]**을 선택합니다. 도구상자가 안 보일 경우, 상단의 **[보기] - [도구상자]**를 선택하면 도구상자 창이 나타납니다.

속성 - frmWBSelect

사전순	항목별
(이름)	frmWBSelect
BackColor	&H8000000E&
BorderColor	&H80000012&

종류 : 텍스트상자  
이름 : txtFilter

종류 : 커맨드버튼  
이름 : btnSelect

파일 선택해 주세요.

병합할 시트 이름 필터

병합할 파일 선택

선택된 파일 병합

종류 : 리스트상자  
이름 : lstWB  
Column Width : 800

사용자정의폼을 생성합니다.

종류 : 커맨드버튼  
이름 : btnMerge

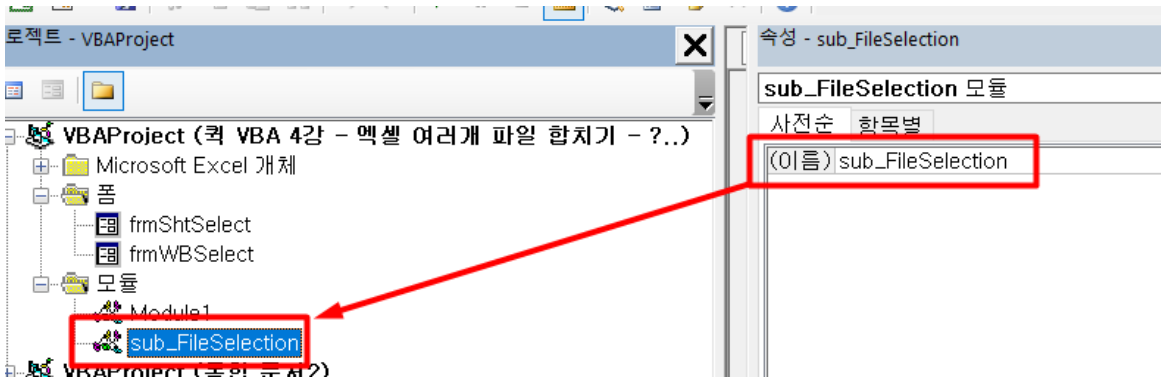


www.oppadu.com

오빠두엑셀 무료 E-Book

## 2. 파일선택창 모듈 추가하기

매크로 편집창에서 [삽입] - [모듈]을 클릭하여 새로운 모듈을 생성합니다. 모듈의 이름은 'sub\_FileSelection'으로 변경합니다.



새로운 모듈을 삽입한 뒤, 모듈의 이름을 변경합니다.

이전 퀵 VBA 1 강에서 다루었던 파일선택창 마스터코드를 복사하여 붙여넣기합니다. 파일선택창 마스터코드에 대한 자세한 내용은 [관련 포스트를 참고](#)하세요.

```
Public Function Multiple_FileDialog(Optional Title As String = "파일을 선택하세요", Optional
FilterName As String = "엑셀파일", _
Optional FilterExt As String = "*.xls; *.xlsx; *.xlsm", Optional InitialFolder As String
= "", _
Optional InitialView As MsoFileDialogView = msoFileDialogViewList, Optional
MultiSelection As Boolean = True) As String

Dim FDG As FileDialog
Dim Selected As Integer: Dim i As Integer
Dim ReturnStr As String

Set FDG = Application.FileDialog(msoFileDialogFilePicker)

With FDG
.Title = Title
.Filters.Add FilterName, FilterExt
.InitialView = InitialView
.InitialFileName = InitialFolder
.AllowMultiSelect = MultiSelection
Selected = .Show

If Selected = -1 Then
For i = 1 To FDG.SelectedItems.Count - 1
ReturnStr = ReturnStr & FDG.SelectedItems(i) & ", "
```



```
Next i
ReturnStr = ReturnStr & FDG.SelectedItems(.SelectedItems.Count)

Multiple_FileDialog = ReturnStr
ElseIf Selected = 0 Then
MsgBox "선택된 파일이 없으므로 프로그램을 종료합니다."
End
End If

End With

End Function
```

### 3. '병합할 파일 선택' 버튼 명령문 작성

유저폼에서 '병합할 파일 선택' 버튼을 더블클릭하여 클릭 이벤트 명령문을 작성합니다.

```
Private Sub btnSelect_Click()

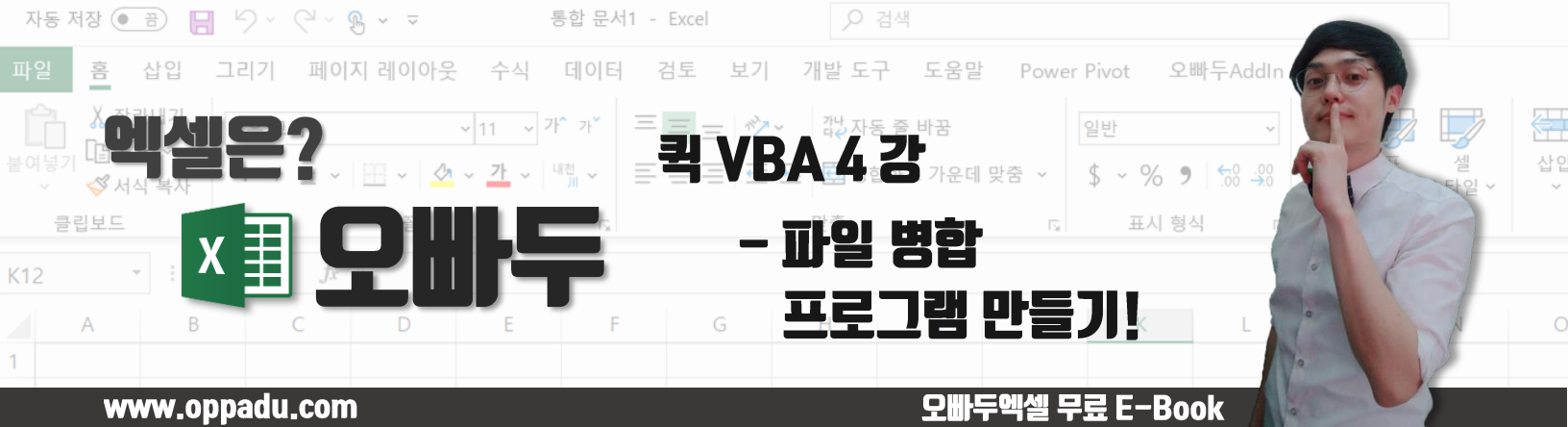
Dim strFilePath As String
Dim varFilePaths As Variant: Dim varFilePath As Variant

strFilePath = Multiple_FileDialog
varFilePaths = Split(strFilePath, ", ")

Me.lstWB.Clear

For Each varFilePath In varFilePaths
Me.lstWB.AddItem varFilePath
Next

End Sub
```



#### 4. '선택된 파일 병합' 버튼 명령문 작성

유저폼에서 '선택된 파일 병합' 버튼을 더블클릭하여 클릭 이벤트 명령문을 작성합니다.

##### 4-A. 변수설정

명령문에서 사용할 변수를 설정합니다. 이전 **퀵 VBA 3강, 시트합치기 명령문**에서 사용했던 변수세트와 동일하게 변수를 생성합니다. 변수설정방법에 대한 자세한 내용은 기초강의에서 다룬 적이 있으니 [관련 포스트를 참고](#)하세요.

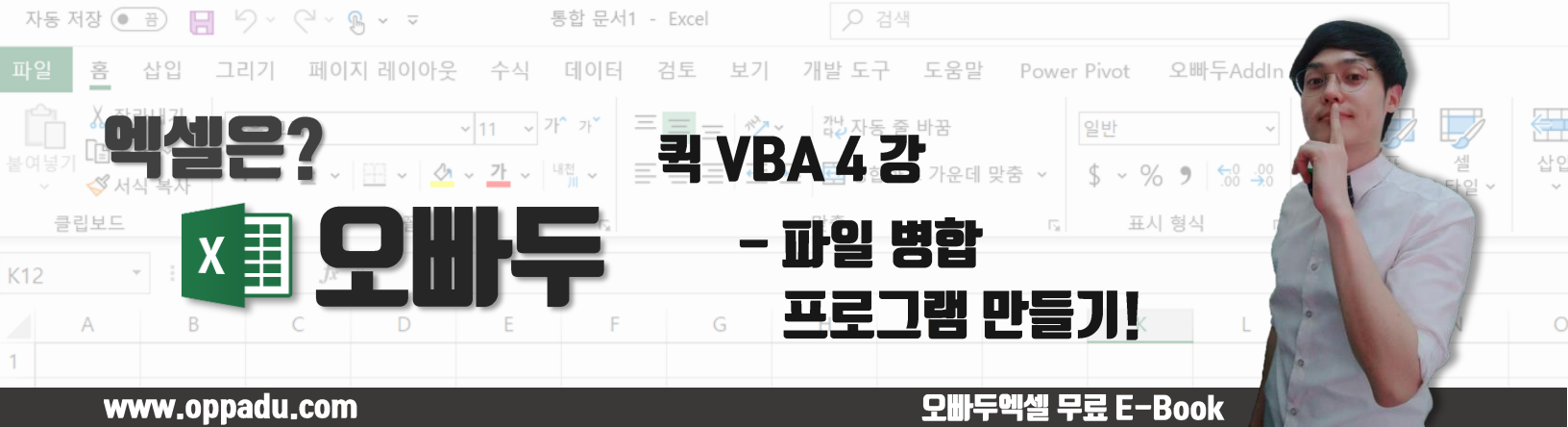
- |           |              |  |
|-----------|--------------|--|
| 1. WB     | as Workbook  | : 병합할 엑셀파일입니다.                         |
| 2. WS     | as Worksheet | : 병합할 엑셀파일에 작성된 각각의 워크 시트입니다.          |
| 3. toWS   | as Worksheet | : 각 WS 의 내용을 취합할 활성화된 워크 시트입니다.        |
| 4. rng    | as Range     | : 각 WS 에서 취합될 범위입니다.                   |
| 5. i      | as Long      | : 리스트상자에 선택된 엑셀파일들의 순번입니다.             |
| 6. j      | as Long      | : 복사한 rng 를 toWS 에 붙여넣기 할 행 번호입니다.     |
| 7. endCol | as Long      | : rng 의 넓이를 계산하기 위한 마지막 열번호입니다.        |
| 8. endRow | as Long      | : rng 의 넓이를 계산하기 위한 마지막 행번호입니다.        |
| 9. strWS  | as String    | : 이전 시트합치기 명령문에서만 사용된 변수이므로 지워도 무방합니다. |

##### 4-B. 오류방지

유저폼의 리스트상자에 선택된 엑셀파일이 없이 '파일 병합' 버튼을 눌러 명령문을 실행할 경우, 참조할 파일주소가 없으므로 오류가 발생하게 됩니다. 따라서 **리스트상자에 선택된 항목이 없을 경우 '파일을 선택하세요'** 라는 안내 메시지를 띄워줍니다.

```

'// 오류방지
If Me.lstWB.ListCount = 0 Then
    MsgBox "병합할 파일을 선택하세요."
    Exit Sub
End If
    
```



#### 4-C. 리스트상자에 선택된 엑셀파일을 하나씩 실행/종료하기

파일 선택하기 명령문을 통해 리스트상자에 입력된 각각의 **엑셀파일들을 실행하고 종료하는 명령문**을 작성합니다.

```
For i = 0 To Me.lstWB.ListCount - 1
    Set WB = Application.Workbooks.Open(Me.lstWB.List(i))

    ' // 각 엑셀파일에서 실행될 명령문 작성

WB.Close
Next
```

#### 4-D. 실행된 엑셀파일의 모든 워크시트의 이름 비교하기

실행된 **엑셀파일의 모든 워크시트를 하나씩 돌아가며 이름을 비교**합니다. IF 함수와 LIKE 연산자를 이용하여 각 시트이름을 비교한 뒤, 만약 해당 워크시트의 이름이 우리가 원하는 조건을 만족할 경우 명령문을 실행하도록 코드를 작성합니다.

```
For Each WS In WB.Worksheets
    If WS.Name Like Me.txtFilter.Value & "*" Then

        ' // 해당 워크시트에서 실행될 명령문 작성

    End If
Next
```

#### 4-E. 선택된 시트 내용 합치기

이전 시트합치기 강의에서 사용했던 명령문을 그대로 사용하여 **선택된 시트의 내용을 병합**합니다.

```
With WS
    endCol = .Cells(1, .Columns.Count).End(xlToLeft).Column
    endRow = .Cells(.Rows.Count, 1).End(xlUp).Row
    Set rng = .Range(.Cells(2, 1), .Cells(endRow, endCol))
    rng.Copy toWS.Cells(j, 1)
    j = j + rng.Rows.Count
End With
```



#### 4-F. 안내메세지 출력

파일 병합이 완료되었다는 안내메세지를 띄운 후, 사용자 정의폼을 닫아줍니다.

```
'// 안내메세지
MsgBox "파일 병합이 완료 되었습니다."
Unload Me
```

#### 4-G. 스크린 업데이트 중단/활성화 (매크로 동작속도 개선)

더욱 빠른 매크로 동작을 원할경우, Application.ScreenUpdating 속성을 활성/비활성화 하여 매크로 동작속도를 개선할 수 있습니다.

```
'// 스크린업데이트 중단
Application.ScreenUpdating = False

'// .....
'// 실행할 명령문 작성
'// .....

'//스크린 업데이트 활성화
Application.ScreenUpdating = True
```

#### 4-H. 파일 병합하기 버튼에 사용된 전체 명령문

```
Private Sub btnMerge_Click()

Dim WB As Workbook
Dim WS As Worksheet: Dim toWS As Worksheet
Dim rng As Range
Dim i As Long: i = 0: Dim j As Long
Dim endCol As Long: Dim endRow As Long
Dim strWS As String

'// 스크린업데이트 중단
Application.ScreenUpdating = False

'// 오류방지
If Me.lstWB.ListCount = 0 Then
MsgBox "병합할 파일을 선택하세요."
```



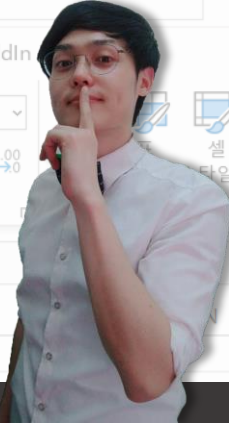
엑셀은?



오빠두

퀵 VBA 4강

- 파일 병합 프로그램 만들기!



www.oppadu.com

오빠두엑셀 무료 E-Book

```

Exit Sub
End If

'// 파일병합
Set toWS = ActiveSheet
j = toWS.Cells(toWS.Rows.Count, 1).End(xlUp).Row

For i = 0 To Me.lstWB.ListCount - 1
    Set WB = Application.Workbooks.Open(Me.lstWB.List(i))
    For Each WS In WB.Worksheets
        If WS.Name Like Me.txtFilter.Value & "*" Then
            With WS
                endCol = .Cells(1, .Columns.Count).End(xlToLeft).Column
                endRow = .Cells(.Rows.Count, 1).End(xlUp).Row
                Set rng = .Range(.Cells(2, 1), .Cells(endRow, endCol))
                rng.Copy toWS.Cells(j, 1)
                j = j + rng.Rows.Count
            End With
        End If
    Next
    WB.Close
Next

'// 안내메세지
MsgBox "파일 병합이 완료 되었습니다."
Unload Me

'//스크린 업데이트 활성화
Application.ScreenUpdating = True

End Sub

```

### 5. 유저폼 실행을 위한 단축키 생성

유저폼을 불러오기 위한 명령문을 작성합니다. 기존에 생성하였던 Module1 모듈에 아래의 명령문을 추가합니다.

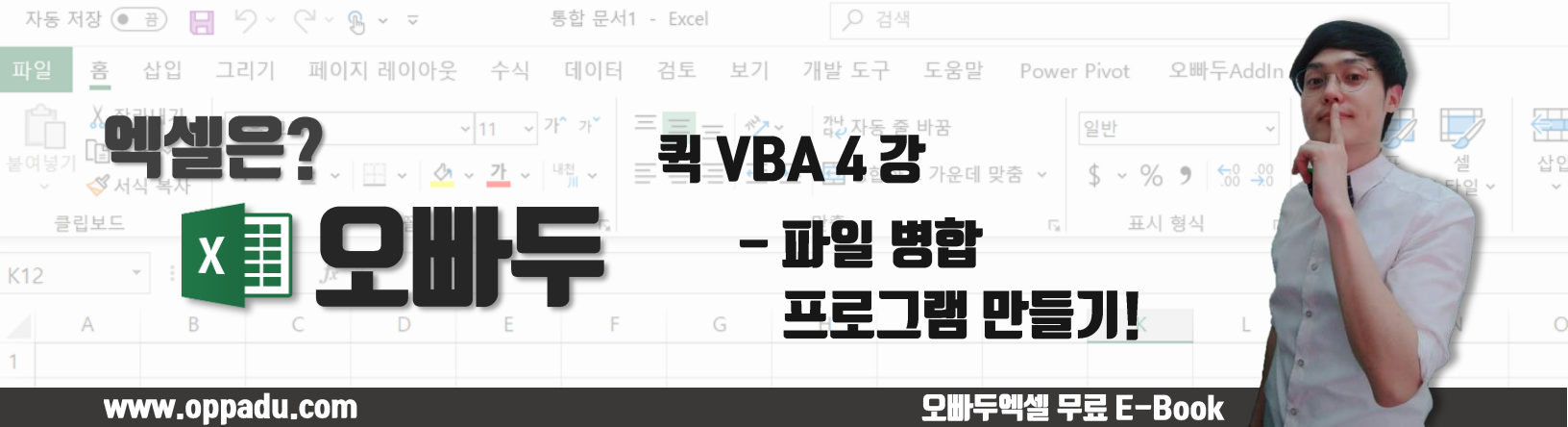
```

Sub Merge_Workbook()

frmWBSelect.Show

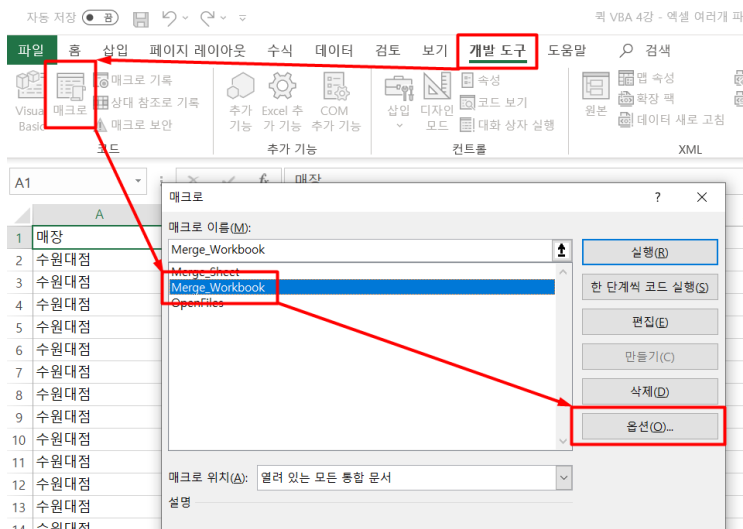
End Sub

```



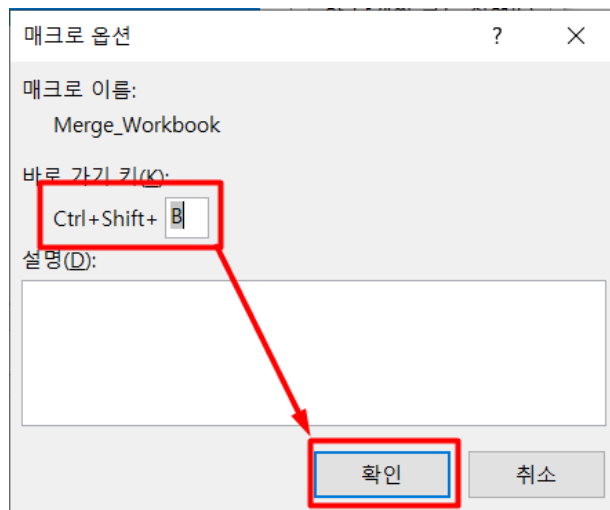
다시 엑셀 시트로 돌아온 뒤, 상단의 [개발도구] - [매크로] 를 선택합니다. 상단의 [개발도구]가 안 보일 경우, 개발도구 활성화 [관련 포스트를 참고](#)하세요.

[매크로]에서 우리가 방금 추가한 Merge\_Workbook 을 선택한 후, [옵션]을 클릭합니다.



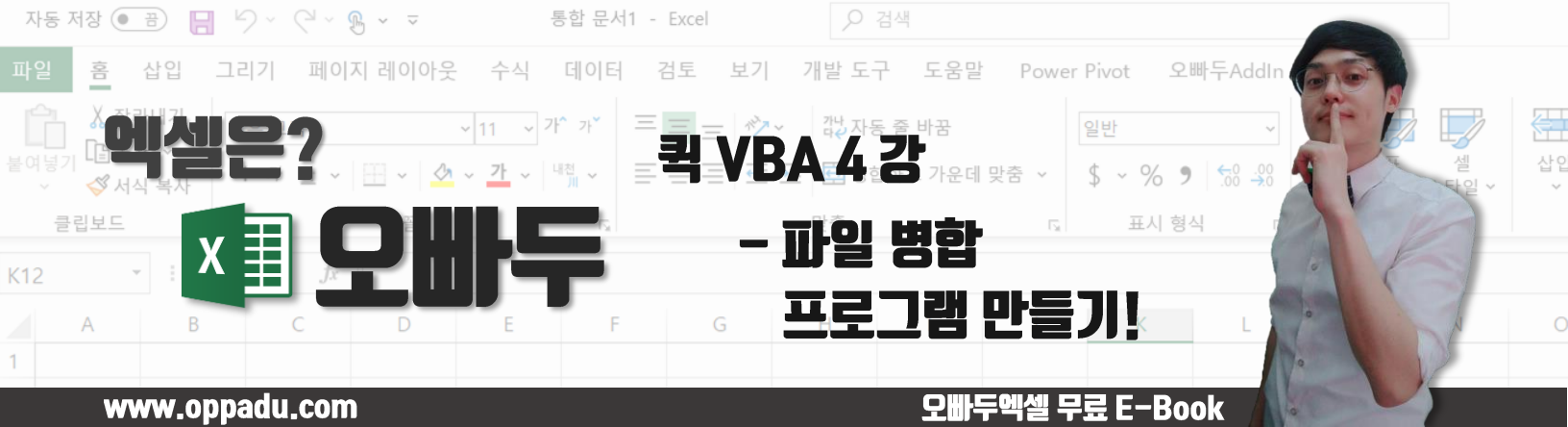
[개발도구] - [매크로] 를 클릭하여 방금 추가한 'Merge\_Workbook'의 옵션을 선택합니다.

옵션에서 해당 매크로를 실행하기 위해 단축키를 **Ctrl + Shift + B** 키로 지정한 뒤 [확인] 버튼을 눌러 마무리합니다.



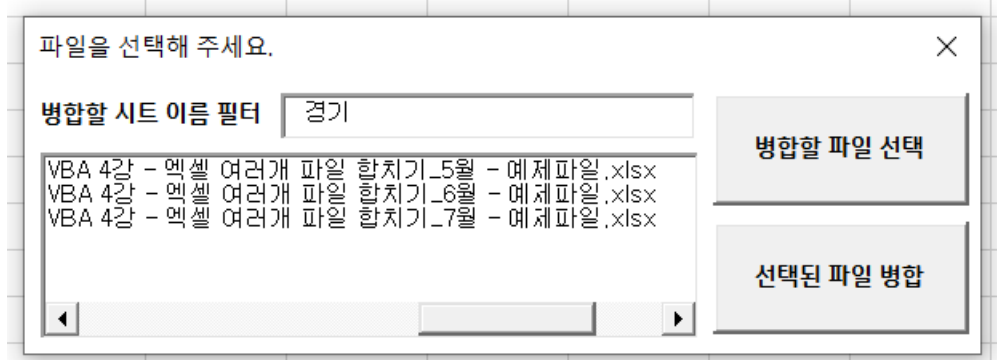
Merge\_Workbook 의 단축키를 Ctrl + Shift + B 로 지정합니다.





## 6. 파일 합치기 매크로 작성 완료 및 테스트

축하합니다. 파일 합치기 매크로 작성이 완료되었습니다. 이제 단축키 Ctrl + Shift + B 키를 눌러 언제든지 원하는 파일 및 그중에서도 원하는 시트만 선택하여 내용을 병합할 수 있습니다.



파일 합치기 프로그램 만들기가 완료되었습니다.