



# 오빠두엑셀 기초강좌 E-BOOK

1

Free Tutorial

2

Business Excel

3

Basic to Expert

한국어

#엑셀 VBA1-7

» VBA 개체

Method

(함수)



Copyright©2018 by 오빠두.  
All pictures cannot be copied without permission.

## 1. VBA에서 Method(함수)란 무엇인가?

VBA 기초 강의 잘 따라오고 계시나요? 이전 강의까지 개체(Object)를 자동차에, 속성(Property)을 자동차의 속성에 비유하였습니다. 그럼 함수(Method)는 무엇일까요?



개체(Object:=자동차)가 무엇인지 속성(Property)을 통해서 표현했다면, 함수(Method)는 **자동차의 동작**이라고 표현할 수 있습니다. 아래에 Method의 몇 가지 예제를 보겠습니다.

1. 시동걸기 (Activate)
2. 문열기 (Open)
3. 문닫기 (Close)
4. 움직이기 (Run)
5. 부품정비 (Find) ...

즉 개체/속성/함수는 하나의 묶음이며 아래와 같이 명사/형용사/동사로 비유할 수 있습니다. 우리가 일상생활에서 명사만 가지고 의사소통을 할 수 없으므로 형용사와 동사를 보태서 사용하듯, 개체의 Property와 Method는 개체를 보조하여 동작합니다.

개체(Object)	:	자동차	= 명사 (Noun)
속성(Property)	:	속성	= 형용사 (Adjective)
함수(Method)	:	동작	= 동사 (Verb)

## 2. Method에는 인수(Argument 또는 Parameter)가 붙습니다.

엑셀 워크시트에서 자주 사용하는 VLOOKUP 함수를 보겠습니다.

```
= VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])
// 엑셀 워크시트에서 사용되는 모든 함수에는 인수(Parameter)가 따라 붙습니다.
// 인수에는 필수인수(반드시 입력해야 하는 값)이 있고 선택인수(선택적으로 입력할 수 있는 값, 기본값이 정해져 있음)이 있습니다.
```



VBA에서 사용하는 함수(Method)에서도 인수를 사용합니다. 하지만 몇몇 함수는 인수없이 동작합니다.

```
= 자동차.Run
= 자동차.Activate 등..
Rng.Show
Rng.Activate
Rng.Calculate
Rng.AutoFit
// 몇몇 함수는 인수없이 간단하게 입력합니다.
```

= 자동차.DoorClose (문의\_위치, [자동잠금])

```
rng.PasteSpecial  
End PasteSpecial([Paste As XIPasteType = xlPasteAll], [Operation As XIPasteSpecialOperation = xlPasteSpecialOperationNone], [SkipBlanks], [Transpose])
```

'// 하지만 많은 함수는 워크시트에서 사용하는 함수처럼 인수를 포함합니다.

### 3. 같지만 다른 Method(함수)

몇몇 Method(함수)는 같은 이름은 갖지만, 어떠한 개체(Object)아래에서 사용되는가에 따라 다른 동작을 하거나 다른 인수를 가질 수 있습니다. 아래 2 가지 예제를 같이 보겠습니다.

#### 3-A. Delete 함수

```
Sub Test()  
Dim WS As Worksheet  
Dim Rng As Range  
Set WS = ThisWorkbook.Worksheets("Sheet1")  
Set Rng = WS.Range("A1")  
WS.Delete  
Rng.Delete  
    Delete([Shift])
```

##### A-1. Worksheet 개체의 Delete (WS.Delete)

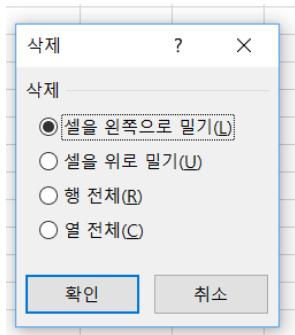
: 인수 (Argument)가 붙지 않습니다.

: Delete 동작 실행 전 Dialog Box 가 실행됩니다. (따라서 일반적으로 Worksheet 의 Delete 를 실행하기 전 Application.DisplayAlerts 를 False 로 바꿔줌으로써 알림창 실행으로 인해 실행이 중단되는 것을 방지합니다.)

```
Application.DisplayAlerts = False  
WS.Delete  
Application.DisplayAlerts = True
```

##### A-2. Range 개체의 Delete (Rng.Delete)

: 추가로 인수(Argumet)가 붙습니다. Range 개체의 Delete 함수는 밀기방향이 인수로 입력됩니다.



### 3-B. Copy 함수

WS.Copy |  
Copy([Before], [After])  
Rng.Copy |

Rng.Copy |  
Copy([Destination])

#### B-1. Worksheet 개체의 Copy (WS.Copy)

: 붙여넣기 할 위치(워크시트)가 인수로 입력됩니다.

: Before / After 중 둘 중 하나만 인수로 입력할 수 있습니다.

#### B-2. Range 개체의 Copy (Rng.Copy)

: 붙여넣기 할 위치(범위)가 인수로 입력됩니다.

### 4. Method 입력 방법

VBA에서 Method(함수)를 입력하는데 2 가지 방법이 있습니다. 각각의 방법을 살펴보겠습니다.

#### 4-A. 순서에 맞춰 콤마(,)로 나눠 입력하는 방법

- : 엑셀 워크시트의 함수 입력방식과 똑같습니다.
- : 입력하는 인수의 종류가 적을 경우 두번째 방법보다 훨씬 간단하게 작성 할 수 있습니다.
- : 필수인수(대괄호"[]"에 들어있지 않은 인수)는 콤마로 입력을 건너뛸 수 없습니다.

```
Sub Test()
```

```
Dim WS As Worksheet
```

```
Dim Rng As Range
```

```
Set WS = ThisWorkbook.Worksheets("Sheet1")
```

```
Set Rng = WS.Range("A1")
```

```
Rng.PasteSpecial , , True
```

'// 콤마로 모든 선택인수는 건너뛰고 필요한 인수 값만 입력합니다.

```
Rng.PasteSpecial , , True
    PasteSpecial([Paste As XlPasteType = xlPasteAll], [Operation As XlPasteSpecialOperation = xlPasteSpecialOperationNone], [SkipBlanks], [Transpose])
```

```
WS.SaveAs "파일이름" , , "abc"
```

```
WS.SaveAs "파일이름" , , "abc"
    SaveAs(Filename As String, [FileFormat], [Password], [WriteResPassword], [ReadOnlyRecommended], [CreateBackup], [AddToMru], [TextCodepage],
    [TextVisualLayout], [Local])
```

```
End Sub
```

#### 4-B. 인수(Argument)를 직접 지정해서 입력하는 방법

- : 입력해야 할 인수의 종류가 많은 경우, 원하는 인수를 직접 지정해서 입력할 수 있습니다.
- : 첫번째 방법보다 사용자가 더욱 이해하기 쉬운 코딩이 가능한 반면, 직접 인수 값을 지정해줘야 하므로 코드를 작성하는데 더 많은 시간이 소요됩니다.
- : 인수이름을 적은 뒤, 콜론과 등호 (:=)로 구분합니다.

```
Sub Test()
```

```
Dim WS As Worksheet
```

```
Dim Rng As Range
```

```
Set WS = ThisWorkbook.Worksheets("Sheet1")
```

```
Set Rng = WS.Range("A1")
```

```
Rng.PasteSpecial Transpose:=True
```

```
WS.SaveAs Filename:="파일이름", Password:="abc"
```

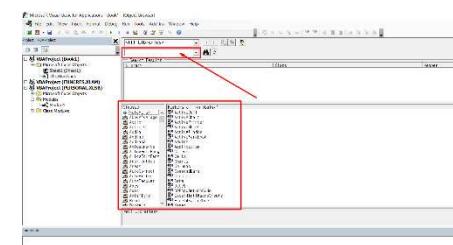
'// 작성해야 할 인수가 많을 경우, 원하는 인수만 직접 지정하여 입력하는 것이 훨씬 간결하고 편리할 때가 있습니다.

```
End Sub
```

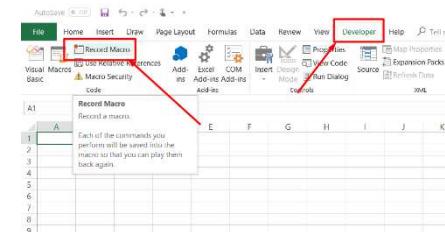
### 5. Method 검색방법

다양한 방법이 있겠지만 저는 주로 3 가지 방법을 사용합니다.

1. VBA 편집기에서 키보드 F2 키를 눌러 라이브러리 창을 불러옵니다. 원하는 개체 또는 클래스를 검색하면 해당 항목에 대한 모든 Property 및 Method 그리고 Event에 대해 확인할 수 있습니다.



2. 엑셀에서 Macro Recording (매크로 녹화기능)을 활용합니다. 매크로 녹화기능을 통해 코드를 작성하게 되면 모든 행동 하나하나가 상당히 자세하게 기록됩니다. 따라서 매크로 녹화기능을 통해 작성된 코드를 분석하면 원하는 행동에 대한 Key Property 또는 Method를 쉽게 확인 할 수 있습니다.



개발자툴(Developer)를 활성화하는 방법은 [여기](#)를 참고하세요.

3. StackOverflow 와 같은 프로그래밍 언어 커뮤니티를 잘 활용하면, 때로는 생각했던 것 이상의 결과물을 얻을 수도 있습니다.

## 6. 예제 코드 전문

```
Sub Test()  
  
Dim WS As Worksheet  
Dim Rng As Range  
Dim PasteRng As Range  
  
Set WS = ThisWorkbook.Worksheets("Sheet1")  
Set Rng = Application.InputBox("범위를 선택하세요", Type:=8)  
  
Rng.Copy  
  
WS.Range("C1").PasteSpecial xlPasteAll, , , True  
WS.Range("C2").PasteSpecial Paste:=xlPasteValues, Transpose:=True  
  
WS.UsedRange.Columns.AutoFit  
  
End Sub
```